Cisco Embedded Wireless Controller on Catalyst Access Points Configuration Guide, IOS XE Bengaluru 17.5.x

First Published: 2021-03-31

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

CONTENTS

Preface

This preface describes the conventions of this document and information on how to obtain other documentation. It also provides information on what's new in Cisco product documentation.

Document Conventions

This document uses the following conventions:

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

Convention	Description
{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 may use the following conventions for reader alerts:

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

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

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

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

Warning IMPORTANT SAFETY INSTRUCTIONS

Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Read the installation instructions before using, installing, or connecting the system to the power source. Use the statement number provided at the end of each warning statement to locate its translation in the translated safety warnings for this device. Statement 1071

SAVE THESE INSTRUCTIONS

Related Documentation

Note Before installing or upgrading the deviceCiscoEmbedded Wireless Controller, refer to the release notes.

Communications, Services, and Additional Information

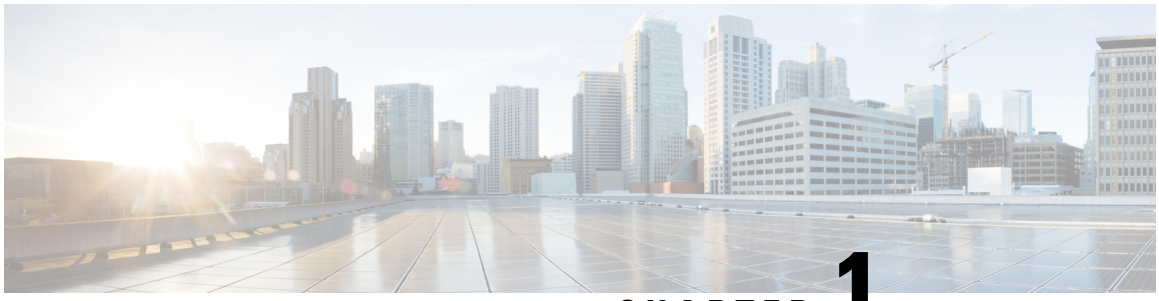
- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions, and services, visit Cisco DevNet.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a gateway to the Cisco bug-tracking system, which maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. The BST provides you with detailed defect information about your products and software.

Documentation Feedback

To provide feedback about Cisco technical documentation, use the feedback form available in the right pane of every online document.



CHAPTER 1

Overview of Cisco Embedded Wireless Controller on Catalyst Access Points

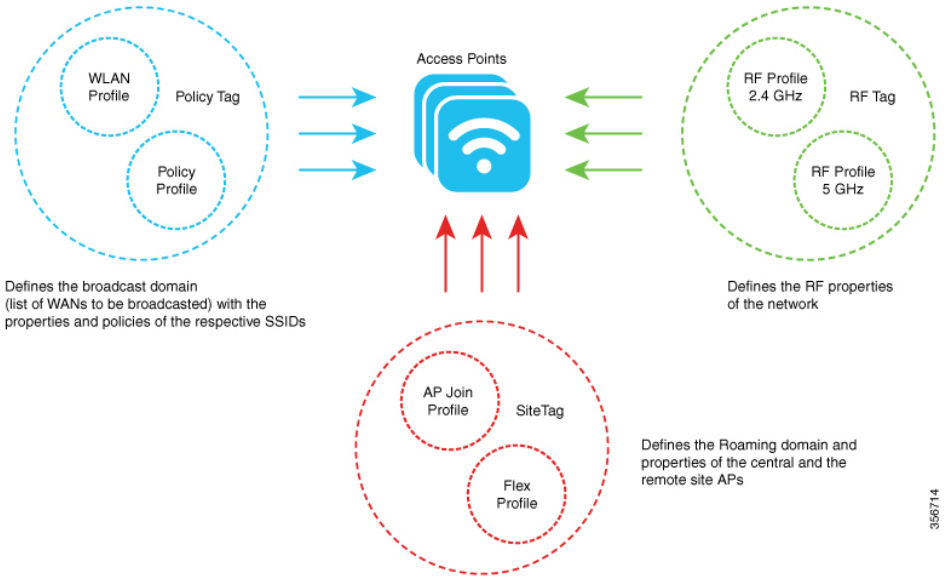
Cisco Embedded Wireless Controller on Catalyst Access Points are the next generation of wireless controllers built for the Intent-based networking. The Cisco controllers are IOS XE based and integrates the RF Excellence from Aironet with Intent-based Networking capabilities of IOS XE to create the best-in-class wireless experience for your evolving and growing organization.

The controllers are deployable in physical form factors and can be managed using Cisco Catalyst Center, Netconf/YANG, web-based GUI, or CLI.

The configuration data model is based on design principles of reusability, simplified provisioning, enhanced flexibility and modularization to help manage networks as they scale up and simplify the management of dynamically changing business and IT requirements.

Elements of the New Configuration Model

The following diagram depicts the elements of the new configuration model.



Tags

The property of a tag is defined by the property of the policies associated to it, which in turn is inherited by an associated client or an AP. There are various type of tags, each of which is associated to different profiles. Every tag has a default that is created when the system boots up.

Profiles

Profiles represent a set of attributes that are applied to the clients associated to the APs or the APs themselves. Profiles are reusable entities that can be used across tags.

Configuration Workflow

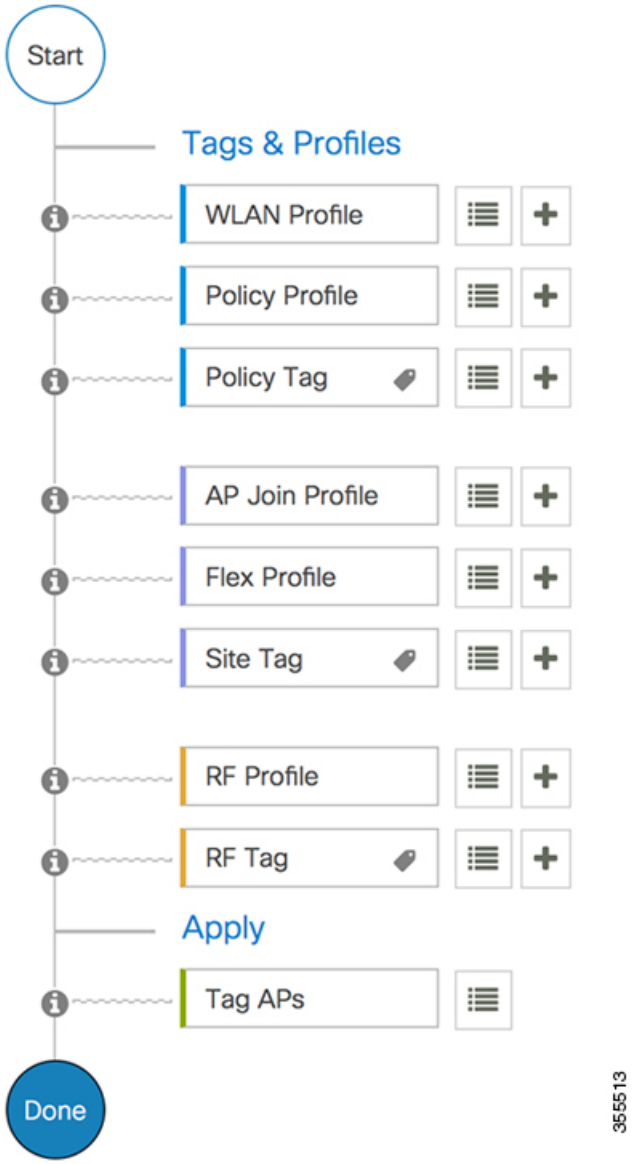
The following set of steps defines the logical order of configuration. Apart from the WLAN profile, all the profiles and tags have a default object associated with it.

1. Create the following profiles:
   - WLAN
   - Policy
   - AP Join
   - Flex
   - RF
2. Create the following tags:
   - Policy
   - Site

- RF

3. Associate tags to an AP.

Figure 1: Configuration Workflow



Initial Setup

Setting up the Controller

The initial configuration wizard in Cisco Embedded Wireless Controller on Catalyst Access Points is a simplified, out-of-the-box installation and configuration interface for controller. This section provides

instructions to set up a controller to operate in a small, medium, or large network wireless environment, where access points can join and together as a simple solution provide various services, such as corporate employee or guest wireless access on the network.

Note From Cisco IOS XE Amsterdam 17.1.x onwards, date and time will not reflect in the web UI unless it is synched with Network Time Protocol (NTP).

Note When the AP has rebooted in the EWC mode, it broadcasts a provisioning SSID ending with the last digits of the MAC address. You can connect to the provisioning SSID using the PSK **password**.

You can then open a browser and you are redirected to mywifi.cisco.com which takes you to the AP web UI. Enter the username as **webui** and password as **cisco**.

Note We recommend that you use the **wireless ewc-ap factory-reset** command to reset the EWC device to Day0 state (with the configuration wizard). This command also resets all the APs and EWC-APs in the network to Day0 state. You can use the **erase startup-config** command to remove the configuration from the device. However, this is not synced to other devices in the network.

Note After completing the Day0 wizard, the internal AP disjoins, and rejoins after one minute.

Note The wireless management must be the AP Gigabit port and you cannot have several SVIs configured in IOS-XE.

Note You must run the **write memory** command after copying a new TAR file.

Interactive Help

The Cisco Catalyst 9800 Series Wireless Controller GUI features an interactive help that walks you through the GUI and guides you through complex configurations.

You can start the interactive help in the following ways:

- By hovering your cursor over the blue flap at the right-hand corner of a window in the GUI and clicking **Interactive Help**.
- By clicking **Walk-me Thru** in the left pane of a window in the GUI.

- By clicking **Show me How** displayed in the GUI. Clicking **Show me How** triggers a specific interactive help that is relevant to the context you are in.

  For instance, **Show me How** in **Configure** > **AAA** walks you through the various steps for configuring a RADIUS server. Choose **Configuration**> **Wireless Setup** > **Advanced** and click **Show me How** to trigger the interactive help that walks you through the steps relating to various kinds of authentication.

The following features have an associated interactive help:

- Configuring AAA
- Configuring FlexConnect Authentication
- Configuring 802.1x Authentication
- Configuring Local Web Authentication
- Configuring OpenRoaming
- Configuring Mesh APs

Note If the WalkMe launcher is unavailable on Safari, modify the settings as follows:

1. Choose **Preferences > Privacy**.
2. In the **Website tracking** section, uncheck the **Prevent cross-site tracking** check box to disable this action.
3. In the **Cookies and website data** section, uncheck the **Block all cookies** check box to disable this action.

Resetting Cisco Embedded Wireless Controller on Catalyst Access Points

To reset the controller on Catalyst APs to factory defaults, follow the steps given below:

Procedure

Step 1 Unplug the Access Point from its power source.

Step 2 Plug in the console cable and open serial session on your computer or laptop.

Step 3 Press and hold the **Mode/Reset** button on the AP.

Step 4 Plug in the AP back to its power source while still pressing the **Mode/Reset** button.

Step 5 Continue holding the button until a prompt is displayed in the serial session on your computer or laptop.

Note The console session also displays for how long the button has been pressed. At least 20 seconds of button press is required for a complete restart.

What to do next

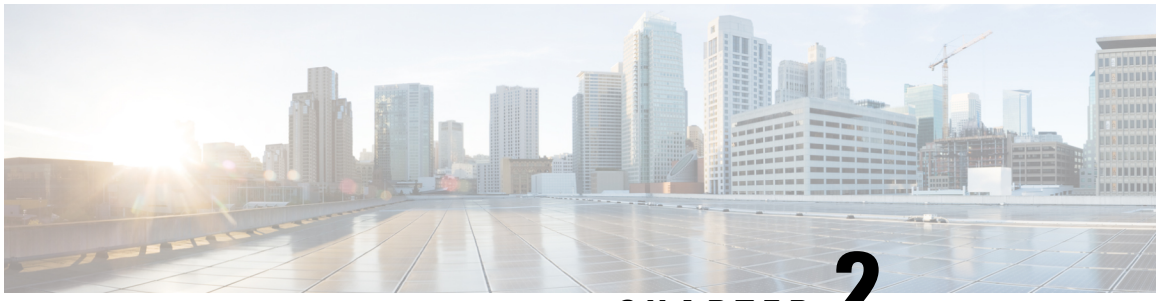
When the AP reboots, use the default credentials **Cisco/Cisco** to log in.

Password Recovery

For password recovery, you must do a factory reset of the AP. For more information about resetting factory defaults, see the Resetting Cisco Embedded Wireless Controller on Catalyst Access Points section.

PART I

System Configuration



CHAPTER 2

System Configuration

Information About New Configuration Model

The configuration of Cisco Embedded Wireless Controller on Catalyst Access Points is simplified using different tags, namely rf-tag, policy-tag, and site-tag. The access points would derive their configuration from the profiles that are contained within the tags.

Profiles are a collection of feature-specific attributes and parameters applied to tags. The rf-tag contains the radio profiles, the policy-tag contains the WLAN profile and policy profile, and the site-tag contains the flex profile and ap-join profile.

Policy Tag

The policy tag constitutes mapping of the WLAN profile to the policy profile. The WLAN profile defines the wireless characteristics of the WLAN. The policy profile defines the network policies and the switching policies for the client (Quality of Service [QoS] is an exception which constitutes AP policies as well).

The policy tag contains the map of WLAN policy profile. There can be a maximum of 16 such entries per policy tag. Changes to the map entries are effected based on the status of the WLAN profile and policy profile. For example, if a map (WLAN1 and Policy1) is added to the policy tag, and both the WLAN profile and the policy profile are enabled, the definitions are pushed to the APs using the policy tag. However, if one of them is in disabled state, the definition is not pushed to the AP. Similarly, if a WLAN profile is already being broadcast by an AP, it can be deleted using the no form of the command in the policy tag.

Site Tag

The site tag defines the properties of a site and contains the flex profile and the AP join profile. The attributes that are specific to the corresponding flex or remote site are part of the flex profile. Apart from the flex profile, the site tag also comprises attributes that are specific to the physical site (and hence cannot be a part of the profile that is a reusable entity). For example, the list of primary APs for efficient upgrade is a part of a site tag rather than that of a flex profile.

If a flex profile name or an AP profile name is changed in the site tag, the AP is forced to rejoin the controller by disconnecting the Datagram Transport Layer Security (DTLS) session. When a site tag is created, the AP and flex profiles are set to default values (default-ap-profile and default-flex-profile).

RF Tag

The RF tag contains the 2.4 GHz and 5 GHz RF profiles. The default RF tag contains the global configuration. Both these profiles contain the same default values for global RF profiles for the respective radios.

Profiles

Profiles are a collection of feature-specific attributes and parameters applied to tags. Profiles are reusable entities that can be used across tags. Profiles (used by tags) define the properties of the APs or its associated clients.

WLAN Profile

WLAN profiles are configured with same or different service set identifiers (SSIDs). An SSID identifies the specific wireless network for the controller to access. Creating WLANs with the same SSID allows to assign different Layer 2 security policies within the same wireless LAN.

To distinguish WLANs having the same SSID, create a unique profile name for each WLAN. WLANs with the same SSID must have unique Layer 2 security policies so that clients can select a WLAN based on the information advertised in the beacon and probe responses. The switching and network policies are not part of the WLAN definition.

Policy Profile

Policy profile broadly consists of network and switching policies. Policy profile is a reusable entity across tags. Anything that is a policy for a client that is applied on an AP or controller is moved to the policy profile, for example, VLAN, ACL, QoS, session timeout, idle timeout, AVC profile, bonjour profile, local profiling, device classification, BSSID QoS, and so on. However, all the wireless-related security attributes and features on the WLAN are grouped under the WLAN profile.

Flex Profile

Flex profile contains policy attributes and remote site-specific parameters. For example, the EAP profiles that can be used when the AP acts as an authentication server for local RADIUS server information, VLAN-ACL mapping, VLAN name-to-ID mapping, and so on.

AP Join Profile

The default AP join profile values will have the global AP parameters and the AP group parameters. The AP join profile contains attributes that are specific to AP, such as CAPWAP, IPv4 and IPv6, UDP Lite, High Availability, Retransmit config parameters, Global AP failover, Hyperlocation config parameters, Telnet and SSH, 11u parameters, and so on.

Note Telnet is not supported for the following Cisco AP models: 1542D, 1542I, 1562D, 1562E, 1562I, 1562PS, 1800S, 1800T, 1810T, 1810W,1815M, 1815STAR, 1815TSN, 1815T, 1815W, 1832I, 1840I, 1852E, 1852I, 2802E, 2802I, 2802H, 3700C, 3800, 3802E, 3802I, 3802P, 4800, IW6300, ESW6300, 9105AXI, 9105AXW, 9115AXI, 9115AXE, 9117I, APVIRTUAL, 9120AXI, 9120AXE, 9124AXI, 9124AXD, 9130AXI, 9130AXE, 9136AXI, 9162I, 9164I, and 9166I.

RF Profile

RF profile contains the common radio configuration for the APs. RF profiles are applied to all the APs that belong to an AP group, where all the APs in that group have the same profile settings.

Association of APs

APs can be associated using different ways. The default option is by using Ethernet MAC address, where the MAC is associated with policy-tag, site tag, and RF tag.

In filter-based association, APs are mapped using regular expressions. A regular expression (regex) is a pattern to match against an input string. Any number of APs matching that regex will have policy-tag, site tag, and RF tag mapped to them, which is created as part of the AP filter.

In AP-based association, tag names are configured at the PnP server and the AP stores them and sends the tag name as part of discovery process.

In location-based association, tags are mapped as per location and are pushed to any AP Ethernet MAC address mapped to that location.

Modifying AP Tags

Modifying an AP tag results in DTLS connection reset, forcing the AP to rejoin the controller. If only one tag is specified in the configuration, default tags are used for other types, for example, if only policy tag is specified, the default-site-tag and default-rf-tag will be used for site tag and RF tag.

Configuring a Wireless Profile Policy (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the **Policy Profile** page, click **Add**.

Step 3 In the **Add Policy Profile** window, in **General** tab, enter a name and description for the policy profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces. Do not use spaces as it causes system instability.

Step 4 To enable the policy profile, set **Status** as **Enabled**.

Step 5 In the WLAN Switching Policy section, choose the following, as required:

- No Central Switching: Tunnels both the wireless user traffic and all control traffic via CAPWAP to the centralized controller where the user traffic is mapped to a dynamic interface/VLAN on the controller. This is the normal CAPWAP mode of operation.
- Central Authentication: Tunnels client data to the controller, as the controller handles client authentication.
- No Central DHCP: The DHCP packets received from AP are centrally switched to the controller and then forwarded to the corresponding VLAN based on the AP and the SSID.
- Central Association Enable: When central association is enabled, all switching is done on the controller.
- Flex NAT/PAT: Enables Network Address Translation(NAT) and Port Address Translation (PAT) mode.

Step 6 Click **Save & Apply to Device**.

Configuring a Wireless Profile Policy (CLI)

Follow the procedure given below to configure a wireless profile policy:

Note When a client moves from an old controller to a new controller (managed by Cisco Prime Infrastructure), the old IP address of the client is retained, if the IP address is learned by ARP or data gleaning. To avoid this scenario, ensure that you enable **ipv4 dhcp required** command in the policy profile. Otherwise, the IP address gets refreshed only after a period of 24 hours.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-policy* **Example:** `Device(config)# wireless profile policy rr-xyz-policy-1`	Configures WLAN policy profile and enters wireless policy configuration mode.
Step 3	**idle-timeout** *timeout* **Example:** `Device(config-wireless-policy)# idle-timeout 1000`	(Optional) Configures the duration of idle timeout, in seconds.

	Command or Action	Purpose
Step 4	**vlan** *vlan-id* **Example:** `Device(config-wireless-policy)# vlan 24`	Configures VLAN name or VLAN ID.
Step 5	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Saves the configuration and exits configuration mode and returns to privileged EXEC mode.
Step 6	**show wireless profile policy summary** **Example:** `Device# show wireless profile policy summary`	Displays the configured policy profiles. **Note** (Optional) To view detailed information about a policy profile, use the **show wireless profile policy detailed** *policy-profile-name* command.

Configuring a Flex Profile

Follow the procedure given below to set a flex profile:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile flex** *flex-profile* **Example:** `Device(config)# wireless profile flex rr-xyz-flex-profile`	Configures a Flex profile and enters Flex profile configuration mode.
Step 3	**description** **Example:** `Device(config-wireless-flex-profile)# description xyz-default-flex-profile`	(Optional) Enables default parameters for the flex profile.
Step 4	**arp-caching** **Example:** `Device(config-wireless-flex-profile)# arp-caching`	(Optional) Enables ARP caching.
Step 5	**end** **Example:**	Saves the configuration and exits configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
	`Device(config-wireless-flex-profile)# end`	
Step 6	**show wireless profile flex summary** **Example:** `Device# show wireless profile flex summary`	(Optional) Displays the flex-profile parameters. **Note** To view detailed parameters about the flex profile, use the **show wireless profile flex detailed** *flex-profile-name* command.

Configuring an AP Profile (GUI)

Before you begin

The default AP join profile values will have the global AP parameters and the AP group parameters. The AP join profile contains attributes that are specific to AP, such as CAPWAP, IPv4/IPv6, UDP Lite, High Availability, retransmit configuration parameters, global AP failover, Hyperlocation configuration parameters, Telnet/SSH, 11u parameters, and so on.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** page, click **Add**.

The **Add AP Join Profile** page is displayed.

Step 3 In the **General** tab, enter a name and description for the AP join profile.

Step 4 Check the **LED State** check box to set the LED state of all APs connected to the device to blink so that the APs are easily located.

Step 5 In the **Client** tab and **Statistics Timer** section, enter the time in seconds that the AP sends its 802.11 statistics to the controller.

Step 6 In the **TCP MSS Configuration** section, check the **Adjust MSS Enable** check box to enter value for Adjust MSS. You can enter or update the maximum segment size (MSS) for transient packets that traverse a router. TCP MSS adjustment enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments with the SYN bit set.

In a CAPWAP environment, a lightweight access point discovers a device by using CAPWAP discovery mechanisms, and then sends a CAPWAP join request to the device. The device sends a CAPWAP join response to the access point that allows the access point to join the device.

When the access point joins the device, the device manages its configuration, firmware, control transactions, and data transactions.

Step 7 In the **AP** tab, you can configure the following:

- General

a) In the **General** tab, check the **Switch Flag** check box to enable switches.

b) Check the **Power Injector State** check box if power injector is being used. Power Injector increases wireless LAN deployment flexibility of APs by providing an alternative powering option to local power, inline power-capable multiport switches, and multiport power patch panels.
c) From the **Power Injector Type** drop-down list, choose power injector type from the following options:

- Installed: If you want the AP to examine and remember the MAC address of the currently connected switch port. (This selection assumes that a power injector is connected.)
- Override: To enable the AP to operate in high-power mode without first verifying a matching MAC address.

d) In the **Injector Switch MAC** field, enter the MAC address of the switch.
e) From the **EAP Type** drop-down list, choose the EAP type as *EAP-FAST*, *EAP-TLS*, or *EAP-PEAP*.
f) From the **AP Authorization Type** drop-down list, choose the type as either *CAPWAP DTLS +* or *CAPWAP DTLS*.
g) In the **Client Statistics Reporting Interval** section, enter the interval for 5 GHz and 2.4 GHz radios in seconds.
h) Check the **Enable** check box to enable extended module.
i) From the **Profile Name** drop-down list, choose a profile name.
j) Click **Save & Apply to Device**.

- Hyperlocation: Cisco Hyperlocation is a location solution that allows to track the location of wireless clients with the accuracy of one meter. Selecting this option disables all other fields in the screen, except NTP Server.

a) In the **Hyperlocation** tab, check the **Enable Hyperlocation** check box.
b) Enter the **Detection Threshold** value to filter out packets with low RSSI. The valid range is –100 dBm to –50 dBm.
c) Enter the **Trigger Threshold** value to set the number of scan cycles before sending a BAR to clients. The valid range is 0 to 99.
d) Enter the **Reset Threshold** value to reset value in scan cycles after trigger. The valid range is 0 to 99.
e) Enter the **NTP Server** IP address.
f) Click **Save & Apply to Device**.

- BLE: If your APs are Bluetooth Low Energy (BLE) enabled, they can transmit beacon messages that are packets of data or attributes transmitted over a low energy link. These BLE beacons are frequently used for health monitoring, proximity detection, asset tracking, and in-store navigation. For each AP, you can customize BLE Beacon settings configured globally for all APs.

a) In the **BLE** tab, enter a value in the **Beacon Interval** field to indicate how often you want your APs to send out beacon advertisements to nearby devices. The range is from 1 to 10, with a default of 1.
b) In the **Advertised Attenuation Level** field, enter the attenuation level. The range is from 40 to 100, with a default of 59.
c) Click **Save & Apply to Device**.

Step 8 In the **Management** tab, you can configure the following:

- Device

a) In the **Device** tab, enter the **IPv4/IPv6 Address** of the TFTP server, **TFTP Downgrade** section.
b) In the **Image File Name** field, enter the name of the software image file.
c) From the **Facility Value** drop-down list, choose the appropriate facility.

d) Enter the IPv4 or IPv6 address of the host.
e) Choose the appropriate **Log Trap Value**.
f) Enable Telnet and/or SSH configuration, if required.
g) Enable core dump, if required.
h) Click **Save & Apply to Device**.

- User

a) In the **User** tab, enter username and password details.
b) Choose the appropriate password type.
c) In the **Secret** field, enter a custom secret code.
d) Choose the appropriate secret type.
e) Choose the appropriate encryption type.
f) Click **Save & Apply to Device**.

- Credentials

a) In the **Credentials** tab, enter local username and password details.
b) Choose the appropriate local password type.
c) Enter 802.1x username and password details.
d) Choose the appropriate 802.1x password type.
e) Enter the time in seconds after which the session should expire.
f) Enable local credentials and/or 802.1x credentials as required.
g) Click **Save & Apply to Device**.

a) In the **CDP Interface** tab, enable the CDP state, if required.
b) Click **Save & Apply to Device**.

Step 9 In the **Rogue AP** tab, check the **Rogue Detection** check box to enable rogue detection.

Step 10 In the **Rogue Detection Minimum RSSI** field, enter the RSSI value.

This field specifies the minimum RSSI value for which a Rogue AP should be reported. All Rogue APs with RSSI lower than what is configured will not be reported to controller.

Step 11 In the **Rogue Detection Transient Interval** field, enter the transient interval value.

This field indicates how long the Rogue AP should be seen before reporting the controller.

Step 12 In the **Rogue Detection Report Interval** field, enter the report interval value.

This field indicates the frequency (in seconds) of Rogue reports sent from AP to controller.

Step 13 Check the **Rogue Containment Automatic Rate Selection** check box to enable rogue containment automatic rate selection.

Here, the AP selects the best rate for the target Rogue, based on its RSSI.

Step 14 Check the **Auto Containment on FlexConnect Standalone** check box to enable the feature.

Here, the AP will continue containment in case it moves to flexconnect standalone mode.

Step 15 Click **Save & Apply to Device**.

Configuring an AP Profile (CLI)

Follow the procedure given below to configure and AP profile:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *ap-profile* **Example:** `Device(config)# ap profile xyz-ap-profile`	Configures an AP profile and enters AP profile configuration mode. **Note** In an AP profile, the **EAP-FAST** is the default EAP type. **Note** When you delete a named profile, the APs associated with that profile will not revert to the default profile.
Step 3	**description** *ap-profile-name* **Example:** `Device(config-ap-profile)# description "xyz ap profile"`	Adds a description for the ap profile.
Step 4	**cdp** **Example:** `Device(config-ap-profile)# cdp`	Enables CDP for all Cisco APs.
Step 5	**end** **Example:** `Device(config-ap-profile)# end`	Saves the configuration and exits configuration mode and returns to privileged EXEC mode.
Step 6	**show ap profile name***profile-name* **detailed** **Example:** `Device# show ap profile name xyz-ap-profile detailed`	(Optional) Displays detailed information about an AP join profile.

Configuring an RF Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **RF**.

Step 2 On the **RF Profile** page, click **Add**.

Step 3 In the **General** tab, enter a name for the RF profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

Step 4 Choose the appropriate **Radio Band**.

Step 5 To enable the profile, set the status as **Enable**.

Step 6 Enter a **Description** for the RF profile.

Step 7 Click **Save & Apply to Device**.

Configuring an RF Profile (CLI)

Follow the procedure given below to configure an RF profile:

Before you begin

Ensure that you use the same RF profile name that you create here, when configuring the wireless RF tag too. If there is a mismatch in the RF profile name (for example, if the RF tag contains an RF profile that does not exist), the corresponding radios will not come up.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz rf-profile** *rf-profile* **Example:** `Device(config)# ap dot11 24ghz rf-profile rfprof24_1`	Configures an RF profile and enters RF profile configuration mode. **Note** Use the **24ghz** command to configure the 802.11b parameters. Use the **5ghz** command to configure the 802.11a parameters.
Step 3	**default** **Example:** `Device(config-rf-profile)# default`	(Optional) Enables default parameters for the RF profile.

	Command or Action	Purpose
Step 4	**no shutdown** **Example:** `Device(config-rf-profile)# no shutdown`	Enables the RF profile on the device.
Step 5	**end** **Example:** `Device(config-rf-profile)# end`	Exits configuration mode and returns to privileged EXEC mode.
Step 6	**show ap rf-profile summary** **Example:** `Device# show ap rf-profile summary`	(Optional) Displays the summary of the available RF profiles.
Step 7	**show ap rf-profile name** *rf-profile* **detail** **Example:** `Device# show ap rf-profile name rfprof24_1 detail`	(Optional) Displays detailed information about a particular RF profile.

Configuring Policy Tag (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **Policy**.

Step 2 Click **Add** to view the **Add Policy Tag** window.

Step 3 Enter a name and description for the policy tag. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

Step 4 Click **Add** to map WLAN and policy.

Step 5 Choose the WLAN profile to map with the appropriate policy profile, and click the tick icon.

Step 6 Click **Save & Apply to Device**.

Configuring a Policy Tag (CLI)

Follow the procedure given below to configure a policy tag:

Procedure

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

	Command or Action	Purpose
	`Device> enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wireless tag policy** *policy-tag-name* **Example:** `Device(config-policy-tag)# wireless tag policy default-policy-tag`	Configures policy tag and enters policy tag configuration mode. **Note** When performing LWA, the clients connected to a controller gets disconnected intermittently before session timeout.
Step 4	**description** *description* **Example:** `Device(config-policy-tag)# description "default-policy-tag"`	Adds a description to a policy tag.
Step 5	**remote-lan** *name* **policy** *profile-policy-name* {**ext-module**| **port-id** } **Example:** `Device(config-policy-tag)# remote-lan rr-xyz-rlan-aa policy rr-xyz-rlan-policy1 port-id 2`	Maps a remote-LAN profile to a policy profile.
Step 6	**wlan** *wlan-name* **policy** *profile-policy-name* **Example:** `Device(config-policy-tag)# wlan rr-xyz-wlan-aa policy rr-xyz-policy-1`	Maps a policy profile to a WLAN profile.
Step 7	**end** **Example:** `Device(config-policy-tag)# end`	Exits policy tag configuration mode, and returns to privileged EXEC mode.
Step 8	**show wireless tag policy summary** **Example:** `Device# show wireless tag policy summary`	(Optional) Displays the configured policy tags. **Note** To view detailed information about a policy tag, use the **show wireless tag policy detailed** *policy-tag-name* command.

Configuring Wireless RF Tag (GUI)

Procedure

Step 1 a) Choose **Configuration** > **Tags & Profiles** > **Tags** > **RF**.

Step 2 Click **Add** to view the **Add RF Tag** window.

Step 3 Enter a name and description for the RF tag. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

Step 4 Choose the required **5 GHz Band RF Profile** and **2.4 GHz Band RF Profile** to be associated with the RF tag.

Step 5 Click **Update & Apply to Device**.

Configuring Wireless RF Tag (CLI)

Follow the procedure given below to configure a wireless RF tag:

Before you begin

- You can use only two profiles (2.4-GHz and 5-GHz band RF profiles) in an RF tag.
- Ensure that you use the same AP tag name that you created when configuring the AP tag task too.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless tag rf** *rf-tag* **Example:** `Device(config)# wireless tag rf rftag1`	Creates an RF tag and enters wireless RF tag configuration mode.
Step 3	**24ghz-rf-policy** *rf-policy* **Example:** `Device(config-wireless-rf-tag)# 24ghz-rf-policy rfprof24_1`	Attaches an IEEE 802.11b RF policy to the RF tag. To configure a dot11a policy, use the **5ghz-rf-policy** command.
Step 4	**description** *policy-description* **Example:** `Device(config-wireless-rf-tag)# description Test`	Adds a description for the RF tag.

	Command or Action	Purpose
Step 5	**end** **Example:** `Device(config-wireless-rf-tag)# end`	Exits configuration mode and returns to privileged EXEC mode.
Step 6	**show wireless tag rf summary** **Example:** `Device# show wireless tag rf summary`	Displays the available RF tags.
Step 7	**show wireless tag rf detailed** *rf-tag* **Example:** `Device# show wireless tag rf detailed rftag1`	Displays detailed information of a particular RF tag.

Attaching a Policy Tag and Site Tag to an AP (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

The **All Access Points** section displays details of all the APs on your network.

Step 2 To edit the configuration details of an AP, select the row for that AP.

The **Edit AP** window is displayed.

Step 3 In the **General** tab and **Tags** section, specify the appropriate policy, site, and RF tags, that you created on the **Configuration** > **Tags & Profiles** > **Tags** page.

Step 4 Click **Update & Apply to Device**.

Attaching Policy Tag and Site Tag to an AP (CLI)

Follow the procedure given below to attach a policy tag and a site tag to an AP:

Procedure

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

	Command or Action	Purpose
Step 2	**ap** *mac-address* **Example:** `Device(config)# ap F866.F267.7DFB`	Configures a Cisco AP and enters AP profile configuration mode. **Note** The *mac-address* should be a wired mac address.
Step 3	**policy-tag** *policy-tag-name* **Example:** `Device(config-ap-tag)# policy-tag rr-xyz-policy-tag`	Maps a policy tag to the AP.
Step 4	**site-tag** *site-tag-name* **Example:** `Device(config-ap-tag)# site-tag rr-xyz-site`	Maps a site tag to the AP.
Step 5	**rf-tag** *rf-tag-name* **Example:**	Associates the RF tag.
Step 6	**end** **Example:** `Device(config-ap-tag)# end`	Saves the configuration, exits configuration mode, and returns to privileged EXEC mode.
Step 7	**show ap tag summary** **Example:** `Device# show ap tag summary`	(Optional) Displays AP details and the tags associated to it.
Step 8	**show ap name** <*ap-name*> **tag info** **Example:** `Device# show ap name ap-name tag info`	(Optional) Displays the AP name with tag information.
Step 9	**show ap name** <*ap-name*> **tag detail** **Example:** `Device# show ap name ap-name tag detail`	(Optional) Displays the AP name with tag detals.

Time Management

The date and time of the system on EWC is configured when you run the initial wireless express setup wizard. You can change or configure the time from the GUI menu by choosing **Administration** > **Time**.

You can configure a Network Time Protocol (NTP) server to synchronize date and time, if it was not configured during the wireless express setup. Greenwich Mean Time (GMT) is used as the standard for setting the time zone on the controller. You can also update or add the specific NTP server to EWC.

Note EWC APs do not track time when powered off. Therefore, we recommned you to configure NTP to keep a proper time across reboots on the EWC.

AP Filter

Introduction to AP Filter

The introduction of tags in the new configuration model in the Cisco Embedded Wireless Controller on Catalyst Access Points has created multiple sources for tags to be associated with access points (APs). Tag sources can be static configuration, AP filter engine, per-AP PNP, or default tag sources. In addition to this, the precedence of the tags also plays an important role. The AP filter feature addresses these challenges in a seamless and intuitive manner.

AP filters are similar to the access control lists (ACLs) used in the controller and are applied at the global level. You can add AP names as filters, and other attributes can be added as required. Add the filter criteria as part of the discovery requests.

The AP Filter feature organizes tag sources with the right priority, based on the configuration.

You cannot disable the AP filter feature. However, the relative priority of a tag source can be configured using **ap filter-priority** *priority filter-name* command.

Note You can configure tag names at the PnP server (similar to the Flex group and AP group) and the AP stores and send the tag name as part of discovery and join requests.

Set Tag Priority (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Tag Source**.

Step 2 Drag and Drop the Tag Sources to change priorities.

Set Tag Priority

Multiple tag sources might result in ambiguity for network administrators. To address this, you can define priority for tags. When an AP joins the controller, the tags are picked based on priority. If precedence is not set, the defaults are used.

Use the following procedure to set tag priority:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	**ap tag-source-priority** *source-priority* **source {filter | pnp}** **Example:** `Device(config)# ap tag-source-priority 2 source pnp`	Configures AP tag source priority. **Note** It is not mandatory to configure AP filter. It comes with default priorities for Static, Filter, and PnP.
Step 3	**end** **Example:** `Device(config)# end`	Exits configuration mode and returns to privileged EXEC mode.
Step 4	**ap tag-sources revalidate** **Example:** `Device# ap tag-sources revalidate`	Revalidates AP tag sources. The priorities become active only after this command is run. **Note** If you change the priorities for Filter and PnP, and want to evaluate them, run the **revalidate** command.

Create an AP Filter (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Filter**.

Step 2 Click **Add**.

Step 3 In the **Associate Tags to AP** dialog box which is displayed, enter the **Rule Name**, the **AP name regex** and the **Priority**. Optionally, you can also choose the policy tag from the **Policy Tag Name** drop-down list, the site tag from the **Site Tag Name** drop-down list and the RF tag from the **RF Tag Name** drop-down list.

Step 4 Click **Apply to Device**.

Create an AP Filter (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	**ap filter name** *filter_name* **Example:** `Device(config)# ap filter filter-1`	Configures an AP filter.
Step 3	**ap name-regex** *regular-expression* **Example:** `Device(config-ap-filter)# ap name-regex testany`	Configures the AP filter based on regular expression. For example, if you have named an AP as **ap-lab-12**, then you can configure the filter with a regular expression, such as **ap-lab-\d+** , to match the AP name.
Step 4	**tag policy** *policy-tag* **Example:** `Device(config-ap-filter)# tag policy pol-tag1`	Configures a policy tag for this filter.
Step 5	**tag rf** *rf-tag* **Example:** `Device(config-ap-filter)# tag rf rf-tag1`	Configures an RF tag for this filter.
Step 6	**tag site** *site-tag* **Example:** `Device(config-ap-filter)# tag site site1`	Configures a site tag for this filter.
Step 7	**end** **Example:** `Device(config-ap-filter)# end`	Exits configuration mode and returns to privileged EXEC mode.

Set Up and Update Filter Priority (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Filter**.

Step 2

a) If you want to setup a new AP filter, then click **Add**. In the **Associate Tags to AP** dialog box which is displayed, enter the **Rule Name**, the **AP name regex** and the **Priority**. Optionally, you can also select the **Policy Tag Name**, the **Site Tag Name** and the **RF Tag Name**. Click **Apply to Device**.

b) If you want to update the priority of an existing AP filter, click on the Filter and in the **Edit Tags** dialog box and change the **Priority**. In case the Filter is Inactive, no priority can be set to it. Click **Update and Apply to Device**.

Set Up and Update Filter Priority

Follow the procedure given below to set and update filter priority:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap filter priority** *priority* **filter-name** *filter-name* **Example:** `Device(config)# ap filter priority 10 filter-name test1`	Configure AP filter priority. Valid values range from 0 to 1023; 0 is the highest priority. **Note** A filter without a priority is not active. Similarly, you cannot set a filter priority without a filter.
Step 3	**end** **Example:** `Device(config-ap)# end`	Exits configuration mode and returns to privileged EXEC mode.

Verify AP Filter Configuration

The following **show** commands are used to display tag sources and filters, and their priorities.

To view the tag source priorities, use the following command:

```
Device# show ap tag sources

Priority Tag source
--------------------------------
0 Static
1 Filter
2 AP
3 Default
```

To view the available filters, use the following command:

```
Device# show ap filter all

Filter Name            regex            Policy Tag              RF Tag              Site
 Tag
------------------------------------------------------------------------------------------------
```

```
first                   abcd             pol-tag1                rf-tag1
site-tag1
test1                  testany                                                site1

filter1                 testany
```

To view the list of active filters, use the following command:

```
Device# show ap filters active

Priority   Filter Name       regex          Policy Tag              RF Tag
Site Tag
------------------------------------------------------------------------------------------------------------------
10         test1             testany
site1
```

To view the source of an AP tag, use the following command:

```
Device# show ap tag summary

Number of APs: 4

AP Name          AP Mac         Site Tag Name    Policy Tag Name    RF Tag Name
Misconfigured Tag Source
------------------------------------------------------------------------------------------------------------------
AP002A.1034.CA78 002a.1034.ca78 named-site-tag   named-policy-tag   named-rf-tag   No Filter

AP00A2.891C.2480 00a2.891c.2480 named-site-tag   named-policy-tag   named-rf-tag   No Filter

AP58AC.78DE.9946 58ac.78de.9946 default-site-tag default-policy-tag default-rf-tag No AP
AP0081.C4F4.1F34 0081.c4f4.1f34 default-site-tag default-policy-tag default-rf-tag No Default
```

Configuring Access Point for Location Configuration

Information About Location Configuration

During location configuration, you can perform the following:

- Configure a site or location for an AP.
- Configure a set of tags for this location.
- Add APs to this location.

Any location comprises of the following components:

- A set of unique tags, one for each kind, namely: Policy, RF and Site.
- A set of ethernet MAC addresses that applies to the tags.

This feature works in conjunction with the existing tag resolution scheme. The location is considered as a new tag source to the existing system. Similar, to the static tag source.

Prerequisite for Location Configuration

If you configure an access point in one location, you cannot configure the same access point in another location.

Configuring a Location for an Access Point (GUI)

Before you begin

Note When you create local and remote sites in the Basic Setup workflow, corresponding policies and tags are created in the backend. These tags and policies that are created in the Basic Setup cannot be modified using the Advanced workflow, and vice versa.

Procedure

Step 1 Choose **Configuration** > **Wireless Setup** > **Basic**.

Step 2 On the **Basic Wireless Setup** page, click **Add**.

Step 3 In the **General** tab, enter a name and description for the location.

Step 4 Set the **Location Type** as either *Local* or *Flex*.

Step 5 Use the slider to set **Client Density** as *Low*, *Typical* or *High*.

Step 6 Click **Apply**.

Configuring a Location for an Access Point (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap location name** *location_name* **Example:** `Device(config)# ap location name location1`	Configures a location for an access point. Run the **no** form of this command to remove location for an access point.
Step 3	**tag** {**policy** *policy_name*| **rf** *rf_name* | **site** **site_name**} **Example:** `Device(config-ap-location)# tag policy policy_tag`	Configures tags for the location.

	Command or Action	Purpose
	`Device(config-ap-location)# tag rf rf_tag` `Device(config-ap-location)# tag site site_tag`	
Step 4	**location** *description* **Example:** `Device(config-ap-location)# location description`	Adds description to the location.
Step 5	**end** **Example:** `Device(config-ap-location)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Adding an Access Point to the Location (GUI)

Note When the tag source is not set to location, the AP count and AP location tagging will not be correctly reflected on the web UI. To change static tag source on the AP, run the **no ap** *ap-mac* command on the controller to change AP tag source to default (which is location).

Procedure

Step 1 Choose **Configuration** > **Wireless Setup** > **Basic**.

Step 2 On the**Basic Wireless Setup** page, click **Add** to configure the following:

- General
- Wireless Networks
- AP Provisioning

Step 3 In the **AP Provisioning** tab and **Add/Select APs** section, enter the AP MAC address and click the right arrow to add the AP to the associated list.

You can also add a CSV file from your system. Ensure that the CSV has the MAC Address column.

Step 4 Use the search option in the **Available AP List** to select the APs from the Selected AP list and click the right arrow to add the AP to the associated list.

Step 5 Click **Apply**.

Adding an Access Point to the Location (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap location name** *location_name* **Example:** `Device(config)# ap location name location1`	Configures a location for an access point.
Step 3	**ap-eth-mac** *ap_ethernet_mac* **Example:** `Device(config-ap-location)# ap-eth-mac 188b.9dbe.6eac`	Adds an access point to the location.
Step 4	**end** **Example:** `Device(config-ap-location)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode. **Note** After adding an AP to a location, the AP may reset automatically to get the new configuration

Configuring SNMP in Location Configuration

SNMP

EWC does not support SNMP and does not implement the SNMP MIBs of Cisco Catalyst 9800 Series Wireless Controllers, although EWC might respond to some of the object identifiers (OIDs).

Verifying Location Configuration

To view the summary of AP location configuration, use the following command:

```
Device# show ap location summary

Location Name    Description     Policy Tag           RF Tag           Site Tag
-------------------------------------------------------------------------------------------
first            first floor     default-policy-tag   default-rf-tag   default-site-tag
second           second floor    default-policy-tag   default-rf-tag   default-site-tag
```

To view the AP location configuration details for a specific location, use the following command:

```
Device# show ap location details first
```

```
Location Name.....................: first
Location description..............: first floor
Policy tag........................: default-policy-tag
Site tag..........................: default-site-tag
RF tag............................: default-rf-tag

Configured list of APs
005b.3400.0af0
005b.3400.0bf0
```

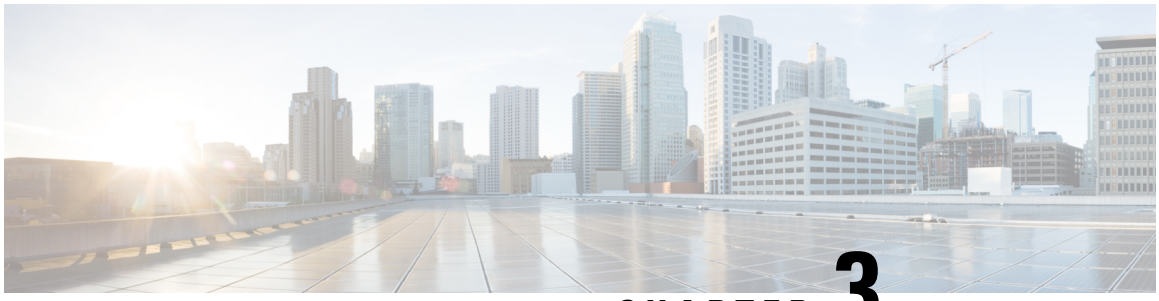
To view the AP tag summary, use the following command:

```
Device# show ap tag summary

Number of APs: 4
AP Name      AP Mac           Site Tag Name      Policy Tag Name      RF Tag Name
Misconfigured  Tag Source
-----------------------------------------------------------------------------------------------------
Asim_5-1     005b.3400.02f0   default-site-tag   default-policy-tag   default-rf-tag   Yes
            Filter
Asim_5-2     005b.3400.03f0   default-site-tag   default-policy-tag   default-rf-tag   No
            Default
Asim_5-9     005b.3400.0af0   default-site-tag   default-policy-tag   default-rf-tag   No
            Location
Asim_5-10    005b.3400.0bf0   default-site-tag   default-policy-tag   default-rf-tag   No
            Location
```

Verifying Location Statistics

To view the AP location statistics, use the following command:

```
Device# show ap location stats

Location name    APs joined     Clients joined     Clients on 11a     Clients on 11b
-----------------------------------------------------------------------------------------------
first             2             0                   3                  4
second            0             0                   0                  0
```



CHAPTER 3

Smart Licensing Using Policy

Introduction to Smart Licensing Using Policy

Smart Licensing Using Policy is an enhanced version of Smart Licensing, with the overarching objective of providing a licensing solution that does not interrupt the operations of your network, rather, one that enables a compliance relationship to account for the hardware and software licenses you purchase and use.

Smart Licensing Using Policy is supported starting with Cisco IOS XE Amsterdam 17.3.2a.

The primary benefits of this enhanced licensing model are:

- Seamless day-0 operations

  After a license is ordered, no preliminary steps, such as registration or generation of keys etc., are required unless you use an export-controlled or enforced license. There are no export-controlled or enforced licenses on Cisco Catalyst Wireless Controllers and product features can be configured on the device right-away.

- Consistency in Cisco IOS XE

  Campus and industrial ethernet switching, routing, and wireless devices that run Cisco IOS XE software, have a uniform licensing experience.

- Visibility and manageability

  Tools, telemetry and product tagging, to know what is in-use.

- Flexible, time series reporting to remain compliant

  Easy reporting options are available, whether you are directly or indirectly connected to Cisco Smart Software Manager (CSSM), or in an air-gapped network.

This document provides conceptual, configuration, and troubleshooting information for Smart Licensing Using Policy on Cisco Catalyst Wireless Controllers.

For a more detailed overview on Cisco Licensing, go to cisco.com/go/licensingguide.

Information About Smart Licensing Using Policy

This section provides conceptual information about Smart Licensing Using Policy, supported products, an overview of each supported topology, and explains how Smart Licensing Using Policy interacts, with other features.

Overview

Smart Licensing Using Policy is a software license management solution that provides a seamless experience with the various aspects of licensing.

- Purchase licenses: Purchase licenses through the existing channels and use the Cisco Smart Software Manager (CSSM) portal to view product instances and licenses.

Note To simplify your implementation of Smart Licensing Using Policy, provide your Smart Account and Virtual Account information when placing an order for new hardware or software. This allows Cisco to install applicable policies and authorization codes (terms explained in the Concepts, on page 38 section below), at the time of manufacturing.

- Use: All licenses on Cisco Catalyst Wireless Controllers are unenforced. This means that you do not have to complete any licensing-specific operations, such as registering or generating keys before you start using the software and the licenses that are tied to it. License usage is recorded on your device with timestamps and the required workflows can be completed at a later date.
- Report license usage to CSSM: Multiple options are available for license usage reporting. You can use Cisco Smart Licensing Utility (CSLU), or report usage information directly to CSSM. For air-gapped networks, a provision for offline reporting where you download usage information and upload it to CSSM, is also available.The usage report is in plain text XML format. See: Sample Resource Utilization Measurement Report, on page 131.
- Reconcile: For situations where delta billing applies (purchased versus consumed).

Supported Products

This section provides information about the Cisco IOS-XE product instances that support Smart Licensing Using Policy. All models (Product IDs or PIDs) in a product series are supported – unless indicated otherwise.

Table 1: Supported Product Instances: Cisco Catalyst Wireless Controllers

Cisco Catalyst Wireless Controllers	When Support for Smart Licensing Using Policy was Introduced
Cisco Catalyst 9800-40 Wireless Controller	Cisco IOS XE Amsterdam 17.3.2a

Cisco Catalyst Wireless Controllers	When Support for Smart Licensing Using Policy was Introduced
Cisco Catalyst 9800-L Wireless Controller	Cisco IOS XE Amsterdam 17.3.2a
Cisco Catalyst 9800-CL Wireless Controller	Cisco IOS XE Amsterdam 17.3.2a
Cisco Catalyst 9800 embedded Wireless Controller	Cisco IOS XE Amsterdam 17.3.2a
Cisco Embedded Wireless Controller on Cisco Catalyst 9100 Access Points (EWC-AP)	Cisco IOS XE Amsterdam 17.3.2a

Architecture

This section explains the various components that can be part of your implementation of Smart Licensing Using Policy. One or more components make up a topology.

Product Instance

A product instance is a single instance of a Cisco product, identified by a Unique Device Identifier (UDI).

A product instance records and reports license usage (RUM reports), and provides alerts and system messages about overdue reports, communication failures, etc. RUM reports and usage data are securely stored in the product instance.

Throughout this document, the term *product instance* refers to all supported physical and virtual product instances - unless noted otherwise. For information about the product instances that are within the scope of this document, see Supported Products, on page 34.

CSLU

Cisco Smart License Utility (CSLU) is a Windows-based reporting utility that provides aggregate licensing workflows. This utility performs the following key functions:

- Provides options relating to how workflows are triggered. The workflows can be triggered by CSLU or by a product instance.
- Collects usage reports from one or more product instances and uploads these usage reports to the corresponding Smart Account or Virtual Account – online, or offline, using files. Similarly, the RUM report ACK is collected online, or offline, and sent back to the product instance.
- Sends authorization code requests to CSSM and receives authorization codes from CSSM, if applicable.

CSLU can be part of your implementation in the following ways:

- Install the windows application, to use CSLU as a standalone tool that is connected to CSSM.
- Install the windows application, to use CSLU as a standalone tool that is disconnected from CSSM. With this option, the required usage information is downloaded to a file and then uploaded to CSSM. This is suited to air-gapped networks.
- Embedded (by Cisco) in a controller such as Cisco DNA Center.

CSSM

Cisco Smart Software Manager (CSSM) is a portal that enables you to manage all your Cisco software licenses from a centralized location. CSSM helps you manage current requirements and review usage trends to plan for future license requirements.

You can access the CSSM Web UI at https://software.cisco.com. Under the **License** tab, click the **Smart Software Licensing** link.

See the Supported Topologies, on page 42 section to know about the different ways in which you can connect to CSSM

In CSSM you can:

- Create, manage, or view virtual accounts.
- Create and manage Product Instance Registration Tokens.
- Transfer licenses between virtual accounts or view licenses.
- Transfer, remove, or view product instances.
- Run reports against your virtual accounts.
- Modify your email notification settings.
- View overall account information.

Controller

A management application or service that manages multiple product instances.

Note Throughout this chapter, and in the context of Smart Licensing Using Policy, the term "controller"or "Controller" always means a management application or service that manages a product instance. The term is not used to refer to Cisco Catalyst Wireless Controllers, which are *product instances*.

On Cisco Catalyst Wireless Controllers, Cisco DNA Center is the supported controller. Information about the controller, product instances that support the controller, and minimum required software versions on the controller and on the product instance is provided below:

Table 2: Support Information for Controller: Cisco DNA Center

Minimum Required Cisco DNA Center Version for Smart Licensing Using Policy[1]	Minimum Required Cisco IOS XE Version[2]	Supported Product Instances
Cisco DNA Center Release 2.2.2	Cisco IOS XE Amsterdam 17.3.2a	• Cisco Catalyst 9800-40 Wireless Controller • Cisco Catalyst 9800-80 Wireless Controller • Cisco Catalyst 9800-L Wireless Controller • Cisco Catalyst 9800-CL Wireless Controller • Cisco Catalyst 9800 embedded Wireless Controller • Cisco Embedded Wireless Controller on Cisco Catalyst 9100 Access Points (EWC-AP)

1 The minimum required software version on the controller. This means support continues on all subsequent releases - unless noted otherwise

2 The minimum required software version on the product instance. This means support continues on all subsequent releases - unless noted otherwise.

For more information about Cisco DNA Center, see the support page at: https://www.cisco.com/c/en/us/support/cloud-systems-management/dna-center/series.html.

SSM On-Prem

Smart Software Manager On-Prem (SSM On-Prem) is an asset manager, which works in conjunction with CSSM. It enables you to administer products and licenses on your premises instead of having to directly connect to CSSM.

Information about the required software versions to implement Smart Licensing Using Policy with SSM On-Prem, is provided below:

Minimum Required SSM On-Prem Version for Smart Licensing Using Policy[3]	Minimum Required Cisco IOS XE Version[4]	Supported Product Instances
Version 8, Release 202102	Cisco IOS XE Amsterdam 17.3.3	• Cisco Catalyst 9800-40 Wireless Controller • Cisco Catalyst 9800-80 Wireless Controller • Cisco Catalyst 9800-L Wireless Controller • Cisco Catalyst 9800-CL Wireless Controller • Cisco Catalyst 9800 embedded Wireless Controller • Cisco Embedded Wireless Controller on Cisco Catalyst 9100 Access Points (EWC-AP)

[3] The minimum required SSM On-Prem version. This means support continues on all subsequent releases - unless noted otherwise

[4] The minimum required software version on the product instance. This means support continues on all subsequent releases - unless noted otherwise.

For more information about SSM On-Prem, see Smart Software Manager On-Prem on the Software Download page. Hover over the .iso image to display the documentation links.

Concepts

This section explains the key concepts of Smart Licensing Using Policy.

License Enforcement Types

A given license belongs to one of three enforcement types. The enforcement type indicates if the license requires authorization before use, or not.

- Unenforced or Not Enforced

  Unenforced licenses *do not* require authorization before use in air-gapped networks, or registration, in connected networks. The terms of use for such licenses are as per the General Terms and Conditions.

  All licenses available on Cisco Catalyst Wireless Controllers are unenforced licenses.

- Enforced

  Licenses that belong to this enforcement type require authorization before use. The required authorization is in the form of an authorization code, which must be installed in the corresponding product instance.

  An example of an enforced license is the Media Redundancy Protocol (MRP) Client license, which is available on Cisco's Industrial Ethernet Switches.

- Export-Controlled

  Licences that belong to this enforcement type are export-restricted by U.S. trade-control laws and these licenses require authorization before use. The required authorization code must be installed in the corresponding product instance for these licenses as well. Cisco may pre-install export-controlled licenses when ordered with hardware purchase.

  An example of an export-controlled license is the High Speed Encryption (HSECK9) license, which is available on certain Cisco Routers.

License Duration

This refers to the duration or term for which a purchased license is valid. A given license may belong to any one of the enforcement types mentioned above and be valid for the following durations:

- Perpetual: There is no expiration date for such a license.

  AIR Network Essentials and AIR Network Advantage licenses are examples of unenforced, perpetual licenses that are available on Cisco Catalyst Wireless Controllers.

- Subscription: The license is valid only until a certain date.

  AIR Digital Network Architecture (DNA) Essentials and AIR DNA Advantage licenses are examples of unenforced subscription licenses that are available on Cisco Catalyst Wireless Controllers.

Authorization Code

The Smart Licensing Authorization Code (SLAC) allows activation and continued use of a license that is export-controlled or enforced.

A SLAC is not required for any of the licenses available on Cisco Catalyst Wireless Controllers, but if you are upgrading from an earlier licensing model to Smart Licensing Using Policy, you may have a Specific License Reservation (SLR) with its own authorization code. The SLR authorization code is supported after upgrade to Smart Licensing Using Policy.

Note While existing SLRs are carried over after upgrade, you cannot request a new SLR in the Smart Licensing Using Policy environment, because the notion of "reservation" does not apply. For an air-gapped network, the No Connectivity to CSSM and No CSLU topology applies instead

For more information about how the SLR authorization code is handled, see Upgrades, on page 53. If you want to return an SLR authorization code, see Removing and Returning an Authorization Code, on page 118.

Policy

A policy provides the product instance with these reporting instructions:

- License usage report acknowledgement requirement (Reporting ACK required): The license usage report is known as a RUM Report and the acknowledgement is referred to as an ACK (See RUM Report and Report Acknowledgement). This is a yes or no value which specifies if the report for this product instance requires CSSM acknowledgement or not. The default policy is always set to "yes".

- First report requirement (days): The first report must be sent within the duration specified here.

  If the value here is zero, no first report is required.

- Reporting frequency (days): The subsequent report must be sent within the duration specified here.

  If the value here is zero, it means no further reporting is required *unless* there is a usage change.

- Report on change (days): In case of a change in license usage, a report must be sent within the duration specified here.

  If the value here is zero, no report is required on usage change.

  If the value here is not zero, reporting *is* required after the change is made. All the scenarios listed below count as changes in license usage on the product instance:

  - Changing licenses consumed (includes changing to a different license, and, adding or removing a license).
  - Going from consuming zero licenses to consuming one or more licenses.
  - Going from consuming one or more licenses to consuming zero licenses.

Note If a product instance has *never* consumed a license, reporting is not required even if the policy has a non-zero value for any of the reporting requirements (First report requirement, Reporting frequency, Report on change).

Understanding Policy Selection

CSSM determines the policy that is applied to a product instance. Only one policy is in use at a given point in time. The policy and its values are based on a number of factors, including the licenses being used.

`Cisco default` is the default policy that is always available in the product instance. If no other policy is applied, the product instance applies this default policy. The table below (Table 3: Policy: Cisco default, on page 40) shows the `Cisco default` policy values.

While you cannot configure a policy, you can request for a customized one, by contacting the Cisco Global Licensing Operations team. Go to Support Case Manager. Click **OPEN NEW CASE** > Select **Software Licensing**. The licensing team will contact you to start the process or for any additional information. Customized policies are also made available through your Smart account in CSSM.

Note To know which policy is applied (the policy in-use) and its reporting requirements, enter the **show license all** command in privileged EXEC mode.

Table 3: Policy: Cisco default

Policy: `Cisco default`	**Default Policy Values**
Export (Perpetual/Subscription) **Note** Applied only to licenses with enforcement type "Export-Controlled".	Reporting ACK required: Yes First report requirement (days): 0 Reporting frequency (days): 0 Report on change (days): 0

Policy: `Cisco default`	**Default Policy Values**
Enforced (Perpetual/Subscription) **Note** Applied only to licenses with enforcement type "Enforced".	Reporting ACK required: Yes First report requirement (days): 0 Reporting frequency (days): 0 Report on change (days): 0
Unenforced/Non-Export Perpetual[5]	Reporting ACK required: Yes First report requirement (days): 365 Reporting frequency (days): 0 Report on change (days): 90
Unenforced/Non-Export Subscription	Reporting ACK required: Yes First report requirement (days): 90 Reporting frequency (days): 90 Report on change (days): 90

[5] For Unenforced/Non-Export Perpetual: the default policy's first report requirement (within 365 days) applies only if you have purchased hardware or software from a distributor or partner.

RUM Report and Report Acknowledgement

A Resource Utilization Measurement report (RUM report) is a license usage report, which fulfils reporting requirements as specified by the policy. RUM reports are generated by the product instance and consumed by CSSM. The product instance records license usage information and all license usage changes in an open RUM report. At system-determined intervals, open RUM reports are closed and new RUM reports are opened to continue recording license usage. A closed RUM report is ready to be sent to CSSM.

A RUM acknowledgement (RUM ACK or ACK) is a response from CSSM and provides information about the status of a RUM report.

The reporting method, that is, how a RUM report is sent to CSSM, depends on the topology you implement.

CSSM displays license usage information as per the last received RUM report.

A RUM report may be accompanied by other requests, such as a trust code request, or a SLAC request. So in addition to the RUM report IDs that have been received, an ACK from CSSM may include authorization codes, trust codes, and policy files.

The policy that is applied to a product instance determines the following aspects of the reporting requirement:

- Whether a RUM report is sent to CSSM and the maximum number of days provided to meet this requirement.
- Whether the RUM report requires an acknowledgement (ACK) from CSSM.
- The maximum number of days provided to report a change in license consumption.

Trust Code

A *UDI-tied public key*, which the product instance uses to

- Sign a RUM report. This prevents tampering and ensures data authenticity.
- Enable secure communication with CSSM.

If a trust code is installed on the product instance, the output of the **show license status** command displays a timestamp in the `Trust Code Installed:` field.

Supported Topologies

This section describes the various ways in which you can implement Smart Licensing Using Policy. For each topology, refer to the accompanying overview to know the how the set-up is designed to work, and refer to the considerations and recommendations, if any.

After Topology Selection

After you have selected a topology, see How to Configure Smart Licensing Using Policy: Workflows by Topology , on page 57. These workflows are only for new deployments. They provide the simplest and fastest way to implement a topology.

If you are migrating from an existing licensing model, see Migrating to Smart Licensing Using Policy, on page 70.

After initial implementation, for any additional configuration tasks you have to perform, for instance, changing the AIR license, or synchronizing RUM reports, see the *Task Library for Smart Licensing Using Policy*.

Note Always check the "Supported topologies" where provided, before you proceed.

Connected to CSSM Through CSLU

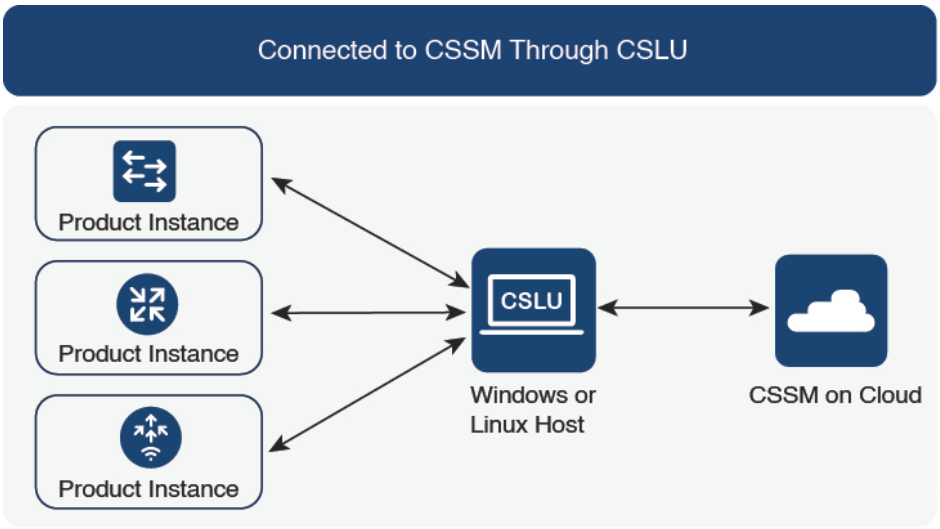
Overview:

Here, product instances in the network are connected to CSLU, and CSLU becomes the single point of interface with CSSM. A product instance can be configured to *push* the required information to CSLU. Alternatively, CSLU can be set-up to *pull* the required information from a product instance at a configurable frequency.

Product instance-initiated communication (push): A product instance initiates communication with CSLU, by connecting to a REST endpoint in CSLU. Data that is sent includes RUM reports and requests for authorization codes, UDI-tied trust codes, and policies. You can configure the product instance to automatically send RUM reports to CSLU at required intervals. This is the default method for a product instance.

CSLU-initiated communication (pull): To initiate the retrieval of information from a product instance, CSLU uses NETCONF, or RESTCONF, or gRPC with YANG models, or native REST APIs, to connect to the product instance. Supported workflows include retrieving RUM reports from the product instance and sending the same to CSSM, authorization code installation, UDI-tied trust code installation, and application of policies.

Figure 2: Topology: Connected to CSSM Through CSLU



Considerations or Recommendations:

Choose the method of communication depending on your network's security policy.

Release-Wise Changes and Enhancements:

This section outlines important release-wise software changes and enhancements that affect this topology.

- RUM report throttling

  In the product instance-initiated mode, the minimum reporting frequency is throttled to one day. This means the product instance does not send more than one RUM report a day. This resolves the problem of too many RUM reports being generated and sent for certain licenses. It also resolves the memory-related issues and system slow-down caused by an excessive generation of RUM reports.

  You can override the throttling restriction by entering the **license smart sync** command in privileged EXEC mode.

  RUM report throttling applies to the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train.

Where to Go Next:

To implement this topology, see Workflow for Topology: Connected to CSSM Through CSLU, on page 57.

Connected Directly to CSSM

Overview:

This topology is available in the earlier version of Smart Licensing and continues to be supported with Smart Licensing Using Policy.

Here, you establish a *direct* and *trusted* connection from a product instance to CSSM. The direct connection, requires network reachability to CSSM. For the product instance to then exchange messages and communicate with CSSM, configure one of the transport options available with this topology (described below). Lastly, the

establishment of trust requires the generation of a token from the corresponding Smart Account and Virtual Account in CSSM, and installation on the product instance.

You can configure a product instance to communicate with CSSM in the following ways:

- Use Smart transport to communicate with CSSM

  Smart transport is a transport method where a Smart Licensing (JSON) message is contained within an HTTPs message, and exchanged between a product instance and CSSM, to communicate. The following Smart transport configuration options are available:

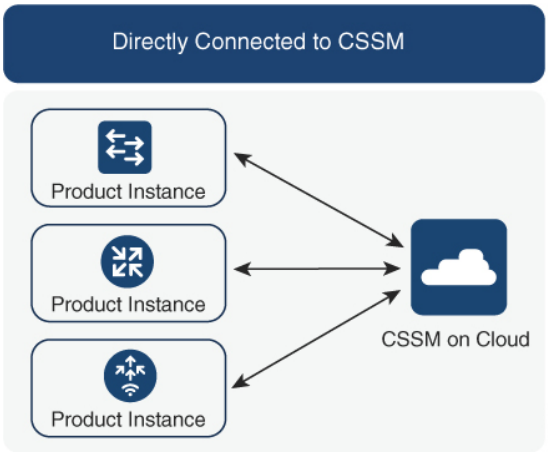
  - Smart transport: In this method, a product instance uses a specific Smart transport licensing server URL. This must be configured exactly as shown in the workflow section.
  - Smart transport through an HTTPs proxy: In this method, a product instance uses a proxy server to communicate with the licensing server, and eventually, CSSM.

- Use Call Home to communicate with CSSM.

  Call Home provides e-mail-based and web-based notification of critical system events. This method of connecting to CSSM is available in the earlier Smart Licensing environment, and continues to be available with Smart Licensing Using Policy. The following Call Home configuration options are available:

  - Direct cloud access: In this method, a product instance sends usage information directly over the internet to CSSM; no additional components are needed for the connection.
  - Direct cloud access through an HTTPs proxy: In this method, a product instance sends usage information over the internet through a proxy server - either a Call Home Transport Gateway or an off-the-shelf proxy (such as Apache) to CSSM.

Figure 3: Topology: Connected Directly to CSSM



Considerations or Recommendations:

Smart transport is the recommended transport method when directly connecting to CSSM. This recommendation applies to:

- New deployments.
- Earlier licensing models. Change configuration after migration to Smart Licensing Using Policy.

- Registered licenses that currently use the Call Home transport method. Change configuration after migration to Smart Licensing Using Policy.
- Evaluation or expired licenses in an earlier licensing model. Change configuration after migration to Smart Licensing Using Policy.

To change configuration after migration, see Workflow for Topology: Connected Directly to CSSM, on page 59 > Product Instance Configuration > Configure a connection method and transport type > Option 1.

Release-Wise Changes and Enhancements:

This section outlines important release-wise software changes and enhancements that affect this topology.

- RUM report throttling

  The minimum reporting frequency for this topology, is throttled to one day. This means the product instance does not send more than one RUM report a day. This resolves the problem of too many RUM reports being generated and sent for certain licenses. It also resolves the memory-related issues and system slow-down caused by an excessive generation of RUM reports.

  You can override the throttling restriction by entering the **license smart sync** command in privileged EXEC mode.

  RUM report throttling applies to the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train.

Where to Go Next:

To implement this topology, see Workflow for Topology: Connected Directly to CSSM, on page 59.

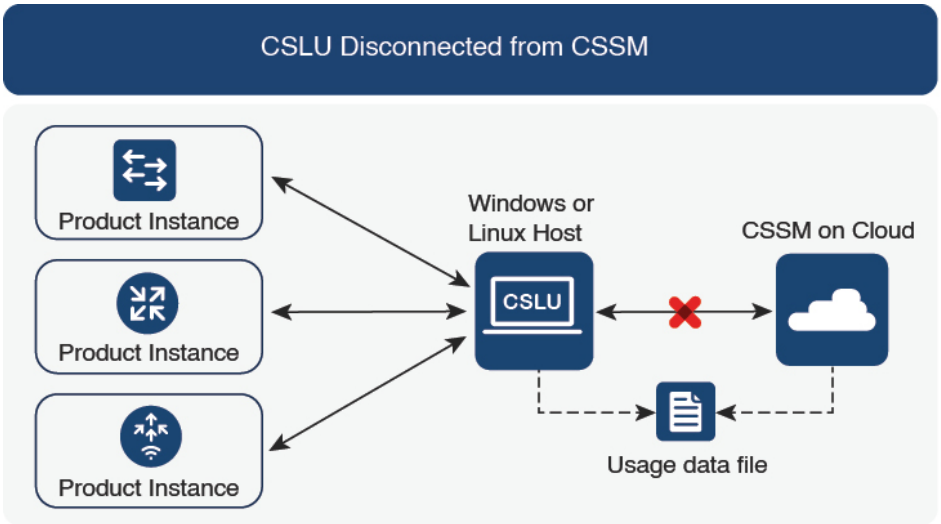
CSLU Disconnected from CSSM

Overview:

Here, a product instance communicates with CSLU, and you have the option of implementing product instance-initiated communication or CSLU-initiated communication (as in the *Connected to CSSM Through CSLU* topology). The other side of the communication, between CSLU and CSSM, is offline. CSLU provides you with the option of working in a mode that is disconnected from CSSM.

Communication between CSLU and CSSM is sent and received in the form of signed files that are saved offline and then uploaded to or downloaded from CSLU or CSSM, as the case may be.

Figure 4: Topology: CSLU Disconnected from CSSM



Considerations or Recommendations:

Choose the method of communication depending on your network's security policy.

Release-Wise Changes and Enhancements:

This section outlines important release-wise software changes and enhancements that affect this topology.

- RUM report throttling

  In the product instance-initiated mode, the minimum reporting frequency is throttled to one day. This means the product instance does not send more than one RUM report a day. This resolves the problem of too many RUM reports being generated and sent for certain licenses. It also resolves the memory-related issues and system slow-down caused by an excessive generation of RUM reports.

  You can override the throttling restriction by entering the **license smart sync** command in privileged EXEC mode.

  RUM report throttling applies to the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train.

Where to Go Next:

To implement this topology, see Workflow for Topology: CSLU Disconnected from CSSM, on page 61.

Connected to CSSM Through a Controller

When you use a controller to manage a product instance, the controller connects to CSSM, and is the interface for all communication to and from CSSM. The supported controller for Cisco Catalyst Wireless Controllers is Cisco DNA Center.

Overview:

If a product instance is managed by Cisco DNA Center as the controller, the product instance records license usage and saves the same, but it is the Cisco DNA Center that initiates communication with the product instance to retrieve RUM reports, report to CSSM, and return the ACK for installation on the product instance.

All product instances that must be managed by Cisco DNA Center must be part of its inventory and must be assigned to a site. Cisco DNA Center uses the NETCONF protocol to provision configuration and retrieve the required information from the product instance - the product instance must therefore have NETCONF enabled, to facilitate this.

In order to meet reporting requirements, Cisco DNA Center retrieves the applicable policy from CSSM and provides the following reporting options:

- Ad hoc reporting: You can trigger an ad hoc report when required.
- Scheduled reporting: Corresponds with the reporting frequency specified in the policy and is automatically handled by Cisco DNA Center.

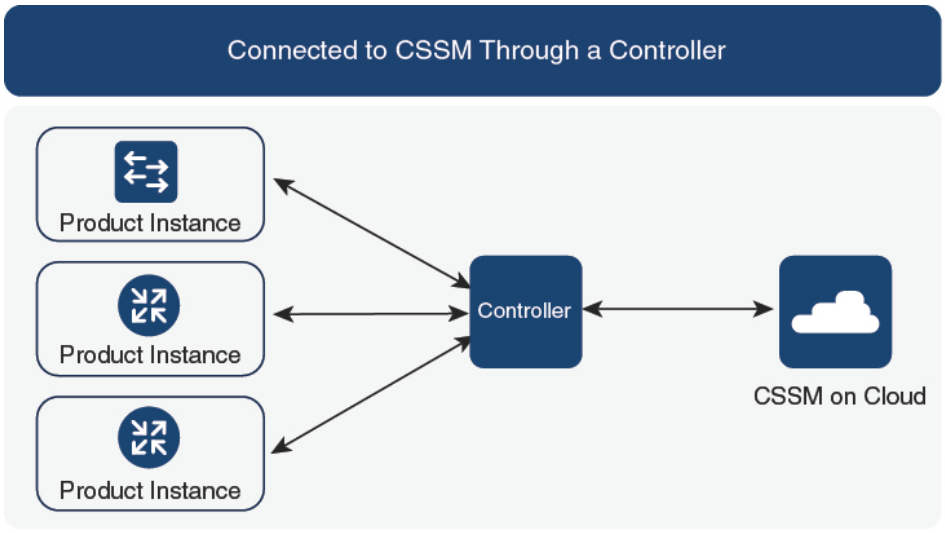
Note Ad hoc reporting must be performed at least once before a product instance is eligible for scheduled reporting.

The first ad hoc report enables Cisco DNA Center to determine the Smart Account and Virtual Account to which subsequent RUM reports must be uploaded. You will receive notifications if ad hoc reporting for a product instance has not been performed even once.

Cisco DNA Center also enables you to install and remove SLAC for export-controlled licenses. Since all available licenses on Cisco Catalyst Wireless Controllers are unenforced licenses, SLAC installation and removal do not apply.

A trust code is *not* required.

Figure 5: Topology: Connected to CSSM Through a Controller



Considerations or Recommendations:

This is the recommended topology if you are using Cisco DNA Center.

Where to Go Next:

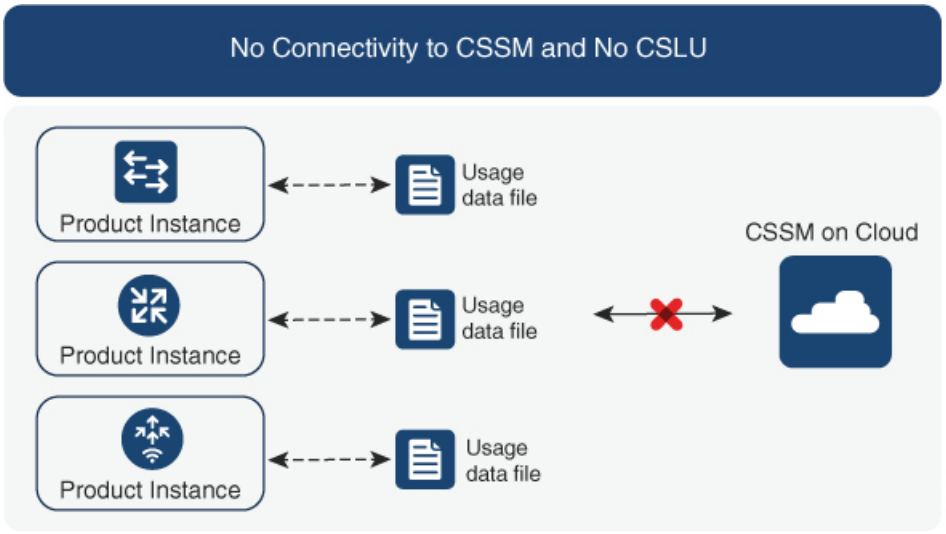
To implement this topology, see Workflow for Topology: Connected to CSSM Through a Controller, on page 63.

No Connectivity to CSSM and No CSLU

Overview:

Here you have a product instance and CSSM disconnected from each other, and without any other intermediary utilities or components. All communication is in the form of uploaded and downloaded files. These files can be RUM reports .

Figure 6: Topology: No Connectivity to CSSM and No CSLU



Considerations or Recommendations:

This topology is suited to a high-security deployment where a product instance cannot communicate online, with anything outside its network.

Where to Go Next:

To implement this topology, see Workflow for Topology: No Connectivity to CSSM and No CSLU, on page 64.

SSM On-Prem Deployment

Overview:

SSM On-Prem is designed to work as an extension of CSSM that is deployed on your premises.

Here, a product instance is connected to SSM On-Prem, and SSM On-Prem becomes the single point of interface with CSSM. Each instance of SSM On-Prem must be made known to CSSM through a mandatory registration and synchronization of the local account in SSM On-Prem, with a Virtual Account in CSSM.

When you deploy SSM On-Prem to manage a product instance, the product instance can be configured to *push* the required information to SSM On-Prem. Alternatively, SSM On-Prem can be set-up to *pull* the required information from a product instance at a configurable frequency.

- Product instance-initiated communication (push): The product instance initiates communication with SSM On-Prem, by connecting to a REST endpoint in SSM On-Prem. Data that is sent includes RUM reports and requests for authorization codes, trust codes, and policies.

  Options for communication between the product instance and SSM On-Prem in this mode:

  - Use a CLI command to push information to SSM On-Prem as and when required.
  - Use a CLI command and configure a reporting interval, to automatically send RUM reports to SSM On-Prem at a scheduled frequency.

- SSM On-Prem-initiated communication (pull): To initiate the retrieval of information from a product instance, SSM On-Prem NETCONF, RESTCONF, and native REST API options, to connect to the product instance. Supported workflows include receiving RUM reports from the product instance and sending the same to CSSM, authorization code installation, trust code installation, and application of policies.

  Options for communication between the product instance and SSM On-Prem in this mode:

  - Collect usage information from one or more product instances as and when required (on-demand).
  - Collect usage information from one or more product instances at a scheduled frequency.

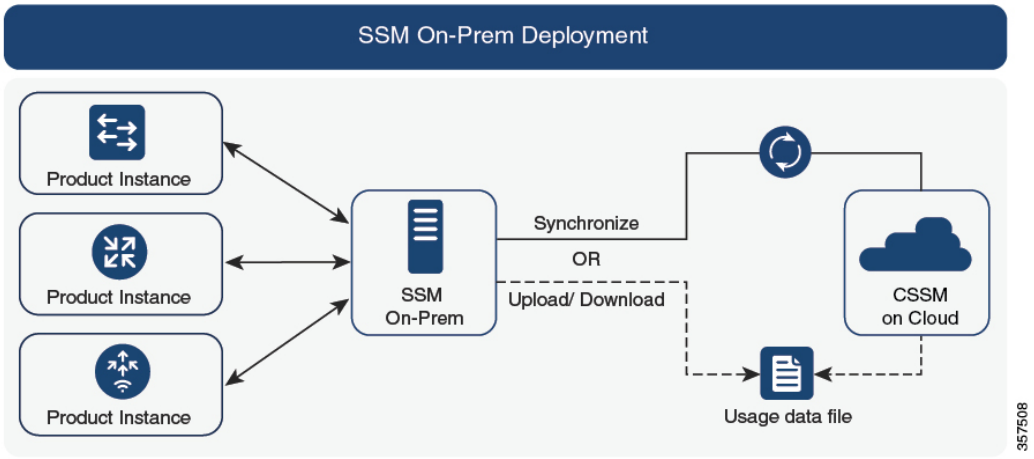
In SSM On-Prem, the reporting interval is set to the default policy on the product instance. You can change this, but only to report more frequently (a narrower interval), or you can install a custom policy if available.

After usage information is available in SSM On-Prem, you must synchronize the same with CSSM, to ensure that the product instance count, license count and license usage information is the same on both, CSSM and SSM On-Prem. Options for usage synchronization between SSM On-Prem and CSSM – for the push *and* pull mode:

- Perform ad-hoc synchronization with CSSM (Synchronize now with Cisco).
- Schedule synchronization with CSSM for specified times.
- Communicate with CSSM through signed files that are saved offline and then upload to or download from SSM On-Prem or CSSM, as the case may be.

Note This topology involves two different kinds of synchronization between SSM On-Prem and CSSM. The first is where the *local account* is synchronized with CSSM - this is for the SSM On-Prem instance to be known to CSSM and is performed by using the **Synchronization** widget in SSM On-Prem. The second is where *license usage* is synchronized with CSSM, either by being connected to CSSM or by downloading and uploading files. You must synchronize the local account before you can synchronize license usage.

Figure 7: Topology: SSM On-Prem Deployment



Considerations or Recommendations:

This topology is suited to the following situations:

- If you want to manage your product instances on your premises, as opposed communicating directly with CSSM for this purpose.
- If your company's policies prevent your product instances from reporting license usage directly to Cisco (CSSM).
- If your product instances are in an air-gapped network and cannot communicate online, with anything outside their network.

Apart from support for Smart Licensing Using Policy, some of the key benefits of SSM On-Prem *Version 8* include:

- Multi-tenancy: One tenant constitutes one Smart Account-Virtual Account pair. SSM On-Prem enables you to manage multiple pairs. Here you create local accounts that reside in SSM On-Prem. Multiple local accounts roll-up to a Smart Account-Virtual Account pair in CSSM. For more information, see the Cisco Smart Software Manager On-Prem User Guide > *About Accounts and Local Virtual Accounts*.

Note The relationship between CSSM and SSM On-Prem instances is still one-to-one.

- Scale: Supports up to a total of 300,000 product instances
- High-Availability: Enables you to run two SSM On-Prem servers in the form of an active-standby cluster. For more information, see the Cisco Smart Software On-Prem Installation Guide > Appendix 4. Managing a High Availability (HA) Cluster in Your System.

  High-Availability deployment is supported on the SSM On-Prem console and the required command details are available in the Cisco Smart Software On-Prem Console Guide.
- Options for online and offline connectivity to CSSM.

SSM On-Prem Limitations:

- Proxy support for communication with CSSM, for the purpose of *license usage* synchronization is available only from Version 8 202108 onwards. The use of a proxy for *local account* synchronization, which is performed by using the **Synchronization** widget, is available from the introductory SSM On-Prem release where Smart Licensing Using Policy is supported.
- SSM On-Prem-initiated communication is not supported on a product instance that is in a Network Address Translation (NAT) set-up. You must use product instance-initiated communication, and further, you must *enable* SSM On-Prem to support a product instance that is in a NAT setup. Details are provided in the workflow for this topology.

Release-Wise Changes and Enhancements:

This section outlines important release-wise software changes and enhancements that affect this topology.

From Cisco IOS XE Cupertino 17.9.1:

- RUM report throttling

  In the product instance-initiated mode, the minimum reporting frequency is throttled to one day. This means the product instance does not send more than one RUM report a day. This resolves the problem of too many RUM reports being generated and sent for certain licenses. It also resolves the memory-related issues and system slow-down caused by an excessive generation of RUM reports.

  You can override the throttling restriction by entering the **license smart sync** command in privileged EXEC mode.

  RUM report throttling applies to the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train.

Where to Go Next:

To implement this topology, see Workflow for Topology: SSM On-Prem Deployment, on page 65

If you are migrating from an existing version of SSM On-Prem, the sequence in which you perform the various upgrade-related activities is crucial. See Migrating to a Version of SSM On-Prem That Supports Smart Licensing Using Policy, on page 89

Interactions with Other Features

High Availability

This section explains considerations that apply to a High Availability configuration, when running a software version that supports Smart Licensing Using Policy. The following High Availability set-ups are within the scope of this document:

A dual-chassis set-up (could be fixed or modular), with the active in one chassis and a standby in the other chassis.

A wireless N+1 topology, where “n” number of wireless controllers act as primary and a “+1” wireless controller acts as the secondary or fallback wireless controller for Access Points (APs). Each Access Point is configured with a primary and a secondary wireless controller. In case of a failure on the primary, all access points that were connected to the primary now fallback to the secondary wireless controller.

Trust Code Requirements in a High Availability Set-Up

The number of trust codes required depends on the number of UDIs. The active product instance can submit requests for all devices in the High Availability set-up and install all the trust codes that are returned in an ACK.

Policy Requirements in a High Availability Set-Up

There are no policy requirements that apply exclusively to a High Availability set-up. As in the case of a standalone product instance, only one policy exists in a High Availability set-up as well, and this is on the active. The policy on the active applies to any standbys in the set-up.

Product Instance *Functions* in a High Availability Set-Up

This section explains general product instance functions in a High Availability set-up, as well as what the product instance does when a new standby or secondary is added to an existing High Available set-up.

For authorization and trust codes: The active product instance can request (if required) and install authorization codes and trust codes for standbys.

For policies: The active product instance synchronizes with the standby.

For reporting: Only the active product instance reports usage. The active reports usage information for all devices in the High Availability set-up. In addition to scheduled reporting, the following events trigger reporting:

- The addition or removal of a standby. The RUM report includes information about the standby that was added or removed.
- A switchover.
- A reload.

When one of the above events occur, the "Next report push" date of the **show license status** privileged EXEC command is updated. But it is the implemented topology and associated reporting method that determine if the report is sent by the product instance or not. For example, if you have implemented a topology where the product instance is disconnected (Transport Type is Off), then the product instance does not send RUM reports even if the "Next report push" date is updated.

For addition or removal of a new standby:

- A product instance that is connected to CSLU, does not take any further action.
- A product instance that is directly connected to CSSM, performs trust synchronization. Trust synchronization involves the following:

  Installation of trust code on the standby if not installed already.

  If a trust code is already installed, the trust synchronization process ensures that the new standby is in the same Smart Account and Virtual Account as the active. If it is not, the new standby is *moved* to the same Smart Account and Virtual Account as the active.

  Installation of an authorization code, policy, and purchase information, if applicable

  Sending of a RUM report with current usage information.

For addition or removal of a secondary:

There are no product instance functions that apply exclusively to the addition or removal of a secondary product instance. Further, all the secondary product instances are in the same Smart Account and Virtual Account as the primary product instance.

Upgrades

This section describes how upgrade or migration to Smart Licensing Using Policy is handled. It clarifies how Smart Licensing Using Policy handles all earlier licensing models including: the earlier version of Smart Licensing, Specific License Reservation (SLR), and how evaluation or expired licenses from any of the earlier licensing models.

To migrate to Smart Licensing Using Policy, you must upgrade to a software version that supports Smart Licensing Using Policy. After you upgrade, Smart Licensing Using Policy is the only supported licensing model and the product instance continues to operate *without any licensing changes*. The Migrating to Smart Licensing Using Policy, on page 70 section provides details and examples for migration scenarios that apply to Cisco Catalyst Wireless Controllers.

Device-led conversion is not supported for migration to Smart Licensing Using Policy.

Identifying the Current Licensing Model Before Upgrade

Before you upgrade to Smart Licensing Using Policy, if you want to know the current licensing model that is effective on the product instance, enter the **show license all** command in privileged EXEC mode.

How Upgrade Affects Enforcement Types for Existing Licenses

When you upgrade to a software version which supports Smart Licensing Using Policy, the way existing licenses are handled, depends primarily on the license enforcement type.

- An unenforced license that was being used before upgrade, continues to be available after the upgrade. All licenses on Cisco Catalyst Wireless Controllers are unenforced licenses. This includes licenses from all earlier licensing models:
  - Smart Licensing
  - Specific License Reservation (SLR), which has an accompanying authorization code. The authorization code continues to be valid after upgrade to Smart Licensing Using Policy and authorizes existing license consumption.
  - Evaluation or expired licenses from any of the above mentioned licensing models.
- An enforced or export-controlled license that was being used before upgrade, continues to be available after upgrade if the required authorization exists.

  There are no export-controlled or enforced licenses on any of the supported Cisco Catalyst Wireless Controllers, therefore, these enforcement types and the requisite SLAC do not apply.

How Upgrade Affects Reporting for Existing Licenses

Existing License	Reporting Requirements After Migration to Smart Licensing Using Policy
Specific License Reservation (SLR)	Required only if there is a change in license consumption. An existing SLR authorization code authorizes existing license consumption after upgrade to Smart Licensing Using Policy.

Existing License	Reporting Requirements After Migration to Smart Licensing Using Policy
Smart Licensing (Registered and Authorized license)	Depends on the policy.
Evaluation or expired licenses	Based on the reporting requirements of the Cisco default policy.

How Upgrade Affects Transport Type for Existing Licenses

The transport type, if configured in your existing set-up, is retained after upgrade to Smart Licensing Using Policy.

When compared to the earlier version of Smart Licensing, additional transport types are available with Smart Licensing Using Policy. There is also a change in the default transport mode. The following table clarifies how this may affect upgrades:

Transport type Before Upgrade	License or License State Before Upgrade	Transport Type After Upgrade
Default (callhome)	evaluation	cslu (default in Smart Licensing Using Policy)
	SLR	off
	registered	callhome
smart	evaluation	off
	SLR	off
	registered	smart

How Upgrade Affects the Token Registration Process

In the earlier version of Smart Licensing, a token was used to register and connect to CSSM. ID token registration is not required in Smart Licensing Using Policy. The token generation feature is still available in CSSM, and is used to *establish trust* when a product instance is directly connected to CSSM. See Connected Directly to CSSM.

Downgrades

To downgrade, you must downgrade the software version on the product instance. This section provides information about downgrades for new deployments and existing deployments (you upgraded to Smart Licensing Using Policy and now want to downgrade).

New Deployment Downgrade

This section describes considerations and actions that apply if a newly purchased product instance with a software version where Smart Licensing Using Policy is enabled by default, is downgraded to a software version where Smart Licensing Using Policy is not supported.

The outcome of the downgrade depends on whether a trust code was installed while still operating in the Smart Licensing Using Policy environment, and further action may be required depending on the release you downgrade to.

If the topology you implemented while in the Smart Licensing Using Policy environment was "Connected Directly to CSSM", then a trust code installation can be expected or assumed, because it is required as part of topology implementation. For any of the other topologies, trust establishment is not mandatory. Downgrading product instances with one of these other topologies will therefore mean that you have to restore licenses to a registered and authorized state by following the procedures that are applicable in the Smart Licensing environment. See the table (*Outcome and Action for New Deployment Downgrade to Smart Licensing*) below.

Table 4: Outcome and Action for New Deployment Downgrade to Smart Licensing

In the Smart Licensing Using Policy Environment	Downgrade to..	Outcome and Further Action
Standalone product instance, connected directly to CSSM, and trust established.	Cisco IOS XE Amsterdam 17.3.1 OR Cisco IOS XE Gibraltar 16.12.4 and later releases in Cisco IOS XE Gibraltar 16.12.x	No further action is required. The product instance attempts to renew trust with CSSM after downgrade. After a successful renewal, licenses are in a registered state and the earlier version of Smart Licensing is effective on the product instance.
	Any other release (other than the ones mentioned in the row above) that supports Smart Licensing	Action is required: You must reregister the product instance. Generate an ID token in the CSSM Web UI and on the product instance, configure the **license smart register idtoken** *idtoken* command in global configuration mode.
High Availability set-up, connected directly to CSSM, and trust established.	Any release that supports Smart Licensing	Action is required: You must reregister the product instance. Generate an ID token in the CSSM Web UI and on the product instance, configure the **license smart register idtoken** *idtoken* **all** command in global configuration mode.
Any other topology. (Connected to CSSM Through CSLU, CSLU Disconnected from CSSM, No Connectivity to CSSM and No CSLU)	Any release that supports Smart Licensing	Action is required. Restore licenses to a registered and authorized state by following the procedures that are applicable in the Smart Licensing environment.

Upgrade and Then Downgrade

This section describes considerations and actions that apply if a product instance is upgraded to a software version that supports Smart Licensing Using Policy and then downgraded to an earlier licensing model.

When you downgrade such a product instance, *license consumption does not change* and any product features you have configured on the product instance are preserved – only the features and functions that are available with Smart Licensing Using Policy are not available anymore. Refer to the corresponding section below to know more about reverting to an earlier licensing model.

Upgrade to Smart Licensing Using Policy and then Downgrade to Smart Licensing

The outcome of the downgrade depends on whether a trust code was installed while you were still operating in the Smart Licensing Using Policy environment, and further action may be required depending on the release you downgrade to. See the table below.

Table 5: Outcome and Action for Upgrade to Smart Licensing Using Policy and then Downgrade to Smart Licensing

In the Smart Licensing Using Policy Environment	Downgrade to..	Outcome and Further Action
Standalone product instance, connected directly to CSSM, and trust established.	Cisco IOS XE Amsterdam 17.3.1 OR Cisco IOS XE Gibraltar 16.12.4 and later releases in Cisco IOS XE Gibraltar 16.12.x	No further action is required. The system recognizes the trust code and converts it back to a registered ID token, and this reverts the license to an AUTHORIZED and REGISTERED state.
	Any other release (other than the ones mentioned in the row above) that supports Smart Licensing	Action is required: You must reregister the product instance. Generate an ID token in the CSSM Web UI and on the product instance, configure the **license smart register idtoken***idtoken* command in global configuration mode.
High Availability set-up, connected directly to CSSM, and trust established.	Any release that supports Smart Licensing	Action is required: You must reregister the product instance. Generate an ID token in the CSSM Web UI and on the product instance, configure the **license smart register idtoken** *idtoken* **all** command in global configuration mode.
Any other topology (Connected to CSSM Through CSLU, CSLU Disconnected from CSSM, No Connectivity to CSSM and No CSLU)	Any release that supports Smart Licensing.	Action is required. Restore licenses to a registered and authorized state by following the procedures that are applicable in the Smart Licensing environment.

Note Licenses that were in an evaluation or expired state in the Smart Licensing environment, revert to that same state after downgrade.

Upgrade to Smart Licensing Using Policy and then Downgrade to SLR

To revert to SLR, all that is required is for the image to be downgraded. The license remains reserved and authorized – no further action is required.

However, if you have returned an SLR while in the Smart Licensing Using Policy environment, then you must repeat the process of procuring an SLR as required, in the supported release.

How to Configure Smart Licensing Using Policy: Workflows by Topology

This section provides the simplest and fastest way to implement a topology.

Note These workflows are meant for new deployments only. If you are migrating from an existing licensing model, see Migrating to Smart Licensing Using Policy, on page 70.

Workflow for Topology: Connected to CSSM Through CSLU

Depending on whether you want to implement a product instance-initiated or CSLU-initiated method of communication, complete the corresponding sequence of tasks:

- Tasks for Product Instance-Initiated Communication
- Tasks for CSLU-Initiated Communication

Tasks for Product Instance-Initiated Communication

CSLU Installation → CSLU Preference Settings → Product Instance Configuration

1. ***CSLU Installation***

   Where task is performed: A Windows host (laptop, destop, or a Virtual Machine (VM)

   Download the file from Smart Software Manager > **Smart Licensing Utility**.

   Refer to Cisco Smart License Utility Quick Start Setup Guide for help with installation and set-up.

2. ***CSLU Preference Settings***

   Where tasks are performed: CSLU

   a. Logging into Cisco (CSLU Interface), on page 91

   b. Configuring a Smart Account and a Virtual Account (CSLU Interface), on page 91

   c. Adding a Product-Initiated Product Instance in CSLU (CSLU Interface), on page 92

3. ***Product Instance Configuration***

   Where tasks are performed: Product Instance

   a. Ensuring Network Reachability for Product Instance-Initiated Communication, on page 92

   b. Ensure that transport type is set to **cslu**.

      CSLU is the default transport type. If you have configured a different option, enter the **license smart transport cslu** command in global configuration mode. Save any changes to the configuration file.

      ```
      Device(config)# license smart transport cslu
      Device(config)# exit
      Device# copy running-config startup-config
      ```

   c. Specify how you want CSLU to be discovered (*choose one*):

      - Option 1:

        No action required. Name server configured for Zero-touch DNS discovery of `cslu-local`

        Here, if you have configured DNS (the name server IP address is configured on the product instance), and the DNS server has an entry where hostname `cslu-local` is mapped to the CSLU IP address, then no further action is required. The product instance automatically discovers hostname `cslu-local`.

      - Option 2:

        No action required. Name server and domain configured for Zero-touch DNS discovery of `cslu-local.<domain>`

        Here if you have configured DNS (the name server IP address and domain is configured on the product instance), and the DNS server has an entry where `cslu-local.<domain>` is mapped to the CSLU IP address, then no further action is required. The product instance automatically discovers hostname `cslu-local`.

      - Option 3:

        Configure a specific URL for CSLU.

        Enter the **license smart url cslu** *http://<cslu_ip_or_host>:8182/cslu/v1/pi* command in global configuration mode. For `<cslu_ip_or_host>`, enter the hostname or the IP address of the windows host where you have installed CSLU. 8182 is the port number and it is the only port number that CSLU uses.

        ```
        Device(config)# license smart url cslu http://192.168.0.1:8182/cslu/v1/pi
        Device(config)# exit
        Device# copy running-config startup-config
        ```

Result:

Since the product instance initiates communication, it automatically sends out the first RUM report at the scheduled time, as per the policy. CSLU forwards the RUM report to CSSM and retrieves the ACK, which also contains the trust code. The ACK is applied to the product instance the next time the product instance contacts CSLU.

In the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train: The product instance does not send more than one RUM report a day. You can override this for an on-demand synchronization between the product instance and CSSM, by entering the **license smart sync** command in privileged EXEC mode.

To know when the product instance will be sending the next RUM report, enter the **show license all** command in privileged EXEC mode and in the output, check the date in the `Next report push` field.

In case of a change in license usage, see Configuring an AIR License, on page 128 to know how it affects reporting.

Tasks for CSLU-Initiated Communication

CSLU Installation → CSLU Preference Settings→ Product Instance Configuration → Usage Synchronization

1. ***CSLU Installation***

   Where task is performed: A Windows host (laptop, destop, or a Virtual Machine (VM)

   Download the file from Smart Software Manager > **Smart Licensing Utility**.

   Refer to Cisco Smart License Utility Quick Start Setup Guide for help with installation and set-up.

2. ***CSLU Preference Settings***

   Where tasks is performed: CSLU

   a. Logging into Cisco (CSLU Interface), on page 91

   b. Configuring a Smart Account and a Virtual Account (CSLU Interface), on page 91

   c. Adding a CSLU-Initiated Product Instance in CSLU (CSLU Interface), on page 94

3. ***Product Instance Configuration***

   Where tasks is performed: Product Instance

   Ensuring Network Reachability for CSLU-Initiated Communication, on page 96

4. ***Usage Synchronization***

   Where tasks is performed: Product Instance

   Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94

Result:

Since CSLU is logged into CSSM, the reports are automatically sent to the associated Smart Account and Virtual Account in CSSM and CSSM will send an ACK to CSLU as well as to the product instance. It gets the ACK from CSSM and sends this back to the product instance for installation. The ACK from CSSM contains the trust code and SLAC if this was requested.

In case of a change in license usage, see Configuring an AIR License, on page 128 to know how it affects reporting.

Workflow for Topology: Connected Directly to CSSM

Smart Account Set-Up → Product Instance Configuration → Trust Establishment with CSSM

1. ***Smart Account Set-Up***

   Where task is performed: CSSM Web UI, https://software.cisco.com/

Ensure that you have a user role with proper access rights to a Smart Account and the required Virtual Accounts.

2. ***Product Instance Configuration***

   Where tasks are performed: Product Instance

   a. Set-Up product instance connection to CSSM: Setting Up a Connection to CSSM , on page 111

   b. Configure a connection method and transport type (choose one)

   - Option 1:

     Smart transport: Set transport type to **smart** and configure the corresponding URL.

     If the transport mode is set to **license smart transport smart**, and you configure **license smart url default**, the Smart URL (https://smartreceiver.cisco.com/licservice/license) is automatically configured. Save any changes to the configuration file.

     ```
     Device(config)# license smart transport smart
     Device(config)# license smart url default
     Device(config)# exit
     Device# copy running-config startup-config
     ```

   - Option 2:

     Configure Smart transport through an HTTPs proxy. See Configuring Smart Transport Through an HTTPs Proxy, on page 113

   - Option 3:

     Configure Call Home service for direct cloud access. See Configuring the Call Home Service for Direct Cloud Access, on page 114.

   - Option 4:

     Configure Call Home service for direct cloud access through an HTTPs proxy. See Configuring the Call Home Service for Direct Cloud Access through an HTTPs Proxy Server, on page 117.

3. ***Trust Establishment with CSSM***

   Where task is performed: CSSM Web UI and then the product instance

   a. Generate one token for each *Virtual Account* you have. You can use same token for all the product instances that are part of one Virtual Account: Generating a New Token for a Trust Code from CSSM, on page 121

   b. Having downloaded the token, you can now install the trust code on the product instance: Installing a Trust Code, on page 122

Result:

After establishing trust, CSSM returns a policy. The policy is automatically installed on all product instances of that Virtual Account. The policy specifies if and how often the product instance reports usage.

In the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train: The product instance does not send more than one RUM report a day. You can override this for an on-demand synchronization between the product instance and CSSM, by entering the **license smart sync** command in privileged EXEC mode.

To change the reporting interval, configure the **license smart usage interval** command in global configuration mode. For syntax details see the *license smart (privileged EXEC)* command in the Command Reference for the corresponding release.

In case of a change in license usage, see Configuring an AIR License, on page 128 to know how it affects reporting.

Workflow for Topology: CSLU Disconnected from CSSM

Depending on whether you want to implement a product instance-initiated or CSLU-initiated method of communication. Complete the corresponding table of tasks below.

- Tasks for Product Instance-Initiated Communication
- Tasks for CSLU-Initiated Communication

Tasks for Product Instance-Initiated Communication

CSLU Installation → CSLU Preference Settings → Product Instance Configuration→ Usage Synchronization

1. ***CSLU Installation***

   Where task is performed: A Windows host (laptop, destop, or a Virtual Machine (VM)

   Download the file from Smart Software Manager > **Smart Licensing Utility**.

   Refer to Cisco Smart License Utility Quick Start Setup Guide for help with installation and set-up.

2. ***CSLU Preference Settings***

   Where tasks are performed: CSLU

   a. In the CSLU Preferences tab, click the **Cisco Connectivity** toggle switch to **off**. The field switches to “Cisco Is Not Available”.

   b. Configuring a Smart Account and a Virtual Account (CSLU Interface), on page 91

   c. Adding a Product-Initiated Product Instance in CSLU (CSLU Interface), on page 92

3. ***Product Instance Configuration***

   Where tasks are performed: Product Instance

   a. Ensuring Network Reachability for Product Instance-Initiated Communication, on page 92

   b. Ensure that transport type is set to **cslu**.

      CSLU is the default transport type. If you have configured a different option, enter the **license smart transport cslu** command in global configuration mode. Save any changes to the configuration file.

      ```
      Device(config)# license smart transport cslu
      Device(config)# exit
      Device# copy running-config startup-config
      ```

   c. Specify how you want CSLU to be discovered (*choose one*)

      - Option 1:

        No action required. Name server configured for Zero-touch DNS discovery of `cslu-local`

Here, if you have configured DNS (the name server IP address is configured on the product instance), and the DNS server has an entry where hostname `cslu-local` is mapped to the CSLU IP address, then no further action is required. The product instance automatically discovers hostname `cslu-local`.

- Option 2:

  No action required. Name server and domain configured for Zero-touch DNS discovery of `cslu-local.<domain>`

  Here if you have configured DNS (the name server IP address and domain is configured on the product instance), and the DNS server has an entry where `cslu-local.<domain>` is mapped to the CSLU IP address, then no further action is required. The product instance automatically discovers hostname `cslu-local`.

- Option 3:

  Configure a specific URL for CSLU.

  Enter the **license smart url cslu** *http://<cslu_ip_or_host>:8182/cslu/v1/pi* command in global configuration mode. For `<cslu_ip_or_host>`, enter the hostname or the IP address of the windows host where you have installed CSLU. 8182 is the port number and it is the only port number that CSLU uses.

```
Device(config)# license smart url cslu http://192.168.0.1:8182/cslu/v1/pi
Device(config)# exit
Device# copy running-config startup-config
```

4. ***Usage Synchronization***

   Where tasks are performed: CSLU and CSSM

   Since the product instance initiates communication, it automatically sends out the first RUM report at the scheduled time, as per the policy. You can also enter the **license smart sync** privileged EXEC command to trigger this. Along with this first report, if applicable, it sends a request for a UDI-tied trust code. Since CSLU is disconnected from CSSM, perform the following tasks to send the RUM Reports to CSSM.

   a. Export to CSSM (CSLU Interface), on page 95

   b. Uploading Data or Requests to CSSM and Downloading a File, on page 123

   c. Import from CSSM (CSLU Interface), on page 96

Result:

The ACK is applied to the product instance the next time the product instance contacts CSLU.

In the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train: The product instance does not send more than one RUM report a day. You can override this for an on-demand synchronization between the product instance and CSSM, by entering the **license smart sync** command in privileged EXEC mode.

To know when the product instance will be sending the next RUM report, enter the **show license all** command in privileged EXEC mode and in the output, check the date for the `Next report push` field.

In case of a change in license usage, see Configuring an AIR License, on page 128 to know how it affects reporting.

Tasks for CSLU-Initiated Communication

CSLU Installation → CSLU Preference Settings → Product Instance Configuration → Usage Synchronization

1. ***CSLU Installation***

   Where task is performed: A Windows host (laptop, destop, or a Virtual Machine (VM)

   Download the file from Smart Software Manager > **Smart Licensing Utility**.

   Refer to Cisco Smart License Utility Quick Start Setup Guide for help with installation and set-up.

2. ***CSLU Preference Settings***

   Where tasks is performed: CSLU

   a. In the CSLU Preferences tab, click the **Cisco Connectivity** toggle switch to **off**. The field switches to “Cisco Is Not Available”.

   b. Configuring a Smart Account and a Virtual Account (CSLU Interface), on page 91

   c. Adding a CSLU-Initiated Product Instance in CSLU (CSLU Interface), on page 94

   d. Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94

3. ***Product Instance Configuration***

   Where task is performed: Product Instance

   Ensuring Network Reachability for CSLU-Initiated Communication, on page 96

4. ***Usage Synchronization***

   Where tasks are performed: CSLU and CSSM

   Collect usage data from the product instance. Since CSLU is disconnected from CSSM, you then save usage data which CSLU has collected from the product instance to a file. Along with this first report, if applicable, an authorization code and a UDI-tied trust code request is included in the RUM report. Then, from a workstation that is connected to Cisco, upload it to CSSM. After this, download the ACK from CSSM. In the workstation where CSLU is installed and connected to the product instance, upload the file to CSLU.

   a. Export to CSSM (CSLU Interface), on page 95

   b. Uploading Data or Requests to CSSM and Downloading a File, on page 123

   c. Import from CSSM (CSLU Interface), on page 96

Result:

The ACK you have imported from CSSM contains the trust code and SLAC if this was requested. The uploaded ACK is applied to the product instance the next time CSLU runs an update.

In case of a change in license usage, see Configuring an AIR License, on page 128 to know how it affects reporting.

Workflow for Topology: Connected to CSSM Through a Controller

To deploy Cisco DNA Center as the controller, complete the following workflow:

Product Instance Configuration → Cisco DNA Center Configuration

1. ***Product Instance Configuration***

   Where task is performed: Product Instance

   Enable NETCONF. Cisco DNA Center uses the NETCONF protocol to provision configuration and retrieve the required information from the product instance - the product instance must therefore have NETCONF enabled, to facilitate this.

   For more information, see the Programmability Configuration Guide, Cisco IOS XE Amsterdam 17.3.x. In the guide, go to *Model-Driven Programmability* > *NETCONF Protocol*.

2. ***Cisco DNA Center Configuration***

   Where tasks is performed: Cisco DNA Center GUI

   An outline of the tasks you must complete and the accompanying documentation reference is provided below. The document provides detailed steps you have to complete in the Cisco DNA Center GUI:

   a. Set-up the Smart Account and Virtual Account.

      Enter the same log in credentials that you use to log in to the CSSM Web UI. This enables Cisco DNA Center to establish a connection with CSSM.

      See the *Cisco DNA Center Administrator Guide* of the required release (Release 2.2.2 onwards) > *Manage Licenses* > *Set Up License Manager*.

   b. Add the required product instances to Cisco DNA Center inventory and assign them to a site.

      This enables Cisco DNA Center to push any necessary configuration, including the required certificates, for Smart Licensing Using Policy to work as expected.

      See the *Cisco DNA Center User Guide* of the required release (Release 2.2.2 onwards) > *Display Your Network Topology* > *Assign Devices to a Site*.

Result:

After you implement the topology, *you* must trigger the very first ad hoc report in Cisco DNA Center, to establish a mapping between the Smart Account and Virtual Account, and product instance. See the *Cisco DNA Center Administrator Guide* of the required release (Release 2.2.2 onwards) > *Manage Licenses* > *Upload Resource Utilization Details to CSSM*. Once this is done, Cisco DNA Center handles subsequent reporting based on the reporting policy.

If multiple policies are available, Cisco DNA Center maintains the narrowest reporting interval. You can change this, but only to report more frequently (a narrower interval). See the *Cisco DNA Center Administrator Guide* of the required release (Release 2.2.2 onwards) > *Manage Licenses* > *Modify License Policy*.

If you want to change the license level after this, see the *Cisco DNA Center Administrator Guide* of the required release (Release 2.2.2 onwards) > *Manage Licenses* > *Change License Level*.

Workflow for Topology: No Connectivity to CSSM and No CSLU

Since you do not have to configure connectivity to any other component, the list of tasks required to set-up the topology is a small one. See, the ***Results*** section at the end of the workflow to know how you can complete requisite usage reporting after you have implemented this topology.

Product Instance Configuration

Where task is performed: Product Instance

Set transport type to **off**.

Enter the **license smart transport off** command in global configuration mode. Save any changes to the configuration file.

```
Device(config)# license smart transport off
Device(config)# exit
Device# copy running-config startup-config
```

Result:

All communication to and from the product instance is disabled. To report license usage you must save RUM reports to a file on the product instance. From a workstation that has connectivity to the Internet and Cisco, upload the file to CSSM:

1. Generate and save RUM reports

   Enter the **license smart save usage** command in provileged EXEC mode. In the example below, all RUM reports are saved to the flash memory of the product instance, in file `all_rum.txt`.

   In the example below, the file is first saved to bootflash and then copied to a TFTP location:

   ```
   Device# license smart save usage all file bootflash:all_rum.txt
   Device# copy bootflash:all_rum.txt tftp://10.8.0.6/all_rum.txt
   ```

2. Upload usage data to CSSM: Uploading Data or Requests to CSSM and Downloading a File, on page 123.
3. Install the ACK on the product instance: Installing a File on the Product Instance, on page 124

If you want to change license usage, see Configuring an AIR License, on page 128.

If you want to return an SLR authorization code, see Removing and Returning an Authorization Code, on page 118.

Workflow for Topology: SSM On-Prem Deployment

Depending on whether you want to implement a product instance-initiated (push) or SSM On-Prem-initiated (pull) method of communication, complete the corresponding sequence of tasks.

Tasks for Product Instance-Initiated Communication

SSM On-Prem Installation → Addition and Validation of Product Instances (Only if Applicable) → Product Instance Configuration→ Initial Usage Synchronization

1. ***SSM On-Prem Installation***

   Where task is performed: A physical server such as a Cisco UCS C220 M3 Rack Server, or a hardware-based server that meets the necessary requirements.

   Download the file from Smart Software Manager > **Smart Software Manager On-Prem**.

   Refer to the *Cisco Smart Software On-Prem Installation Guide* and the *Cisco Smart Software On-Prem User Guide* for help with installation.

   Installation is complete when you have deployed SSM On-Prem, configured a common name on SSM On-Prem (**Security Widget** > **Certificates**), synchronized the NTP server (**Settings** widget > **Time Settings**), and created, registered, and synchronized (**Synchronization** widget) the SSM On-Prem local account with your Smart Account and Virtual Account in CSSM.

Note Licensing functions in the **On-Prem Licensing Workspace** are greyed-out until you complete the creation, registration, and synchronization of the local account with your Smart Account in CSSM. The *local account*synchronization with CSSM is for the SSM On-Prem instance to be known to CSSM, and is different from usage synchronization which is performed in **4. Initial Usage Synchronization** below.

2. ***Addition and Validation of Product Instances***

   Where tasks are performed: SSM On-Prem UI

   This step ensures that the product instances are validated and mapped to the applicable Smart Account and Virtual account in CSSM. This step is required only in the following cases:

   - If you want your product instances to be added and validated in SSM On-Prem before they are reported in CSSM (for added security).
   - If you have created local virtual accounts (in addition to the default local virtual account) in SSM On-Prem. In this case you must provide SSM On-Prem with the Smart Account and Virtual Account information for the product instances in these local virtual accounts, so that SSM On-Prem can report usage to the correct license pool in CSSM.

   a. Assigning a Smart Account and Virtual Account (SSM On-Prem UI), on page 100

   b. Validating Devices (SSM On-Prem UI), on page 101

Note If your product instance is in a NAT set-up, also enable support for a NAT Setup when you enable device validation – both toggle switches are in the same window.

3. ***Product Instance Configuration***

   Where tasks are performed: Product Instance and the SSM On-Prem UI

   Remember to save any configuration changes on the product instance, by entering the **copy running-config startup-config** command in privileged EXEC mode.

   a. Ensuring Network Reachability for Product Instance-Initiated Communication, on page 101

   b. Retrieving the Transport URL (SSM On-Prem UI), on page 104

   c. Setting the Transport Type, URL, and Reporting Interval, on page 125

      The transport type configuration for CSLU and SSM On-Prem are the same (**license smart transport cslu** command in global configuration mode), but the URLs are different.

4. ***Initial Usage Synchronization***

   Where tasks are performed: Product instance, SSM On-Prem, CSSM

   a. Synchronize the product instance with SSM On-Prem.

      On the product instance, enter the **license smart sync** {**all** | **local**} command, in privileged EXEC mode. This synchronizes the product instance with SSM On-Prem, to send and receive any pending data. For example:

```
Device# license smart sync local
```

You can verify this in the SSM On-Prem UI. Log in and select the **Smart Licensing** workspace. Navigate to the **Inventory** > **SL Using Policy** tab. In the **Alerts** column of the corresponding product instance, the following message is displayed: Usage report from product instance.

Note If you have not performed Step 2 above (Addition and Validation of Product Instances), completing this sub-step will add the product instance to the SSM On-Prem database.

b. Synchronize usage information with CSSM (*choose one*):

- Option 1:

  SSM On-Prem is connected to CSSM: In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Reports** > **Usage Schedules** > **Synchronize now with Cisco**.

- Option 2:

  SSM On-Prem is not connected to CSSM: See Exporting and Importing Usage Data (SSM On-Prem UI), on page 104.

Result:

You have completed initial usage synchronization. Product instance and license usage information is now displayed in SSM On-Prem.

For subsequent reporting, you have the following options:

- To synchronize data between the product instance and SSM On-Prem:

  Schedule periodic synchronization between the product instance and the SSM On-Prem, by configuring the reporting interval. Enter the **license smart usage interval** *interval_in_days* command in global configuration mode.

  In the Cisco IOS XE Amsterdam 17.3.6 and later releases of the 17.3.x train: The product instance does not send more than one RUM report a day. You can override this for an on-demand synchronization between the product instance and CSSM, by entering the **license smart sync** command in privileged EXEC mode.

  To know when the product instance will be sending the next RUM report, enter the **show license all** command in privileged EXEC mode and in the output, check the `Next report push:` field.

- To synchronize usage information with CSSM schedule periodic synchronization, or , upload and download the required files:

  - Schedule periodic synchronization with CSSM. In the SSM On-Prem UI, navigate to **Reports** > **Usage Schedules** > **Synchronization schedule with Cisco**. Enter the following frequency information and save:

    - **Days**: Refers to how *often* synchronization occurs. For example, if you enter 2, synchronization occurs once every two days.

    - **Time of Day:**: Refers to the time at which synchronization occurs, in the 24-hour notation system. For example, if you enter 14 hours and 0 minutes, synchronization occurs at 2 p.m. (1400) in your local time zone.

- Upload and download the required files for reporting: Exporting and Importing Usage Data (SSM On-Prem UI), on page 104).

Tasks for SSM On-Prem Instance-Initiated Communication

SSM On-Prem Installation → Product Instance Addition → Product Instance Configuration→ Initial Usage Synchronization

1. ***SSM On-Prem Installation***

   Where task is performed: A physical server such as a Cisco UCS C220 M3 Rack Server, or a hardware-based server that meets the necessary requirements.

   Download the file from Smart Software Manager > **Smart Software Manager On-Prem**.

   Refer to the *Cisco Smart Software On-Prem Installation Guide* and the *Cisco Smart Software On-Prem User Guide* for help with installation.

   Installation is complete when you have deployed SSM On-Prem, configured a common name on SSM On-Prem (**Security Widget** > **Certificates**), synchronized the NTP server (**Settings** widget > **Time Settings**), and created, registered, and synchronized (**Synchronization** widget) the SSM On-Prem local account with your Smart Account and Virtual Account in CSSM.

Note Licensing functions in the **On-Prem Licensing Workspace** are greyed-out until you complete the creation, registration, and synchronization of the local account with your Smart Account in CSSM. The *local account* synchronization with CSSM is for the SSM On-Prem instance to be known to CSSM, and is different from usage synchronization which is performed in **4. Initial Usage Synchronization** below.

2. ***Product Instance Addition***

   Where task is performed: SSM On-Prem UI

   Depending on whether you want to add a single product instance or multiple product instances, follow the corresponding sub-steps: Adding One or More Product Instances (SSM On-Prem UI), on page 105.

3. ***Product Instance Configuration***

   Where tasks are performed: Product Instance and the SSM On-Prem UI

   Remember to save any configuration changes on the product instance, by entering the **copy running-config startup-config** command in privileged EXEC mode: Ensuring Network Reachability for SSM On-Prem-Initiated Communication, on page 106.

4. ***Initial Usage Synchronization***

   Where tasks are performed: SSM On-Prem UI, and CSSM

   a. Retrieve usage information from the product instance.

      In the SSM On-Prem UI, navigate to **Reports** > **Synchronisation pull schedule with the devices** > **Synchronise now with the device**.

      In the **Alerts** column, the following message is displayed: Usage report from product instance.

Tip It takes 60 seconds before synchronization is triggered. To view progress, navigate to the **On-Prem Admin Workspace**, and click the **Support Centre** widget. The system logs here display progress.

b. Synchronize usage information with CSSM (*choose one*)

- Option 1:

  SSM On-Prem is connected to CSSM: In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Reports** > **Usage Schedules** > **Synchronize now with Cisco**.

- Option 2:

  SSM On-Prem is not connected to CSSM. See: Exporting and Importing Usage Data (SSM On-Prem UI), on page 104.

Result:

You have completed initial usage synchronization. Product instance and license usage information is now displayed in SSM On-Prem. SSM On-Prem automatically sends the ACK back to the product instance. To verify that the product instance has received the ACK, enter the **show license status** command in privileged EXEC mode, and in the output, check the date for the `Last ACK received` field.

For subsequent reporting, you have the following options:

- To retrieve usage information from the product instance, you can:
  - In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Reports** > **Usage Schedules** > **Synchronize now with Cisco**.
  - Schedule periodic retrieval of information from the product instance by configuring a frequency. In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Reports** > **Usage Schedules** > **Synchronisation pull schedule with the devices**. Enter values in the following fields:
    - **Days**: Refers to how *often* synchronization occurs. For example, if you enter 2, synchronization occurs once every two days.
    - **Time of Day:**: Refers to the time at which synchronization occurs, in the 24-hour notation system. For example, if you enter 14 hours and 0 minutes, synchronization occurs at 2 p.m. (1400).
  - Collect usage data from the product instance without being connected to CSSM. In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Inventory** > **SL Using Policy** tab. Select one or more product instances by enabling the coresponding check box. Click **Actions for Selected...** > **Collect Usage**. On-Prem connects to the selected Product Instance(s) and collects the usage reports. These usage reports are then stored in On-Prem's local library. These reports can then be transferred to Cisco if On-Prem is connected to Cisco, or (if you are not connected to Cisco) you can manually trigger usage collection by selecting **Export/Import All..** > **Export Usage to Cisco**.
- To synchronize usage information with CSSM, you can:
  - Schedule periodic synchronization with CSSM. In the SSM On-Prem UI, navigate to **Reports** > **Usage Schedules** > **Synchronization schedule with Cisco**. Enter the following frequency information and save:

- **Days**: Refers to how *often* synchronization occurs. For example, if you enter 2, synchronization occurs once every two days.
- **Time of Day:**: Refers to the time at which synchronization occurs, in the 24-hour notation system. For example, if you enter 14 hours and 0 minutes, synchronization occurs at 2 p.m. (1400).

- Upload and download the required files for reporting: Exporting and Importing Usage Data (SSM On-Prem UI), on page 104).

Migrating to Smart Licensing Using Policy

To upgrade to Smart Licensing Using Policy, you must upgrade the software version (image) on the product instance to a supported version.

Before you Begin

Ensure that you have read the Upgrades, on page 53 section, to understand how Smart Licensing Using Policy handles all earlier licensing models.

Smart Licensing Using Policy is introduced in Cisco IOS XE Amsterdam 17.3.2a. This is therefore the minimum required version for Smart Licensing Using Policy.

Note that all the licenses that you are using prior to migration will be available after upgrade. This means that not only registered and authorized licenses (including reserved licenses), but also evaluation licenses will be migrated. The advantage with migrating registered and authorized licenses is that you will have fewer configuration steps to complete after migration, because your configuration is retained after upgrade (transport type configuration and configuration for connection to CSSM, all authorization codes). This ensures a smoother transition to the Smart Licensing Using Policy environment.

Device-led conversion is not supported for migration to Smart Licensing Using Policy.

Upgrading the Wireless Controller Software

For information about the upgrade procedure:

- For Cisco Embedded Wireless Controller on Cisco Catalyst 9100 Access Points, see the *Software Upgrade* section in the Cisco Embedded Wireless Controller on Catalyst Access Points Online Help
- For all other supported wireless controllers, see the *System Upgrade > Upgrading the Cisco Catalyst 9800 Wireless Controller Software* section of the Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide for the required release.

You can use the procedure to upgrade in install mode or ISSU (ISSU only on supported platforms and supported releases)

After Upgrading the Software Version

- Complete topology implementation.

  If a transport mode is available in your pre-upgrade set-up, this is retained after you upgrade. Only in some cases, like with evaluation licenses or with licensing models where the notion of a transport type

does not exist, the default (**cslu**) is applied - in these cases you may have a few more steps to complete before you are set to operate in the Smart Licensing Using Policy environment.

No matter which licensing model you upgrade from, you can change the topology after upgrade.

- Synchronize license usage with CSSM

  No matter which licensing model you are upgrading from and no matter which topology you implement, synchronize your usage information with CSSM. For this you have to follow the reporting method that applies to the topology you implement. This initial synchronization ensures that up-to-date usage information is reflected in CSSM and a custom policy (if available), is applied. The policy that is applicable after this synchronization also indicates subsequent reporting requirements. These rules are also tabled here: How Upgrade Affects Reporting for Existing Licenses, on page 53

Note After initial usage synchronization is completed, reporting is required only if the policy, or, system messages indicate that it is.

Sample Migration Scenarios

Sample migration scenarios have been provided considering the various existing licensing models and licenses. All scenarios provide sample outputs before and after migration, any CSSM Web UI changes to look out for (as an indicator of a successful migration or further action), and how to identify and complete any necessary post-migration steps.

Note For SSM On-Prem, the sequence in which you perform the various upgrade-related activities is crucial. So only for this scenario, the migration sequence has been provided - and not an example.

Example: Smart Licensing to Smart Licensing Using Policy

The following is an example of a Cisco Catalyst 9800-CL Wireless Controller migrating from Smart Licensing to Smart Licensing Using Policy.

- Table 6: Smart Licensing to Smart Licensing Using Policy: show Commands, on page 71
- The CSSM Web UI After Migration, on page 75
- Reporting After Migration, on page 78

The **show** command outputs below call-out key fields to check, before and after migration.

Table 6: Smart Licensing to Smart Licensing Using Policy: show Commands

Before Upgrade (Smart Licensing)	After Upgrade (Smart Licensing Using Policy)
show license summary The `Status` and `License Authorization` fields show that the license is `REGISTERED` and `AUTHORIZED`.	**show license summary** The `Status` field shows that the licenses are now `IN USE` instead of registered and authorized.

Before Upgrade (Smart Licensing)	After Upgrade (Smart Licensing Using Policy)

`Device# show license summary`
 	
`Smart Licensing is ENABLED`
 	
`Registration:`
`  Status: REGISTERED`
`  Smart Account: SA-Eg-Company-02`
`  Virtual Account: Dept-02`
`  Export-Controlled Functionality: ALLOWED`
`  Last Renewal Attempt: None`
`  Next Renewal Attempt: May 01 08:19:02 2021 IST`
 	
`License Authorization:`
`  Status: AUTHORIZED`
`  Last Communication Attempt: SUCCEEDED`
`  Next Communication Attempt: Dec 02 08:19:09 2020 IST`
 	
`License Usage:`
`  License  Entitlement tag  Count Status`
`---`
`  AP Perpetual Network... (DNA_NWSTACK_E)  1 AUTHORIZED`
`  Aironet DNA Essentia... (AIR-DNA-E)  1 AUTHORIZED`
	`Device# show license summary`
	
	`License Usage:`
	`  License  Entitlement Tag  Count Status`
	`---`
	`  air-network-essentials  (DNA_NWSTACK_E)  1 IN USE`
	`  air-dna-essentials  (AIR-DNA-E)  1 IN USE`

Before Upgrade (Smart Licensing)	After Upgrade (Smart Licensing Using Policy)
show license usage One perpetual and one subscription license are being used before upgrade.	**show license usage** All licenses are migrated and the `Enforcement Type` field displays NOT ENFORCED. There are no export-controlled or enforced licenses on Cisco Catalyst Wireless Controllers.

Before Upgrade (Smart Licensing)	After Upgrade (Smart Licensing Using Policy)

`Device# show license usage`
 	
`License Authorization:`
`  Status: AUTHORIZED on Nov 02 08:21:29 2020 IST`
 	
`AP Perpetual Networkstack Essentials (DNA_NWSTACK_E):`
`  Description: AP Perpetual Network Stack entitled with DNA-E`
`  Count: 1`
`  Version: 1.0`
`  Status: AUTHORIZED`
`  Export status: NOT RESTRICTED`
 	
`Aironet DNA Essentials Term Licenses (AIR-DNA-E):`
`  Description: DNA Essentials for Wireless`
`  Count: 1`
`  Version: 1.0`
`  Status: AUTHORIZED`
`  Export status: NOT RESTRICTED`
	`Device# show license usage`
	
	`License Authorization:`
	`  Status: Not Applicable`
	
	`air-network-essentials (DNA_NWSTACK_E):`
	`  Description: air-network-essentials`
	`  Count: 1`
	`  Version: 1.0`
	`  Status: IN USE`
	`  Export status: NOT RESTRICTED`
	`  Feature Name: air-network-essentials`
	`  Feature Description: air-network-essentials`
	`  Enforcement type: NOT ENFORCED`
	`  License type: Perpetual`
	
	`air-dna-essentials (AIR-DNA-E):`
	`  Description: air-dna-essentials`
	`  Count: 1`
	`  Version: 1.0`
	`  Status: IN USE`
	`  Export status: NOT RESTRICTED`
	`  Feature Name: air-dna-essentials`
	`  Feature Description: air-dna-essentials`
	`  Enforcement type: NOT ENFORCED`
	`  License type: Perpetual`

Before Upgrade (Smart Licensing)	After Upgrade (Smart Licensing Using Policy)
show license status	**show license status** The `Transport:` field shows that the transport type, which was configured before update, is retained after upgrade. The `Policy:` header and details show that a custom policy was available in the Smart Account or Virtual Account – this has also been automatically installed on the product instance. (After establishing trust, CSSM returns a policy. The policy is then automatically installed.) The `Usage Reporting:` header: The `Next report push:` field provides information about when the product instance will send the next RUM report to CSSM. The `Trust Code Installed:` field shows that the ID token is successfully converted and a trusted connected has been established with CSSM.

Before Upgrade (Smart Licensing)	After Upgrade (Smart Licensing Using Policy)

Device# **show license status**
 	
Smart Licensing is ENABLED
 	
Utility:
  Status: DISABLED
 	
Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED
 	
Transport:
  Type: Callhome
 	
Registration:
  Status: REGISTERED
  Smart Account: SA-Eg-Company-02
  Virtual Account: Dept-02
  Export-Controlled Functionality: ALLOWED
  Initial Registration: SUCCEEDED on Nov 02 08:19:02 2020 IST
  Last Renewal Attempt: None
  Next Renewal Attempt: May 01 08:19:01 2021 IST
  Registration Expires: Nov 02 08:14:06 2021 IST
 	
License Authorization:
  Status: AUTHORIZED on Nov 02 08:21:29 2020 IST
  Last Communication Attempt: SUCCEEDED on Nov 02 08:21:29 2020 IST
  Next Communication Attempt: Dec 02 08:19:09 2020 IST
  Communication Deadline: Jan 31 08:14:15 2021 IST
 	
Export Authorization Key:
  Features Authorized:
  <none>
	Device# **show license status**
	Utility:
	  Status: DISABLED
	
	Smart Licensing Using Policy:
	  Status: ENABLED
	
	Data Privacy:
	  Sending Hostname: yes
	  Callhome hostname privacy: DISABLED
	  Smart Licensing hostname privacy: DISABLED
	  Version privacy: DISABLED
	
	Transport:
	  Type: Callhome
	
	Policy:
	  Policy in use: Installed On Nov 02 09:09:47 2020 IST
	  Policy name: **SLE Policy**
	  Reporting ACK required: yes (Customer Policy)
	  Unenforced/Non-Export Perpetual Attributes:
	  First report requirement (days): 60 (Customer Policy)
	  Reporting frequency (days): 60 (Customer Policy)
	  Report on change (days): 60 (Customer Policy)
	  Unenforced/Non-Export Subscription Attributes:
	  First report requirement (days): 30 (Customer Policy)
	  Reporting frequency (days): 30 (Customer Policy)
	  Report on change (days): 30 (Customer Policy)
	  Enforced (Perpetual/Subscription) License Attributes:
	  First report requirement (days): 0 (CISCO default)
	  Reporting frequency (days): 90 (Customer Policy)
	  Report on change (days): 90 (Customer Policy)
	  Export (Perpetual/Subscription) License Attributes:
	  First report requirement (days): 0 (CISCO default)
	  Reporting frequency (days): 90 (Customer Policy)
	  Report on change (days): 90 (Customer Policy)
	
	Miscellaneous:
	  Custom Id: <empty>
	
	Usage Reporting:
	  Last ACK received: Nov 02 09:09:47 2020 IST
	  Next ACK deadline: Jan 01 09:09:47 2021 IST
	  Reporting push interval: 30 days
	  Next ACK push check: Nov 02 09:13:54 2020 IST
	  Next report push: Dec 02 09:05:45 2020 IST
	  Last report push: Nov 02 09:05:45 2020 IST
	  Last report file write: <none>
	
	Trust Code Installed:
	  Active: PID:C9800-CL-K9,SN:93BBAH93MGS
	  INSTALLED on Nov 02 08:59:26 2020 IST
	  Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN
	  INSTALLED on Nov 02 09:00:45 2020 IST

Before Upgrade (Smart Licensing)	After Upgrade (Smart Licensing Using Policy)
show license udi	**show license udi** This is a High Availability set-up and the command displays all UDIs in the set-up. There is no change in the sample output before and after migration.

`Device# show license udi`
`UDI: PID:C9800-CL-K9,SN:93BBAH93MGS`
 	
`HA UDI List:`
`Active:PID:C9800-CL-K9,SN:93BBAH93MGS`
`Standby:PID:C9800-CL-K9,SN:9XECPSUU4XN`
	`Device# show license udi`
	`UDI: PID:C9800-CL-K9,SN:93BBAH93MGS`
	
	`HA UDI List:`
	`Active:PID:C9800-CL-K9,SN:93BBAH93MGS`
	`Standby:PID:C9800-CL-K9,SN:9XECPSUU4XN`

The CSSM Web UI After Migration

Log in to the CSSM Web UI at https://software.cisco.com and click **Smart Software Licensing**. Under **Inventory** > **Product Instances**.

The product instance previously displayed with the host name (Catalyst 9800CL Cloud Wireless Controller in this example) is now displayed with the UDI instead. All migrated UDIs are displayed, that is, PID:C9800-CL-K9,SN:93BBAH93MGS, and PID:C9800-CL-K9,SN:9XECPSUU4XN.

Only the active product instance reports usage, therefore, PID:C9800-CL-K9,SN:93BBAH93MGS displays license consumption information under **License Usage**. The standby does not report usage and the **License Usage** for the standby displays No Records Found.

Figure 8: Smart Licensing to Smart Licensing Using Policy: Hostname of Product Instance on the CSSM Web UI Before Migration

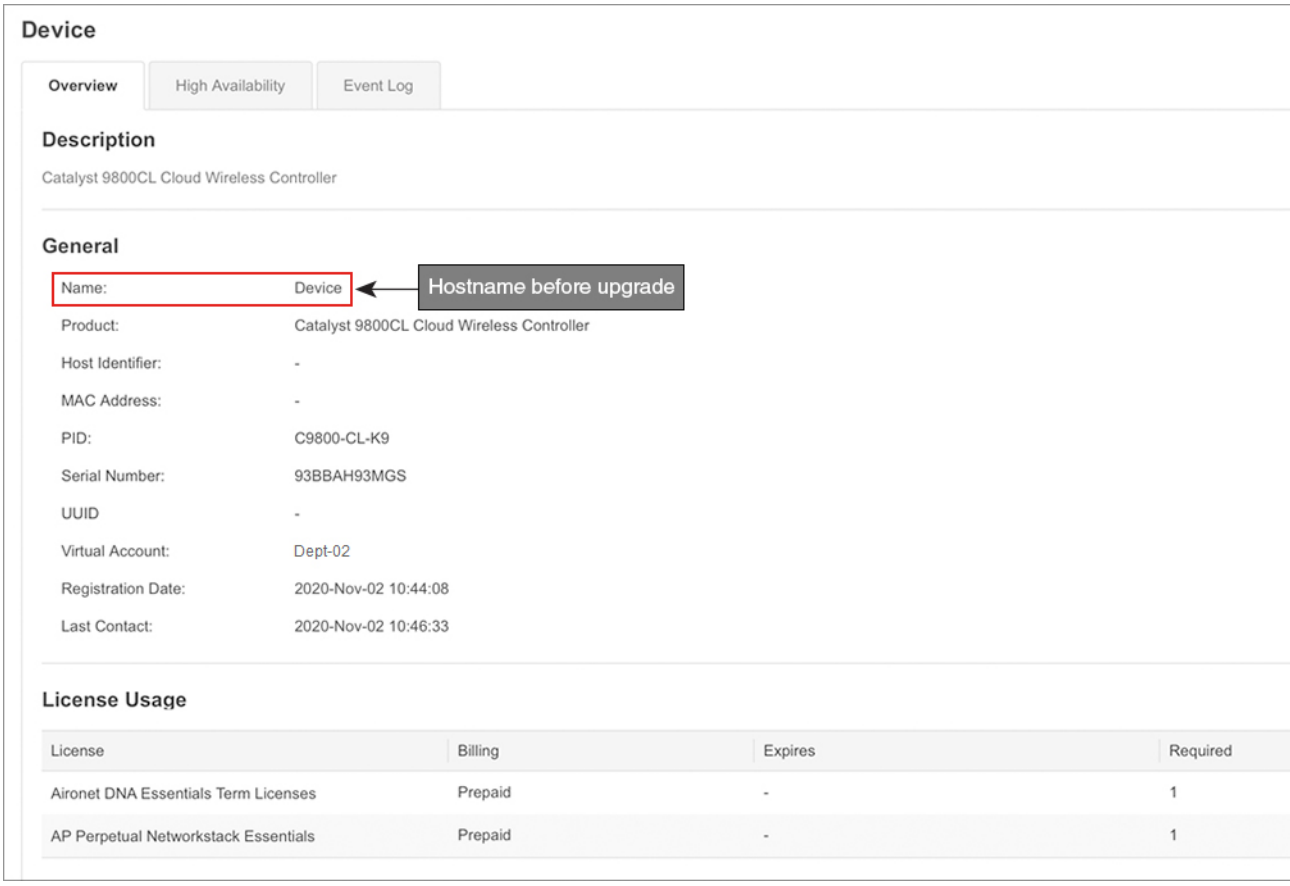


Figure 9: Smart Licensing to Smart Licensing Using Policy: UDI and License Usage Under Active Product Instance After Migration

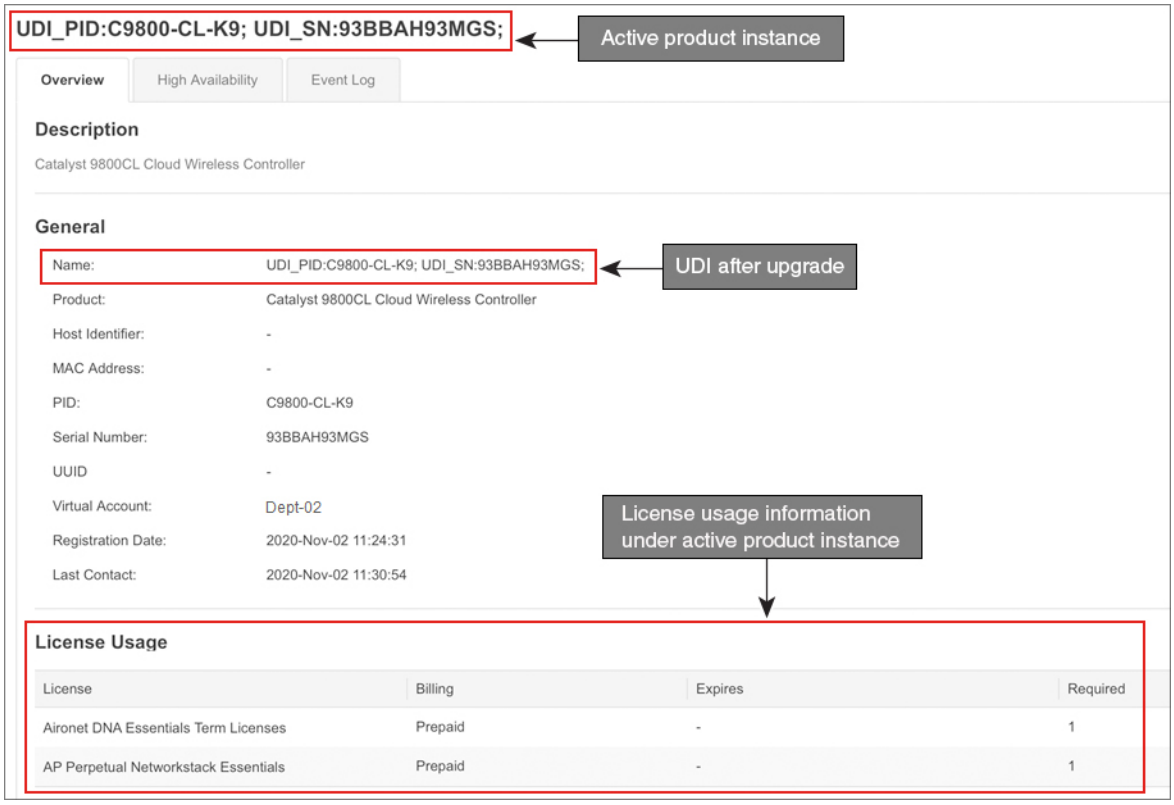
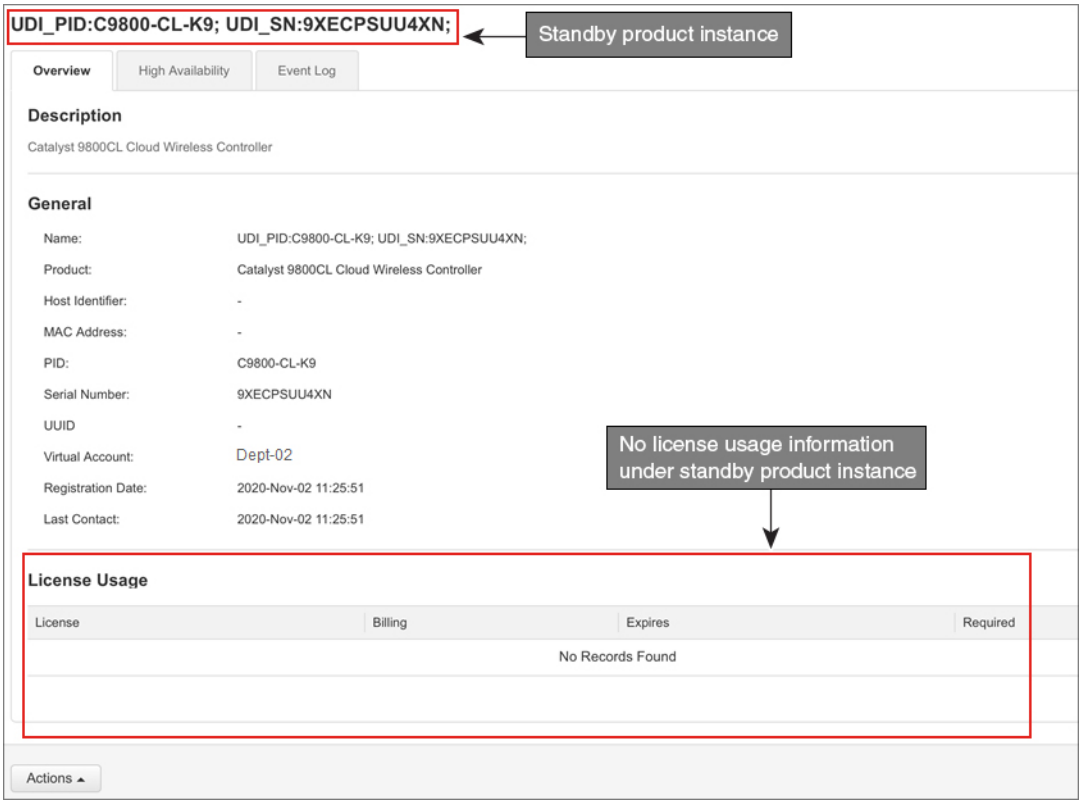


Figure 10: Smart Licensing to Smart Licensing Using Policy: Standby Product Instance After Migration



It is always the active that reports usage, so if the active in this High Availabilty set-up changes, the new active product instance will display license consumption information and report usage.

Reporting After Migration

The product instance sends the next RUM report to CSSM, based on the policy.

If you want to change your reporting interval to report more frequently: on the product instance, configure the **license smart usage interval** command in global configuration mode. For syntax details see the *license smart (global config)* command in the Command Reference for the corresponding release.

Example: SLR to Smart Licensing Using Policy

The following is an example of a Cisco Catalyst 9800-CL Wireless Controller migrating from Specific License Reservation (SLR) to Smart Licensing Using Policy. This is a High Availability set-up with an active and standby.

License conversion is automatic and authorization codes are migrated. No further action is required to complete migration. After migration the No Connectivity to CSSM and No CSLU, on page 48 topology is effective. For information about the SLR authorization code in the Smart Licensing Using Policy environment, see Authorization Code, on page 39.

- Table 7: SLR to Smart Licensing Using Policy: show Commands, on page 79
- The CSSM Web UI After Migration, on page 83

• Reporting After Migration, on page 85

The **show** command outputs below call-out key fields to check, before and after migration.

Table 7: SLR to Smart Licensing Using Policy: show Commands

Before Upgrade (SLR)	After Upgrade (Smart Licensing Using Policy)

show license summary
 	
The `Registration` and `License Authorization` status fields show that the license was `REGISTERED - SPECIFIC LICENSE RESERVATION` and `AUTHORIZED - RESERVED`.
 	
`Device# show license summary`
 	
`Smart Licensing is ENABLED`
`License Reservation is ENABLED`
 	
`Registration:`
 	
`  Status: REGISTERED - SPECIFIC LICENSE RESERVATION`
`  Export-Controlled Functionality: ALLOWED`
 	
`License Authorization:`
`  Status: AUTHORIZED - RESERVED`
 	
`License Usage:`
`  License  Entitlement tag  Count Status`
 	
`---`
 	
`  AP Perpetual Network... (DNA_NWStack)`
`1 AUTHORIZED`
`  Aironet DNA Advantag... (AIR-DNA-A)`
`1 AUTHORIZED`
	show license summary
	
	Licenses are migrated , but none of the APs have joined the controller, current consumption (Count) is therefore zero, and the `Status` field shows that the licenses are `NOT IN USE`.
	
	`Device# show license summary`
	`License Reservation is ENABLED`
	
	`License Usage:`
	`  License  Entitlement Tag  Count Status`
	
	`---`
	
	`  Aironet DNA Advantag... (AIR-DNA-A)  0 NOT IN USE`
	`  AP Perpetual Network... (DNA_NWStack)  0 NOT IN USE`

Before Upgrade (SLR)	After Upgrade (Smart Licensing Using Policy)
show license reservation	**show license authorization** The `Last Confirmation code:` field shows that the SLR authorization code is successfully migrated for the active and standby product instances in the High Availability set-up. The `Specified license reservations:` header shows that a perpetual license (AP Perpetual Networkstack Advantage) and a subscription license (Aironet DNA Advantage Term Licenses) are the migrated SLR licenses.

Before Upgrade (SLR)	After Upgrade (Smart Licensing Using Policy)

```
Device# show license reservation
License reservation: ENABLED

Overall status:
  Active: PID:C9800-CL-K9,SN:93BBAH93MGS
       Reservation status: SPECIFIC INSTALLED on Nov 02
 03:16:01 2020 IST
       Export-Controlled Functionality: ALLOWED
       Last Confirmation code: 102fc949
  Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN
       Reservation status: SPECIFIC INSTALLED on Nov 02
 03:15:45 2020 IST
       Export-Controlled Functionality: ALLOWED
       Last Confirmation code: ad4382fe

Specified license reservations:
  Aironet DNA Advantage Term Licenses (AIR-DNA-A):
    Description: DNA Advantage for Wireless
    Total reserved count: 20
    Term information:
      Active: PID:C9800-CL-K9,SN:93BBAH93MGS
        License type: TERM
          Start Date: 2020-OCT-14 UTC
          End Date: 2021-APR-12 UTC
          Term Count: 5
        License type: TERM
          Start Date: 2020-JUN-18 UTC
          End Date: 2020-DEC-15 UTC
          Term Count: 5
      Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN
        License type: TERM
          Start Date: 2020-OCT-14 UTC
          End Date: 2021-APR-12 UTC
          Term Count: 10
  AP Perpetual Networkstack Advantage (DNA_NWStack):
    Description: AP Perpetual Network Stack entitled
with DNA-A
    Total reserved count: 20
    Term information:
      Active: PID:C9800-CL-K9,SN:93BBAH93MGS
        License type: TERM
          Start Date: 2020-OCT-14 UTC
          End Date: 2021-APR-12 UTC
          Term Count: 5
        License type: TERM
          Start Date: 2020-JUN-18 UTC
          End Date: 2020-DEC-15 UTC
          Term Count: 5
      Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN
        License type: TERM
          Start Date: 2020-OCT-14 UTC
          End Date: 2021-APR-12 UTC
          Term Count: 10
```

Before Upgrade (SLR)	After Upgrade (Smart Licensing Using Policy)

	Device# **show license authorization**
	Overall status:
	  Active: PID:C9800-CL-K9,SN:93BBAH93MGS
	  Status: SPECIFIC INSTALLED on Nov 02 03:16:01 2020 IST
	  Last Confirmation code: 102fc949
	  Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN
	  Status: SPECIFIC INSTALLED on Nov 02 03:15:45 2020 IST
	  Last Confirmation code: ad4382fe
	
	Specified license reservations:
	  Aironet DNA Advantage Term Licenses (AIR-DNA-A):
	  Description: DNA Advantage for Wireless
	  Total reserved count: 20
	  Enforcement type: NOT ENFORCED
	  Term information:
	  Active: PID:C9800-CL-K9,SN:93BBAH93MGS
	  Authorization type: SPECIFIC INSTALLED on Nov 02 03:15:45 2020 IST
	  License type: TERM
	  Start Date: 2020-OCT-14 UTC
	  End Date: 2021-APR-12 UTC
	  Term Count: 5
	  Authorization type: SPECIFIC INSTALLED on Nov 02 03:15:45 2020 IST
	  License type: TERM
	  Start Date: 2020-JUN-18 UTC
	  End Date: 2020-DEC-15 UTC
	  Term Count: 5
	  Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN
	  Authorization type: SPECIFIC INSTALLED on Nov 02 03:15:45 2020 IST
	  License type: TERM
	  Start Date: 2020-OCT-14 UTC
	  End Date: 2021-APR-12 UTC
	  Term Count: 10
	  AP Perpetual Networkstack Advantage (DNA_NWStack):
	  Description: AP Perpetual Network Stack entitled with DNA-A
	  Total reserved count: 20
	  Enforcement type: NOT ENFORCED
	  Term information:
	  Active: PID:C9800-CL-K9,SN:93BBAH93MGS
	  Authorization type: SPECIFIC INSTALLED on Nov 02 03:15:45 2020 IST
	  License type: TERM
	  Start Date: 2020-OCT-14 UTC
	  End Date: 2021-APR-12 UTC
	  Term Count: 5
	  Authorization type: SPECIFIC INSTALLED on Nov 02 03:15:45 2020 IST
	  License type: TERM
	  Start Date: 2020-JUN-18 UTC
	  End Date: 2020-DEC-15 UTC
	  Term Count: 5
	  Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN
	  Authorization type: SPECIFIC INSTALLED on Nov 02 03:15:45 2020 IST
	  License type: TERM
	  Start Date: 2020-OCT-14 UTC
	  End Date: 2021-APR-12 UTC

Before Upgrade (SLR)	After Upgrade (Smart Licensing Using Policy)

	`Term Count: 10`
	
	`Purchased Licenses:`
	`No Purchase Information Available`

Before Upgrade (SLR)	After Upgrade (Smart Licensing Using Policy)
show license status	**show license status** Under the `Transport:` header, the `Type:` field displays that the transport type is set to off. Under the `Usage Reporting:` header, the `Next report push:` field displays if and when the next RUM report must be uploaded to CSSM.

Before Upgrade (SLR)	After Upgrade (Smart Licensing Using Policy)

-	Device# **show license status**
	
	Utility:
	  Status: DISABLED
	
	Smart Licensing Using Policy:
	  Status: ENABLED
	
	Data Privacy:
	  Sending Hostname: yes
	  Callhome hostname privacy: DISABLED
	  Smart Licensing hostname privacy: DISABLED
	  Version privacy: DISABLED
	
	Transport:
	  Type: Transport Off
	
	Policy:
	  Policy in use: Merged from multiple sources.
	  Reporting ACK required: yes (CISCO default)
	  Unenforced/Non-Export Perpetual Attributes:
	  First report requirement (days): 365 (CISCO default)
	
	  Reporting frequency (days): 0 (CISCO default)
	  Report on change (days): 90 (CISCO default)
	  Unenforced/Non-Export Subscription Attributes:
	  First report requirement (days): 90 (CISCO default)
	
	  Reporting frequency (days): 90 (CISCO default)
	  Report on change (days): 90 (CISCO default)
	  Enforced (Perpetual/Subscription) License Attributes:
	
	  First report requirement (days): 0 (CISCO default)
	  Reporting frequency (days): 0 (CISCO default)
	  Report on change (days): 0 (CISCO default)
	  Export (Perpetual/Subscription) License Attributes:
	  First report requirement (days): 0 (CISCO default)
	  Reporting frequency (days): 0 (CISCO default)
	  Report on change (days): 0 (CISCO default)
	
	Miscellaneous:
	  Custom Id: <empty>
	
	Usage Reporting:
	  Last ACK received: <none>
	  Next ACK deadline: <none>
	  Reporting push interval: 0 (no reporting)
	  Next ACK push check: Nov 01 20:31:46 2020 IST
	  Next report push: <none>
	  Last report push: <none>
	  Last report file write: <none>
	
	Trust Code Installed: <none>

The CSSM Web UI After Migration

Log in to the CSSM Web UI at https://software.cisco.com and click **Smart Software Licensing**. Under **Inventory** > **Product Instances**.

There are no changes in the **Product Instances** tab. The Last Contact column displays "Reserved Licenses" since there has been no usage reporting yet. After the requisite RUM report is uploaded and acknowledged "Reserved Licenses" is no longer displayed and license usage is displayed only in the active product instance.

Figure 11: SLR to Smart Licensing Using Policy: Active Product Instance Before Upgrade

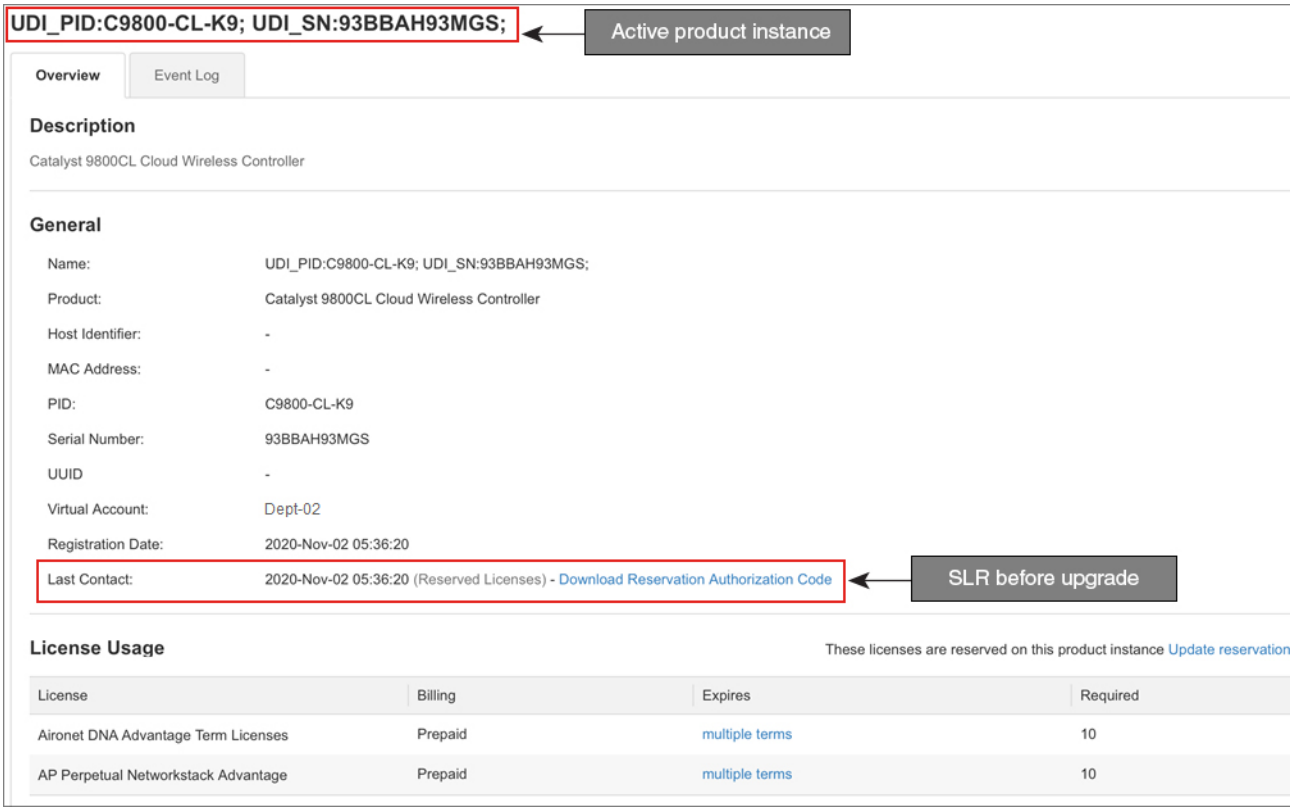
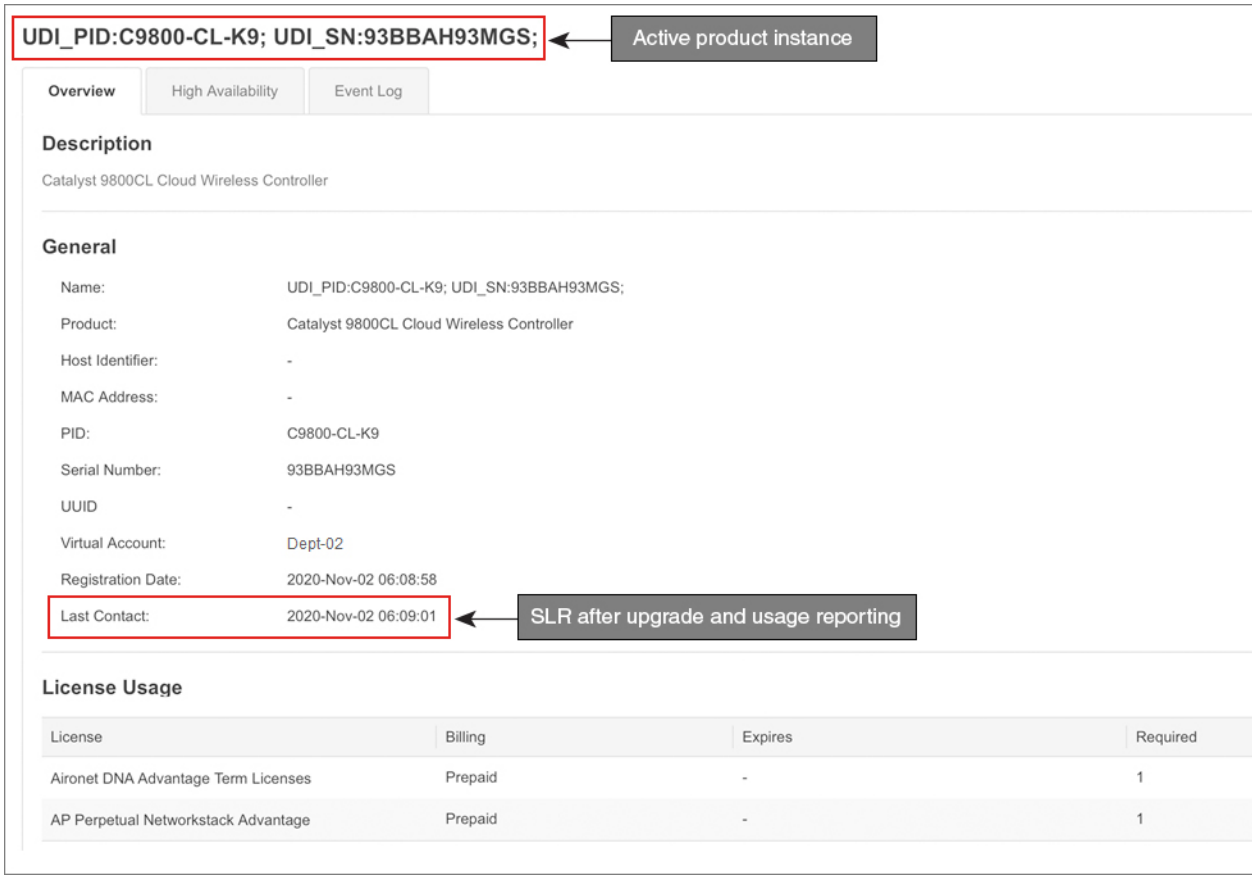


Figure 12: SLR to Smart Licensing Using Policy: Active Product Instance After Upgrade



Reporting After Migration

SLR licenses require reporting only when there is a change in license consumption (For example, when using a subscription license which is for specified term).

In an air-gapped network, use the `Next report push:` date in the **show license status** output to know when the next usage report must be sent. This ensures that the product instance and CSSM are synchronized.

Since all communication to and from the product instance is disabled, to report license usage you must save RUM reports to a file and upload it to CSSM (from a workstation that has connectivity to the internet, and Cisco):

1. Generate and save RUM reports

   Enter the **license smart save usage** command in provileged EXEC mode. In the example below, all RUM reports are saved to the flash memory of the product instance, in file `all_rum.txt`. For syntax details see the *license smart (privileged EXEC)* command in the Command Reference. In the example, the file is first saved to bootflash and then copied to a TFTP location:

   ```
   Device# license smart save usage all bootflash:all_rum.txt
   Device# copy bootflash:all_rum.txt tftp://10.8.0.6/all_rum.txt
   ```

2. Upload usage data to CSSM: Uploading Data or Requests to CSSM and Downloading a File, on page 123

3. Install the ACK on the product instance: Installing a File on the Product Instance, on page 124

Example: Evaluation or Expired to Smart Licensing Using Policy

The following is an example of a Cisco Catalyst 9800-CL Wireless Controller with evaluation expired licenses (Smart Licensing) that are migrated to Smart Licensing Using Policy.

The notion of evaluation licenses does not apply to Smart Licensing Using Policy. When the software version is upgraded to one that supports Smart Licensing Using Policy, all licenses are displayed as IN USE and the Cisco default policy is applied to the product instance. Since all licenses on Cisco Catalyst Wireless Controllers are unenforced (enforcement type), no functionality is lost.

- Table 8: Evaluation or Expired to Smart Licensing Using Policy: show Commands, on page 86
- The CSSM Web UI After Migration, on page 89
- Reporting After Migration, on page 89

The table below calls out key changes or new fields to check for in the **show** command outputs, after upgrade to Smart Licensing Using Policy

Table 8: Evaluation or Expired to Smart Licensing Using Policy: show Commands

Before Upgrade (Smart Licensing, Evaluation Mode)	After Upgrade (Smart Licensing Using Policy)

show license summary
Licenses are UNREGISTERED and in `EVAL MODE`.
 	
`Device# show license summary`
`Smart Licensing is ENABLED`
 	
`Registration:`
`  Status: UNREGISTERED`
`  Export-Controlled Functionality: NOT ALLOWED`
 	
`License Authorization:`
`  Status: EVAL EXPIRED`
 	
`License Usage:`
`  License  Entitlement tag  Count Status`
 	
`----------`
 	
`  (DNA_NWStack)  1 EVAL EXPIRED`
`  (AIR-DNA-A)  1 EVAL EXPIRED`
	show license summary
	All licenses are migrated and `IN USE`. There are no EVAL MODE licenses.
	
	`Device# show license summary`
	`License Usage:`
	`  License  Entitlement Tag  Count Status`
	
	`----------`
	
	`  air-network-advantage  (DNA_NWStack)  1 IN USE`
	`  air-dna-advantage  (AIR-DNA-A)  1 IN USE`

Before Upgrade (Smart Licensing, Evaluation Mode)	After Upgrade (Smart Licensing Using Policy)
show license usage	**show license usage** The `Enforcement Type` field displays NOT ENFORCED. (There are no export-controlled or enforced licenses on Cisco Catalyst Wireless Controllers).

Before Upgrade (Smart Licensing, Evaluation Mode)	After Upgrade (Smart Licensing Using Policy)

`Device# show license usage`
`License Authorization:`
`  Status: EVAL EXPIRED on Apr 14 18:20:46 2020 UTC`
 	
`(DNA_NWStack):`
`  Description:`
`  Count: 1`
`  Version: 1.0`
`  Status: EVAL EXPIRED`
`  Export status: NOT RESTRICTED`
 	
`(AIR-DNA-A):`
`  Description:`
`  Count: 1`
`  Version: 1.0`
`  Status: EVAL EXPIRED`
`  Export status: NOT RESTRICTED`
	`Device# show license usage`
	`License Authorization:`
	`  Status: Not Applicable`
	
	`air-network-advantage (DNA_NWStack):`
	`  Description: air-network-advantage`
	`  Count: 1`
	`  Version: 1.0`
	`  Status: IN USE`
	`  Export status: NOT RESTRICTED`
	`  Feature Name: air-network-advantage`
	`  Feature Description: air-network-advantage`
	`  Enforcement type: NOT ENFORCED`
	`  License type: Perpetual`
	
	`air-dna-advantage (AIR-DNA-A):`
	`  Description: air-dna-advantage`
	`  Count: 1`
	`  Version: 1.0`
	`  Status: IN USE`
	`  Export status: NOT RESTRICTED`
	`  Feature Name: air-dna-advantage`
	`  Feature Description: air-dna-advantage`
	`  Enforcement type: NOT ENFORCED`
	`  License type: Perpetual`

Before Upgrade (Smart Licensing, Evaluation Mode)	After Upgrade (Smart Licensing Using Policy)
show license status	**show license status** The `Transport:` field displays that the default type is set, but a URL or a method for the product instance to discover CSLU is not specified. The `Trust Code Installed:` field displays that a trust code is not installed. The `Policy:` header and details show that the `Cisco default` policy is applied. Under the `Usage Reporting:` header, the `Next report push:` field provides information about when the next RUM report must be sent to CSSM.

Before Upgrade (Smart Licensing, Evaluation Mode)	After Upgrade (Smart Licensing Using Policy)

Device# **show license status**
 	
Smart Licensing is ENABLED
 	
Utility:
  Status: DISABLED
 	
Data Privacy:
  Sending Hostname: yes
  Callhome hostname privacy: DISABLED
  Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED
 	
Transport:
  Type: Callhome
 	
Registration:
  Status: UNREGISTERED
  Export-Controlled Functionality: NOT ALLOWED
 	
License Authorization:
  Status: EVAL EXPIRED on Apr 14 18:20:46 2020 UTC
 	
Export Authorization Key:
  Features Authorized:
  <none>
	Device# **show license status**
	Utility:
	  Status: DISABLED
	
	Smart Licensing Using Policy:
	  Status: ENABLED
	
	Data Privacy:
	  Sending Hostname: yes
	  Callhome hostname privacy: DISABLED
	  Smart Licensing hostname privacy: DISABLED
	  Version privacy: DISABLED
	
	Transport:
	  Type: cslu
	  Cslu address: <empty>
	  Proxy:
	  Not Configured
	
	Policy:
	  Policy in use: Merged from multiple sources.
	  Reporting ACK required: yes (CISCO default)
	  Unenforced/Non-Export Perpetual Attributes:
	  First report requirement (days): 365 (CISCO default)
	
	  Reporting frequency (days): 0 (CISCO default)
	  Report on change (days): 90 (CISCO default)
	  Unenforced/Non-Export Subscription Attributes:
	  First report requirement (days): 90 (CISCO default)
	
	  Reporting frequency (days): 90 (CISCO default)
	  Report on change (days): 90 (CISCO default)
	  Enforced (Perpetual/Subscription) License Attributes:
	
	  First report requirement (days): 0 (CISCO default)
	  Reporting frequency (days): 0 (CISCO default)
	  Report on change (days): 0 (CISCO default)
	  Export (Perpetual/Subscription) License Attributes:
	  First report requirement (days): 0 (CISCO default)
	  Reporting frequency (days): 0 (CISCO default)
	  Report on change (days): 0 (CISCO default)
	
	Miscellaneous:
	  Custom Id: <empty>
	
	Usage Reporting:
	  Last ACK received: <none>
	  Next ACK deadline: <none>
	  Reporting push interval: 0 (no reporting)
	  Next ACK push check: <none>
	  Next report push: <none>
	  Last report push: <none>
	  Last report file write: <none>
	
	Trust Code Installed: <none>

The CSSM Web UI After Migration

Log in to the CSSM Web UI at https://software.cisco.com and click **Smart Software Licensing**. Under **Inventory** > **Product Instances**, the Last Contact field for the migrated product instances display an updated timestamp after migration.

Reporting After Migration

Implement any one of the supported topologies, and fulfil reporting requirements. See Supported Topologies, on page 42 and How to Configure Smart Licensing Using Policy: Workflows by Topology , on page 57. The reporting method you can use depends on the topology you implement.

Migrating to a Version of SSM On-Prem That Supports Smart Licensing Using Policy

If you are using a version of SSM On-Prem that is earlier than the minimum required version (See SSM On-Prem, on page 37), you can use this section as an outline of the process and sequence you have to follow to migrate the SSM On-Prem version and the product instance.

1. Upgrade SSM On-Prem.

   Upgrade to the minimum required Version 8, Release 202102 or a later version.

   Refer to the Cisco Smart Software Manager On-Prem Migration Guide.

2. Upgrade the product instance.

   For information about the minimum required software version, see SSM On-Prem, on page 37.

   For information about the upgrade procedure, see Upgrading the Wireless Controller Software, on page 70.

3. Re-Register a local account with CSSM

   Online and Offline options are available. Refer to the Cisco Smart Software Manager On-Prem Migration Guide > *Re-Registering a local Account (Online Mode)* or *Manually Re-Registering a Local Account (Offline Mode)* .

   Once re-registration is complete, the following events occur automatically:

   - SSM On-Prem responds with new transport URL that points to the tenant in SSM On-Prem.
   - The transport type configuration on the product instance changes from from **call-home** or **smart**, to **cslu**. The transport URL is also updated automatically.

4. Save configuration changes on the product instance, by entering the **copy running-config startup-config** command in privileged EXEC mode.

5. Clear older On-Prem Smart Licensing certificates on the product instance and reload the product instance. Do not save configuration changes after this.

Note This step is required only if the software version running on the product instance is Cisco IOS XE Amsterdam 17.3.x or Cisco IOS XE Bengaluru 17.4.x.

Enter the **licence smart factory reset** and then the **reload** commands in privileged EXEC mode.

```
Device# licence smart factory reset
Device# reload
```

6. Perform usage synchronization

   a. On the product instance, enter the **license smart sync** {**all**|**local**} command, in privileged EXEC mode. This synchronizes the product instance with SSM On-Prem, to send and receive any pending data.

      ```
      Device(config)# license smart sync local
      ```

      You can verify this in the SSM On-Prem UI. Go to **Inventory** > **SL Using Policy**. In the **Alerts** column, the following message is displayed: Usage report from product instance.

   b. Synchronize usage information with CSSM (*choose one*)

      - Option 1:

        SSM On-Prem is connected to CSSM: In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Reports** > **Usage Schedules** > **Synchronize now with Cisco**.

      - Option 2:

        SSM On-Prem is not connected to CSSM. See Exporting and Importing Usage Data (SSM On-Prem UI), on page 104.

Result:

You have completed migration and initial usage synchronization. Product instance and license usage information is now displayed in SSM On-Prem.

For subsequent reporting, you have the following options:

- To synchronize data between the product instance and SSM On-Prem:
  - Schedule periodic synchronization between the product instance and SSM On-Prem, by configuring the reporting interval. Enter the **license smart usage interval** *interval_in_days* command in global configuration mode.

    To know when the product instance will be sending the next RUM report, enter the **show license all** command in privileged EXEC mode and in the output, check the `Next report push:` field.

  - Enter the **license smart sync** privileged EXEC command, for ad hoc or on-demand synchronization between the product instance and SSM On-Prem.

- To synchronize usage information with CSSM:
  - Schedule periodic synchronization with CSSM. In the SSM On-Prem UI, navigate to **Reports** > **Usage Schedules** > **Synchronization schedule with Cisco**. Enter the following frequency information and save:
    - **Days**: Refers to how *often* synchronization occurs. For example, if you enter 2, synchronization occurs once every two days.
    - **Time of Day:**: Refers to the time at which synchronization occurs, in the 24-hour notation system. For example, if you enter 14 hours and 0 minutes, synchronization occurs at 2 p.m. (1400) in your local time zone.

- Upload and download the required files for reporting. See Exporting and Importing Usage Data (SSM On-Prem UI), on page 104.

Task Library for Smart Licensing Using Policy

This section is a grouping of tasks that apply to Smart Licensing Using Policy. It includes tasks performed on a product instance, on the CSLU interface, and on the CSSM Web UI.

To implement a particular topology, refer to the corresponding workflow to know the sequential order of tasks that apply. See How to Configure Smart Licensing Using Policy: Workflows by Topology , on page 57.

To perform any additional configuration tasks, for instance, to configure a different license, or use an add-on license, or to configure a narrower reporting interval, refer to the corresponding task here. Check the "Supported Topologies" where provided, before you proceed.

Logging into Cisco (CSLU Interface)

Depending on your needs, when working in CSLU, you can either be in connected or disconnected mode. To work in the connected mode, complete these steps to connect with Cisco.

Procedure

Step 1 From the CSLU Main screen, click **Login to Cisco** (located at the top right corner of the screen).

Step 2 Enter: **CCO User Name** and **CCO Password**.

Step 3 In the CSLU Preferences tab, check that the Cisco connectivity toggle displays “Cisco Is Available”.

Configuring a Smart Account and a Virtual Account (CSLU Interface)

Both the Smart Account and Virtual Account are configured through the Preferences tab. Complete the following steps to configure both Smart and Virtual Accounts for connecting to Cisco.

Procedure

Step 1 Select the **Preferences Tab** from the CSLU home screen.

Step 2 Perform these steps for adding both a Smart Account and Virtual Account:

a) In the Preferences screen navigate to the **Smart Account** field and add the **Smart Account Name**.

b) Next, navigate to the **Virtual Account** field and add the **Virtual Account Name**.

If you are connected to CSSM (In the Preferences tab, **Cisco is Available**), you can select from the list of available SA/VAs.

If you are not connected to CSSM (In the Preferences tab, **Cisco Is Not Available**), enter the SA/VAs manually.

Note SA/VA names are case sensitive.

Step 3 Click **Save**. The SA/VA accounts are saved to the system

Only one SA/VA pair can reside on CSLU at a time. You cannot add multiple accounts. To change to another SA/VA pair, repeat Steps 2a and 2b then Save. A new SA/VA account pair replaces the previous saved pair

Adding a Product-Initiated Product Instance in CSLU (CSLU Interface)

Complete these steps to add a device-created Product Instance using the Preferences tab.

Procedure

Step 1 Select the **Preferences** tab.

Step 2 In the Preferences screen, de-select the **Validate Device** check box.

Step 3 Set the **Default Connect Method** to **Product Instance Initiated** and then click **Save**.

Ensuring Network Reachability for Product Instance-Initiated Communication

This task provides *possible* configurations that may be required to ensure network reachability for product instance-initiated communication. Steps marked as "(Required)" are required for all product instances, all other steps my be required or optional, depending the kind of product instance and network requirements. Configure the applicable commands:

Before you begin

Supported topologies: Connected to CSSM Through CSLU (product instance-initiated communication).

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-type-number* **Example:** `Device (config)# interface gigabitethernet0/0`	Enters interface configuration mode and specifies the Ethernet interface, subinterface, or VLAN to be associated with the VRF.

	Command or Action	Purpose
Step 4	**vrf forwarding** *vrf-name* **Example:** `Device(config-if)# vrf forwarding Mgmt-vrf`	Associates the VRF with the Layer 3 interface. This command activates multiprotocol VRF on an interface
Step 5	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 192.168.0.1 255.255.0.0`	Defines the IP address for the VRF.
Step 6	**negotiation auto** **Example:** `Device(config-if)# negotiation auto`	Enables auto-negotiation operation for the speed and duplex parameters of an interface. **Note** Cisco Catalyst 9800-L-F Wireless Controller 10G Ports do not support in an auto-negotiation operation.
Step 7	**end** **Example:** `Device(config-if)# end`	Exits the interface configuration mode and enters global configuration mode.
Step 8	**ip http client source-interface** *interface-type-number* **Example:** `Device(config)# ip http client source-interface gigabitethernet0/0`	Configures a source interface for the HTTP client.
Step 9	**ip route** *ip-address ip-mask subnet mask* **Example:** `Device(config)# ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1`	(Required) Configures a route and gateway on the product instance. You can configure either a static route or a dynamic route.
Step 10	{**ip** | **ipv6**} **name-server** *server-address 1 ...server-address 6*] **Example:** `Device(config)# Device(config)# ip name-server vrf mgmt-vrf 173.37.137.85`	Configures Domain Name System (DNS) on the VRF interface.
Step 11	**ip domain lookup source-interface** *interface-type-number* **Example:** `Device(config)# ip domain lookup source-interface gigabitethernet0/0`	Configures the source interface for the DNS domain lookup.

	Command or Action	Purpose
Step 12	**ip domain name** *domain-name* **Example:** `Device(config)# ip domain name example.com`	Configure DNS discovery of your domain. In accompanying example, the name-server creates entry `cslu-local.example.com`.

Adding a CSLU-Initiated Product Instance in CSLU (CSLU Interface)

Using the CSLU interface, you can configure the connect method to be CSLU Initiated. This connect method (mode) enables CSLU to retrieve Product Instance information from the Product Instance.

Note The default Connect Method is set in the **Preferences** tab.

Complete these steps to add a Product Instance from the Inventory tab

Procedure

Step 1 Go to the **Inventory** tab and from the Product Instances table, select **Add Single Product**.

Step 2 Enter the **Host** (IP address of the Host).

Step 3 Select the **Connect Method** and select one of the CSLU Initiated connect methods.

Step 4 In the right panel, click **Product Instance Login Credentials**. The left panel of the screen changes to show the User Name and Password fields.

Step 5 Enter the product instance **User Name** and **Password**.

Step 6 Click **Save**.

The information is saved to the system and the device is listed in the Product Instances table with the Last Contact listed as never.

Collecting Usage Reports: CSLU Initiated (CSLU Interface)

CSLU also allows you to manually trigger the gathering of usage reports from devices.

After configuring and selecting a product instance (selecting **Add Single Product**, filling in the **Host** name and selecting a CSLU-initiated connect method), click **Actions for Selected** > **Collect Usage**. CSLU connects to the selected product instances and collects the usage reports. These usage reports are stored in CSLU's local library. These reports can then be transferred to Cisco if CSLU is connected to Cisco, or (if you are not connected to Cisco) you can manually trigger usage collection by selecting **Data** > **Export to CSSM**.

If you are working in CSLU-initiated mode, complete these steps to configure CSLU to collect RUM reports from Product Instances.

Procedure

Step 1 Click the **Preference** tab and enter a valid **Smart Account** and **Virtual Account**, and then selectan appropriate CSLU-initiated collect method. (If there have been any changes in Preferences, make sure you click **Save**).

Step 2 Click the **Inventory** tab and select one or more product instances.

Step 3 Click **Actions for Selected** > **Collect Usage**.

RUM reports are retrieved from each selected device and stored in the CSLU local library. The Last Contacted column is updated to show the time the report was received, and the Alerts column shows the status.

If CSLU is currently logged into Cisco the reports will be automatically sent to the associated Smart Account and Virtual Account in Cisco and Cisco will send an acknowledgement to CSLU as well as to the product instance. The acknowledgement will be listed in the alerts column of the Product Instance table. To manually transfer usage reports Cisco, from the CSLU main screen select **Data** > **Export to CSSM**.

Step 4 From the **Export to CSSM** modal, select the local directory where the reports are to be stored. (<CSLU_WORKING_Directory>/data/default/rum/unsent)

At this point, the usage reports are saved in your local directory (library). To upload these usage reports to Cisco, follow the steps described in Uploading Data or Requests to CSSM and Downloading a File, on page 123.

Note The Windows operating system can change the behavior of a usage report file properties by dropping the extension when that file is renamed. The behavior change happens when you rename the downloaded file and the renamed file drops the extension. For example, the downloaded default file named `UD_xxx.tar` is renamed to `UD_yyy`. The file loses its TAR extension and cannot function. To enable the usage file to function normally, after re-naming a usage report file, you must also add the TAR extension back to the file name, for example `UD_yyy.tar`.

Export to CSSM (CSLU Interface)

The Download All for Cisco menu option is a manual process used for offline purposes. Complete these steps to use the Download For Cisco menu option

Procedure

Step 1 Go to the **Preferences** tab, and turn off the **Cisco Connectivity** toggle switch.

The field switches to “Cisco Is Not Available”.

Step 2 From the main menu in the CSLU home screen navigate to **Data** > **Export to CSSM**.

Step 3 Select the file from the modal that opens and click **Save**. You now have the file saved.

Note At this point you have a DLC file, RUM file, or both.

Step 4 Go to a station that has connectivity to Cisco, and complete the following: Uploading Data or Requests to CSSM and Downloading a File, on page 123

Once the file is downloaded, you can import it into CSLU, see Import from CSSM (CSLU Interface), on page 96.

Import from CSSM (CSLU Interface)

Once you have received the ACK or other file (such as an authorization code) from Cisco, you are ready to Upload that file to your system. This procedure can be used for workstations that are offline. Complete these steps to select and upload files from Cisco.

Procedure

Step 1 Ensure that you have downloaded the file to a location that is accessible to CSLU.

Step 2 From the main menu in the CSLU home screen, navigate to **Data** > **Import from CSSM**.

Step 3 An Import from CSSM modal open for you to either:

- Drag and Drop a file that resides on your local drive, or
- Browse for the appropriate *.xml file, select the file and click **Open**.

If the upload is successful, you will get message indicating that the file was successfully sent to the server. If the upload is not successful, you will get an import error.

Step 4 When you have finished uploading, click the **x** at the top right corner of the modal to close it.

Ensuring Network Reachability for CSLU-Initiated Communication

This task provides *possible* configurations that may be required to ensure network reachability for CSLU-initiated communication. Steps marked as "(Required)" are required for all product instances, all other steps may be required or optional, depending the kind of product instance and network requirements. Configure the applicable commands:

Before you begin

Supported topologies: Connected to CSSM Through CSLU (CSLU-initiated communication).

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**aaa new model** **Example:** `Device(config)# aaa new model`	(Required) Enable the authentication, authorization, and accounting (AAA) access control model.
Step 4	**aaa authentication login default local** **Example:** `Device(config)# aaa authentication login default local`	(Required) Sets AAA authentication to use the local username database for authentication.
Step 5	**aaa authorization exec default local** **Example:** `Device(config)# aaa authorization exec default local`	Sets the parameters that restrict user access to a network. The user is allowed to run an EXEC shell.
Step 6	ip routing **Example:** `Device(config)# ip routing`	Enables IP routing.
Step 7	{**ip** | **ipv6**} **name-server** *server-address 1 ...server-address 6*] **Example:** `Device(config)# ip name-server vrf Mgmt-vrf 192.168.1.100 192.168.1.200 192.168.1.300`	(Optional) 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 device sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 8	**ip domain lookup source-interface** *interface-type-number* **Example:** `Device(config)# ip domain lookup source-interface gigabitethernet0/0`	Enables DNS-based hostname-to-address translation on your device. 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 9	**ip domain name** *name* **Example:** `Device(config)# ip domain name vrf Mgmt-vrf cisco.com`	Defines a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name).
Step 10	**no username** *name* **Example:**	(Required) Clears the specified username, if it exists. For *name* , enter the same username you will create in the next step. This ensures

	Command or Action	Purpose
	`Device(config)# no username admin`	that a duplicate of the username you are going to create in the next step does not exist. If you plan to use REST APIs for CSLU-initiated retrieval of RUM reports, you have to log in to CSLU. Duplicate usernames may cause the feature to work incorrectly if there are duplicate usernames in the system.

Step 11	**username** *name* **privilege** *level* **password** *password*
	Example:
	`Device(config)# username admin privilege 15`
	`password 0 lab`
		(Required) Establishes a username-based authentication system.
		The **privilege** keyword sets the privilege level for the user. A number between 0 and 15 that specifies the privilege level for the user.
		The password allows access to the name argument. A password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
		This enables CSLU to use the product instance native REST.
		Note Enter this username and password in CSLU (Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94 → *Step 4. f.* CSLU can then collect RUM reports from the product instance.

Step 12	**interface** *interface-type-number* **Example:** `Device (config)# interface gigabitethernet0/0`	Enters interface configuration mode and specifies the Ethernet interface, subinterface, or VLAN to be associated with the VRF.
Step 13	**vrf forwarding** *vrf-name* **Example:** `Device(config-if)# vrf forwarding Mgmt-vrf`	Associates the VRF with the Layer 3 interface. This command activates multiprotocol VRF on an interface
Step 14	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 192.168.0.1 255.255.0.0`	Defines the IP address for the VRF.
Step 15	**negotiation auto** **Example:** `Device(config-if)# negotiation auto`	Enables auto-negotiation operation for the speed and duplex parameters of an interface.

	Command or Action	Purpose
Step 16	**no shutdown** **Example:** `Device(config-if)# no shutdown`	Restarts a disabled interface.
Step 17	**end** **Example:** `Device(config-if)# end`	Exits the interface configuration mode and enters global configuration mode.
Step 18	**ip http server** **Example:** `Device(config)# ip http server`	(Required) Enables the HTTP server on your IP or IPv6 system, including a Cisco web browser user interface. The HTTP server uses the standard port 80, by default.

Step 19	**ip http authentication local**
	Example:
	`ip http authentication local`
	`Device(config)#`
		(Required) Specifies a particular authentication method for HTTP server users.
		The **local** keyword means that the login user name, password and privilege level access combination specified in the local system configuration (by the username global configuration command) should be used for authentication and authorization.

Step 20	**ip http secure-server** **Example:** `Device(config)# ip http server`	(Required) Enables a secure HTTP (HTTPS) server. The HTTPS server uses the Secure Sockets Layer (SSL) version 3.0 protocol.
Step 21	**ip http max-connections** **Example:** `Device(config)# ip http max-connections 16`	(Required) Configures the maximum number of concurrent connections allowed for the HTTP server. Enter an integer in the range from 1 to 16. The default is 5.
Step 22	**ip tftp source-interface** *interface-type-number* **Example:** `Device(config)# ip tftp source-interface GigabitEthernet0/0`	Specifies the IP address of an interface as the source address for TFTP connections.
Step 23	**ip route** *ip-address ip-mask subnet mask* **Example:** `Device(config)# ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1`	Configures a route and gateway on the product instance. You can configure either a static route or a dynamic route.

Step 24	**logging host**
	Example:
	`Device(config)# logging host 172.25.33.20`
	`vrf Mgmt-vrf`
		Logs system messages and debug output to a remote host.

	Command or Action	Purpose
Step 25	**end** **Example:** `Device(config)# end`	Exits the global configuration mode and enters priveleged EXEC mode.
Step 26	**show ip http server session-module** **Example:** `Device# show ip http server session-module`	(Required) Verifies HTTP connectivity. In the output, check that `SL_HTTP` is active. Additionally, you can also perform the following checks : • From device where CSLU is installed, verify that you can ping the product instance. A successful ping confirms that the product instance is reachable. • From a Web browser on the device where CSLU is installed verify `https://<product-instance-ip>/`. This ensures that the REST API from CSLU to the product instance works as expected.

Assigning a Smart Account and Virtual Account (SSM On-Prem UI)

You can use this procedure to import one or more product instances along with corresponding Smart Account and Virtual Account information, into the SSM On-Prem database. This enables SSM On-Prem to map product instances that are part of local virtual accounts (other than the default local virtual account), to the correct license pool in CSSM:

Before you begin

Supported topologies: SSM On-Prem Deployment (product instance-initiated communication).

Procedure

Step 1 Log into the SSM On-Prem and select the **Smart Licensing** workspace.

Step 2 Navigate to **Inventory** > **SL Using Policy** > **Export/Import All** > **Import Product Instances List**

The **Upload Product Instances** window is displayed.

Step 3 Click **Download** to download the .csv template file and enter the required information for all the product instances in the template.

Step 4 Once you have filled-out the template, click **Inventory** > **SL Using Policy** > **Export/Import All** > **Import Product Instances List**.

The **Upload Product Instances** window is displayed.

Step 5 Now, click **Browse** and upload the filled-out .csv template.

Smart Account and Virtual Account information for all uploaded product instances is now available in SSM On-Prem.

Validating Devices (SSM On-Prem UI)

When device validation is enabled, RUM reports from an unknown product instance (not in the SSM On-Prem database) are rejected.

By default, devices are not validated. Complete the following steps to enable it:

Before you begin

Supported topologies: SSM On-Prem Deployment (product instance-initiated communication).

Procedure

Step 1 In the **On-Prem License Workspace** window, click **Admin Workspace** and log in, if prompted.

The **On-Prem Admin Workspace** window is displayed.

Step 2 Click the **Settings** widget.

The **Settings** window is displayed.

Step 3 Navigate to the **CSLU** tab and turn-on the **Validate Device** toggle switch.

RUM reports from an unknown product instance will now be rejected. If you haven't already, you must now add the required product instances to the SSM On-Prem database before sending RUM reports. See Assigning a Smart Account and Virtual Account (SSM On-Prem UI), on page 100

Ensuring Network Reachability for Product Instance-Initiated Communication

This task provides *possible* configurations that may be required to ensure network reachability for product instance-initiated communication. Steps marked as "(Required)" are required for all product instances, all other steps my be required or optional, depending the kind of product instance and network requirements. Configure the applicable commands:

Note Ensure that you configure steps 13, 14, and 15 exactly as shown below. These commands must be configured to ensure that the correct trustpoint is used and that the necessary certificates are accepted for network reachability.

Before you begin

Supported topologies: SSM On-Prem Deployment(product instance-initiated communication).

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-type-number* **Example:** `Device (config)# interface gigabitethernet0/0`	Enters interface configuration mode and specifies the Ethernet interface, subinterface, or VLAN to be associated with the VRF.
Step 4	**vrf forwarding** *vrf-name* **Example:** `Device(config-if)# vrf forwarding Mgmt-vrf`	Associates the VRF with the Layer 3 interface. This command activates multiprotocol VRF on an interface
Step 5	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 192.168.0.1 255.255.0.0`	Defines the IP address for the VRF.
Step 6	**negotiation auto** **Example:** `Device(config-if)# negotiation auto`	Enables auto-negotiation operation for the speed and duplex parameters of an interface.
Step 7	**end** **Example:** `Device(config-if)# end`	Exits the interface configuration mode and enters global configuration mode.
Step 8	**ip http client source-interface** *interface-type-number* **Example:** `Device(config)# ip http client source-interface gigabitethernet0/0`	Configures a source interface for the HTTP client.
Step 9	**ip route** *ip-address ip-mask subnet mask* **Example:** `Device(config)# ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1`	(Required) Configures a route and gateway on the product instance. You can configure either a static route or a dynamic route.

	Command or Action	Purpose
Step 10	{**ip** | **ipv6**} **name-server** *server-address 1 ...server-address 6*] **Example:** `Device(config)# Device(config)# ip name-server vrf mgmt-vrf 198.51.100.1`	Configures Domain Name System (DNS) on the VRF interface.
Step 11	**ip domain lookup source-interface** *interface-type-number* **Example:** `Device(config)# ip domain lookup source-interface gigabitethernet0/0`	Configures the source interface for the DNS domain lookup.
Step 12	**ip domain name** *domain-name* **Example:** `Device(config)# ip domain name example.com`	Configure DNS discovery of your domain. In the accompanying example, the name-server creates entry `cslu-local.example.com`.

Step 13	**crypto pki trustpoint SLA-TrustPoint**
	Example:
	`Device(config)# crypto pki trustpoint SLA-TrustPoint`
	`Device(ca-trustpoint)#`
		(Required) Declares that the product instance should use trustpoint "SLA-TrustPoint" and enters the ca-trustpoint configuration mode. The product instance does not recognize any trustpoints until you declare a trustpoint using this command.

Step 14	enrollment terminal **Example:** `Device(ca-trustpoint)# enrollment terminal`	Required) Specifies the certificate enrollment method.
Step 15	**revocation-check none** **Example:** `Device(ca-trustpoint)# revocation-check none`	(Required) Specifes a method that is to be used to ensure that the certificate of a peer is not revoked. For the SSM On-Prem Deployment topology, enter the **none** keyword. This means that a revocation check will not be performed and the certificate will always be accepted.

Step 16	**exit**
	Example:
	`Device(ca-trustpoint)# exit`
	`Device(config)# exit`
		Exits the ca-trustpoint configuration mode and then the global configuration mode and returns to privileged EXEC mode.

Step 17	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.

Retrieving the Transport URL (SSM On-Prem UI)

You must configure the transport URL on the product instance when you deploy the product instance-initiated communication with SSM On-Prem deployment. This task show you how to easily copy the complete URL including the tenant ID from SSM On-Prem.

Before you begin

Supported topologies: SSM On-Prem Deployment (product instance-initiated communication).

Procedure

Step 1 Log into SSM On-Prem and select the **Smart Licensing** workspace.

Step 2 Navigate to the **Inventory** tab and from the dropdown list of local virtual accounts (top right corner), select the *default local virtual account*. When you do, the area under the **Inventory** tab displays **Local Virtual Account: Default**.

Step 3 Navigate to the **General** tab.

The **Product Instance Registration Tokens** area is displayed.

Step 4 In the **Product Instance Registration Tokens** area click **CSLU Transport URL**.

The **Product Registration URL** pop-window is displayed.

Step 5 Copy the entire URL and save it in an accessible place.

You will require the URL when you configure the transport type and URL on the product instance.

Step 6 Configure the transport type and URL. See: Setting the Transport Type, URL, and Reporting Interval, on page 125.

Exporting and Importing Usage Data (SSM On-Prem UI)

You can use this procedure to complete usage synchronization between SSM On-Prem and CSSM when SSM On-Prem is disconnected from CSSM.

Before you begin

Supported topologies:

- SSM On-Prem Deployment (SSM On-Prem-initiated communication)
- SSM On-Prem Deployment (product instance-initiated communication).

Reporting data must be available in SSM On-Prem. You must have either pushed the nessary reporting data from the product instance to SSM On-Prem (product instance-initiated communication) or retrieved the necessary reporting data from the product instance (SSM On-Prem-initiated communication).

Procedure

Step 1 Log into SSM On-Prem and select **Smart Licensing**.

Step 2 Navigate to **Inventory** > **SL Using Policy** tab.

Step 3 In the **SL Using Policy** tab area, click **Export/Import All...** > **Export Usage to Cisco**.

This generates one .tar file with *all* the usage reports available in the SSM On-Prem server.

Step 4 Complete this task in CSSM: Uploading Data or Requests to CSSM and Downloading a File, on page 123.

At the end of this task you will have an ACK file to import into SSM On-Prem.

Step 5 Again navigate to the **Inventory** > **SL Using Policy** tab.

Step 6 In the **SL Using Policy** tab area, click **Export/Import All...** > **Import From Cisco** . Upload the .tar ACK file.

To verify ACK import, in the **SL Using Policy** tab area check the **Alerts** column of the corresponding product instance. The following message is displayed: Acknowledgement received from CSSM.

Adding One or More Product Instances (SSM On-Prem UI)

You can use this procedure to add one product instance or to import and add multiple product instances. It enables SSM On-Prem to retrieve information from the product instance.

Before you begin

Supported topologies: SSM On-Prem Deployment (SSM On-Prem-initiated communication).

Procedure

Step 1 Log into the SSM On-Prem UI and click **Smart Licensing**.

Step 2 Navigate to **Inventory** tab. Select a local virtual account from the drop-down list in the top right corner.

Step 3 Navigate to the **SL Using Policy** tab.

Step 4 Add a single product or import multiple product instances (*choose one*).

- **To add a single product instance**:
  - **a.** In the **SL Using Policy** tab area, click **Add Single Product**.
  - **b.** In the **Host** field, enter the IP address of the host (product instance).
  - **c.** From the **Connect Method** dropdown list, select an appropriate SSM On-Prem-initiated connect method.

    The available connect methods for SSM On-Prem-initiated communication are: NETCONF, RESTCONF, and REST API.
  - **d.** In the right panel, click **Product Instance Login Credentials**.

    The **Product Instance Login Credentials** window is displayed

Note You need the login credentials only if a product instance requires a SLAC.

e. Enter the **User ID** and **Password**, and click **Save**.

This is the same user ID and password that you configured as part of commands required to establish network reachability (Ensuring Network Reachability for SSM On-Prem-Initiated Communication, on page 106).

Once validated, the product instance is displayed in the listing in the **SL Using Policy** tab area.

- **To import multiple product instances:**

a. In **SL Using Policy** tab, click **Export/Import All...** > **Import Product Instances List**.

The **Upload Product Instances** window is displayed.

b. Click **Download** to download the predefined .csv template.

c. Enter the required information for all the product instances in the .csv template.

In the template, ensure that you provide **Host**, **Connect Method** and **Login Credentials** for all product instances.

The available connect methods for SSM On-Prem-initiated communication are: NETCONF, RESTCONF, and REST API.

Login credentials refer to the user ID and password that you configured as part of commands required to establish network reachability (Ensuring Network Reachability for SSM On-Prem-Initiated Communication, on page 106).

d. Again navigate to **Inventory** > **SL Using Policy** tab. Click **Export/Import All....** > **Import Product Instances List**.

The **Upload Product Instances** window is displayed.

e. Now upload the filled-out .csv template.

Once validated, the product instances are displayed in the listing in the **SL Using Policy** tab.

Ensuring Network Reachability for SSM On-Prem-Initiated Communication

This task provides *possible* configurations that may be required to ensure network reachability for SSM On-Prem-initiated communication. Steps marked as "(Required)" are required for all product instances, all other steps may be required or optional, depending the kind of product instance and network requirements. Configure the applicable commands:

Note Ensure that you configure steps 25, 26, and 27 exactly as shown below. These commands must be configured to ensure that the correct trustpoint is used and that the necessary certificates are accepted for network reachability.

Before you begin

Supported topologies: SSM On-Prem Deployment (SSM On-Prem-initiated communication).

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa new model** **Example:** `Device(config)# aaa new model`	(Required) Enable the authentication, authorization, and accounting (AAA) access control model.
Step 4	**aaa authentication login default local** **Example:** `Device(config)# aaa authentication login default local`	(Required) Sets AAA authentication to use the local username database for authentication.
Step 5	**aaa authorization exec default local** **Example:** `Device(config)# aaa authorization exec default local`	Sets the parameters that restrict user access to a network. The user is allowed to run an EXEC shell.
Step 6	ip routing **Example:** `Device(config)# ip routing`	Enables IP routing.
Step 7	{**ip** | **ipv6**} **name-server** *server-address 1 ...server-address 6*] **Example:** `Device(config)# ip name-server vrf Mgmt-vrf 192.168.1.100 192.168.1.200 192.168.1.300`	(Optional) 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 device sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 8	**ip domain lookup source-interface** *interface-type-number* **Example:** `Device(config)# ip domain lookup source-interface gigabitethernet0/0`	Enables DNS-based hostname-to-address translation on your device. This feature is enabled by default. If your network devices require connectivity with devices in networks for which you do not

	Command or Action	Purpose
		control name assignment, you can dynamically assign device names that uniquely identify your devices by using the global Internet naming scheme (DNS).
Step 9	**ip domain name** *name* **Example:** `Device(config)# ip domain name vrf Mgmt-vrf cisco.com`	Defines a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name).
Step 10	**no username** *name* **Example:** `Device(config)# no username admin`	(Required) Clears the specified username, if it exists. For *name* , enter the same username you will create in the next step. This ensures that a duplicate of the username you are going to create in the next step does not exist. If you plan to use REST APIs for SSM On-Prem-initiated retrieval of RUM reports, you have to log in to SSM On-Prem. Duplicate usernames may cause the feature to work incorrectly if there are present in the system.

Step 11	**username** *name* **privilege** *level* **password** *password*
	Example:
	`Device(config)# username admin privilege 15`
	`password 0 lab`
		(Required) Establishes a username-based authentication system.
		The **privilege** keyword sets the privilege level for the user. A number between 0 and 15 that specifies the privilege level for the user.
		The password allows access to the name argument. A password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command.
		This enables SSM On-Prem to use the product instance native REST.
		Note Enter this username and password in SSM On-Prem (Adding One or More Product Instances (SSM On-Prem UI), on page 105). This enables SSM On-Prem to collect RUM reports from the product instance.

Step 12	**interface** *interface-type-number* **Example:** `Device (config)# interface gigabitethernet0/0`	Enters interface configuration mode and specifies the Ethernet interface, subinterface, or VLAN to be associated with the VRF.

	Command or Action	Purpose
Step 13	**vrf forwarding** *vrf-name* **Example:** `Device(config-if)# vrf forwarding Mgmt-vrf`	Associates the VRF with the Layer 3 interface. This command activates multiprotocol VRF on an interface
Step 14	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 192.168.0.1 255.255.0.0`	Defines the IP address for the VRF.
Step 15	**negotiation auto** **Example:** `Device(config-if)# negotiation auto`	Enables auto-negotiation operation for the speed and duplex parameters of an interface.
Step 16	**no shutdown** **Example:** `Device(config-if)# no shutdown`	Restarts a disabled interface.
Step 17	**end** **Example:** `Device(config-if)# end`	Exits the interface configuration mode and enters global configuration mode.
Step 18	**ip http server** **Example:** `Device(config)# ip http server`	(Required) Enables the HTTP server on your IP or IPv6 system, including a Cisco web browser user interface. The HTTP server uses the standard port 80, by default.

Step 19	**ip http authentication local**
	Example:
	`ip http authentication local`
	`Device(config)#`
		(Required) Specifies a particular authentication method for HTTP server users.
		The **local** keyword means that the login user name, password and privilege level access combination specified in the local system configuration (by the username global configuration command) should be used for authentication and authorization.

Step 20	**ip http secure-server** **Example:** `Device(config)# ip http server`	(Required) Enables a secure HTTP (HTTPS) server. The HTTPS server uses the Secure Sockets Layer (SSL) version 3.0 protocol.
Step 21	**ip http max-connections** **Example:** `Device(config)# ip http max-connections 16`	(Required) Configures the maximum number of concurrent connections allowed for the HTTP server. Enter an integer in the range from 1 to 16. The default is 5.

	Command or Action	Purpose
Step 22	**ip tftp source-interface** *interface-type-number* **Example:** `Device(config)# ip tftp source-interface GigabitEthernet0/0`	Specifies the IP address of an interface as the source address for TFTP connections.
Step 23	**ip route** *ip-address ip-mask subnet mask* **Example:** `Device(config)# ip route vrf mgmt-vrf 192.168.0.1 255.255.0.0 192.168.255.1`	Configures a route and gateway on the product instance. You can configure either a static route or a dynamic route.
Step 24	**logging host** **Example:** `Device(config)# logging host 172.25.33.20 vrf Mgmt-vrf`	Logs system messages and debug output to a remote host.

Step 25	**crypto pki trustpoint SLA-TrustPoint**
	Example:
	`Device(config)# crypto pki trustpoint SLA-TrustPoint`
	`Device(ca-trustpoint)#`
		(Required) Declares that the product instance should use trustpoint "SLA-TrustPoint" and enters the ca-trustpoint configuration mode. The product instance does not recognize any trustpoints until you declare a trustpoint using this command.

Step 26	enrollment terminal **Example:** `Device(ca-trustpoint)# enrollment terminal`	Required) Specifies the certificate enrollment method.
Step 27	**revocation-check none** **Example:** `Device(ca-trustpoint)# revocation-check none`	(Required) Specifes a method that is to be used to ensure that the certificate of a peer is not revoked. For the SSM On-Prem Deployment topology, enter the **none** keyword. This means that a revocation check will not be performed and the certificate will always be accepted.

Step 28	**end**
	Example:
	`Device(ca-trustpoint)# exit`
	`Device(config)# end`
		Exits the ca-trustpoint configuration mode and then the global configuration mode and returns to privileged EXEC mode.

Step 29	**show ip http server session-module** **Example:** `Device# show ip http server session-module`	(Required) Verifies HTTP connectivity. In the output, check that `SL_HTTP` is active. Additionally, you can also perform the following checks : • From device where SSM On-Prem is installed, verify that you can ping the product instance. A successful ping

	Command or Action	Purpose
		confirms that the product instance is reachable. • From a Web browser on the device where SSM On-Prem is installed verify `https://<product-instance-ip>/`. This ensures that the REST API from SSM On-Prem to the product instance works as expected.
Step 30	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.

Setting Up a Connection to CSSM

The following steps show how to set up a Layer 3 connection to CSSM to verify network reachability. Steps marked as "(Required)" are required for all product instances, all other steps may be required or optional, depending the kind of product instance and network requirements. Configure the applicable commands:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	{**ip** | **ipv6**} **name-server** *server-address 1 ...server-address 6*] **Example:** `Device(config)# ip name-server 209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230`	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 device sends DNS queries to the primary server first. If that query fails, the backup servers are queried.
Step 4	**ip name-server vrf Mgmt-vrf** *server-address 1...server-address 6* **Example:** `Device(config)# ip name-server vrf Mgmt-vrf`	(Optional) Configures DNS on the VRF interface. You can specify up to six name servers. Separate each server address with a space.

	Command or Action	Purpose
	`209.165.201.1 209.165.200.225 209.165.201.14 209.165.200.230`	**Note** This command is an alternative to the **ip name-server** command.
Step 5	**ip domain lookup source-interface** *interface-type interface-number* **Example:** `Device(config)# ip domain lookup source-interface Vlan100`	Configures the source interface for the DNS domain lookup.
Step 6	**ip domain name** *domain-name* **Example:** `Device(config)# ip domain name example.com`	Configures the domain name.
Step 7	**ip host tools.cisco.com** *ip-address* **Example:** `Device(config)# ip host tools.cisco.com 209.165.201.30`	Configures static hostname-to-address mappings in the DNS hostname cache if automatic DNS mapping is not available.

Step 8	**interface** *interface-type-number*
	Example:
	`Device(config)# interface Vlan100`
	`Device(config-if)# ip address 192.0.2.10 255.255.255.0`
	`Device(config-if)# exit`
		Configures a Layer 3 interface. Enter an interface type and number or a VLAN.

Step 9	**ntp server** *ip-address* [**version** *number*] [**key** *key-id*] [**prefer**] **Example:** `Device(config)# ntp server 198.51.100.100 version 2 prefer`	(Required) Activates the NTP service (if it has not already been activated) and enables the system to synchronize the system software clock with the specified NTP server. This ensures that the device time is synchronized with CSSM. Use the **prefer** keyword if you need to use this command multiple times and you want to set a preferred server. Using this keyword reduces switching between servers.

Step 10	**switchport access vlan** *vlan_id*
	Example:
	`Device(config)# interface GigabitEthernet1/0/1`
	`Device(config-if)# switchport access`
		Enables the VLAN for which this access port carries traffic and sets the interface as a nontrunking nontagged single-VLAN Ethernet interface.

	Command or Action	Purpose

	`vlan 100`
	`Device(config-if)# switchport mode access`
	`Device(config-if)# exit`
	`OR`
	`Device(config)#`
		Note This step is to be configured only if the switchport access mode is required. The **switchport access vlan** command may apply to Catalyst switching product instances, for example, and for routing product instances you may want to configure the **ip address** *ip-address mask* command instead.

Step 11	**ip route** *ip-address ip-mask subnet mask* **Example:** `Device(config)# ip route 192.0.2.0 255.255.255.255 192.0.2.1`	Configures a route on the device. You can configure either a static route or a dynamic route.
Step 12	**ip http client source-interface** *interface-type-number* **Example:** `Device(config)# ip http client source-interface Vlan100`	(Required) Configures a source interface for the HTTP client. Enter an interface type and number or a VLAN.
Step 13	**exit** **Example:** `Device(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.
Step 14	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.

Configuring Smart Transport Through an HTTPs Proxy

To use a proxy server to communicate with CSSM when using the Smart transport mode, complete the following steps:

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**license smart transport smart** **Example:** `Device(config)# license smart transport smart`	Enables Smart transport mode.
Step 4	**license smart url default** **Example:** `Device(config)# license smart transport default`	Automatically configures the Smart URL (https://smartreceiver.cisco.com/licservice/license). For this option to work as expected, the transport mode in the previous step must be configured as **smart**.

Step 5	**license smart proxy** {**address** *address_hostname* | **port** *port_num*}
	Example:
	`Device(config)# license smart proxy address 192.168.0.1`
	`Device(config)# license smart proxy port 3128`
		Configures a proxy for the Smart transport mode. When a proxy is configured, licensing messages are sent to the proxy along with the final destination URL (CSSM). The proxy sends the message on to CSSM. Configure the proxy address and port number separately:
		• **address** *address_hostname*: Specifies the proxy address. Enter the IP address or hostname of the proxy server.
		• **port** *port_num*: Specifies the proxy port. Enter the proxy port number.

Configuring the Call Home Service for Direct Cloud Access

The Call Home service provides email-based and web-based notification of critical system events to CSSM. To configure the transport mode, enable the Call Home service, and configure a destination profile (A destination profile contains the required delivery information for an alert notification. At least one destination profile is required.), complete the following steps:

Note All steps are required unless specifically called-out as "(Optional)".

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**license smart transport callhome** **Example:** `Device(config)# license smart transport callhome`	Enables Call Home as the transport mode.
Step 4	**license smart url** *url* **Example:** `Device(config)# license smart url https://tools.cisco.com/its/service/oddce/services/DDCEService`	For the **callhome** transport mode, configure the CSSM URL exactly as shown in the example.
Step 5	**service call-home** **Example:** `Device(config)# service call-home`	Enables the Call Home feature.
Step 6	**call-home** **Example:** `Device(config)# call-home`	Enters Call Home configuration mode.
Step 7	**no http secure server-identity-check** **Example:** `Device(config-call-home)# no http secure server-identity-check`	Disables server identity check when HTTP connection is established.
Step 8	**contact-email-address** *email-address* **Example:** `Device(config-call-home)# contact-email-addr username@example.com`	Assigns customer's email address and enables Smart Call Home service full reporting capability and sends a full inventory message from Call-Home TAC profile to Smart Call Home server to start full registration process. You can enter up to 200 characters in email address format with no spaces.

Step 9	**profile** *name*
	Example:
	`Device(config-call-home)# profile CiscoTAC-1`
	`Device(config-call-home-profile)#`
		Enters the Call Home destination profile configuration submode for the specified destination profile.
		By default:
		• The CiscoTAC-1 profile is inactive. To use this profile with the Call Home service, you must enable the profile.
		• The CiscoTAC-1 profile sends a full report of all types of events subscribed in the profile. The alternative is to additionally configure `Device(cfg-call-home-profile)# anonymous-reporting-only`

	Command or Action	Purpose
		anonymous-reporting-only. When this is set, only crash, inventory, and test messages will be sent. Use the **show call-home profile all** command to check the profile status.
Step 10	**active** **Example:** `Device(config-call-home-profile)# active`	Enables the destination profile.

Step 11	**destination transport-method http**{**email** |**http**}
	Example:
	`Device(config-call-home-profile)# destination transport-method http`
	`AND`
	`Device(config-call-home-profile)# no destination transport-method email`
		Enables the message transport method. In the example, Call Home service is enabled via HTTP and transport via email is disabled.
		
		The **no** form of the command disables the method.

Step 12	**destination address** { **email** *email_address* |**http** *url*}
	Example:
	`Device(config-call-home-profile)# destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService`
	`AND`
	`Device(config-call-home-profile)# no destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService`
		Configures the destination e-mail address or URL to which Call Home messages are sent. When entering a destination URL, include either **http://** (default) or **https://**, depending on whether the server is a secure server.
		
		In the example provided here, a **http://** destination URL is configured; and the **no** form of the command is configured for **https://**.

Step 13	**exit** **Example:** `Device(config-call-home-profile)# exit`	Exits Call Home destination profile configuration mode and returns to Call Home configuration mode.
Step 14	**exit** **Example:** `Device(config-call-home)# end`	Exits Call Home configuration mode and returns to privileged EXEC mode.
Step 15	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.
Step 16	**show call-home profile** {*name* |**all**}	Displays the destination profile configuration for the specified profile or all configured profiles.

Configuring the Call Home Service for Direct Cloud Access through an HTTPs Proxy Server

The Call Home service can be configured through an HTTPs proxy server. This configuration requires no user authentication to connect to CSSM.

Note Authenticated HTTPs proxy configurations are not supported.

To configure and enable the Call Home service through an HTTPs proxy, complete the following steps:

Note All steps are required unless specifically called-out as "(Optional)".

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**license smart transport callhome** **Example:** `Device(config)# license smart transport callhome`	Enables Call Home as the transport mode.
Step 4	**service call-home** **Example:** `Device(config)# service call-home`	Enables the Call Home feature.
Step 5	**call-home** **Example:** `Device(config)# call-home`	Enters Call Home configuration mode.
Step 6	**http-proxy** *proxy-address* **proxy-port** *port-number* **Example:** `Device(config-call-home)# http-proxy 198.51.100.10 port 5000`	Configures the proxy server information to the Call Home service.

	Command or Action	Purpose
Step 7	**exit** **Example:** `Device(config-call-home)# exit`	Exits Call Home configuration mode and enters global configuration mode.
Step 8	**exit** **Example:** `Device(config)# exit`	Exits global configuration mode and enters privileged EXEC mode.
Step 9	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.

Removing and Returning an Authorization Code

To remove and return an SLR authorization code, complete the following steps.

Before you begin

Supported topologies: all

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**show license summary** **Example:** `Device# show license summary`	Ensure that the license that you want to remove and return is not in-use. If it is in-use, you must first disable the feature.

Step 3	**license smart authorization return** {**all** | **local**} {**offline** [*path*] | **online**}
	Example:
	`Device# license smart authorization return all online`
	
	`Enter this return code in Cisco Smart Software Manager portal:`
	`UDI: PID:C9800-CL-K9,SN:93BBAH93MGS`
	`Return code:`
	`CqaUPW-WSPYiq-ZNU2ci-SnWydS-hBCXHP-MuyPqy-PJ1GiG-tPTGQj-S2h`
	`UDI: PID:C9800-CL-K9,SN:9XECPSUU4XN`
	`Return code:`
	`CNLwxR-eWiAEJ-XaTEQg-j4rrYW-dSRz9j-37VpcP-`
		Returns an authorization code back to the license pool in CSSM. A return code is displayed after you enter this command.
		Specify the product instance:
		• **all**: Performs the action for all connected product instances in a High Availability set-up.
		• **local**: Performs the action for the active product instance. This is the default option.
		Specify if you are connected to CSSM or not:

	Command or Action	Purpose
	imjuLD-mNeA4k-TXA OR Device# license smart authorization return local offline Enter this return code in Cisco Smart Software Manager portal: UDI: PID:C9800-CL-K9,SN:93BBAH93MGS Return code: CqaUPW-WSPYiq-ZNU2ci-SnWydS-hBCXHP-MuyPqy-PJ1GiG-tPTGQj-S2h UDI: PID:C9800-CL-K9,SN:9XECPSUU4XN Return code: CNLwxR-eWiAEJ-XaTEQg-j4rrYW-dSRz9j-37VpcP-imjuLD-mNeA4k-TXA OR Device# license smart authorization return local offline bootflash:return-code.txt	• If connected to CSSM, enter **online**. The code is automatically returned to CSSM and a confirmation is returned and installed on the product instance. If you choose this option, the return code is automatically submitted to CSSM. • If not connected to CSSM, enter **offline**[*path*]. If you enter only the **offline** keyword, you must copy the return code that is displayed on the CLI and enter it in CSSM. If you specify a file name and path, the return code is saved in the specified location. The file format can be any readable format. For example: `Device# license smart authorization return local offline bootflash:return-code.txt`. For software versions Cisco IOS XE Cupertino 17.7.1 and later only: After you save the return request in a file, you can upload the file to CSSM in the same location and in the same way as you upload a RUM report: Uploading Data or Requests to CSSM and Downloading a File, on page 123. To enter the return code in CSSM, complete this task: Removing the Product Instance from CSSM, on page 120. Proceed with the next step only after you complete this step.
Step 4	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 5	**no license smart reservation** **Example:** `Device(config)# no license smart reservation`	Disables SLR configuration on the product instance. You must complete the authorization code return process in Step 3 above - whether online or offline, before you enter the no license smart reservation command in this step. Otherwise, the return may not be reflected in CSSM or in the show command, and you will have to contact your Cisco technical support representative to rectify the problem.

	Command or Action	Purpose
Step 6	**exit** **Example:** `Device(config)# exit`	Returns to privileged EXEC mode.

Step 7	**show license all**
	Example:
	`Device# show license all`
	`<output truncated>`
	`License Authorizations`
	`==========`
	`Overall status:`
	`Active: PID:C9800-CL-K9,SN:93BBAH93MGS`
	`Status: NOT INSTALLED`
	`Last return code:`
	`CopUPW-WSPYiq-ZNU2ci-SnWydS-hBCXHP-MuyPqy-PJlGiG-tPIGQj-S2h`
	`Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN`
	`Status: NOT INSTALLED`
	`Last return code:`
	`CNLwxR-eWiAEJ-XaIEQg-j4rrYW-dSRz9j-37VpcP-imjuLD-mNeA4k-IXA`
	`<output truncated>`
		Displays licensing information. Check the `License Authorizations` header in the output. If the return process is completed correctly, the `Last return code:` field displays the return code.

Removing the Product Instance from CSSM

To remove a product instance and return all licenses to the license pool, complete the following task:

Before you begin

Supported topologies: No Connectivity to CSSM and No CSLU

If you are removing a product instance that is using reserved licenses (SLR) ensure that you have generated a return code as shown in Removing and Returning an Authorization Code, on page 118. (Enter it in Step 7 in this task).

Procedure

Step 1 Log in to the CSSM Web UI at https://software.cisco.com and click **Smart Software Licensing**.

Log in using the username and password provided by Cisco.

Step 2 Click the **Inventory** tab.

Step 3 From the **Virtual Account** drop-down list, choose your Virtual Account.

Step 4 Click the **Product Instances** tab.

The list of product instances that are available is displayed.

Step 5 Locate the required product instance from the product instances list. Optionally, you can enter a name or product type string in the search tab to locate the product instance.

Step 6 In the **Actions** column of the product instance you want to remove, click the **Remove** link.

- If the product instance is *not* using a license with an SLR authorization code then the **Confirm Remove Product Instance** window is displayed.
- If the product instance *is* using a license with an SLR authorization code, then the **Remove Product Instance** window, with a field for return code entry is displayed.

Step 7 In the **Reservation Return Code** field, enter the return code you generated.

Note This step applies only if the product instance is using a license with an SLR authorization code.

Step 8 Click **Remove Product Instance**.

The license is returned to the license pool and the product instance is removed.

Generating a New Token for a Trust Code from CSSM

To generate a token to request a trust code, complete the following steps.

Generate one token for each *Virtual Account* you have. You can use same token for all the product instances that are part of one Virtual Account.

Before you begin

Supported topologies: Connected Directly to CSSM

Procedure

Step 1 Log in to the CSSM Web UI at https://software.cisco.com and click **Smart Software Licensing**.

Log in using the username and password provided by Cisco.

Step 2 Click the **Inventory** tab.

Step 3 From the **Virtual Account** drop-down list, choose the required virtual account

Step 4 Click the **General** tab.

Step 5 Click **New Token**. The **Create Registration Token** window is displayed.

Step 6 In the **Description** field, enter the token description

Step 7 In the **Expire After** field, enter the number of days the token must be active.

Step 8 (Optional) **In the Max. Number of Uses** field, enter the maximum number of uses allowed after which the token expires.

Step 9 Click **Create Token**.

Note If you enter a value here, ensure that you stagger the installation of the trust code on the product instances, which is the next part of the process. If you want to simultaneously install the trust code on a large number of product instances, we recommend that you leave this field blank. Entering a limit here and simultaneously installing it on a large number of devices causes a bottleneck in the processing of these requests in CSSM and installation on some devices may fail, with the following error: `Failure Reason: Server error occurred: LS_LICENGINE_FAIL_TO_CONNECT`.

Step 10 You will see your new token in the list. Click **Actions** and download the token as a `.txt` file.

Installing a Trust Code

To manually install a trust code, complete the following steps

Before you begin

Supported topologies:

- Connected Directly to CSSM

Procedure

	Command or Action	Purpose
Step 1	Generating a New Token for a Trust Code from CSSM, on page 121	In case you have not completed this already, generate and download a trust code file from CSSM.
Step 2	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted
Step 3	**license smart trust idtoken** *id_token_value* {**local** | **all**} [**force**] **Example:** `Device# license smart trust idtoken NGMwMjk5mYtNZaxMS00NzMZmtgWm all force`	Enables you to establish a trusted connection with CSSM. For *id_token_value*, enter the token you generated in CSSM. Enter one of following options: • **local**: Submits the trust request only for the active device in a High Availability set-up. This is the default option. • **all**: Submits the trust request for all devices in a High Availability set-up. Enter the **force** keyword to submit the trust code request in spite of an existing trust code on the product instance. Trust codes are node-locked to the UDI of the product instance. If a UDI is already registered, CSSM does not allow a new registration for the same UDI. Entering the **force** keyword sets a force flag in the message sent to CSSM to create a new trust code even if one already exists.

	Command or Action	Purpose

Step 4	**show license status**
	Example:
	`<output truncated>`
	`Trust Code Installed:`
	`  Active: PID:C9800-CL-K9,SN:93BBAH93MGS`
	`    INSTALLED on Nov 02 08:59:26 2020 IST`
	`  Standby: PID:C9800-CL-K9,SN:9XECPSUU4XN`
	`    INSTALLED on Nov 02 09:00:45 2020 IST`
		Displays date and time if trust code is installed. Date and time are in the local time zone. See field `Trust Code Installed:`.

Downloading a Policy File from CSSM

If you have requested a custom policy or if you want to apply a policy that is different from the default that is applied to the product instance, complete the following task:

Before you begin

Supported topologies:

- No Connectivity to CSSM and No CSLU
- CSLU Disconnected from CSSM

Procedure

Step 1 Log in to the CSSM Web UI at https://software.cisco.com and click **Smart Software Licensing**.

Log in using the username and password provided by Cisco.

Step 2 Follow this directory path: **Reports** > **Reporting Policy**.

Step 3 Click **Download**, to save the `.xml` policy file.

You can now install the file on the product instance. See Installing a File on the Product Instance, on page 124

Uploading Data or Requests to CSSM and Downloading a File

You can use this task to:

- To upload a RUM report to CSSM and download an ACK.
- To upload a SLAC or SLR authorization code return request.

  This applies only to the *No Connectivity to CSSM and No CSLU* topology and is supported starting with Cisco IOS XE Cupertino 17.7.1.

To upload a RUM report to CSSM and download an ACK *when the product instance is not connected to CSSM or CSLU*, complete the following task:

Before you begin

Supported topologies:

- No Connectivity to CSSM and No CSLU
- CSLU Disconnected from CSSM
- SSM On-Prem Deployment (Product instance-initiated communication and SSM On-Prem-initiated communication)

Procedure

Step 1 Log in to the CSSM Web UI at https://software.cisco.com.

Log in using the username and password provided by Cisco.

Step 2 Select the **Smart Account** (upper left-hand corner of the screen) that will receive the report.

Step 3 Select **Smart Software Licensing** → **Reports** → **Usage Data Files**.

Step 4 Click **Upload Usage Data**. Browse to the file location (RUM report in tar format), select, and click **Upload Data**.

Upload a RUM report (`.tar` format), or a SLAC return request file (`.txt` format).

You cannot delete a usage report in CSSM, after it has been uploaded.

Step 5 From the Select Virtual Accounts pop-up, select the **Virtual Account** that will receive the uploaded file. The file is uploaded to Cisco and is listed in the Usage Data Files table in the Reports screen showing the File Name, time is was Reported, which Virtual Account it was uploaded to, the Reporting Status, Number of Product Instances reported, and the Acknowledgement status.

Step 6 In the Acknowledgement column, click **Download** to save the `.txt` ACK file for the report you uploaded.

Wait for the ACK to appear in the Acknowledgement column. If there many RUM reports or requests to process, CSSM may take a few minutes.

Depending on the topology you have implemented, you can now install the file on the product instance, or transfer it to CSLU, or import it into SSM On-Prem.

Installing a File on the Product Instance

To install a SLAC, or policy, or ACK, on the product instance *when the product instance is not connected to CSSM or CSLU*, complete the following task:

Before you begin

Supported topologies: No Connectivity to CSSM and No CSLU

You must have the corresponding file saved in a location that is accessible to the product instance.

- For a policy, see Downloading a Policy File from CSSM, on page 123
- For an ACK, see Uploading Data or Requests to CSSM and Downloading a File, on page 123

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted
Step 2	**copy** *source* **bootflash:***file-name* **Example:** `Device# copy tftp://10.8.0.6/example.txt bootflash:`	Copies the file from its source location or directory to the flash memory of the product instance. • *source*: This is the location of the source file or directory to be copied. The source can be either local or remote • **bootflash:**: This is the destination for boot flash memory.
Step 3	**license smart import bootflash:** *file-name* **Example:** `Device# license smart import bootflash:example.txt`	Imports and installs the file on the product instance. After installation, a system message displays the type of file you just installed.
Step 4	**show license all** **Example:** `Device# show license all`	Displays license authorization, policy and reporting information for the product instance.

Setting the Transport Type, URL, and Reporting Interval

To configure the mode of transport for a product instance, complete the following task:

Before you begin

Supported topologies: all

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal**	

	Command or Action	Purpose
	Example: `Device# configure terminal`	
Step 3	**license smart transport** {**automatic** | **callhome** | **cslu** | **off** | **smart**} **Example:** `Device(config)# license smart transport cslu`	Configures a mode of transport for the product instance to use. Choose from the following options: • **automatic**: Sets the transport mode **cslu**. • **callhome**: Enables Call Home as the transport mode. • **cslu**: This is the default transport mode. Enter this keyword if you are using CSLU *or* SSM On-Prem, with product instance-initiated communication. While the transport mode keyword is the same for CSLU and SSM On-Prem, the transport URLs are different. See **license smart url cslu** *cslu_or_on-prem_url* in the next step. • **off**: Disables all communication from the product instance. • **smart**: Enables Smart transport.

Step 4	**license smart url** {*url* | **cslu** *cslu_or_on-prem_url* | **default** | **smart** *smart_url* | **utility** *smart_url*}
	Example:
	`Device(config)# license smart url cslu http://192.168.0.1:8182/cslu/v1/pi`
		Sets a URL for the configured transport mode. Depending on the transport mode you've chosen in the previous step, configure the corresponding URL here:
		• *url*: If you have configured the transport mode as **callhome**, configure this option. Enter the CSSM URL exactly as follows:
		`https://tools.cisco.com/its/service/oddce/services/DDCEService`
		The **no license smart url***url* command reverts to the default URL.
		• **cslu** *cslu_or_on-prem_url*: If you have configured the transport mode as **cslu**, configure this option with the URL for CSLU or SSM On-Prem, as applicable.
		  • If you are using CSLU, enter the URL as follows:
		`http://<cslu_ip_or_host>:8182/cslu/v1/pi`
		For `<cslu_ip_or_host>`, enter the hostname or the IP address of the windows host where you have

	Command or Action	Purpose

		installed CSLU. 8182 is the port number and it is the only port number that CSLU uses.
		
		The **no license smart url cslu** *cslu_url* command reverts to `http://cslu-local:8182/cslu/v1/pi`
		
		• If you are using SSM On-Prem, enter the URL as follows:
		
		`http://<ip>/cslu/v1/pi/<tenant ID>`
		
		For <ip>, enter the hostname or the IP address of the server where you have installed SSM On-Prem. The <tenantID> must be the default local virtual account ID.
		
		Tip You can retrieve the entire URL from SSM On-Prem. See Retrieving the Transport URL (SSM On-Prem UI), on page 104
		
		The **no license smart url cslu** *cslu_url* command reverts to `http://cslu-local:8182/cslu/v1/pi`
		
		• **default**: Depends on the configured transport mode. Only the **smart** and **cslu** transport modes are supported with this option.
		
		If the transport mode is set to **cslu**, and you configure **license smart url default**, the CSLU URL is configured automatically (`https://cslu-local:8182/cslu/v1/pi`).
		
		If the transport mode is set to **smart**, and you configure **license smart url default**, the Smart URL is configured automatically (`https://smartreceiver.cisco.com/licservice/license`).
		
		• **smart** *smart_url*: If you have configured the transport type as **smart**, configure this option. Enter the URL exactly as follows:
		
		`https://smartreceiver.cisco.com/licservice/license`
		
		When you configure this option, the system automatically creates a duplicate

	Command or Action	Purpose
		of the URL in **license smart url** *url*. You can ignore the duplicate entry, no further action is required. The **no license smart url smart***smart_url* command reverts to the default URL. • **utility** *smart_url*: Although available on the CLI, this option is not supported.
Step 5	**license smart usage interval** *interval_in_days* **Example:** `Device(config)# license smart usage interval 40`	(Optional) Sets the reporting interval in days. By default the RUM report is sent every 30 days. The valid value range is 1 to 3650. If you do not configure an interval, the reporting interval is determined entirely by the policy value.
Step 6	**exit** **Example:** `Device(config)# exit`	Exits global configuration mode and returns to privileged EXEC mode.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves your entries in the configuration file.

Configuring an AIR License

In the Smart Licensing Using Policy environment, you can use this task to configure a license, or change the license being used on the product instance, or configure an add-on license on the product instance. For example, if you are currently using AIR Network Advantage and you also want to use features available with a corresponding Digital Networking Architecture (DNA) Advantage license, you can configure the same using this task. Or for example, if you do not want to use an add-on license any more, reconfigure this command to use only the AIR Network Advantage license.

Information about available licenses can be found Smart Account or Virtual Account. The available licenses may be one of the following:

- AIR Network Essential
- AIR Network Advantage
- AIR DNA Essential
- AIR DNA Advantage

Starting with Cisco IOS XE Bengaluru 17.4.1, *only for EWC-APs*, you can opt-out of purchasing an AIR DNA license. The option to opt-out of AIR DNA licenses is available only through the Cisco Commerce portal. When you opt-out, Smart Licensing Using Policy functionality is disabled.

For a new product instance, this means:

Condition	Required Action	Outcome or Result
You opt-out of AIR DNA licenses	None.	Use only AIR Network Essentials. Smart Licensing Using Policy functionality is disabled on the product instance and for your Smart Account and Virtual Account in CSSM. License usage is not recorded, and no reporting requirements apply.
You purchase AIR DNA licenses	Enter the **license air level** command in global configuration mode and configure the corresponding AIR DNA license. Reload to use the corresponding license. Implement one of the supported topologies and fulfill reporting requirements. For information about implementing a topology, see the Supported Topologies section in this document.	Use the purchased AIR DNA and AIR Network license. Smart Licensing Using Policy functionality is enabled on the product instance and for your Smart Account and Virtual Account in CSSM.

For an existing product instance, this means:

Condition	Required Action	Outcome or Result
You are using an AIR DNA license	None.	No change. You are already in the Smart Licensing Using Policy environment.
You do not want to renew the DNA license on term expiry	On term expiry, enter the **license air level** command in global configuration mode and configure AIR Network Essentials or AIR Network Advantage. Reload to use the corresponding license.	If you had AIR DNA Essentials, you now use AIR Network Essentials. If you had AIR DNA Advantage, you now use AIR Network Advantage. Smart Licensing Using Policy functionality is disabled on the product instance and for your Smart Account and Virtual Account in CSSM. License usage is not recorded, and no reporting requirements apply.

To configure or change the license in-use, follow this procedure:

Before you begin

Supported topologies: all

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables the privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 3	**license air level** {**air-network-advantage** [**addon air-dna-advantage**] | **air-network-essentials** [**addon air-dna-essentials**] } **Example:** `Device(config)# license air level air-network-essentials addon air-dna-essentials`	Activates the configured license on the product instance. In the accompanying example, the product instance activates the AIR DNA Essentials (along with the AIR Network Essential) license after reload. **Note** Prior to Cisco IOS XE Bengaluru 17.4.1, the default for EWC-APs was AIR DNA Essentials. Starting with 17.4.1, the default is AIR Network Essentials.
Step 4	**exit** **Example:** `Device(config)# exit`	Returns to the privileged EXEC mode.
Step 5	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves configuration changes.
Step 6	**reload** **Example:** `Device# reload`	Reloads the device.

Step 7	**show version**
	Example:
	`Device# show version`
	`Cisco IOS XE Software, Version 17.03.02`
	`Cisco IOS Software [Amsterdam], C9800-CL Software (C9800-CL-K9_IOSXE), Version 17.3.2, RELEASE SOFTWARE`
	`<output truncated>`
	`AIR License Level: AIR DNA Essentials`
	`Next reload AIR license Level: AIR DNA Essentials`
	
	`Smart Licensing Status: Registration Not Applicable/Not Applicable`
	`<output truncated>`
		Displays currently used license and the license that is effective at the next reload information.

What to do next

After you configure a license level, the change is effective after a reload. To know if reporting is required, refer to the output of the **show license status** privileged EXEC command and check the `Next ACK deadline:` and `Next report push:` fields.

Note The change in license usage is recorded on the product instance. The next steps relating to reporting - if required - depend on your current topology.

- Connected to CSSM Through CSLU
  - Product Instance-initiated communication: The product instance triggers reporting and installs the returning ACK. CSLU sends the RUM report to CSSM and collects the ACK from CSSM.
  - CSLU-initiated communication: You have to collect usage from the CSLU interface: Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94. CSLU sends the RUM report to CSSM and collects the ACK from CSSM.
- Connected Directly to CSSM: The product instance triggers reporting and installs the returning ACK.
- CSLU Disconnected from CSSM:
  - Product Instance-initiated communication: The product instance triggers reporting. You then have to report usage in the disconnected mode: Export to CSSM (CSLU Interface), on page 95 > Uploading Data or Requests to CSSM and Downloading a File, on page 123 > Import from CSSM (CSLU Interface), on page 96.
  - CSLU-initiated communication: You have to collect usage from the CSLU interface and report usage in the disconnected mode: Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94> Export to CSSM (CSLU Interface), on page 95 > Uploading Data or Requests to CSSM and Downloading a File, on page 123 > Import from CSSM (CSLU Interface), on page 96.
- No Connectivity to CSSM and No CSLU: License usage is recorded on the product instance. You must save RUM reports to a file on the product instance, and from a workstation that has connectivity to the internet, and Cisco, upload it to CSSM: Enter **license smart save usage** privileged EXEC command to save usage > Uploading Data or Requests to CSSM and Downloading a File, on page 123 > Installing a File on the Product Instance, on page 124.

Sample Resource Utilization Measurement Report

The following is a sample Resource Utilization Measurement (RUM) report, in XML format (See RUM Report and Report Acknowledgement, on page 41). Several such reports may be concatenated to form one report.

```
<?xml version="1.0" encoding="UTF-8"?>
  <smartLicense>
```

```
  </smartLicense>
```

Troubleshooting Smart Licensing Using Policy

This section provides the list of Smart Licensing Using Policy-related system messages you may encounter, possible reasons for failure, and recommended action.

System Message Overview

The system software sends system messages to the console (and, optionally, to a logging server on another system). Not all system messages mean problems with your system. Some messages are informational, and others can help diagnose problems with communications lines, internal hardware, or the system software.

How to Read System Messages

System log messages can contain up to 80 characters. Each system message begins with a percent sign (%) and is structured as follows:

```
%FACILITY-SEVERITY-MNEMONIC: Message-text
```

%FACILITY

Two or more uppercase letters that show the facility to which the message refers. A facility can be a hardware device, a protocol, or a module of the system software

SEVERITY

A single-digit code from 0 to 7 that reflects the severity of the condition. The lower the number, the more serious the situation.

Table 9: Message Severity Levels

Severity Level	Description
0 - emergency	System is unusable.
1 - alert	Immediate action required.
2 - critical	Critical condition.
3 - error	Error condition.
4 - warning	Warning condition.
5 - notification	Normal but significant condition.
6 - informational	Informational message only.
7 - debugging	Message that appears during debugging only.

MNEMONIC

A code that uniquely identifies the message.

Message-text

Message-text is a text string describing the condition. This portion of the message sometimes contains detailed information about the event, including terminal port numbers, network addresses, or addresses that correspond to locations in the system memory address space. Because the information in these variable fields changes from message to message, it is represented here by short strings enclosed in square brackets ([ ]). A decimal number, for example, is represented as [dec].

Table 10: Variable Fields in Messages

Severity Level	Description
[char]	Single character
[chars]	Character string
[dec]	Decimal number
[enet]	Ethernet address (for example, 0000.FEED.00C0)
[hex]	Hexadecimal number
[inet]	Internet address (for example, 10.0.2.16)
[int]	Integer
[node]	Address or node name
[t-line]	Terminal line number in octal (or in decimal if the decimal-TTY service is enabled)
[clock]	Clock (for example, 01:20:08 UTC Tue Mar 2 1993

System Messages

This section provides the list of Smart Licensing Using Policy-related system messages you may encounter, possible reasons for failure (incase it is a failure message), and recommended action (if action is required).

For all error messages, if you are not able to solve the problem, contact your Cisco technical support representative with the following information:

The message, exactly as it appears on the console or in the system log.

The output from the **show license tech support**, **show license history message**, and the **show platform software sl-infra** privileged EXEC commands.

- %SMART_LIC-3-POLICY_INSTALL_FAILED
- %SMART_LIC-3-AUTHORIZATION_INSTALL_FAILED
- %SMART_LIC-3-COMM_FAILED
- %SMART_LIC-3-COMM_RESTORED
- %SMART_LIC-3-POLICY_REMOVED
- %SMART_LIC-3-TRUST_CODE_INSTALL_FAILED
- %SMART_LIC-4-REPORTING_NOT_SUPPORTED

- %SMART_LIC-6-POLICY_INSTALL_SUCCESS
- %SMART_LIC-6-AUTHORIZATION_INSTALL_SUCCESS
- %SMART_LIC-6-AUTHORIZATION_REMOVED
- %SMART_LIC-6-REPORTING_REQUIRED
- %SMART_LIC-6-TRUST_CODE_INSTALL_SUCCESS

```
Error Message %SMART_LIC-3-POLICY_INSTALL_FAILED: The installation of a new
licensing policy has failed: [chars].
```

Explanation: A policy was installed, but an error was detected while parsing the policy code, and installation failed. [chars] is the error string with details of the failure.

Possible reasons for failure include:

- A signature mismatch: This means that the system clock is not accurate.
- A timestamp mismatch: This means the system clock on the product instance is not synchronized with CSSM.

Note The device should have a valid clock and the NTP configuration.

Recommended Action:

For both possible failure reasons, ensure that the system clock is accurate and synchronized with CSSM. Configure the **ntp server** command in global configuration mode. For example:

```
Device(config)# ntp server 198.51.100.100 version 2 prefer
```

If the above does not work and policy installation still fails, and contact your Cisco technical support representative.

```
Error Message %SMART_LIC-3-AUTHORIZATION_INSTALL_FAILED: The install of a new
licensing authorization code has failed on [chars]: [chars].
```

This message is not applicable to Cisco Catalyst Access, Core, and Aggregation Switches, because there are no enforced or export-controlled licenses on these product instances.

```
Error Message %SMART_LIC-3-COMM_FAILED: Communications failure with the [chars] :
[chars]
```

Explanation: Smart Licensing communication either with CSSM, or CSLU, or SSM On-Prem failed. The first [chars] is the currently configured transport type, and the second [chars] is the error string with details of the failure. This message appears for every communication attempt that fails.

Possible reasons for failure include:

- CSSM, CSLU, SSM On-Prem is not reachable: This means that there is a network reachability problem.

- 404 host not found: This means the CSSM server is down.

For topologies where the product instance initiates the sending of RUM reports (Connected to CSSM Through CSLU: Product Instance-Initiated Communication, Connected Directly to CSSM, CSLU Disconnected from CSSM: Product Instance-Initiated Communication, and SSM On-Prem Deployment: Product Instance-Initiated Communication) if this communication failure message coincides with scheduled reporting (**license smart usage interval** *interval_in_days* global configuration command), the product instance attempts to send out the RUM report for up to four hours after the scheduled time has expired. If it is still unable to send out the report (because the communication failure persists), the system resets the interval to 15 minutes. Once the communication failure is resolved, the system reverts the reporting interval to last configured value.

Recommended Action:

Troubleshooting steps are provided for when CSSM is not reachable or there is a missing client certificate, when CSLU is not reachable, and when SSM On-Prem is not reachable.

- If CSSM is not reachable and the configured transport type is **smart**:

  1. Check if the smart URL is configured correctly. Use the **show license status** command in privileged EXEC mode, to check if the URL is exactly as follows: https://smartreceiver.cisco.com/licservice/license. If it is not, reconfigure the **license smart url smart** *smar_URL* command in global configuration mode.
  2. Check DNS resolution. Verify that the product instance can ping `smartreceiver.cisco.com` or the nslookup translated IP. The following example shows how to ping the translated IP

     ```
     Device# ping 171.70.168.183
     Type escape sequence to abort.
     Sending 5, 100-byte ICMP Echos to 171.70.168.183, timeout is 2 seconds:
     !!!!!
     Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
     ```

- If CSSM is not reachable and the configured transport type is **callhome**:

  1. Check if the URL is entered correctly. Use the **show license status** command in privileged EXEC mode, to check if the URL is exactly as follows: https://tools.cisco.com/its/service/oddce/services/DDCEService.
  2. Check if Call Home profile `CiscoTAC-1` is active and destination URL is correct. Use the **show call-home profile all** command in privileged EXEC mode:

     ```
     Current smart-licensing transport settings:
      Smart-license messages: enabled
      Profile: CiscoTAC-1 (status: ACTIVE)
      Destination  URL(s):  https://tools.cisco.com/its/service/oddce/services/DDCEService
     ```

  3. Check DNS Resolution. Verify that the product instance can ping `tools.cisco.com`, or the nslookup translated IP.

     ```
     Device# ping tools.cisco.com
     Type escape sequence to abort.
     Sending 5, 100-byte ICMP Echos to 173.37.145.8, timeout is 2 seconds:
     !!!!!
     Success rate is 100 percent (5/5), round-trip min/avg/max = 41/41/42 ms
     ```

     If the above does not work check the following: if the product instance is set, if the product instance IP network is up. To ensure that the network is up, configure the **no shutdown** command in interface configuration mode.

Check if the device is subnet masked with a subnet IP, and if the DNS IP is confgured.

4. Verify that the HTTPs client source interface is correct.

   Use the **show ip http client** command in privileged EXEC mode to display current configuration. Use **ip http client source-interface** command in global configuration mode to reconfigure it.

   In case the above does not work, double-check your routing rules, and firewall settings.

- If CSLU is not reachable:

1. Check if CSLU discovery works.

   - Zero-touch DNS discovery of `cslu-local` or DNS discovery of your domain..

     In the **show license all** command output, check if the `Last ACK received:` field. If this has a recent timestamp it means that the product instance has connectivity with CSLU. If it is not, proceed with the following checks:

     Check if the product instance is able to ping `cslu-local`. A successful ping confirms that the product instance is reachable.

     If the above does not work, configure the name server with an entry where hostname `cslu-local` is mapped to the CSLU IP address (the windows host where you installed CSLU). Configure the **ip domain name** *domain-name* and **ip name-server** *server-address* commands in global configuration mode. Here the CSLU IP is 192.168.0.1 and name-server creates entry `cslu-local.example.com`:

     ```
     Device(config)# ip domain name example.com
     Device(config)# ip name-server 192.168.0.1
     ```

   - CSLU URL is configured.

     In the **show license all** command output, under the `Transport:` header check the following: The `Type:` must be `cslu`and `Cslu address:` must have the hostname or the IP address of the windows host where you have installed CSLU. Check if the rest of the address is configured as shown below and check if the port number is 8182.

     ```
     Transport:
       Type: cslu
       Cslu address: http://192.168.0.1:8182/cslu/v1/pi
     ```

     If it is not, configure the **license smart transport cslu** and **license smart url cslu** *http://<cslu_ip_or_host>:8182/cslu/v1/pi* commands in global configuration mode

2. For CSLU-initiated communication, in addition to the CSLU discovery checks listed above, check the following:

   Verify HTTP connectivity. Use the **show ip http server session-module** command in privileged EXEC mode. In the output, under header `HTTP server current connections:`, check that `SL_HTTP` is active. If it is not re-configure the **ip http** commands as mentioned in Ensuring Network Reachability for CSLU-Initiated Communication, on page 96

   From a Web browser on the device where CSLU is installed, verify `https://<product-instance-ip>/`. This ensures that the REST API from CSLU to the product instance works as expected.

- If SSM On-Prem is not reachable:

1. For product instance-initiated communication, check if the SSM On-Prem transport type and URL are configured correctly.

   In the **show license all** command output, under the `Transport:` header check the following: The `Type:` must be `cslu`and `Cslu address:` must have the hostname or the IP address of the server where you have installed SSM On-Prem and <tenantID> of the *default* local virtual account. See the example below:

   ```
   Transport:
     Type: cslu
     Cslu address: https://192.168.0.1/cslu/v1/pi/on-prem-default
   ```

   Check if you have the correct URL from SSM On-Prem (Retrieving the Transport URL (SSM On-Prem UI), on page 104) and then configure **license smart transport cslu** and **license smart url cslu** *http://<ip>/cslu/v1/pi/<tenant ID>* commands in global configuration mode.

   Check that you have configured any other required commands for your network as mentioned in Ensuring Network Reachability for Product Instance-Initiated Communication, on page 101.

2. For SSM On-Prem-initiated communication, check HTTPs connectivity.

   Use the **show ip http server session-module** command in privileged EXEC mode. In the output, under header `HTTP server current connections:`, check that `SL_HTTP` is active. If it is not re-configure the **ip http** commands as mentioned in Ensuring Network Reachability for SSM On-Prem-Initiated Communication, on page 106.

3. Check trustpoint and that certificates are accepted.

   For both forms of communication in an SSM On-Prem Deployment, ensure that the correct trustpoint is used and that the necessary certificates are accepted:

   ```
   Device(config)# crypto pki trustpoint SLA-TrustPoint
   Device(ca-trustpoint)#
   Device(ca-trustpoint)# enrollment terminal
   Device(ca-trustpoint)# revocation-check none
   Device(ca-trustpoint)# end
   Device# copy running-config startup-config
   ```

If the above does not work and policy installation still fails, contact your Cisco technical support representative.

```
Error Message  %SMART_LIC-3-COMM_RESTORED: Communications with the [chars] restored.
[chars] - depends on the transport type
        - Cisco Smart Software Manager (CSSM)
        - Cisco Smart License utility (CSLU)
Smart Agent communication with either the Cisco Smart Software Manager (CSSM) or the Cisco
 Smart License
utility (CSLU) has been restored. No action required.
```

Explanation: Product instance communication with either the CSSM, or CSLU, or SSM On-Prem is restored.

Recommended Action: No action required.

`Error Message %SMART_LIC-3-POLICY_REMOVED: The licensing policy has been removed.`

Explanation: A previously installed *custom* licensing policy has been removed. The `Cisco default` policy is then automatically effective. This may cause a change in the behavior of smart licensing.

Possible reasons for failure include:

If you have entered the **license smart factory reset** command in privileged EXEC mode all licensing information including the policy is removed.

Recommended Action:

If the policy was removed intentionally, then no further action is required.

If the policy was removed inadvertantly, you can reapply the policy. Depending on the topology you have implemented, follow the corresponding method to retrieve the policy:

- Connected Directly to CSSM:

  Enter **show license status**, and check field `Trust Code Installed:`. If trust is established, then CSSM will automatically return the policy again. The policy is automatically re-installed on product instances of the corresponding Virtual Account.

  If trust has not been established, complete these tasks: Generating a New Token for a Trust Code from CSSM, on page 121 and Installing a Trust Code, on page 122. When you have completed these tasks, CSSM will automatically return the policy again. The policy is then automatically installed on all product instances of that Virtual Account.

- Connected to CSSM Through CSLU:

  - For product instance-initiatied communication), enter the **license smart sync** command in privileged EXEC mode. The synchronization request causes CSLU to push the missing information (a policy or authurization code) to the product instance.
  - For CSLU-initiated communication, complete this task: Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94. This causes CSLU to detect and re-furnish the missing policy in an ACK response.

- CSLU Disconnected from CSSM:

  - For product instance-initiatied communication), enter the **license smart sync** command in privileged EXEC mode. The synchronization request causes CSLU to push the missing information (a policy or authurization code) to the product instance. Then complete these tasks in the given order:Export to CSSM (CSLU Interface), on page 95 > Uploading Data or Requests to CSSM and Downloading a File, on page 123 > Import from CSSM (CSLU Interface), on page 96.
  - For CSLU-initiated communication, complete this task: Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94. This causes CSLU to detect and re-furnish the missing policy in an ACK response. Then complete these tasks in the given order: Export to CSSM (CSLU Interface), on page 95 > Uploading Data or Requests to CSSM and Downloading a File, on page 123 > Import from CSSM (CSLU Interface), on page 96.

- No Connectivity to CSSM and No CSLU

  If you are in an entirely air-gapped network, from a workstation that has connectivity to the internet and CSSM complete this task: Downloading a Policy File from CSSM, on page 123.

  Then complete this task on the product instance: Installing a File on the Product Instance, on page 124.

- SSM On-Prem Deployment
  - For product instance-initiatied communication), enter the **license smart sync** command in privileged EXEC mode. The causes the product instance to synchronize with SSM On-Prem and restore any required or missing information. Then synchronize SSM On-Prem with CSSM if required:
  - For SSM On-Prem-initiated communication: In the SSM On-Prem UI, navigate to **Reports** > **Synchronisation pull schedule with the devices** > **Synchronise now with the device**.

  For both forms of communication in an SSM On-Prem Deployment, synchronize with CSSM using either option:
  - SSM On-Prem is connected to CSSM: In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Reports** > **Usage Schedules** > **Synchronize now with Cisco**.
  - SSM On-Prem is not connected to CSSM: Exporting and Importing Usage Data (SSM On-Prem UI), on page 104.

```
Error Message %SMART_LIC-3-TRUST_CODE_INSTALL_FAILED: The install of a new licensing
trust code has failed on [chars]: [chars].
```

Explanation: Trust code installation has failed. The first [chars] is the UDI where trust code installation was attempted. The second [chars] is the error string with details of the failure.

Possible reasons for failure include:

- A trust code is already installed: Trust codes are node-locked to the UDI of the product instance. If the UDI is already registered, and you try to install another one, installation fails.
- Smart Account-Virtual Account mismatch: This means the Smart Account or Virtual Account (for which the token ID was generated) does not include the product instance on which you installed the trust code. The token generated in CSSM, applies at the Smart Account or Virtual Account level and applies only to all product instances in that account.
- A signature mismatch: This means that the system clock is not accurate.
- Timestamp mismatch: This means the product instance time is not synchronized with CSSM, and can cause installation to fail.

Recommended Action:

- A trust code is already installed: If you want to install a trust code inspite of an existing trust code on the product instance, re-configure the **license smart trust idtoken** *id_token_value* { **local** | **all** } [**force**] command in privileged EXEC mode, and be sure to include the **force** keyword this time. Entering the **force** keyword sets a force flag in the message sent to CSSM to create a new trust code even if one already exists.
- Smart Account-Virtual Account mismatch:

  Log in to the CSSM Web UI at https://software.cisco.com and click **Smart Software Licensing**>**Inventory** > **Product Instances**.

  Check if the product instance on which you want to generate the token is listed in the selected Virtual Account. If it is, proceed to the next step. If not, check and select the correct Smart Account and Virtual

Account. Then complete these tasks again: Generating a New Token for a Trust Code from CSSM, on page 121 and Installing a Trust Code, on page 122.

- Timestamp mismatch and signature mismatch: Configure the **ntp server** command in global configuration mode. For example:

```
Device(config)# ntp server 198.51.100.100 version 2 prefer
```

```
Error Message    %SMART_LIC-4-REPORTING_NOT_SUPPORTED: The CSSM OnPrem that this
product instance is connected to is down rev and does not support the enhanced policy and
usage
reporting mode.
```

Explanation: Cisco Smart Software Manager On-Prem (formerly known as Cisco Smart Software Manager satellite) is supported in the Smart Licensing Using Policy environment starting with Cisco IOS XE Amsterdam 17.3.3 only (See SSM On-Prem, on page 37). In *unsupported* releases, the product instance will behave as follows:

- Stop sending registration renewals and authorization renewals.
- Start recording usage and saving RUM reports locally.

Recommended Action:

You have the following options:

- Refer to and implement one of the supported topologies instead. See: Supported Topologies, on page 42.
- Upgrade to a release where SSM On-Prem is supported with Smart Licensing Using Policy. See Migrating to a Version of SSM On-Prem That Supports Smart Licensing Using Policy, on page 89.

```
Error Message %SMART_LIC-6-POLICY_INSTALL_SUCCESS: A new licensing policy
was successfully installed.
```

Explanation: A policy was installed in one of the following ways:

- Using Cisco IOS commands.
- CSLU-initiated communication.
- As part of an ACK response.

Recommended Action: No action is required. If you want to know which policy is applied (the policy in-use) and its reporting requirements, enter the **show license all** command in privileged EXEC mode.

```
Error Message %SMART_LIC-6-AUTHORIZATION_INSTALL_SUCCESS: A new licensing
authorization code was successfully installed on: [chars].
```

This message is not applicable to Cisco Catalyst Access, Core, and Aggregation Switches, because there are no enforced or export-controlled licenses on these product instances.

```
Error Message %SMART_LIC-6-AUTHORIZATION_REMOVED: A licensing authorization code has
 been removed from [chars]
```

Explanation: [chars] is the UDI where the authorization code was installed. The authorization code has been removed. This removes the licenses from the product instance and may cause a change in the behavior of smart licensing and the features using licenses.

Recommended Action: No action is required. If you want to see the current state of the license, enter the **show license all** command in privileged EXEC mode.

```
Error Message %SMART_LIC-6-REPORTING_REQUIRED: A Usage report acknowledgement
will be required in [dec] days.
```

Explanation: This is an alert which means that RUM reporting to Cisco is required. [dec] is the amount of time (in days) left to meet this reporting requirements.

Recommended Action: Ensure that RUM reports are sent within the requested time. The topology you have implemented determines the reporting method.

- Connected to CSSM Through CSLU
  - For product instance-initiatied communication: Enter the **license smart sync** command in privileged EXEC mode. If CSLU is currently logged into CSSM the reports will be automatically sent to the associated Smart Account and Virtual Account in CSSM.
  - For CSLU-initiated communication, complete this task: Collecting Usage Reports: CSLU Initiated (CSLU Interface), on page 94.
- Connected Directly to CSSM: Enter the **license smart sync** command in privileged EXEC mode.
- Connected to CSSM Through a Controller: If the product instance is managed by a controller, the controller will send the RUM report at the scheduled time.

  If you are using Cisco DNA Center as the controller, you have the option of ad-hoc reporting. See the *Cisco DNA Center Administrator Guide* of the required release (Release 2.2.2 onwards) > *Manage Licenses* > *Upload Resource Utilization Details to CSSM*.
- CSLU Disconnected from CSSM: If the product instance is connected to CSLU, synchronize with the product instance as shown for "Connected to CSSM Through CSLU"above, then complete these tasks: Export to CSSM (CSLU Interface), on page 95, Uploading Data or Requests to CSSM and Downloading a File, on page 123, and Import from CSSM (CSLU Interface), on page 96.
- No Connectivity to CSSM and No CSLU: Enter the **license smart save usage** command in privileged EXEC mode, to save the required usage information in a file. Then, from a workstation where you have

connectivity to CSSM, complete these tasks: Uploading Data or Requests to CSSM and Downloading a File, on page 123 > Installing a File on the Product Instance, on page 124.

- SSM On-Prem Deployment:

  Synchronize the product instance with SSM On-Prem:

  - For product instance-initiatied communication: Enter the **license smart sync** command in privileged EXEC mode. If CSLU is currently logged into CSSM the reports will be automatically sent to the associated Smart Account and Virtual Account in CSSM.
  - For SSM On-Prem-initiated communication, complete this task: In the SSM On-Prem UI, navigate to **Reports** > **Synchronisation pull schedule with the devices** > **Synchronise now with the device**.

  Synchronize usage information with CSSM (*choose one*)

  - SSM On-Prem is connected to CSSM: In the SSM On-Prem UI, Smart Licensing workspace, nagivate to **Reports** > **Usage Schedules** > **Synchronize now with Cisco**.
  - SSM On-Prem is not connected to CSSM: Exporting and Importing Usage Data (SSM On-Prem UI), on page 104.

```
Error Message %SMART_LIC-6-TRUST_CODE_INSTALL_SUCCESS: A new licensing trust code
 was successfully installed on [chars].
```

Explanation:[chars] is the UDI where the trust code was successfully installed.

Recommended Action: No action is required. If you want to verify that the trust code is installed, enter the **show license status** command in privileged EXEC mode. Look for the updated timestamp under header `Trust Code Installed:` in the output.

Additional References for Smart Licensing Using Policy

Topic	Document Title
For complete syntax and usage information for the commands used in this chapter, see the Command Reference of the corresponding release.	Cisco Catalyst 9800 Series Wireless Controller Command Reference
Cisco Smart Software Manager Help	Smart Software Manager Help
Cisco Smart License Utility (CSLU) installation and user guides	Cisco Smart License Utility Quick Start Setup Guide Cisco Smart License Utility User Guide

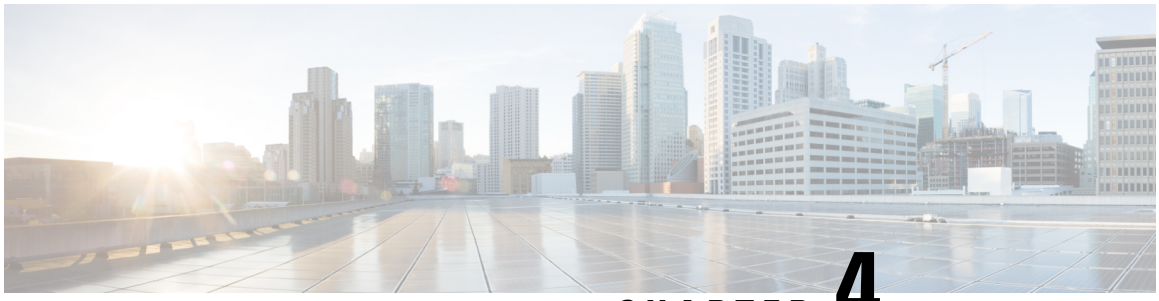
Feature History for Smart Licensing Using Policy

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Gibraltar 16.10.1	Smart Licensing	A cloud-based, software license management solution that allows you to manage and track the status of your license, hardware, and software usage trends.
Cisco IOS XE Amsterdam 17.3.2a	Smart Licensing Using Policy	An enhanced version of Smart Licensing, with the overarching objective of providing a licensing solution that does not interrupt the operations of your network, rather, one that enables a compliance relationship to account for the hardware and software licenses you purchase and use. Starting with this release, Smart Licensing Using Policy is automatically enabled on the device. This is also the case when you upgrade to this release. By default, your Smart Account and Virtual Account in CSSM is enabled for Smart Licensing Using Policy.
	Cisco DNA Center Support for Smart Licensing Using Policy	Cisco DNA Center supports Smart Licensing Using Policy functionality starting with Cisco DNA Center Release 2.2.2. When you use Cisco DNA Center to manage a product instance, Cisco DNA Center connects to CSSM, and is the interface for all communication to and from CSSM. For information about the comptabile controller and product instance versions, see Controller, on page 36. For information about this topology, see Connected to CSSM Through a Controller, on page 46 and Workflow for Topology: Connected to CSSM Through a Controller, on page 63.

Release	Feature	Feature Information
Cisco IOS XE Amsterdam 17.3.3	Smart Software Manager On-Prem (SSM On-Prem) Support for Smart Licensing Using Policy	SSM On-Prem is an asset manager, which works in conjunction with CSSM. It enables you to administer products and licenses on your premises instead of having to directly connect to CSSM. For information about the comptabile SSM On-Prem and product instance versions, see: SSM On-Prem, on page 37. For an overview of this topology, and to know how to implement it see SSM On-Prem Deployment, on page 48 and Workflow for Topology: SSM On-Prem Deployment, on page 65. For information about migrating from an exisiting version of SSM On-Prem, to one that supports Smart Licensing Using Policy, see Migrating to a Version of SSM On-Prem That Supports Smart Licensing Using Policy, on page 89.
Cisco IOS XE Bengaluru 17.4.1	Option to opt-out of AIR DNA licenses and change in default license level for EWC-APs.	The option to opt-out of purchasing an AIR DNA license was introduced. This option is available only through the Cisco Commerce portal. When you opt-out, you use only the AIR Network Essentials license, and Smart Licensing Using Policy functionality is disabled on the product instance. For more information, see the *Configuring an AIR License* section in this guide. Starting with this release, the default license on an EWC-AP was also changed to AIR Network Essentials.



CHAPTER 4

Conversion and Migration

Conversion and Migration in Embedded Wireless Controller Capable APs

The Cisco Embedded Wireless Controller on Catalyst Access Points is not supported on any non-802.11ax (non-11ax) based access points (AP). It is only supported on 802.11ax (11ax) based APs. The embedded wireless controller is the only supported form of Cisco Mobility Express on 11ax based APs.

The conversion enables you to convert the 11ax APs running CAPWAP to embedded wireless controller and vice-versa.

Types of Conversion

The types of conversion scenarios supported are:

- AP Conversion – The following AP conversions are supported:
  - Converting a CAPWAP AP to Embedded Wireless Controller - This conversion is required when you have an AP with a CAPWAP image, and you want to use the AP to deploy a embedded wireless controller based network. In order to do this, you must convert the CAPWAP AP to a embedded wireless controller.
  - Converting an Embedded Wireless Controller AP to a CAPWAP AP – This conversion is required if you want to migrate the APs from an embedded wireless controller network to a non-embedded wireless controller network; or if you do not want the APs to participate in the primary AP election process.
- Network Conversion

- SKU Conversion

Note The request for conversion of an EWC non-capable AP, (for example, Cisco Aironet 1830 Series Access Points), to the EWC mode, is now verified and rejected, because the AP cannot be converted.

Access Point Conversion

This section gives the details of converting a CAPWAP access point to anembedded wireless controller.

Converting a CAPWAP AP to an Embedded Wireless Controller Capable AP

Note Before converting from CAPWAP to embedded wireless controller (EWC), ensure that you upgrade the corresponding AP with the CAPWAP image in Cisco AireOS Release 8.10.105.0. If this upgrade is not performed, the conversion will fail.

To convert an 802.11ax AP with a CAPWAP image to an embedded wireless controller capable image, either download the controller image based on the automated image download process, use the conversion command, or convert through the WebUI.

Note When the AP is embedded wireless controller capable, the AP can participate in the primary AP election process. Only if the AP is elected as a primary, can it perform the controller functionality.

Converting an Embedded Wireless Controller Capable AP to a CAPWAP AP

To convert an 802.11ax AP from the embedded wireless controller network to a non-embedded wireless controller network, set the AP type to CAPWAP using the conversion command or the WebUI, respectively, and then plug it on to the controller network so that it joins the controller. If the image on the controller is different from the image on the AP, a new CAPWAP image is requested from the controller.

Converting a Single AP to CAPWAP or Embedded Wireless Controller Capable AP (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `>enable`	Enters privileged EXEC mode.

	Command or Action	Purpose
Step 2	**wireless ewc ap ap-type** *ap-name* {**capwap** | **ewc**} **Example:** `Device#wireless ewc-ap ap ap-type ap-name capwap`	Changes the AP to CAPWAP type or to the embedded controller type.

Example

```
wireless ewc-ap ap ap-type ap-name {capwap | ewc}
```

AP Conversion Deployment Scenarios

1. Standalone 802.11ax CAPWAP AP to start an embedded wireless controller network:

802.11ax AP	Embedded Wireless Controller Capable APs	Use-Case	Automatic Conversion
Standalone 802.11ax CAPWAP AP	Network does not exist.	To use a the standalone 802.11ax CAPWAP AP as the first AP for setting up the embedded wireless controller network.	Automatic conversion is not possible. You must download both the controller and the AP image using the supported image transfer protocols with AP command: `ap-type {capwap | ewc-ap} [<sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip> Controller ImagePath>]`

2. Non-802.11ax CAPWAP AP joining an existing embedded wireless controller network:

CAPWAP AP	Embedded Wireless Controller Capable APs	Use-Case	Automatic Conversion
CAPWAP AP - Neither AireOS-Mobility Express capable, or, embedded wireless controller capable AP, or, AireOS-Mobility Express capable Wave 2 APs.	Existing network	To bring in a CAPWAP AP which is not embedded wireless controller capable, into an existing embedded wireless controller network, to add one more AP to the existing network.	Yes, automatic conversion is possible. This is automatically taken care through the AP Join image download process.

3. 802.11ax AP joining an existing embedded wireless controller network:

Embedded Wireless Controller Capable AP	Embedded Wireless Controller Network	Use-Case	Automatic Conversion

802.11ax AireOS-CAPWAP AP or 802.11ax Catalyst CAPWAP AP or 802.11ax embedded wireless controller capable AP	Existing network	To bring in an 802.11ax AP from an AireOS-CAPWAP network, or, a CAPWAP network, or, from another embedded wireless controller network into an existing embedded wireless controller network, to add one more AP to the existing network.	Yes, automatic conversion takes place.
			This is automatically taken care through the AP Join image download process.
			If the AP type is explicitly set to CAPWAP, then the AP continues to act as a CAPWAP AP unless it is converted back again to embedded wireless controller AP using the AP command, Controller command, or the WebUI.
			The following command is used for conversion as well as AP image download:
			`ap-type {capwap | ewc-ap} [<sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip>Controller ImagePath>]`
			The following command is used to convert a specific AP to CAPWAP or embedded wireless controller:
			`wireless ewc-ap ap ap-type ap-name {capwap | ewc-ap}`

4. 802.11ax embedded wireless controller AP joining an AireOS CAPWAP network or a CAPWAP network:

802.11 AX Embedded Wireless Controller Capable AP	Embedded Wireless Controller Network	Use-Case	Automatic Conversion

802.11ax AP which was earlier an embedded wireless controller AP	Existing network	To bring an existing 802.11ax embedded wireless controller AP and add it to the CAPWAP network or the AireOS-CAPWAP network to add one more AP to the existing network.	It is recommended to convert the AP to CAPWAP type before bringing it to the CAPWAP network. This conversion can be done manually by using the AP command, the Controller command, Controller WebUI, or by using the DHCP option.
			After conversion, the normal image download process should be followed.
			`ap-type {capwap | ewc-ap} [<sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip>Controller ImagePath>]`
			`wireless ewc-ap ap ap-type` *`ap-name`* `{capwap | ewc-ap}`

Network Conversion

This section describes network conversion thorugh the conversion command and the network conversion deployment scenarios.

Converting the Network (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `>enable`	Enters privileged EXEC mode.

	Command or Action	Purpose
Step 2	**Wireless ewc-ap ap capwap** *primary-controller-name* {**A:B:C:D** | **X:X:X:X::X**} **Example:** `Device#wireless ewc-ap ap capwap wlc-name 10.0.0.0`	Specifies the wireless controller name and IP address to which all the APs currently connected to the embedded wireless controller network should join.

Network Conversion Deployment Scenarios

1. Converting an existing centralized CAPWAP network or AireOS CAPWAP network to the embedded wireless controller network

Existing Network	Embedded Wireless Controller Network	Use-Case	Automatic Conversion
CAPWAP Network: Centralized CAPWAP network or AireOS-CAPWAP network with at least one 802.11ax AP.	Network does not exist.	To convert the existing centralized CAPWAP network or the AireOS-CAPWAP network to theembedded wireless controller network.	No, automatic conversion does not take place. You need to pick one 802.11ax AP to download both the controller and AP image using the supported image transfer protocols with the AP command. `ap-type {capwap | ewc-ap} <sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip> Controller ImagePath>]`

2. Converting an existing embedded wireless controller network to an AireOS CAPWAP network or to a centralized CAPWAP network

Existing Network	Embedded Wireless Controller Network	Use-Case	Automatic Conversion
Embedded wireless controller network with many APs.	Existing network	To convert the existing embedded wireless controller network to an AireOS-CAPWAP network or to a centralized CAPWAP network.	No automatic conversion. You must convert all the APs or one AP at a time using the controller command to specify the IP address of the controller to which the AP has to join. You can also use the WebUI to convert the selected APs or all the APs by specifying the IP address of the controller to which the AP has to join.

SKU Conversion Scenarios

1. 802.11ax Embedded Wireless Controller SKU instead of CAPWAP SKU

SKU	Network	Use-Case	Automatic Conversion
802.11ax embedded wireless controller SKU instead of CAPWAP SKU	Network does not exist.	For an order placed for 802.11ax embedded wireless controller SKU instead of CAPWAP SKU, it should be converted to CAPWAP SKU.	No automatic conversion available. You can use **DHCP option 43** to point to the Catalyst 9800 controller so that the APs join the Catalyst 9800 controller as a CAPWAP AP.

	SKU	Network	Use-Case	Automatic Conversion
2.	802.11ax CAPWAP SKU instead of the embedded wireless controller SKU.	Network does not exist.	For an order placed for the 802.11ax CAPWAP SKU instead of the embedded wireless controller SKU and now would like to convert it to embedded wireless controller SKU.	No automatic conversion available. You should pick one 802.11ax AP to download both the controller and AP image using the supported image transfer protocols with AP command.`ap-type ewc-ap <sftp/tftp>://<server ip>/<AP imagepath> <sftp/tftp>://<server ip> Controller ImagePath>`

Converting AireOS Mobility Express Network to Embedded Wireless Controller Network

Procedure

Step 1 Remove the **Next Preferred Master** configuration from the existing AireOS Mobility Express network and save the configuration.

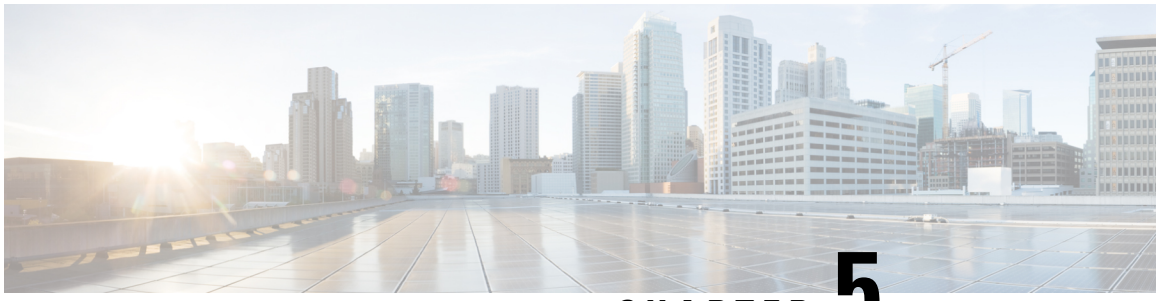
Step 2 Power down all the APs in the AireOS Mobility Express network including the primary AP.

Step 3 Power-on the 11 AX AP with the embedded wireless controller SKU so that it launches the controller.

Step 4 Provision the 11 AX AP with the required configuration (if the box is in Day-0, provision the mandatory configuration to get to Day-1).

Step 5 Copy, Translate, and Apply all the AireOS Mobility Express configurations to the 11 AX embedded wireless controller AP, add image download configuration.

Step 6 Power-on all the APs in the AireOS Mobility Express network. All the APs from the earlier AireOS Mobility Express network will join as regular APs in the embedded wireless controller network.



CHAPTER 5

Best Practices

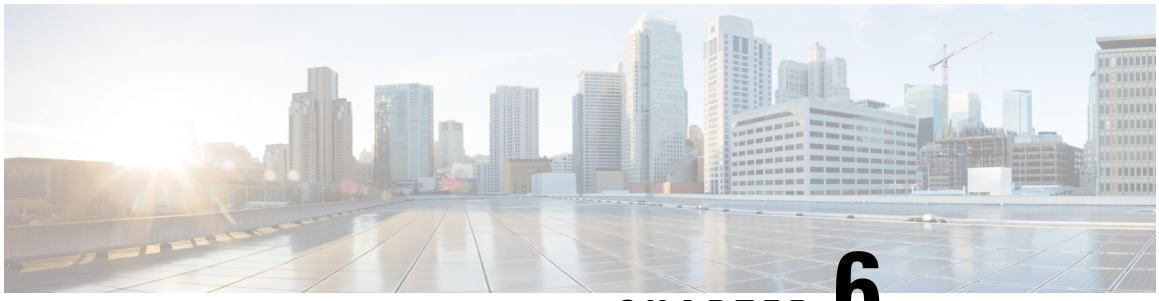
Introduction

This chapter covers the best practices recommended for configuring a typical Cisco Catalyst 9800 Series wireless infrastructure. The objective is to provide common settings that you can apply to most wireless network implementations. However, not all networks are the same. Therefore, some of the tips might not be applicable to your installation. Always verify them before you perform any changes on a live network.

For more information, see Cisco Catalyst 9800 Series Configuration Best Practices guide.

PART II

Lightweight Access Points



CHAPTER 6

Country Codes

Information About 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 within that regulatory domain (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 Japanese Country Codes

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 –U, –P, and –Q radios to join the controller
- J4: Allows 2.4G JPQU and 5G PQU 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.

Prerequisites for Configuring Country Codes

- Generally, you should configure one country code per device; you configure one code that matches the physical location of the device and its access points. You can configure up to 20 country codes per device. This multiple-country support enables you to manage access points in various countries from a single device.

- When the multiple-country feature is used, all the devices 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 will operate on. This list is independent of which countries have been configured on the RF group members.
- For devices in the Japan regulatory domain, you must have at least one access point with a -J regulatory domain joined to your device.
- You cannot delete any country code using the configuration command **wireless country country-code** if the specified country was configured using the **ap country list** command and vice-versa.

Configuring Country Codes (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points** > **Country**.

Step 2 On the **Country** page, select the check box for each country where your access points are installed. If you selected more than one check box, a message is displayed indicating that RRM channels and power levels are limited to common channels and power levels.

Step 3 Click **Apply**.

How to Configure Country Codes

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**show wireless country supported** **Example:** `Device# show wireless country supported`	Displays a list of all the available country codes.
Step 3	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 4	**ap dot11 24ghz shutdown** **Example:** `Device(config)# ap dot11 24ghz shutdown`	Disables the 802.11b/g network.
Step 5	**ap dot11 5ghz shutdown** **Example:** `Device(config)# ap dot11 5ghz shutdown`	Disables the 802.11a network.
Step 6	**ap country** *country_code* **Example:** `Device(config)# ap country IN`	
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 8	**show wireless country channels** **Example:** `Device# show wireless country channels`	Displays the list of available channels for the country codes configured on your device. **Note** Perform Steps 9 through 17 only if you have configured multiple country codes in Step 6.
Step 9	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 10	**no ap dot11 5ghz shutdown** **Example:** `Device(config)# no ap dot11 5ghz shutdown`	Enables the 802.11a network.
Step 11	**no ap dot11 24ghz shutdown** **Example:** `Device(config)# no ap dot11 24ghz shutdown`	Enables the 802.11b/g network.
Step 12	**end** **Example:** `Device(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:** `Device# ap name AP02 shutdown`	Disables the access point. **Note** Ensure that you disable only the access point for which you are configuring country codes.

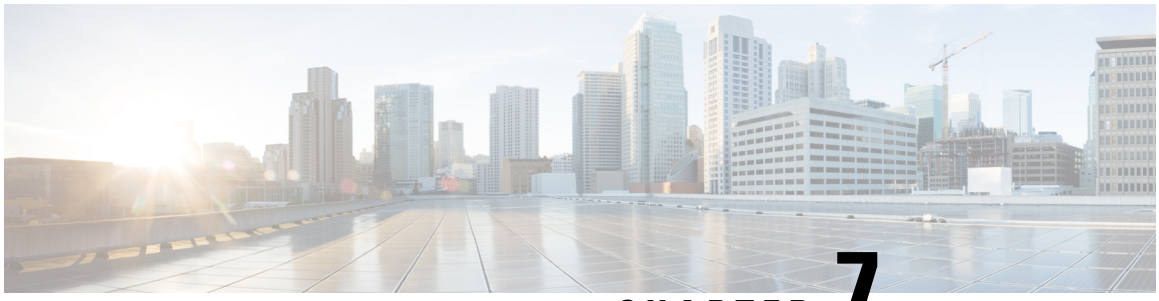
	Command or Action	Purpose
Step 14	**ap name** *cisco-ap* **no shutdown** **Example:** `Device# ap name AP02 no shutdown`	Enables the access point.

Configuration Examples for Configuring Country Codes

Viewing Channel List for Country Codes

These examples show how to display the list of available channels for the country codes on your device:

```
Device# 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
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          : . . . . . . . . . . . . . . . . . . . . . . . . . .

Device# show wireless country configured

Configured Country.......................... US - United States
Configured Country Codes
US - United States 802.11a Indoor,Outdoor/ 802.11b Indoor,Outdoor/ 802.11g Indoor,Outdoor
```



CHAPTER 7

AP Priority

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.

The following are some guidelines for configuring failover priority for access points:

- 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.
- Failover priority is not in effect during the regular operation of your wireless network. It takes effect only if there are more associations requests to controller than the avaiable AP capacity on the controller.
- AP priority is checked while connecting to the controller when the controller is in full scale or the primary controller fails, the APs fallback to the secondary controller.
- You can enable failover priority on your network and assign priorities to the individual access points.
- By default, all access points are set to priority level 1, which is the lowest priority level. Therefore, you need to assign a priority level only to those access points that warrant a higher priority.

Setting AP Priority (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 Click the Access Point.

Step 3 In the **Edit AP** dialog box, go to **High Availability** tab.

Step 4 Choose the priority from the **AP failover priority** drop-down list.

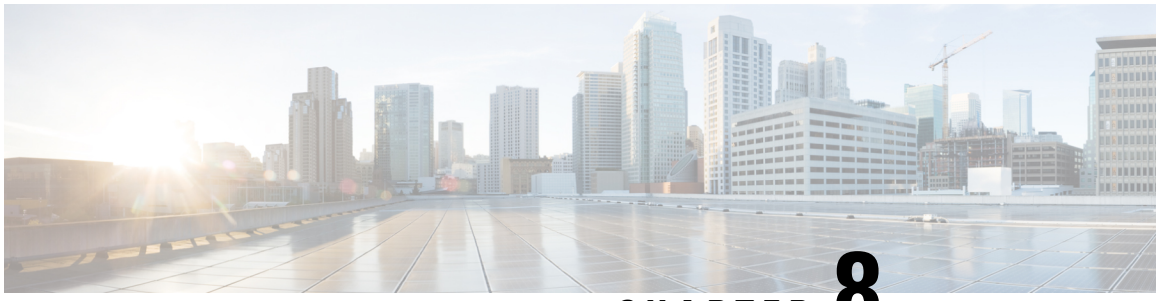
Step 5 Click **Update and Apply to Device**.

Setting AP Priority

Note Priority of access points ranges from 1 to 4, with 4 being the highest.

Procedure

	Command or Action	Purpose
Step 1	**ap name** *ap-name* **priority** *priority* **Example:** `Device# ap name AP44d3.ca52.48b5 priority 1`	Specifies the priority of an access point.
Step 2	**show ap config general** **Example:** `Device# show ap config general`	Displays common information for all access points.
Step 3	**show ap name** *ap-name* **config general** **Example:** `Device# show ap name AP44d3.ca52.48b5 config general`	Displays the configuration of a particular access point.



CHAPTER 8

802.11 Parameters for Cisco Access Points

2.4-GHz Radio Support

Configuring 2.4-GHz Radio Support for the Specified Slot Number

Before you begin

Note The term *802.11b radio* or *2.4-GHz radio* will be used interchangeably.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap name** *ap-name* **dot11 24ghz slot 0 SI** **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 SI`	Enables Spectrum Intelligence (SI) for the dedicated 2.4-GHz radio hosted on slot 0 for a specific access point. For more information, *Spectrum Intelligence* section in this guide.

	Command or Action	Purpose
		Here, **0** refers to the Slot ID.
Step 3	**ap name** *ap-name* **dot11 24ghz slot 0 antenna** {**ext-ant-gain** *antenna_gain_value* | **selection** [**internal** | **external**]} **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 antenna selection internal`	Configures 802.11b antenna hosted on slot 0 for a specific access point. • **ext-ant-gain**: Configures the 802.11b external antenna gain. *antenna_gain_value*- Refers to the external antenna gain value in multiples of .5 dBi units. The valid range is from 0 to 4294967295. • **selection**: Configures the 802.11b antenna selection (internal or external). **Note** • For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration. • For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model. • Cisco Catalyst 9120E and 9130E APs support self-identifying antennas (SIA). Cisco Catalyst 9115E APs do not support SIA antennas. Although Cisco Catalyst 9115E APs work with SIA antennas, the APs do not auto-detect SIA antennas nor add the correct external gain.
Step 4	**ap name** *ap-name* **dot11 24ghz slot 0 beamforming** **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 beamforming`	Configures beamforming for the 2.4-GHz radio hosted on slot 0 for a specific access point.

	Command or Action	Purpose
Step 5	**ap name** *ap-name* **dot11 24ghz slot 0 channel** {*channel_number* | **auto**} **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 channel auto`	Configures advanced 802.11 channel assignment parameters for the 2.4-GHz radio hosted on slot 0 for a specific access point.
Step 6	**ap name** *ap-name* **dot11 24ghz slot 0 cleanair** **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 cleanair`	Enables CleanAir for 802.11b radio hosted on slot 0 for a specific access point.
Step 7	**ap name** *ap-name* **dot11 24ghz slot 0 dot11n antenna** {**A** | **B** | **C** | **D**} **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 dot11n antenna A`	Configures 802.11n antenna for 2.4-GHz radio hosted on slot 0 for a specific access point. Here, **A**: Is the antenna port A. **B**: Is the antenna port B. **C**: Is the antenna port C. **D**: Is the antenna port D.
Step 8	**ap name** *ap-name* **dot11 24ghz slot 0 shutdown** **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 shutdown`	Disables 802.11b radio hosted on slot 0 for a specific access point.
Step 9	**ap name** *ap-name* **dot11 24ghz slot 0 txpower** {*tx_power_level* | **auto**} **Example:** `Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 txpower auto`	Configures transmit power level for 802.11b radio hosted on slot 0 for a specific access point. • *tx_power_level*: Is the transmit power level in dBm. The valid range is from 1 to 8. • **auto**: Enables auto-RF.

5-GHz Radio Support

Configuring 5-GHz Radio Support for the Specified Slot Number

Before you begin

Note The term *802.11a radio* or *5-GHz radio* will be used interchangeably in this document.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap name** *ap-name* **dot11 5ghz slot 1 SI** **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 SI`	Enables Spectrum Intelligence (SI) for the dedicated 5-GHz radio hosted on slot 1 for a specific access point. Here, **1** refers to the Slot ID.
Step 3	**ap name** *ap-name* **dot11 5ghz slot 1 antenna ext-ant-gain** *antenna_gain_value* **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna ext-ant-gain`	Configures external antenna gain for 802.11a radios for a specific access point hosted on slot 1. *antenna_gain_value*—Refers to the external antenna gain value in multiples of .5 dBi units. The valid range is from 0 to 4294967295. **Note** • For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration. • For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model. • Cisco Catalyst 9120E and 9130E APs support self-identifying antennas (SIA). Cisco Catalyst 9115E APs do not support SIA antennas. Although Cisco Catalyst 9115E APs work with SIA antennas, the APs do not auto-detect SIA antennas nor add the correct external gain.

	Command or Action	Purpose
Step 4	**ap name** *ap-name* **dot11 5ghz slot 1 antenna mode** [**omni** | **sectorA** | **sectorB**] **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna mode sectorA`	Configures the antenna mode for 802.11a radios for a specific access point hosted on slot 1.
Step 5	**ap name** *ap-name* **dot11 5ghz slot 1 antenna selection** [**internal** | **external**] **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna selection internal`	Configures the antenna selection for 802.11a radios for a specific access point hosted on slot 1.
Step 6	**ap name** *ap-name* **dot11 5ghz slot 1 beamforming** **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 beamforming`	Configures beamforming for the 5-GHz radio hosted on slot 1 for a specific access point.
Step 7	**ap name** *ap-name* **dot11 5ghz slot 1 channel** {*channel_number* | **auto** | **width** [**20** | **40** | **80** | **160**]} **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 channel auto`	Configures advanced 802.11 channel assignment parameters for the 5-GHz radio hosted on slot 1 for a specific access point. Here, *channel_number*- Refers to the channel number. The valid range is from 1 to 173.
Step 8	**ap name** *ap-name* **dot11 5ghz slot 1 cleanair** **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 cleanair`	Enables CleanAir for 802.11a radio hosted on slot 1 for a given or specific access point.
Step 9	**ap name** *ap-name* **dot11 5ghz slot 1 dot11n antenna** {**A** | **B** | **C** | **D**} **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 dot11n antenna A`	Configures 802.11n for 5-GHz radio hosted on slot 1 for a specific access point. Here, **A**- Is the antenna port A. **B**- Is the antenna port B. **C**- Is the antenna port C. **D**- Is the antenna port D.
Step 10	**ap name** *ap-name* **dot11 5ghz slot 1 rrm channel** *channel* **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 rrm channel 2`	Is another way of changing the channel hosted on slot 1 for a specific access point. Here, *channel*- Refers to the new channel created using 802.11h channel announcement. The valid range is from 1 to 173, provided 173 is

	Command or Action	Purpose
		a valid channel in the country where the access point is deployed.
Step 11	**ap name** *ap-name* **dot11 5ghz slot 1 shutdown** **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 shutdown`	Disables 802.11a radio hosted on slot 1 for a specific access point.
Step 12	**ap name** *ap-name* **dot11 5ghz slot 1 txpower** {*tx_power_level* | **auto**} **Example:** `Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 txpower auto`	Configures 802.11a radio hosted on slot 1 for a specific access point. • *tx_power_level*- Is the transmit power level in dBm. The valid range is from 1 to 8. • **auto**- Enables auto-RF.

Information About Dual-Band Radio Support

The Dual-Band (XOR) radio in Cisco 2800, 3800, 4800, and the 9120 series AP models offer the ability to serve 2.4–GHz or 5–GHz bands or passively monitor both the bands on the same AP. These APs can be configured to serve clients in 2.4–GHz and 5–GHz bands, or serially scan both 2.4–GHz and 5–GHz bands on the flexible radio while the main 5–GHz radio serves clients.

Cisco APs models up and through the Cisco 9120 APs are designed to support dual 5–GHz band operations with the *i* model supporting a dedicated Macro/Micro architecture and the *e* and *p* models supporting Macro/Macro. The Cisco 9130AXI APs and the Cisco 9136 APs support dual 5-GHz operations as Micro/Messo cell.

When a radio moves between bands (from 2.4-GHz to 5-GHz and vice versa), clients need to be steered to get an optimal distribution across radios. When an AP has two radios in the 5–GHz band, client steering algorithms contained in the Flexible Radio Assignment (FRA) algorithm are used to steer a client between the same band co-resident radios.

The XOR radio support can be steered manually or automatically:

- Manual steering of a band on a radio—The band on the XOR radio can only be changed manually.
- Automatic client and band steering on the radios is managed by the FRA feature that monitors and changes the band configurations as per site requirements.

Note RF measurement will not run when a static channel is configured on slot 1. Due to this, the dual band radio slot 0 will move only with 5–GHz radio and not to the monitor mode.

When slot 1 radio is disabled, RF measurement will not run, and the dual band radio slot 0 will be only on 2.4–GHz radio.

Configuring Default XOR Radio Support

Before you begin

Note The default radio points to the XOR radio hosted on slot 0.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap name** *ap-name* **dot11 dual-band antenna ext-ant-gain** *antenna_gain_value* **Example:** `Device# ap name ap-name dot11 dual-band antenna ext-ant-gain 2`	Configures the 802.11 dual-band antenna on a specific Cisco access point. *antenna_gain_value*: The valid range is from 0 to 40.
Step 3	**ap name** *ap-name* **[no] dot11 dual-band shutdown** **Example:** `Device# ap name ap-name dot11 dual-band shutdown`	Shuts down the default dual-band radio on a specific Cisco access point. Use the **no** form of the command to enable the radio.
Step 4	**ap name** *ap-name* **dot11 dual-band role manual client-serving** **Example:** `Device# ap name ap-name dot11 dual-band role manual client-serving`	Switchs to client–serving mode on the Cisco access point.
Step 5	**ap name** *ap-name* **dot11 dual-band band 24ghz** **Example:** `Device# ap name ap-name dot11 dual-band band 24ghz`	Switchs to 2.4-GHz radio band.
Step 6	**ap name** *ap-name* **dot11 dual-band txpower** {*transmit_power_level* | **auto**} **Example:**	Configures the transmit power for the radio on a specific Cisco access point.

	Command or Action	Purpose
	`Device# ap name ap-name dot11 dual-band txpower 2`	**Note** When an FRA-capable radio (slot 0 on 9120 AP[for instance]) is set to Auto, you cannot configure static channel and Txpower on this radio. If you want to configure static channel and Txpower on this radio, you will need to change the radio role to Manual Client-Serving mode.
Step 7	**ap name** *ap-name* **dot11 dual-band channel** *channel-number* **Example:** `Device# ap name ap-name dot11 dual-band channel 2`	Enters the channel for the dual band. *channel-number*—The valid range is from 1 to 173.
Step 8	**ap name** *ap-name* **dot11 dual-band channel auto** **Example:** `Device# ap name ap-name dot11 dual-band channel auto`	Enables the auto channel assignment for the dual-band.
Step 9	**ap name** *ap-name* **dot11 dual-band channel width** {**20 MHz** | **40 MHz** | **80 MHz** | **160 MHz**} **Example:** `Device# ap name ap-name dot11 dual-band channel width 20 MHz`	Chooses the channel width for the dual band.
Step 10	**ap name** *ap-name* **dot11 dual-band cleanair** **Example:** `Device# ap name ap-name dot11 dual-band cleanair`	Enables the Cisco CleanAir feature on the dual-band radio.

Step 11	**ap name** *ap-name* **dot11 dual-band cleanair band** {**24 GHz | 5 GMHz**}
	Example:
	`Device# ap name ap-name dot11 dual-band cleanair band 5 GHz`
	`Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz`
		Selects a band for the Cisco CleanAir feature.
		Use the **no** form of this command to disable the Cisco CleanAir feature.

Step 12	**ap name** *ap-name* **dot11 dual-band dot11n antenna** {**A | B | C | D**} **Example:**	Configures the 802.11n dual-band parameters for a specific access point.

	Command or Action	Purpose
	`Device# ap name ap-name dot11 dual-band dot11n antenna A`	
Step 13	**show ap name** *ap-name* **auto-rf dot11 dual-band** **Example:** `Device# show ap name ap-name auto-rf dot11 dual-band`	Displays the auto-RF information for the Cisco access point.
Step 14	**show ap name** *ap-name* **wlan dot11 dual-band** **Example:** `Device# show ap name ap-name wlan dot11 dual-band`	Displays the list of BSSIDs for the Cisco access point.

Configuring XOR Radio Support for the Specified Slot Number (GUI)

Procedure

Step 1 Click **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** section, select the AP for which you want to configure dual-band radios.

The AP name, MAC address, CleanAir capability and slot information for the AP are displayed. If the Hyperlocation method is HALO, the antenna PID and antenna design information are also displayed.

Step 3 Click **Configure**.

Step 4 In the **General** tab, set the **Admin Status** as required.

Step 5 Set the **CleanAir Admin Status** field to Enable or Disable.

Step 6 Click **Update & Apply to Device**.

Configuring XOR Radio Support for the Specified Slot Number

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.

	Command or Action	Purpose
Step 2	**ap name** *ap-name* **dot11 dual-band slot 0 antenna ext-ant-gain** *external_antenna_gain_value* **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 antenna ext-ant-gain 2`	Configures dual-band antenna for the XOR radio hosted on slot 0 for a specific access point. *external_antenna_gain_value* - Is the external antenna gain value in multiples of .5 dBi unit. The valid range is from 0 to 40. **Note** • For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration. • For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model.
Step 3	**ap name** *ap-name* **dot11 dual-band slot 0 band** {**24ghz** | **5ghz**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 band 24ghz`	Configures current band for the XOR radio hosted on slot 0 for a specific access point.
Step 4	**ap name** *ap-name* **dot11 dual-band slot 0 channel** {*channel_number* | **auto** | **width** [**160** | **20** | **40** | **80**]} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 channel 3`	Configures dual-band channel for the XOR radio hosted on slot 0 for a specific access point. *channel_number*- The valid range is from 1 to 165.
Step 5	**ap name** *ap-name* **dot11 dual-band slot 0 cleanair band** {**24Ghz** | **5Ghz**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 cleanair band 24Ghz`	Enables CleanAir features for dual-band radios hosted on slot 0 for a specific access point.
Step 6	**ap name** *ap-name* **dot11 dual-band slot 0 dot11n antenna** {**A** | **B** | **C** | **D**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 dot11n antenna A`	Configures 802.11n dual-band parameters hosted on slot 0 for a specific access point. Here, **A**- Enables antenna port A. **B**- Enables antenna port B.

	Command or Action	Purpose
		C- Enables antenna port C. **D**- Enables antenna port D.
Step 7	**ap name** *ap-name* **dot11 dual-band slot 0 role** {**auto** | **manual** [**client-serving** | **monitor**]} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 role auto`	Configures dual-band role for the XOR radio hosted on slot 0 for a specific access point. The following are the dual-band roles: • **auto**- Refers to the automatic radio role selection. • **manual**- Refers to the manual radio role selection.

Step 8	**ap name** *ap-name* **dot11 dual-band slot 0 shutdown**
	Example:
	`Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 shutdown`
	`Device# ap name AP-SIDD-A06 [no] dot11 dual-band slot 0 shutdown`
		Disables dual-band radio hosted on slot 0 for a specific access point.
		Use the **no** form of this command to enable the dual-band radio.

Step 9	**ap name** *ap-name* **dot11 dual-band slot 0 txpower** {*tx_power_level* | **auto**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 txpower 2`	Configures dual-band transmit power for XOR radio hosted on slot 0 for a specific access point. • *tx_power_level*- Is the transmit power level in dBm. The valid range is from 1 to 8. • **auto**- Enables auto-RF.

Receiver Only Dual-Band Radio Support

Information About Receiver Only Dual-Band Radio Support

This feature configures the dual-band Rx-only radio features for an access point with dual-band radios.

This dual-band Rx-only radio is dedicated for Analytics, Hyperlocation, Wireless Security Monitoring, and BLE AoA*.

This radio will always continue to serve in monitor mode, therefore, you will not be able to make any channel and *tx-rx* configurations on the 3rd radio.

Configuring Receiver Only Dual-Band Parameters for Access Points

Enabling CleanAir with Receiver Only Dual-Band Radio on a Cisco Access Point (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** settings, click the AP for which you want to configure the dual-band radios.

Step 3 In the **General** tab, enable the **CleanAir** toggle button.

Step 4 Click **Update & Apply to Device**.

Enabling CleanAir with Receiver Only Dual-Band Radio on a Cisco Access Point

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.

Step 2	**ap name** *ap-name* **dot11 rx-dual-band slot 2 cleanair band** {**24Ghz** | **5Ghz**}
	Example:
	`Device# ap name AP-SIDD-A06 dot11 rx-dual-band slot 2 cleanair band 24Ghz`
	`Device# ap name AP-SIDD-A06 [no] dot11 rx-dual-band slot 2 cleanair band 24Ghz`
		Enables CleanAir with receiver only (Rx-only) dual-band radio on a specific access point.
		Here, 2 refers to the slot ID.
		Use the **no** form of this command to disable CleanAir.

Disabling Receiver Only Dual-Band Radio on a Cisco Access Point (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** settings, click the AP for which you want to configure the dual-band radios.

Step 3 In the **General** tab, disable the **CleanAir Status** toggle button.

Step 4 Click **Update & Apply to Device**.

Disabling Receiver Only Dual-Band Radio on a Cisco Access Point

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.

Step 2	**ap name** *ap-name* **dot11 rx-dual-band slot 2 shutdown**
	Example:
	`Device# ap name AP-SIDD-A06 dot11 rx-dual-band slot 2 shutdown`
	`Device# ap name AP-SIDD-A06 [no] dot11 rx-dual-band slot 2 shutdown`
		Disables receiver only dual-band radio on a specific Cisco access point.
		Here, 2 refers to the slot ID.
		Use the **no** form of this command to enable receiver only dual-band radio.

Configuring Client Steering (CLI)

Before you begin

Enable Cisco CleanAir on the corresponding dual-band radio.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wireless macro-micro steering transition-threshold balancing-window** *number-of-clients(0-65535)* **Example:** `Device(config)# wireless macro-micro steering transition-threshold balancing-window 10`	Configures the micro-macro client load–balancing window for a set number of clients.
Step 4	**wireless macro-micro steering transition-threshold client count** *number-of-clients(0-65535)*	Configures the macro-micro client parameters for a minimum client count for transition.

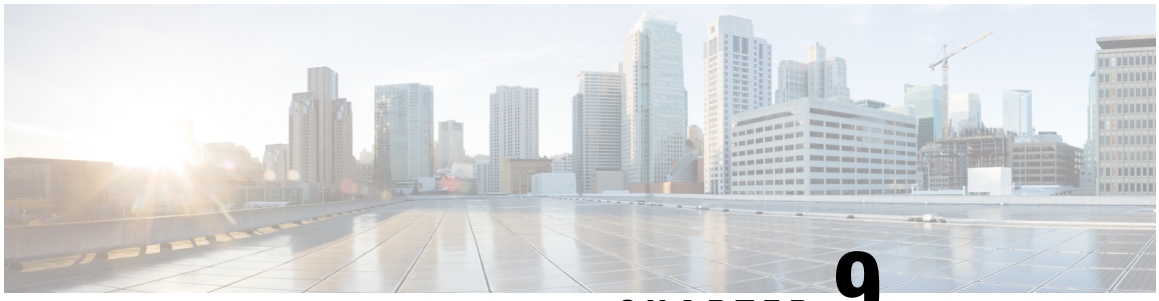
	Command or Action	Purpose
	Example: `Device(config)# wireless macro-micro steering transition-threshold client count 10`	
Step 5	**wireless macro-micro steering transition-threshold macro-to-micro** *RSSI-in-dBm(–128—0)* **Example:** `Device(config)# wireless macro-micro steering transition-threshold macro-to-micro -100`	Configures the macro–to–micro transition RSSI.
Step 6	**wireless macro-micro steering transition-threshold micro-to-macro** *RSSI-in-dBm(–128—0)* **Example:** `Device(config)# wireless macro-micro steering transition-threshold micro-to-macro -110`	Configures the micro–to–macro transition RSSI.
Step 7	**wireless macro-micro steering probe-suppression aggressiveness** *number-of-cycles(–128—0)* **Example:** `Device(config)# wireless macro-micro steering probe-suppression aggressiveness -110`	Configures the number of probe cycles to be suppressed.
Step 8	**wireless macro-micro steering probe-suppression hysteresis** *RSSI-in-dBm* **Example:** `Device(config)# wireless macro-micro steering probe-suppression hysteresis -5`	Configures the macro-to-micro probe in RSSI. The range is between –6 to –3.
Step 9	**wireless macro-micro steering probe-suppression probe-only** **Example:** `Device(config)# wireless macro-micro steering probe-suppression probe-only`	Enables probe suppression mode.
Step 10	**wireless macro-micro steering probe-suppression probe-auth** **Example:** `Device(config)# wireless macro-micro steering probe-suppression probe-auth`	Enables probe and single authentication suppression mode.

	Command or Action	Purpose
Step 11	**show wireless client steering** **Example:** `Device# show wireless client steering`	Displays the wireless client steering information.

Verifying Cisco Access Points with Dual-Band Radios

To verify the access points with dual-band radios, use the following command:

```
Device# show ap dot11 dual-band summary

AP Name Subband Radio      Mac   Status Channel Power Level Slot ID Mode
-------------------------------------------------------------------------------
4800   All 3890.a5e6.f360 Enabled (40)* *1/8     (22 dBm)      0   Sensor
4800   All 3890.a5e6.f360 Enabled N/A    N/A       2               Monitor
```



CHAPTER 9

802.1x Support

Introduction to the 802.1X Authentication

IEEE 802.1X port-based authentication is configure on a device to prevent unauthorized devices from gaining access to the network. The device can combine the function of a router, switch, and access point, depending on the fixed configuration. Any device connecting to a switch port where 802.1X authentication is enabled must go through relevant EAP authentication model to start exchanging traffic.

Currently, the Cisco Wave 2 and Wi-Fi 6 (802.11AX) APs support 802.1X authentication with switch port for EAP-FAST, EAP-TLS and EAP-PEAP methods. Now, you can enable configurations and provide credentials to the AP from the embedded controller.

Note If the AP is dot1x EAP-FAST, when the AP reboots, it should perform an anonymous PAC provision. For performing PAC provision, the ADH cipher suites should be used to establish an authenticated tunnel. If the ADH cipher suites are not supported by radius servers, AP will fail to authenticate on reload.

EAP-FAST Protocol

In the EAP-FAST protocol developed by Cisco, in order to establish a secured TLS tunnel with RADIUS, the AP requires a strong shared key (PAC), either provided via in-band provisioning (in a secured channel) or via out-band provisioning (manual).

Note The EAP-FAST type configuration requires 802.1x credentials configuration for AP, since AP will use EAP-FAST with MSCHAP Version 2 method.

Note Local EAP is not supported on the Cisco 7925 phones.

Note In Cisco Wave 2 APs, for 802.1x authentication using EAP-FAST after PAC provisioning (caused by the initial connection or after AP reload), ensure that you configure the switch port to trigger re-authentication using one of the following commands: **authentication timer restart** *num* or **authentication timer reauthenticate** *num*.

EAP-TLS/EAP-PEAP Protocol

The EAP-TLS protocol or EAP-PEAP protocol provides certificate based mutual EAP authentication.

In EAP-TLS, both the server and the client side certificates are required, where the secured shared key is derived for the particular session to encrypt or decrypt data. Whereas, in EAP-PEAP, only the server side certificate is required, where the client authenticates using password based protocol in a secured channel.

Note The EAP-PEAP type configuration requires Dot1x credentials configuration for AP; and the AP also needs to go through LSC provisioning. AP uses the PEAP protocol with MSCHAP Version 2 method.

Limitations of the 802.1X Authentication

- 802.1X is not supported on dynamic ports or Ethernet Channel ports.
- 802.1X is not supported in a mesh AP scenario.
- There is no recovery from the embedded controller on credential mismatch or the expiry/invalidity of the certificate on AP. The 802.1X authentication has to be disabled on the switch port to connect the AP back to fix the configurations.
- There are no certificate revocation checks implemented on the certificates installed in AP.
- Only one Locally Significant Certificates (LSC) can be provisioned on the AP and the same certificate must be used for CAPWAP DTLS session establishment with embedded controller and the 802.1X authentication with the switch. If global LSC configuration on the embedded controller is disabled; AP deletes LSC which is already provisioned.
- If clear configurations are applied on the AP, then the AP will lose the 802.1X EAP type configuration and the LSC certificates. AP should again go through staging process if 802.1X is required.

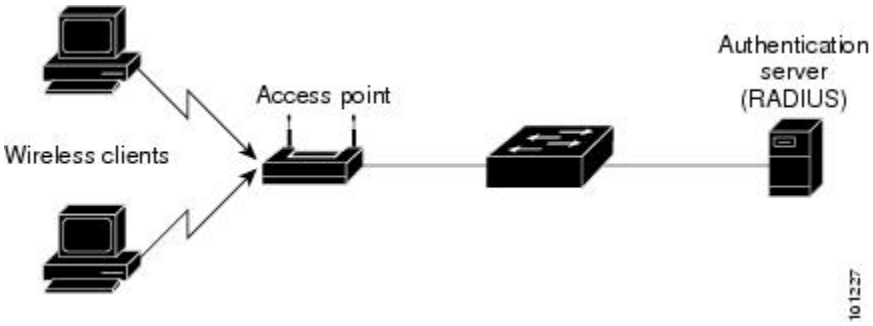
- 802.1X for trunk port APs on multi-host authentication mode is supported. Network Edge Authentication Topology (NEAT) is not supported on COS APs.

Topology - Overview

The 802.1X authentication events are as follows:

1. The AP acts as the 802.1X supplicant and is authenticated by the switch against the RADIUS server which supports EAP-FAST along with EAP-TLS and EAP-PEAP. When dot1x authentication is enabled on a switch port, the device connected to it authenticates itself to receive and forward data other than 802.1X traffic.
2. In order to authenticate with EAP-FAST method, the AP requires the credentials of the RADIUS server. It can be configured at the embedded controller, from where it will be passed on to the AP via configuration update request. For, EAP-TLS or EAP-PEAP the APs use the certificates (device/ID and CA) made significant by the local CA server.

Figure 13: Figure: 1 Topology for 802.1X Authentication



Configuring 802.1X Authentication Type and LSC AP Authentication Type (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** page, click **Add**.

The **Add AP Join Profile** page is displayed.

Step 3 In the **AP** > **General** tab, navigate to the **AP EAP Auth Configuration** section.

Step 4 From the **EAP Type** drop-down list, choose the EAP type as *EAP-FAST*, *EAP-TLS*, or *EAP-PEAP* to configure the dot1x authentication type.

Step 5 From the **AP Authorization Type** drop-down list, choose the type as either CAPWAP DTLS + or CAPWAP DTLS.

Step 6 Click **Save & Apply to Device**.

Configuring 802.1X Authentication Type and LSC AP Authentication Type

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ap profile** *profile-name* **Example:** `Device(config)# ap profile new-profile`	Specify a profile name.
Step 4	**dot1x {max-sessions | username | eap-type | lsc-ap-auth-state}** **Example:** `Device(config-ap-profile)# dot1x eap-type`	Configures the dot1x authentication type. **max-sessions**: Configures the maximum 802.1X sessions initiated per AP. **username**: Configures the 802.1X username for all Aps. **eap-type:** Configures the dot1x authentication type with the switch port. **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.
Step 5	**dot1x eap-type {EAP-FAST | EAP-TLS | EAP-PEAP}** **Example:** `Device(config-ap-profile)# dot1x eap-type`	Configures the dot1x authentication type: EAP-FAST, EAP-TLS, or EAP-PEAP.
Step 6	**dot1x lsc-ap-auth-state {CAPWAP-DTLS | Dot1x-port-auth | Both}** **Example:** `Device(config-ap-profile)#dot1x lsc-ap-auth-state Dot1x-port-auth`	Configures the LSC authentication state on the AP. **CAPWAP-DTLS**: Uses LSC only for CAPWAP DTLS. **Dot1x-port-auth**: Uses LSC only for dot1x authentication with port.

	Command or Action	Purpose
		Both: Uses LSC for both CAPWAP-DTLS and Dot1x authentication with port.
Step 7	**end** **Example:** `Device(config-ap-profile)# end`	Exits the AP profile configuration mode and enters privileged EXEC mode.

Configuring the 802.1X Username and Password (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.
Step 2 On the **AP Join** page, click the name of the AP Join profile or click **Add** to create a new one.
Step 3 Click the **Management** tab and then click the **Credentials** tab.
Step 4 Enter the local username and password details.
Step 5 Choose the appropriate local password type.
Step 6 Enter 802.1X username and password details.
Step 7 Choose the appropriate 802.1X password type.
Step 8 Enter the time in seconds after which the session should expire.
Step 9 Enable local credentials and/or 802.1X credentials as required.
Step 10 Click **Update & Apply to Device**.

Configuring the 802.1X Username and Password (CLI)

The following procedure configures the 802.1X password for all the APs:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ap profile** *profile-name* **Example:** `Device(config)# ap profile new-profile`	Specify a profile name.

	Command or Action	Purpose
Step 4	**dot1x {max-sessions | username | eap-type | lsc-ap-auth-state}** **Example:** `Device(config-ap-profile)# dot1x eap-type`	Configures the dot1x authentication type. **max-sessions**: Configures the maximum 802.1X sessions initiated per AP. **username**: Configures the 802.1X username for all Aps. **eap-type:** Configures the dot1x authentication type with the switch port. **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.
Step 5	**dot1x username** <username> **password {0 | 8}** <password> **Example:** `Device(config-ap-profile)#dot1x username username password 0 password`	Configures the dot1x password for all the APs. 0: Specifies an unencrypted password will follow. 8: Specifies an AES encrypted password will follow.

Enabling 802.1X on the Switch Port

The following procedure enables 802.1X on the switch port:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables AAA.
Step 4	**aaa authentication dot1x {default | listname} method1[method2...]** **Example:** `Device(config)# aaa authentication dot1x default group radius`	Creates a series of authentication methods that are used to determine user privilege to access the privileged command level so that the device can communicate with the AAA server.

	Command or Action	Purpose
Step 5	**aaa authourization network group** **Example:** `aaa authourization network group`	Enables AAA authorization for network services on 802.1X.
Step 6	**dot1x system-auth-control** **Example:** `Device(config)# dot1x system-auth-control`	Globally enables 802.1X port-based authentication.
Step 7	**interface type slot/port** **Example:** `Device(config)# interface fastethernet2/1`	Enters interface configuration mode and specifies the interface to be enabled for 802.1X authentication.
Step 8	**authentication port-control {auto | force-authorized | force-unauthorized}** **Example:** `Device(config-if)# authentication port-control auto`	Enables 802.1X port-based authentication on the interface. **auto**—Enables IEEE 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 Device requests the identity of the supplicant and begins relaying authentication messages between the supplicant and the authentication server. Each supplicant attempting to access the network is uniquely identified by the Device by using the supplicant MAC address. **force-authorized**-—Disables IEEE 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 IEEE 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 supplicant to authenticate. The Device cannot provide authentication services to the supplicant through the port.
Step 9	**dot1x pae [supplicant | authenticator | both]** **Example:** `Device(config-if)# dot1x pae authenticator`	Enables 802.1X authentication on the port with default parameters.

	Command or Action	Purpose
Step 10	**end** **Example:** `Device(config-if)# end`	Enters privileged EXEC mode.

Verifying 802.1X on the Switch Port

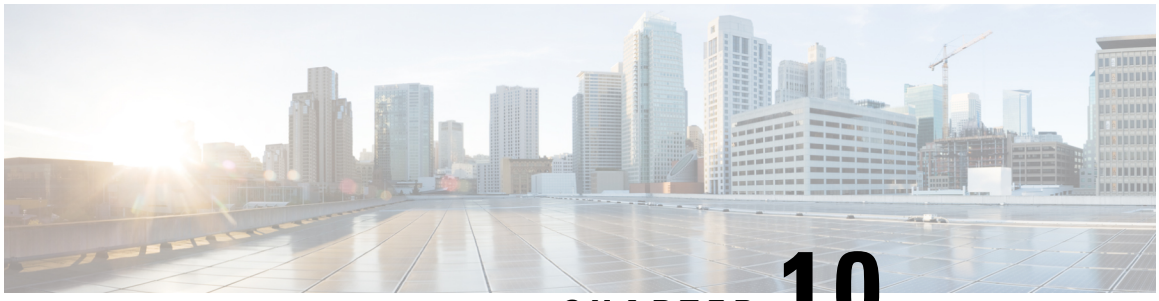
The following show command displays the authentication state of 802.1X on the switch port:

```
Device# show dot1x all
Sysauthcontrol              Enabled
Dot1x Protocol Version      2
Dot1x Info for FastEthernet1
-----------------------------------
PAE                       = AUTHENTICATOR
PortControl               = AUTO
ControlDirection          = Both
HostMode                  = MULTI_HOST
ReAuthentication          = Disabled
QuietPeriod               = 60
ServerTimeout             = 30
SuppTimeout               = 30
ReAuthPeriod              = 3600 (Locally configured)
ReAuthMax                 = 2
MaxReq                    = 2
TxPeriod                  = 30
RateLimitPeriod           = 0
Device#
```

Verifying the Authentication Type

The following show command displays the authentication state of an AP profile:

```
Device#show ap profile <profile-name> detailed ?
  chassis  Chassis
  |        Output modifiers
  <cr>

Device#show ap profile <profile-name> detailed

  AP Profile Name        : default-ap-profile
  Description            : default ap profile
  …
  Dot1x EAP Method       : [EAP-FAST/EAP-TLS/EAP-PEAP/Not-Configured]
  LSC AP AUTH STATE      : [CAPWAP DTLS / DOT1x port auth / CAPWAP DTLS + DOT1x port auth
```



CHAPTER 10

Real-Time Access Points Statistics

Information About Access Point Real-Time Statistics

From Cisco IOS XE Bengaluru 17.5.1 onwards, you can track the CPU utilization and memory usage of an AP, and monitor the health of an AP, by generating real-time statistics for an AP.

SNMP traps are defined for CPU and memory utilization of APs and the controller. An SNMP trap is sent out when the threshold is crossed. The sampling period and statistics interval can be configured using SNMP, YANG, and CLI.

Statistics interval is used to process the data coming from an AP, and the average CPU utilization and memory utilization is computed over time. You can also configure an upper threshold for these statistics. When a statistic value surpasses the upper threshold, an alarm is enabled, and an SNMP trap is triggered.

Configuring Access Point Real-Time Statistics (GUI)

Procedure

Step 1	Choose **Configuration** > **Tags & Profiles** > **AP Join**.
Step 2	Click **Add**. The **Add AP Join Profile** page is displayed.
Step 3	Click the **AP** tab.
Step 4	Under the **AP** tab, click the **AP Statistics** tab.
Step 5	Click the **Monitor Real Time Statistics** toggle button to **Enabled** status.
Step 6	Click the **Trigger Alarm for AP** toggle button to **Enabled** status.

Step 7 In the **CPU Threshold to Trigger Alarm** field, enter the threshold percentage of CPU usage. When the CPU usage crosses this threshold, an alarm is triggered.

Step 8 In the **Memory Threshold to Trigger Alarm** field, enter the threshold percentage of memory usage. When the memory usage exceeds this threshold, an alarm is triggered.

Step 9 In the **Interval to Hold Alarm** field, enter the time, in seconds, for which the alarm is held before it gets triggered.

Step 10 In the **Trap Retransmission Time** field, enter the time, in seconds, between retransmissions of the alarm.

Step 11 In the **Sampling Interval** field, enter the value, in seconds.The sampling interval defines how often data is collected from the AP.

Step 12 In the **Statistics Interval** field, enter the value, in seconds.The statistics interval defines the interval for which statistics are to be calculated for the AP.

Step 13 Click **Apply to Device** to save the configuration.

Configuring Access Point Real-Time Statistics (CLI)

To configure AP real-time statistics for an AP profile, follow the steps given below.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *ap-profile-name* **Example:** `Device(config)# ap profile ap-profile-name`	Configures the AP profile. The default AP join profile name is *default-ap-profile*.
Step 3	**stats-timer** *0-65535* **Example:** `Device(config-ap-profile)# stats-timer 60`	Configures the statistics timer. This command is used to change the frequency of the statistics reports coming from the AP.
Step 4	**statistics ap-system-monitoring enable** **Example:** `Device(config-ap-profile)# statistics ap-system-monitoring enable`	Enables monitoring of AP real-time statistics (CPU and memory).
Step 5	**statistics ap-system-monitoring alarm-enable** **Example:** `Device(config-ap-profile)# statistics ap-system-monitoring alarm-enable`	Enables alarms for AP real-time statistics (CPU and memory).

	Command or Action	Purpose
Step 6	**statistics ap-system-monitoring cpu-threshold** *<0-100> percentage* **Example:** `Device(config-ap-profile)# statistics ap-system-monitoring cpu-threshold 90`	Defines the threshold for CPU usage on the AP (percentage) to trigger alarms.
Step 7	**statistics ap-system-monitoring mem-threshold** *<0-100> percentage* **Example:** `Device(config-ap-profile)# statistics ap-system-monitoring mem-threshold 90`	Define the threshold for used memory usage on the AP (percentage) to trigger an alarm.
Step 8	**exit** **Example:** `Device(config-ap-profile)# exit`	Exits from AP profile configuration mode and returns to global configuration mode.
Step 9	**trapflags ap ap-stats** **Example:** `Device(config)# trapflags ap ap-stats`	Enables or disables sending AP-related traps. Traps are sent when statistics exceed the configured threshold.

Example

The following example shows how to configure AP real-time statistics.

```
Device(config)# ap profile default-policy-profile
    Device(config-ap-profile)# statistics ap-system-monitoring enable
    Device(config-ap-profile)#statistics ap-system-monitoring sampling-interval 90
    Device(config-ap-profile)#statistics ap-system-monitoring stats-interval 120
    Device(config-ap-profile)#statistics ap-system-monitoring alarm-enable
    Device(config-ap-profile)#statistics ap-system-monitoring alarm-hold-time 3
    Device(config-ap-profile)#statistics ap-system-monitoring alarm-retransmit-time 10
    Device(config-ap-profile)#statistics ap-system-monitoring cpu-threshold 90
    Device(config-ap-profile)#statistics ap-system-monitoring mem-threshold 90
Device(config)# trapflags ap ap-stats
```

Note The **sampling-interval**, **stats-interval**, **alarm-enable**, **alarm-hold-time**, and **alarm-retransmit**, keyword configurations are optional.

Monitoring Access Point Real-Time Statistics (GUI)

Procedure

Step 1 Choose **Monitoring** > **Wireless** > **AP Statistics**.

Step 2 Click the **General** tab.

Step 3 Click an AP name. The **General** window is displayed.

Step 4 To view the AP Statistics data, click the **AP Statistics** tab.

The following information is displayed:

- **Memory alarm last send time**: Displays the time of the last memory trap sent.
- **Memory Alarm Status**: Displays the state of the memory alarm. An alarm can be **ACTIVE**, **INACTIVE**, **INACTIVE_SOAKING**, **ACTIVE_SOAKING**. An alarm is soaked until the configured hold time has passed.
- **Memory alarm raise time**: Displays the last time the memory alarm was active.
- **Memory alarm clear time**: Displays the last time the memory alarm was inactive.
- **Last statistics received**: Displays the time of the last statistics report received from the AP.
- **Current CPU Usage**: Displays the latest percentage of CPU usage reported.
- **Average CPU Usage**: Displays the average CPU usage calculated.
- **Current Memory Usage**: Displays the latest percentage of memory usage reported.
- **Average Memory Usage**: Displays the average memory usage calculated.
- **Current window size**: Displays the window size. The window size is calculated by dividing the statistics interval by the sampling interval. The average CPU and memory usage is calculated by the window size.
- **CPU alarm last send time**: Displays the time of the last CPU trap sent.
- **CPU Alarm Status**: Displays the state of the CPU alarm. An alarm can be **ACTIVE**, **INACTIVE**, **INACTIVE_SOAKING**, **ACTIVE_SOAKING**. An alarm is soaked until the configured hold time has passed.
- **CPU alarm raise time**: Displays the last time the CPU alarm was active.
- **CPU alarm clear time**: Displays the last time the CPU alarm was inactive.

Step 5 Click **OK**.

Verifying Access Point Real-Time Statistics

To verify AP real-time statistics, run the **show ap config general | section AP statistics** command:

```
Device# show ap config general | section AP statistics
!Last Statistics
AP statistics : Enabled
Current CPU usage : 4
Average CPU usage : 49
Current memory usage : 35
Average memory usage : 35
Last statistics received : 03/09/2021 15:25:08
!Statistics Configuration
Current window size : 1
Sampling interval : 30
```

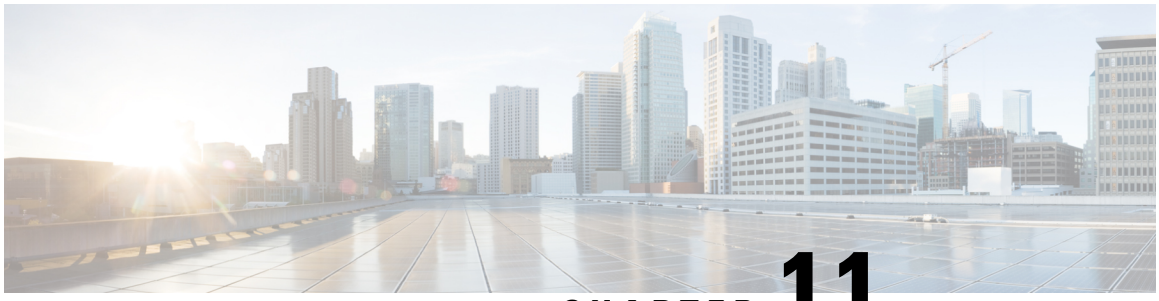
```
Statistics interval : 300
AP statistics alarms : Enabled
!Alarm State - Active, Inactive, Inactive_Soaking, Inactive_Soaking
Memory alarm status : Active
Memory alarm raise time : 03/09/2021 15:24:29
Memory alarm clear time : NA
Memory alarm last send time : 03/09/2021 15:24:59
CPU alarm status : Inactive
CPU alarm raise time : 03/09/2021 15:24:25
CPU alarm clear time : 03/09/2021 15:25:05
CPU alarm last send time : 03/09/2021 15:25:05
!Alarm Configuration
Alarm hold time : 6
Alarm retransmission time : 30
Alarm threshold cpu : 30
Alarm threshold memory : 32
```

To verify the statistics reporting period, run the **show ap config general | i Stats Reporting Period** command:

```
Device# show ap config general | i Stats Reporting Period
Stats Reporting Period : 10
```

PART III

Radio Resource Management



CHAPTER 11

Radio Resource Management

Information About Radio Resource Management

The Radio Resource Management (RRM) software that is embedded in the device acts as a built-in Radio Frequency (RF) engineer to consistently provide real-time RF management of your wireless network. RRM enables devices 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
- Power control transmission
- Dynamic channel assignment
- Coverage hole detection and correction
- RF grouping

Note RRM grouping does not occur when an AP operates in a static channel that is not in the DCA channel list. The Neighbor Discovery Protocol (NDP) is sent only on DCA channels; therefore, when a radio operates on a non-DCA channel, it does not receive NDP on the channel.

Radio Resource Monitoring

RRM automatically detects and configures new devices and lightweight access points as they are added to the network. It then automatically adjusts the associated and nearby lightweight access points to optimize coverage and capacity.

Lightweight access points can scan all the valid channels for the country of operation as well as for channels available in other locations. The access points in local mode go *offchannel* for a period not greater than 70 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 or other critical traffic (in the last 100 ms), access points can defer off-channel measurements. The access points also defer off-channel measurements based on the WLAN scan priority configurations.

Each access point spends only 0.2 percent of its time off channel. This activity is distributed across all the 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 device dynamically controls access point transmit power based on the real-time wireless LAN conditions.

The Transmit Power Control (TPC) algorithm 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 the surrounding access points. This feature is different from coverage hole detection, which is primarily concerned with clients. TPC provides enough RF power to achieve the required coverage levels while avoiding channel interference between access points. We recommend that you select TPCv1; TPCv2 option is deprecated. With TPCv1, you can select the channel aware mode; we recommend that you select this option for 5 GHz, and leave it unchecked for 2.4 GHz.

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 the access points must be mounted in a central hallway, placing the access points close together, but requiring coverage 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 the access points through RF profiles in a RF network.

To set the Maximum Power Level Assignment and Minimum Power Level Assignment, enter the maximum and minimum transmit power used by RRM in the fields in the **Tx Power Control** window. 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, no access point will 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 an e-mail in a café affects the performance of the access point in a neighboring business. Even though these are separate networks, someone sending traffic to the café on channel 1 can disrupt communication in an enterprise using the same channel. Devices can dynamically allocate access point channel assignments to avoid conflict and 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 device'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 simultaneously use 11 or 54 Mbps. By effectively reassigning channels, the device keeps adjacent channels that are separated.

Note We recommend that you use only nonoverlapping channels (1, 6, 11, and so on).

Note Channel change does not require you to shut down the radio.

The device 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 device 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 a part of your wireless LAN, including rogue access points and neighboring wireless networks. Lightweight access points constantly scan all the 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 device. Using the

RRM algorithms, the device 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 device 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 device may choose to avoid this channel. In huge deployments in which all nonoverlapping channels are occupied, the device does its best, but you must consider RF density when setting expectations.

- Load and 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 device can then assign channels to improve the access point that has performed the worst. The load is taken into account when changing the channel structure to minimize the impact on the clients that are 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 *Load and utilization* parameter is disabled by default.

The device 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 In a Dynamic Frequency Selection (DFS) enabled AP environment, ensure that you enable the UNII2 channels option under the DCA channel to allow 100-MHz separation for the dual 5-GHz radios.

The RRM startup mode is invoked in the following conditions:

- In a single-device environment, the RRM startup mode is invoked after the device is upgraded and rebooted.
- In a multiple-device environment, the RRM startup mode is invoked after an RF Group leader is elected.
- You can trigger the RRM startup mode from the CLI.

The 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.

Note DCA algorithm interval is set to 1 hour, but DCA algorithm always runs in default interval of 10 min, channel allocation occurs at 10-min intervals for the first 10 cycles, and channel changes occur as per the DCA algorithm every 10 min. After that the DCA algorithm goes back to the configured time interval. This is common for both DCA interval and anchor time because it follows the steady state.

Note If Dynamic Channel Assignment (DCA)/Transmit Power Control (TPC) is turned off on the RF group member, and auto is set on RF group leader, the channel or TX power on a member gets changed as per the algorithm that is run on the RF group leader.

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 device. 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 device discriminates between coverage holes that can and cannot be corrected. For coverage holes that can be corrected, the device mitigates the coverage hole by increasing the transmit power level for that specific access point. The device 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.

Restrictions for Radio Resource Management

- If an 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.

How to Configure RRM

Configuring Neighbor Discovery Type (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {24ghz | 5ghz} rrm ndp-type** {**protected | transparent**} **Example:** `Device(config)#ap dot11 24ghz rrm`	Configures the neighbor discovery type. By default, the mode is set to "transparent". • **protected**: Sets the neighbor discover type to protected. Packets are encrypted.

	Command or Action	Purpose
	ndp-type protected `Device(config)#`**ap dot11 24ghz rrm ndp-type transparent**	• **transparent**: Sets the neighbor discover type to transparent. Packets are sent as is.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Transmit Power Control

Configuring the Tx-Power Control Threshold (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11** {**24ghz** | **5ghz**} **rrm tpc-threshold** *threshold_value* **Example:** `Device(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:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring the Tx-Power Level (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {24ghz | 5ghz} rrm txpower**{*trans_power_level* | **auto** | **max** | **min** | **once**} **Example:**	Configures the 802.11 tx-power level • **trans_power_level**—Sets the transmit power level.

	Command or Action	Purpose
	`Device(config)#ap dot11 24ghz rrm txpower auto`	• **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:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring 802.11 RRM Parameters

Configuring Advanced 802.11 Channel Assignment Parameters (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11** {**24ghz** | **5ghz**} **rrm channel cleanair-event sensitivity** {**high** | **low** | **medium**} **Example:** `Device(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high`	Configures CleanAir event-driven RRM parameters. • **High**–Specifies the most sensitivity to non-Wi-Fi interference as indicated by the air quality (AQ) value. • **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 3	**ap dot11** {**24ghz** | **5ghz**} **rrm channel dca** { | **anchor-time** | **global** {**auto** | **once**} | **interval** | **min-metric** | **sensitivity** {**high** | **low** | **medium**}} **Example:** `Device(config)#ap dot11 24ghz rrm channel`	Configures Dynamic Channel Assignment (DCA) algorithm parameters for the 802.11 band. • –Enter a channel number to be added to the DCA list.

	Command or Action	Purpose

	`dca interval 2`	• **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.
		• **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 5ghz rrm channel dca chan-width** {**20** | **40** | **80**} **Example:** `Device(config)#ap dot11 5ghz rrm channel dca chan-width best`	Configures the DCA channel bandwidth for all 802.11 radios in the 5-GHz band. Sets the channel bandwidth to 20 MHz, 40 MHz, or 80 MHz, ; 20 MHz is the default value for channel bandwidth. 80 MHz is the default value for best. Set the channel bandwidth to best before configuring the constraints.
Step 5	**ap dot11** {**24ghz** | **5ghz**} **rrm channel device** **Example:** `Device(config)#ap dot11 24ghz rrm channel device`	Configures the persistent non-Wi-Fi device avoidance in the 802.11 channel assignment.
Step 6	**ap dot11** {**24ghz** | **5ghz**} **rrm channel foreign** **Example:** `Device(config)#ap dot11 24ghz rrm channel foreign`	Configures the foreign AP 802.11 interference avoidance in the channel assignment.

	Command or Action	Purpose
Step 7	**ap dot11** {**24ghz** | **5ghz**} **rrm channel load** **Example:** `Device(config)#ap dot11 24ghz rrm channel load`	Configures the Cisco AP 802.11 load avoidance in the channel assignment.
Step 8	**ap dot11** {**24ghz** | **5ghz**} **rrm channel noise** **Example:** `Device(config)#ap dot11 24ghz rrm channel noise`	Configures the 802.11 noise avoidance in the channel assignment.
Step 9	**end** **Example:** `Device(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 (CLI)

Procedure

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

Step 2	**ap dot11 {24ghz | 5ghz} rrm coverage data**{**fail-percentage | packet-count | rssi-threshold**}
	Example:
	`Device(config)#ap dot11 24ghz rrm coverage`
	`data fail-percentage 60`
		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:** `Device(config)#ap dot11 24ghz rrm`	Configures the 802.11 Cisco AP coverage exception level as a percentage that ranges from 0 to 100%.

	Command or Action	Purpose
	`coverage exception global 50`	
Step 4	**ap dot11 {24ghz | 5ghz} rrm coverage level global** *cli_min exception level* **Example:** `Device(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.
Step 5	**ap dot11 {24ghz | 5ghz} rrm coverage voice**{**fail-percentage | packet-count | rssi-threshold**} **Example:** `Device(config)#ap dot11 24ghz rrm coverage voice packet-count 10`	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:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring 802.11 Event Logging (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm logging**{**channel | coverage | foreign | load | noise | performance | txpower**} **Example:** `Device(config)#ap dot11 24ghz rrm logging channel`	Configures event-logging for various parameters. • **channel**—Configures the 802.11 channel change logging mode. • **coverage**—Configures the 802.11 coverage profile logging mode.

	Command or Action	Purpose

	`Device(config)#ap dot11 24ghz rrm logging coverage`
	`Device(config)#ap dot11 24ghz rrm logging foreign`
	`Device(config)#ap dot11 24ghz rrm logging load`
	`Device(config)#ap dot11 24ghz rrm logging noise`
	`Device(config)#ap dot11 24ghz rrm logging performance`
	`Device(config)#ap dot11 24ghz rrm logging txpower`
		• **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:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring 802.11 Statistics Monitoring (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 24ghz | 5ghz rrm monitor channel-list**{**all | country | dca**} **Example:** `Device(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:** `Device(config)#ap dot11 24ghz rrm monitor coverage 600`	Configures the 802.11 coverage measurement interval in seconds that ranges from 60 to 3600.

	Command or Action	Purpose
Step 4	**ap dot11 24ghz | 5ghz rrm monitor load** *interval* **Example:** `Device(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:** `Device(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.
Step 6	**ap dot11 24ghz | 5ghz rrm monitor signal** *interval* **Example:** `Device(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:** `Device(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 (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11** {**24ghz | 5ghz**} **rrm profile clients** *cli_threshold_value* **Example:** `Device(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:**	Sets the threshold value for 802.11 foreign interference that ranges between 0 and 100%.

	Command or Action	Purpose
	`Device(config)#ap dot11 24ghz rrm profile foreign 50`	
Step 4	**ap dot11** {**24ghz** | **5ghz**} **rrm profile noise** *for_noise_threshold_value* **Example:** `Device(config)#ap dot11 24ghz rrm profile noise -65`	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:** `Device(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.
Step 6	**ap dot11** {**24ghz** | **5ghz**} **rrm profile utilization** *rf_util_threshold_value* **Example:** `Device(config)#ap dot11 24ghz rrm profile utilization 75`	Sets the threshold value for 802.11 RF utilization that ranges between 0 to 100%.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring Advanced 802.11 RRM

Enabling Channel Assignment (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap dot11** {**24ghz** | **5ghz**} **rrm channel-update** **Example:** `Device# ap dot11 24ghz rrm channel-update`	Enables the 802.11 channel selection update for each of the Cisco access points. **Note** After you enable **ap dot11 {24ghz | 5ghz} rrm channel-update**, a token is assigned for channel assignment in the DCA algorithm.

Restarting DCA Operation

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap dot11** {**24ghz** | **5ghz**} **rrm dca restart** **Example:** `Device# ap dot11 24ghz rrm dca restart`	Restarts the DCA cycle for 802.11 radio.

Updating Power Assignment Parameters (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap dot11** {**24ghz** | **5ghz**} **rrm txpower update** **Example:** `Device# ap dot11 24ghz rrm txpower update`	Updates the 802.11 transmit power for each of the Cisco access points.

Configuring Rogue Access Point Detection in RF Groups

Configuring Rogue Access Point Detection in RF Groups (CLI)

Before you begin

Ensure that each embedded controller in the RF group has been configured with the same RF group name.

Note The name is used to verify the authentication IE in all beacon frames. If the embedded controller have different names, false alarms will occur.

Procedure

	Command or Action	Purpose
Step 1	**Example:** `Device#`	Perform this step for every access point connected to the embedded controller. • **monitor**:Sets the AP mode to monitor mode. • **clear**: Resets AP mode to local or remote based on the site. • **sensor**: Sets the AP mode to sensor mode. • **sniffer**: Sets the AP mode to wireless sniffer mode.
Step 2	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 3	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 4	**wireless wps ap-authentication** **Example:** `Device (config)# wireless wps ap-authentication`	Enables rogue access point detection.
Step 5	**wireless wps ap-authentication threshold** *value* **Example:** `Device (config)# wireless wps ap-authentication threshold 50`	Specifies when a rogue access point alarm is generated. An alarm occurs when the threshold value (which specifies the number of access point frames with an invalid authentication IE) is met or exceeded within the detection period. The valid threshold range is from 1 to 255, and the default threshold value is 1. To avoid false alarms, you may want to set the threshold to a higher value. **Note** Enable rogue access point detection and threshold value on every embedded controller in the RF group. **Note** If rogue access point detection is not enabled on every embedded controller in the RF group, the access points on the embedded controller with this feature disabled are reported as rogues.

Monitoring RRM Parameters and RF Group Status

Monitoring RRM Parameters

Table 11: Commands for monitoring Radio Resource Management

Commands	Description
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.
show ap dot11 24ghz group	Displays the configuration and statistics of the 802.11b grouping.
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 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 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 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 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.

Verifying RF Group Status (CLI)

This section describes the new commands for RF group status.

The following commands can be used to verify RF group status on the .

Table 12: Verifying Aggressive Load Balancing Command

Command	Purpose

show ap dot11 5ghz group	Displays the controller name which is the RF group leader for the 802.11a RF network.
show ap dot11 24ghz group	Displays the controller name which is the RF group leader for the 802.11b/g RF network.

Examples: RF Group Configuration

This example shows how to configure RF group name:

```
Device# configure terminal
Device(config)# wireless rf-network test1
Device(config)# ap dot11 24ghz shutdown
Device(config)# end
Device # show network profile 5
```

This example shows how to configure rogue access point detection in RF groups:

```
Device#
Device# end
Device# configure terminal
Device(config)# wireless wps ap-authentication
Device(config)# wireless wps ap-authentication threshold 50
Device(config)# end
```

Information About ED-RRM

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.

Configuring ED-RRM on the Cisco Wireless LAN Controller (CLI)

Procedure

Step 1 Trigger spectrum event-driven radio resource management (RRM) to run when a Cisco CleanAir-enabled access point detects a significant level of interference by entering these commands:

ap dot11 {**24ghz** | **5ghz**} **rrm channel cleanair-event** —Configures CleanAir driven RRM parameters for the 802.11 Cisco lightweight access points.

ap dot11 {**24ghz** | **5ghz**} **rrm channel cleanair-event sensitivity** {**low** | **medium** | **high** | **custom**}—Configures CleanAir driven RRM sensitivity for the 802.11 Cisco lightweight access points. Default selection is Medium.

ap dot11 {**24ghz** | **5ghz**} **rrm channel cleanair-event rogue-contribution**—Enables rogue contribution.

ap dot11 {**24ghz** | **5ghz**} **rrm channel cleanair-event rogue-contribution duty-cycle** *thresholdvalue*—Configures threshold value for rogue contribution. The valid range is from 1 to 99, with 80 as the default.

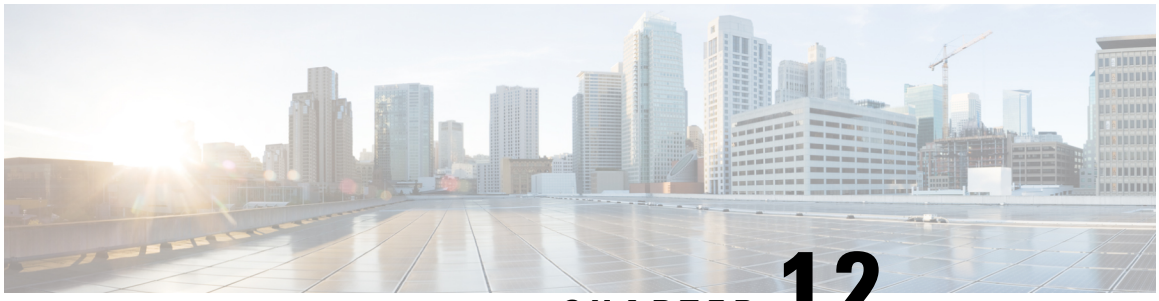
Step 2 Save your changes by entering this command:

write memory

Step 3 See the CleanAir configuration for the 802.11a/n/ac or 802.11b/g/n network by entering this command:

show ap dot11 {**24ghz** | **5ghz**} **cleanair config**

Information similar to the following appears:



CHAPTER 12

Coverage Hole Detection

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 device. 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 device discriminates between coverage holes that can and cannot be corrected. For coverage holes that can be corrected, the device mitigates the coverage hole by increasing the transmit power level for that specific access point. The device 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.

Configuring Coverage Hole Detection (GUI)

Follow the procedure given below to configure client accounting.

Procedure

Step 1 Click **Configuration** > **Radio Configurations** > **RRM**.

On this page, you can configure Radio Resource Management parameters for 802.11a/n/ac (5 GHZ) and 802.11b/g/n (2.4 GHZ) radios, and flexible radio assignment parameters.

Step 2 Check the **Enable Coverage Hole Detection** check box.

Enables coverage hole detection.

Configuring Coverage Hole Detection (CLI)

Coverage Hole Detection (CHD) is based on upstream RSSI metrics observed by the AP.

Follow the procedure given below to configure CHD:

Before you begin

Disable the 802.11 network before applying the configuration.

Procedure

	Command or Action	Purpose
Step 1	**ap dot11 {24ghz | 5ghz} rrm coverage data** {**fail-percentage | packet-count | rssi-threshold**} **Example:** `Device(config)# ap dot11 24ghz rrm coverage data fail-percentage 60`	Configures the 802.11 coverage level 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 2	**ap dot11 {24ghz | 5ghz} rrm coverage exception global** *exception level* **Example:** `Device(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 3	**ap dot11{24ghz | 5ghz}rrm coverage level global** *cli_min exception level* **Example:** `Device(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.
Step 4	**ap dot11 {24ghz | 5ghz} rrm coverage voice** {**fail-percentage | packet-count | rssi-threshold**} **Example:** `Device(config)# ap dot11 24ghz rrm`	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%.

	Command or Action	Purpose
	`coverage` `voice packet-count 10`	• **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 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 6	**show ap dot11 {24ghz | 5ghz} coverage** **Example:** `Device# show ap dot11 5ghz coverage`	Displays the CHD details.

Note If both the number and percentage of failed packets exceed the values entered in the **packet-count** and **fail-rate** commands for a 5-second period, the client is considered to be in a pre-alarm condition. The controller uses this information to distinguish between real and false coverage holes. False positives are generally due to the poor roaming logic implemented on most clients. A coverage hole is detected if both the number and percentage of failed clients meet or exceed the values entered in the **coverage level global** and **coverage exception global** commands over a 90-second period. The controller determines if the coverage hole can be corrected and, if appropriate, mitigates the coverage hole by increasing the transmit power level for that specific access point.

Configuring CHD for RF Tag Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **RRM**.

Step 2 On the **Coverage** tab, select the **Enable Coverage Hole Detection** check box.

Step 3 In the **Data Packet Count** field, enter the number of data packets.

Step 4 In the **Data Packet Percentage** field, enter the percentage of data packets.

Step 5 In the **Data RSSI Threshold** field, enter the actual value in dBm. Value ranges from -60 dBm to -90 dBm; the default value is –80 dBm.

Step 6 In the **Voice Packet Count** field, enter the number of voice data packets.

Step 7 In the **Voice Packet Percentage** field, enter the percentage of voice data packets.

Step 8 In the **Voice RSSI Threshold** field, enter the actual value in dBm. Value ranges from -60 dBm to -90 dBm; the default value is –80 dBm.

Step 9 In the **Minimum Failed Client per AP** field, enter the minimum number of clients on an AP with a signal-to-noise ratio (SNR) below the coverage threshold. Value ranges from 1 to 75 and the default value is 3.

Step 10 In the **Percent Coverage Exception Level per AP** field, enter the maximum desired percentage of clients on an access point's radio operating below the desired coverage threshold and click Apply. Value ranges from 0 to 100% and the default value is 25%.

Step 11 Click **Apply**.

Configuring CHD for RF Profile (CLI)

Follow the procedure given below to configure Coverage Hole Detection (CHD) for RF profile.

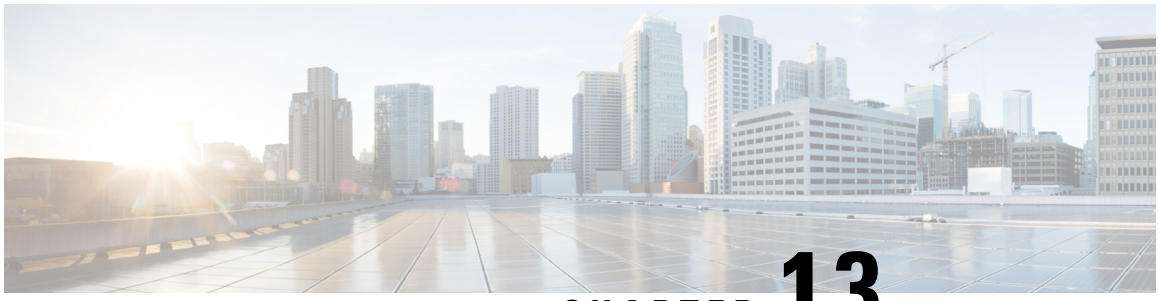
Before you begin

Ensure that the RF profile is already created.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11** {**24ghz** | **5ghz** } **rf-profile** *rf-profile-tag* **Example:** `Device(config)# ap dot11 24ghz rf-profile alpha-rfprofile-24ghz`	Configures the 802.11 coverage hole detection for data packets.
Step 3	**coverage data rssi threshold** *threshold-value* **Example:** `Device(config-rf-profile)# coverage data rssi threshold -80`	Configures the minimum RSSI value for data packets received by the access point. Valid values range from -90 to -60 in dBm.
Step 4	**end** **Example:** `Device(config-rf-profile)# end`	Returns to privileged EXEC mode.
Step 5	**show ap dot11 24ghz rf-profile summary** **Example:**	Displays summary of the available RF profiles.

	Command or Action	Purpose
	`Device# show ap dot11 24ghz rf-profile summary`	



CHAPTER 13

Cisco Flexible Radio Assignment

Information About Flexible Radio Assignment

Flexible Radio Assignment (FRA) takes advantage of the dual-band radios included in APs. The FRA is a new feature added to the RRM to analyze the Neighbor Discovery Protocol (NDP) measurements, which manages the hardware used to determine the role of the new flexible radio (2.4 GHz, 5 GHz, or monitor) in your network.

Traditional legacy dual–band APs always had 2 radio slots, (1 slot per band) and were organized by the band they were serving, that is slot 0= 802.11b,g,n and slot 1=802.11a,n,ac.

XOR Support in 2.4-GHz or 5-GHz Bands

The flexible radio (XOR) offers the ability to serve the 2.4-GHz or the 5-GHz bands, or passively monitor both bands on the same AP. The AP models that are offered are designed to support dual 5-GHz band operations, with the Cisco APs *i* model supporting a dedicated Macro/Micro architecture, and the *e* and *p* models supporting Macro/Macro architecture.

When using FRA with the internal antenna (*i* series models), two 5-GHz radios can be used in a Micro/Macro cell mode. When using FRA with external antenna (*e* and *p* models) the antennas may be placed to enable the creation of two completely separate macro (wide-area cells) or two micro cells (small cells) for HDX or any combination.

FRA calculates and maintains a measurement of redundancy for 2.4-GHz radios and represents this as a new measurement metric called COF (Coverage Overlap Factor).

This feature is integrated into existing RRM and runs in mixed environments with legacy APs. The **AP MODE** selection sets the entire AP (slot 0 and slot1) into one of several operating modes, including:

- Local Mode
- Monitor Mode
- FlexConnect Mode
- Sniffer Mode
- Spectrum Connect Mode

Before XOR was introduced, changing the mode of an AP propagated the change to the entire AP, that is both radio slot 0 and slot 1. The addition of the XOR radio in the slot 0 position provides the ability to operate a single radio interface in many of the previous modes, eliminating the need to place the whole AP into a mode. When this concept is applied to a single radio level, its is called *role*. Three such roles can be assigned now:

- Client Serving
- Either 2.4 GHz(1) or 5 GHz(2)
- Monitor-Monitor mode (3)

Note

- MODE: Assigned to a whole AP (slot 0 and slot 1)
- ROLE: Assigned to a single radio interface (slot 0)

Benefits of the FRA

- Solves the problem of 2.4–GHz over coverage.
- Creating two diverse 5–GHz cells doubles the airtime that is available.
- Permits one AP with one Ethernet drop to function like two 5–GHz APs.
- Introduces the concept of Macro/Micro cells for airtime efficiency.
- Allows more bandwidth to be applied to an area within a larger coverage cell.
- Can be used to address nonlinear traffic.
- Enhances the High-Density Experience (HDX) with one AP.
- XOR radio can be selected by the corresponding user in either band–servicing client mode or monitor mode.

Configuring an FRA Radio (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**[no] ap fra** **Example:** `Device(config)# [no] ap fra`	Enables or disables FRA on the AP.
Step 4	**ap fra interval** **Example:** `Device(config)# ap fra interval 3`	Configures the FRA interval in hours. The range is 1 to 24 hours. **Note** The FRA interval has to be more than the configured RRM interval.
Step 5	**ap fra sensitivity** {**high** | **medium** | **low**} **Example:** `Device(config)# ap fra sensitivity high`	Configures the FRA sensitivity. • **high**: Sets the FRA Coverage Overlap Sensitivity to **high**. • **medium**: Sets the FRA Coverage Overlap Sensitivity to **medium**. • **low**: Sets the FRA Coverage Overlap Sensitivity to **low**.
Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 7	**ap fra revert** {**all** | **auto-only**}{**auto** | **static**} **Example:** `Device# ap fra revert all auto`	Rolls back the XOR Radio state. • **all**: Reverts all XOR Radios • **auto-only**: Revert only XOR radios currently in automatic band selection. • **auto**: Sets the XOR radios in automatic band selection. • **static**: Sets the XOR radio in static 2.4-GHz band.
Step 8	**show ap dot11** {**24ghz** | **5ghz**} **summary** **Example:** `Device# show ap dot11 5ghz summary`	Shows the configuration and statistics of 802.11 Cisco APs

Step 9	Device# **show ap fra**
	Example:
	`Device# show ap fra`
	`FRA State : Disabled`
	`FRA Sensitivity : medium (95%)`
		Shows the current FRA configuration.

	Command or Action	Purpose
	FRA Interval : 1 Hour(s) AP Name MAC Address Slot ID Current-Band COF % Suggested Mode --- AP00A6.CA36.295A 006b.f09c.8290 0 2.4GHz None 2.4GHz COF : Coverage Overlap Factor test_machine#	
Step 10	**show ap name** *ap-name* **config dot11 dual-band** **Example:** Device# **show ap name config dot11 dual-band**	Shows the current 802.11 dual-band parameters in a given AP.

Configuring an FRA Radio (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **RRM** > **FRA**.

Step 2 In the **Flexible Radio Assignment** window, enable FRA status and determine the overlapping 2.4 GHz or 5 GHz coverage for each AP, choose **Enabled** in the **FRA Status** field. By default, the FRA status is disabled.

Step 3 Under the From the **FRA Interval** drop-down list, choose the FRA run interval. The interval values range from 1 hour to 24 hours. You can choose the FRA run interval value only after you enable the FRA status.

Step 4 From the **FRA Sensitivity** drop-down list, choose the percentage of Coverage Overlap Factor (COF) required to consider a radio as redundant. You can select the supported value only after you enable the FRA status.

The supported values are as follows:

- Low: 100 percent
- Medium (default): 95 percent
- High: 90 percent

The **Last Run** and **Last Run Time** fields will show the time FRA was run last and the time it was run.

Step 5 Check the **Client Aware** check box to take decisions on redundancy.

When enabled, the **Client Aware** feature monitors the dedicated 5-GHz radio and when the client load passes a pre-set threshold, automatically changes the Flexible Radio assignment from a monitor role into a 5-GHz role, effectively doubling the capacity of the cell on demand. Once the capacity crisis is over and Wi-Fi load returns to normal, the radios resume their previous roles.

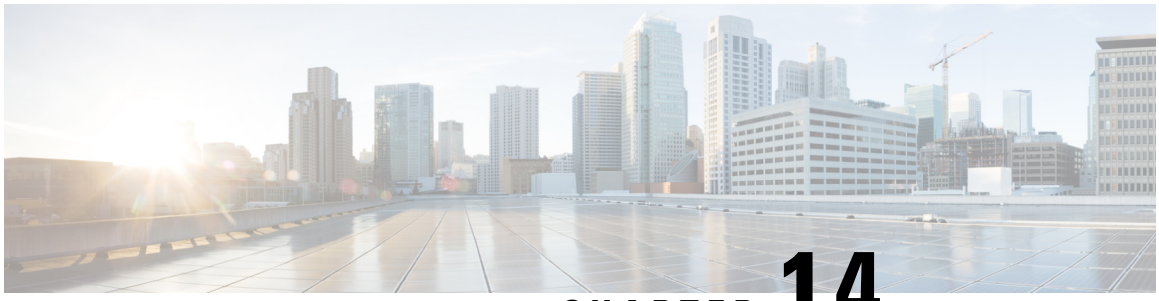
Step 6 In the **Client Select** field, enter a value for client selection. The valid values range between 0 and 100 percent. The default value is 50 percent.

This means that if the dedicated 5-GHz interface reaches 50% channel utilization, this will trigger the monitor role dual-band interface to transition to a 5-GHz client-serving role.

Step 7 In the **Client Reset** field, enter a reset value for the client. The valid values range between 0 and 100 percent. The default value is 5 percent.

Once the AP is operating as a dual 5-GHz AP, this setting indicates the reduction in the combined radios' overall channel utilization required to reset the dual-band radio to monitor role.

Step 8 Click **Apply** to save the configuration.



CHAPTER 14

XOR Radio Support

Information About Dual-Band Radio Support

The Dual-Band (XOR) radio in Cisco 2800, 3800, 4800, and the 9120 series AP models offer the ability to serve 2.4–GHz or 5–GHz bands or passively monitor both the bands on the same AP. These APs can be configured to serve clients in 2.4–GHz and 5–GHz bands, or serially scan both 2.4–GHz and 5–GHz bands on the flexible radio while the main 5–GHz radio serves clients.

Cisco APs models up and through the Cisco 9120 APs are designed to support dual 5–GHz band operations with the *i* model supporting a dedicated Macro/Micro architecture and the *e* and *p* models supporting Macro/Macro. The Cisco 9130AXI APs and the Cisco 9136 APs support dual 5-GHz operations as Micro/Messo cell.

When a radio moves between bands (from 2.4-GHz to 5-GHz and vice versa), clients need to be steered to get an optimal distribution across radios. When an AP has two radios in the 5–GHz band, client steering algorithms contained in the Flexible Radio Assignment (FRA) algorithm are used to steer a client between the same band co-resident radios.

The XOR radio support can be steered manually or automatically:

- Manual steering of a band on a radio—The band on the XOR radio can only be changed manually.
- Automatic client and band steering on the radios is managed by the FRA feature that monitors and changes the band configurations as per site requirements.

Note RF measurement will not run when a static channel is configured on slot 1. Due to this, the dual band radio slot 0 will move only with 5–GHz radio and not to the monitor mode.

When slot 1 radio is disabled, RF measurement will not run, and the dual band radio slot 0 will be only on 2.4–GHz radio.

Configuring Default XOR Radio Support

Before you begin

Note The default radio points to the XOR radio hosted on slot 0.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap name** *ap-name* **dot11 dual-band antenna ext-ant-gain** *antenna_gain_value* **Example:** `Device# ap name ap-name dot11 dual-band antenna ext-ant-gain 2`	Configures the 802.11 dual-band antenna on a specific Cisco access point. *antenna_gain_value*: The valid range is from 0 to 40.
Step 3	**ap name** *ap-name* **[no] dot11 dual-band shutdown** **Example:** `Device# ap name ap-name dot11 dual-band shutdown`	Shuts down the default dual-band radio on a specific Cisco access point. Use the **no** form of the command to enable the radio.
Step 4	**ap name** *ap-name* **dot11 dual-band role manual client-serving** **Example:** `Device# ap name ap-name dot11 dual-band role manual client-serving`	Switchs to client–serving mode on the Cisco access point.
Step 5	**ap name** *ap-name* **dot11 dual-band band 24ghz** **Example:** `Device# ap name ap-name dot11 dual-band band 24ghz`	Switchs to 2.4-GHz radio band.
Step 6	**ap name** *ap-name* **dot11 dual-band txpower** {*transmit_power_level* | **auto**} **Example:**	Configures the transmit power for the radio on a specific Cisco access point.

	Command or Action	Purpose
	`Device# ap name ap-name dot11 dual-band txpower 2`	**Note** When an FRA-capable radio (slot 0 on 9120 AP[for instance]) is set to Auto, you cannot configure static channel and Txpower on this radio. If you want to configure static channel and Txpower on this radio, you will need to change the radio role to Manual Client-Serving mode.
Step 7	**ap name** *ap-name* **dot11 dual-band channel** *channel-number* **Example:** `Device# ap name ap-name dot11 dual-band channel 2`	Enters the channel for the dual band. *channel-number*—The valid range is from 1 to 173.
Step 8	**ap name** *ap-name* **dot11 dual-band channel auto** **Example:** `Device# ap name ap-name dot11 dual-band channel auto`	Enables the auto channel assignment for the dual-band.
Step 9	**ap name** *ap-name* **dot11 dual-band channel width**{**20 MHz** | **40 MHz** | **80 MHz** | **160 MHz**} **Example:** `Device# ap name ap-name dot11 dual-band channel width 20 MHz`	Chooses the channel width for the dual band.
Step 10	**ap name** *ap-name* **dot11 dual-band cleanair** **Example:** `Device# ap name ap-name dot11 dual-band cleanair`	Enables the Cisco CleanAir feature on the dual-band radio.

Step 11	**ap name** *ap-name* **dot11 dual-band cleanair band**{**24 GHz | 5 GMHz**}
	Example:
	`Device# ap name ap-name dot11 dual-band cleanair band 5 GHz`
	`Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz`
		Selects a band for the Cisco CleanAir feature.
		Use the **no** form of this command to disable the Cisco CleanAir feature.

Step 12	**ap name** *ap-name* **dot11 dual-band dot11n antenna** {**A | B | C | D**} **Example:**	Configures the 802.11n dual-band parameters for a specific access point.

	Command or Action	Purpose
	`Device# ap name ap-name dot11 dual-band dot11n antenna A`	
Step 13	**show ap name** *ap-name* **auto-rf dot11 dual-band** **Example:** `Device# show ap name ap-name auto-rf dot11 dual-band`	Displays the auto-RF information for the Cisco access point.
Step 14	**show ap name** *ap-name* **wlan dot11 dual-band** **Example:** `Device# show ap name ap-name wlan dot11 dual-band`	Displays the list of BSSIDs for the Cisco access point.

Configuring XOR Radio Support for the Specified Slot Number (GUI)

Procedure

Step 1 Click **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Dual-Band Radios** section, select the AP for which you want to configure dual-band radios.

The AP name, MAC address, CleanAir capability and slot information for the AP are displayed. If the Hyperlocation method is HALO, the antenna PID and antenna design information are also displayed.

Step 3 Click **Configure**.

Step 4 In the **General** tab, set the **Admin Status** as required.

Step 5 Set the **CleanAir Admin Status** field to Enable or Disable.

Step 6 Click **Update & Apply to Device**.

Configuring XOR Radio Support for the Specified Slot Number

Procedure

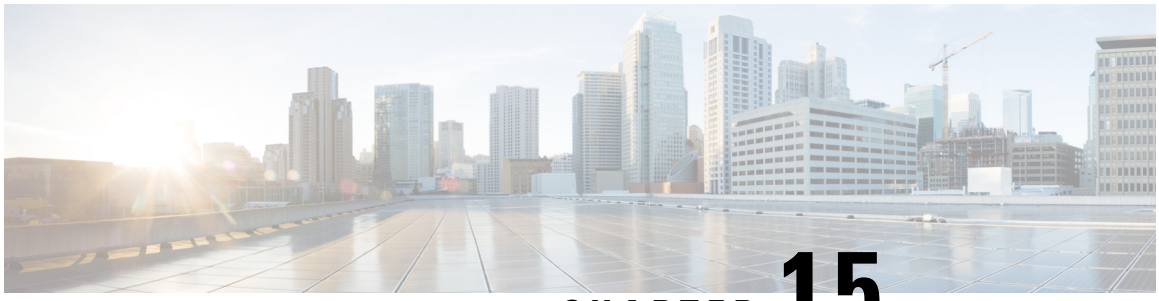
	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.

	Command or Action	Purpose
Step 2	**ap name** *ap-name* **dot11 dual-band slot 0 antenna ext-ant-gain** *external_antenna_gain_value* **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 antenna ext-ant-gain 2`	Configures dual-band antenna for the XOR radio hosted on slot 0 for a specific access point. *external_antenna_gain_value* - Is the external antenna gain value in multiples of .5 dBi unit. The valid range is from 0 to 40. **Note** • For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration. • For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model.
Step 3	**ap name** *ap-name* **dot11 dual-band slot 0 band** {**24ghz** | **5ghz**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 band 24ghz`	Configures current band for the XOR radio hosted on slot 0 for a specific access point.
Step 4	**ap name** *ap-name* **dot11 dual-band slot 0 channel** {*channel_number* | **auto** | **width** [**160** | **20** | **40** | **80**]} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 channel 3`	Configures dual-band channel for the XOR radio hosted on slot 0 for a specific access point. *channel_number*- The valid range is from 1 to 165.
Step 5	**ap name** *ap-name* **dot11 dual-band slot 0 cleanair band** {**24Ghz** | **5Ghz**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 cleanair band 24Ghz`	Enables CleanAir features for dual-band radios hosted on slot 0 for a specific access point.
Step 6	**ap name** *ap-name* **dot11 dual-band slot 0 dot11n antenna** {**A** | **B** | **C** | **D**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 dot11n antenna A`	Configures 802.11n dual-band parameters hosted on slot 0 for a specific access point. Here, **A**- Enables antenna port A. **B**- Enables antenna port B.

	Command or Action	Purpose
		C- Enables antenna port C. **D**- Enables antenna port D.
Step 7	**ap name** *ap-name* **dot11 dual-band slot 0 role** {**auto** | **manual** [**client-serving** | **monitor**]} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 role auto`	Configures dual-band role for the XOR radio hosted on slot 0 for a specific access point. The following are the dual-band roles: • **auto**- Refers to the automatic radio role selection. • **manual**- Refers to the manual radio role selection.

Step 8	**ap name** *ap-name* **dot11 dual-band slot 0 shutdown**
	Example:
	`Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 shutdown`
	`Device# ap name AP-SIDD-A06 [no] dot11 dual-band slot 0 shutdown`
		Disables dual-band radio hosted on slot 0 for a specific access point.
		Use the **no** form of this command to enable the dual-band radio.

Step 9	**ap name** *ap-name* **dot11 dual-band slot 0 txpower** {*tx_power_level* | **auto**} **Example:** `Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 txpower 2`	Configures dual-band transmit power for XOR radio hosted on slot 0 for a specific access point. • *tx_power_level*- Is the transmit power level in dBm. The valid range is from 1 to 8. • **auto**- Enables auto-RF.



CHAPTER 15

Cisco Receiver Start of Packet

Information About Receiver Start of Packet Detection Threshold

The Receiver Start of Packet (Rx SOP) Detection Threshold feature determines the Wi-Fi signal level in dBm at which an access point's radio demodulates and decodes a packet. As the Wi-Fi level increases, the radio sensitivity decreases and the receiver cell size becomes smaller. Reduction of the cell size affects the distribution of clients in the network.

Rx SOP is used to address clients with weak RF links, sticky clients, and client load balancing across access points. Rx SOP helps to optimize the network performance in high-density deployments, such as stadiums and auditoriums where access points need to optimize the nearest and strongest clients.

Restrictions for Rx SOP

- Rx SOP configuration is not applicable to the third radio module pluggable on Cisco Aironet Series APs.
- Rx SOP configurations are supported only in Local, FlexConnect, Bridge, and Flex+Bridge modes.
- Rx SOP configurations are not supported in the FlexConnect+PPPoE, FlexConnect+PPPoE-wIPS, and FlexConnect+OEAP submodes.

The following table shows the permitted range for the Rx SOP threshold.

Table 13: Rx SOP Threshold

Radio Band	Threshold High	Threshold Medium	Threshold Low
2.4 GHz	-79 dBm	-82 dBm	-85 dBm
5 GHz	-76 dBm	-78 dBm	-80 dBm

Configuring Rx SOP (CLI)

Procedure

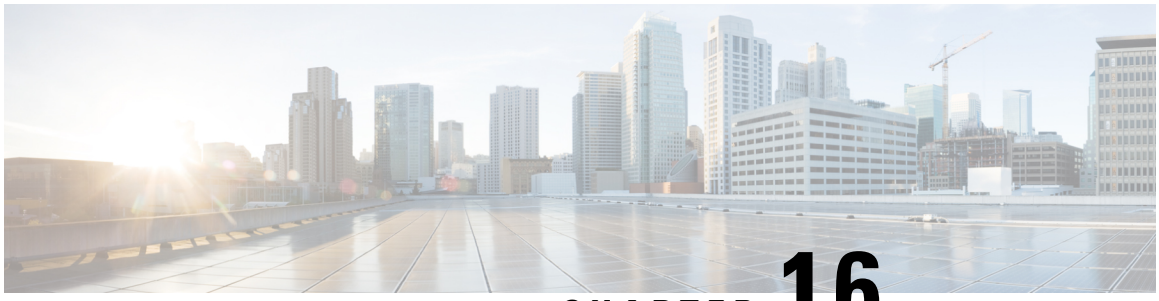
	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {24ghz | 5ghz} rx-sop threshold {auto | custom | high | low | medium}** **Example:** `Device(config)# ap dot11 5ghz rx-sop threshold high`	Configures the 802.11bg/802.11a radio Rx SOP threshold.
Step 3	end	Returns to privileged EXEC mode.
Step 4	**show ap dot11 {24ghz | 5ghz} high-density** **Example:** `Device# show ap dot11 5ghz high-density`	Displays the 802.11bg/802.11a high-density parameters.
Step 5	**show ap summary** **Example:** `Device# show ap summary`	Displays a summary of all the connected Cisco APs.

Customizing RF Profile (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11** {**24ghz | 5ghz** } **rf-profile** *profile-name* **Example:** `Device(config)# ap dot11 24ghz rf-profile AHS_2.4ghz`	Configures the 802.11a and 11b parameters.

	Command or Action	Purpose
Step 3	**high-density rx-sop threshold {auto | custom | high | low | medium}** **Example:** `Device(config-rf-profile)# high-density rx-sop threshold high`	Configures the 802.11bg, 802.11a high-density parameters.
Step 4	**show ap summary** **Example:** `Device# show ap summary`	Displays a summary of all the connected Cisco APs.
Step 5	end	Returns to privileged EXEC mode. **Note** • Irrespective of radio mode, the controller configures the radio with configured RX-SOP value. The AP determines whether to use the configured RX-SOP value. • For the XOR radio (Slot 0), when the AP is in monitor mode the RX-SOP value that gets pushed to AP depends on the band it was operating before moving to monitor mode (basically if radio operating band is 24g then RX-SOP params picked from 24GHz RF profile (or default rf-profile). If it was in 5g then RX-SOP params picked from 5GHz RF profile (or default rf-profile) configured for the AP).



CHAPTER 16

Client Limit

Information About Client Limit

This feature enforces a limit on the number of clients that can to be associated with an AP. Further, you can configure the number of clients that can be associated with each AP radio.

Configuring Client Limit Per WLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click a WLAN from the list of WLANs.

Step 3 Click the **Advanced** tab.

Step 4 Under the **Max Client Connections** settings, enter the client limit for **Per WLAN**, **Per AP Per WLAN**, and **Per AP Radio Per WLAN**.

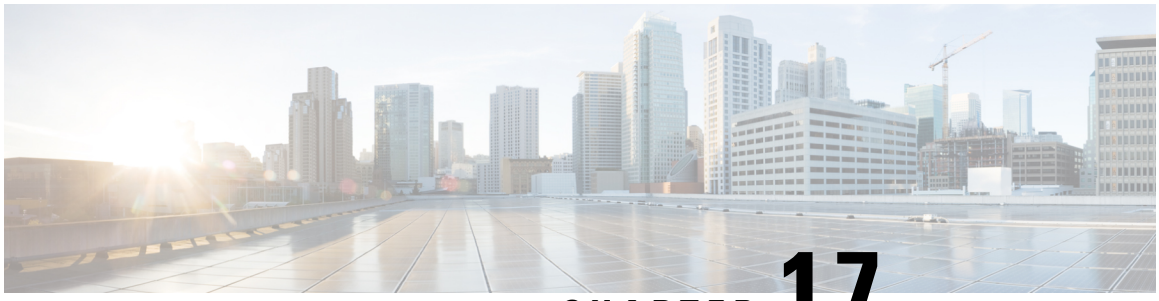
Step 5 Click **Update & Apply to Device**.

Configuring Client Limit Per WLAN (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enters privileged EXEC mode.

	Command or Action	Purpose
	`Device# enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wlan** *wlan-name* **Example:** `Device(config)# wlan ramban`	Specifies the WLAN name.
Step 4	**client association limit** *maximum-clients-per-WLAN* **Example:** `Device(config-wlan)# client association limit 110`	Configures the maximum number of clients that can be associated to the given WLAN. **Note** Depending on the primary AP in the Cisco Embedded Wireless Controller network, the maximum number of clients supported varies. For more information about the client count limit per WLAN in a Cisco Embedded Wireless Controller network, see Table 14: Scale Supported in a Cisco Embedded Wireless Controller Network, on page 236 ***Table 14: Scale Supported in a Cisco Embedded Wireless Controller Network***
Step 5	**client association limit ap** *max-clients-per-AP-per-WLAN* **Example:** `Device(config-wlan)# client association limit ap 120`	Configures the maximum number of clients that can be associated to an AP in the WLAN.
Step 6	**client association limit radio** *max-clients-per-AP-radio-per-WLAN* **Example:** `Device(config-wlan)# client association limit radio 100`	Configures the maximum number of clients that can be associated to an AP radio in the WLAN.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 8	**show wlan id** *wlan-id* **Example:** `Device#  show wlan id 2`	Displays the current configuration of the WLAN and the corresponding client association limits.



CHAPTER 17

IP Theft

Introduction to IP Theft

The IP Theft feature prevents the usage of an IP address that is already assigned to another device. If the controller finds that two wireless clients are using the same IP address, it declares the client with lesser precedence binding as the IP thief and allows the other client to continue. If blocked list is enabled, the client is put on the exclusion list and thrown out.

The IP Theft feature is enabled by default on the controller. The preference level of the clients (new and existing clients in the database) are also used to report IP theft. The preference level is a learning type or source of learning, such as Dynamic Host Configuration Protocol (DHCP), Address Resolution Protocol (ARP), data glean (looking at the IP data packet that shows what IP address the client is using), and so on. The wired clients always get a higher preference level. If a wireless client tries to steal the wired IP, that client is declared as a thief.

The order of preference for IPv4 clients are:

1. DHCPv4
2. ARP
3. Data packets

The order of preference for IPv6 clients are:

1. DHCPv6
2. NDP
3. Data packets

Note The static wired clients have a higher preference over DHCP.

Configuring IP Theft (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies** > **Client Exclusion Policies**.

Step 2 Check the **IP Theft or IP Reuse** check box.

Step 3 Click **Apply**.

Configuring IP Theft

Follow the procedure given below to configure the IP Theft feature:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps client-exclusion ip-theft** **Example:** `Device(config)# wireless wps client-exclusion ip-theft`	Configures the client exclusion policy.

Configuring the IP Theft Exclusion Timer

Follow the procedure given below to configure the IP theft exclusion timer:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-policy* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Configures a WLAN policy profile and enters wireless policy configuration mode.

	Command or Action	Purpose
Step 3	**exclusionlist timeout** *time-in-seconds* **Example:** `Device(config-wireless-policy)# exclusionlist timeout 5`	Specifies the timeout, in seconds. The valid range is from 0-2147483647. Enter zero (0) for no timeout.

Verifying IP Theft Configuration

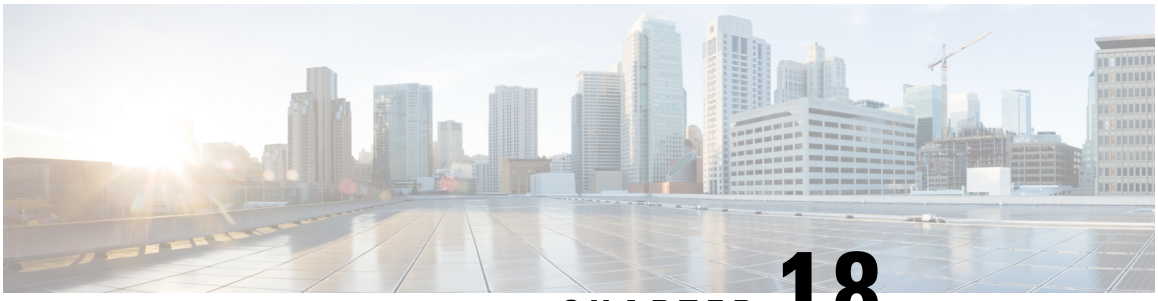
Use the following command to check if the IP Theft feature is enabled or not:

```
Device# show wireless wps summary

Client Exclusion Policy
  Excessive 802.11-association failures   : Enabled
  Excessive 802.11-authentication failures: Enabled
  Excessive 802.1x-authentication         : Enabled
  IP-theft                                : Enabled
  Excessive Web authentication failure    : Enabled
  Cids Shun failure                       : Enabled
  Misconfiguration failure                : Enabled
  Failed Qos Policy                       : Enabled
  Failed Epm                              : Enabled
```

Use the following commands to view additional details about the IP Theft feature:

```
Device# show wireless client summary

Number of Local Clients: 1

MAC Address    AP Name                WLAN State              Protocol Method     Role
-------------------------------------------------------------------------------------
000b.bbb1.0001 SimAP-1                2    Run                11a      None       Local


Number of Excluded Clients: 1

MAC Address    AP Name                WLAN State              Protocol Method
-------------------------------------------------------------------------------------
10da.4320.cce9 charlie2               2    Excluded           11ac     None


Device# show wireless device-tracking database ip

IP                              VLAN  STATE       DISCOVERY   MAC
  ------------------------------------------------------------------
  20.20.20.2                     20    Reachable   Local       001e.14cc.cbff
  20.20.20.6                     20    Reachable   IPv4 DHCP   000b.bbb1.0001


Device# show wireless exclusionlist

Excluded Clients

MAC Address       Description          Exclusion Reason               Time Remaining
-------------------------------------------------------------------------------------
10da.4320.cce9                         IP address theft                   59
```

```
Device# show wireless exclusionlist client mac 12da.4820.cce9 detail

Client State : Excluded
Client MAC Address : 12da.4820.cce9
Client IPv4 Address: 20.20.20.6
Client IPv6 Address: N/A
Client Username: N/A
Exclusion Reason : IP address theft
Authentication Method : None
Protocol: 802.11ac
AP MAC Address : 58ac.780e.08f0
AP Name: charlie2
AP slot : 1
Wireless LAN Id : 2
Wireless LAN Name: mhe-ewlc
VLAN Id : 20
```



CHAPTER 18

Unscheduled Automatic Power Save Delivery

Information About Unscheduled Automatic Power Save Delivery

Unscheduled automatic power save delivery (U-APSD) is a QoS facility that is defined in IEEE 802.11e that extends the battery life of mobile clients. In addition to extending the battery life, this feature reduces the latency of traffic flow that is delivered over the wireless media. Because U-APSD does not require the client to poll each individual packet that is 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.

Viewing Unscheduled Automatic Power Save Delivery (CLI)

Procedure

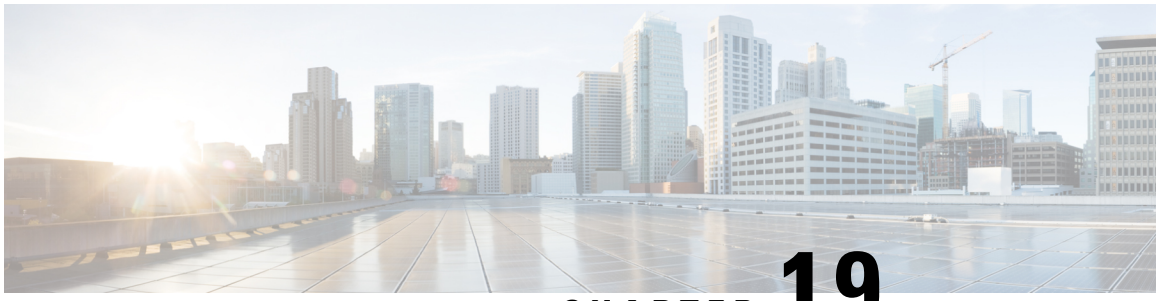
show wireless client mac-address *client_mac* **detail**

Example:

```
Device# show wireless client mac-address 2B:5B:B3:18:56:E9 detail
Output Policy State : Unknown
Output Policy Source : Unknown
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 15
  APSD ACs    : BK(T/D), BE, VI(T/D), VO(T/D)
Power Save : OFF
Current Rate :

--------------------------
BK : Background
BE : Best Effort
VI : Video
VO : Voice.

T: UAPSD Trigger Enabled
```

```
D: UAPSD Delivery Enabled
T/D : UAPSD Trigger and Delivery Enabled
```

Show detailed information of a client by MAC address.



CHAPTER 19

Enabling USB Port on Access Points

USB Port as Power Source for Access Points

Some Cisco APs have a USB port that can act as a source of power for some USB devices. The power can be up to 2.5W; if a USB device draws more than 2.5W of power, the USB port shuts down automatically. The port is enabled when the power draw is 2.5W and lower. Refer to the datasheet of your AP to check if the AP has a USB port that can act as a source of power.

Note The controller records the last five power-overdrawn incidents in its logs.

Caution When unsupported USB device is connected to the Cisco AP, the following message is displayed:

```
The inserted USB module is not a supported device. The behavior of this
USB device and the impact to the Access Point is not guaranteed. If Cisco
determines that a fault or defect can be isolated due to the use of
third-party USB modules installed by a customer or reseller, Cisco may
withhold support under warranty or support program under contract. In the
course of providing support for Cisco networking products, the end user
may be required to install Cisco-supported USB modules in the event Cisco
determines that removing third-party parts will assist Cisco in diagnosing
root cause for troubleshooting purposes. Cisco also reserves the right
to charge the customer per then-current time and material rates for
services provided to the customer when Cisco determines, after having
provided such services, that an unsupported device caused the root cause
of the defective product
```

Configuring an AP Profile (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *ap-profile* **Example:** `Device(config)# ap profile xyz-ap-profile`	Configures an AP profile and enters the AP profile configuration mode. **Note** When you delete a named profile, the APs associated with that profile will not revert to the default profile.
Step 3	**usb-enable** **Example:** `Device(config-ap-profile)# usb-enable`	Enables USB for each AP profile. **Note** By default, the USB for each AP profile is enabled. Use the **no usb-enable** command to disable USB for each AP profile.
Step 4	**end** **Example:** `Device(config-ap-profile)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring USB Settings for an Access Point (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**ap name** *ap-name* **usb-module** **Example:** `Device# ap name AP44d3.xy45.69a1 usb-module`	Enables the USB port on the AP. Use the **ap name** *ap-name* **no usb-module** command to disable the USB port on the AP.

	Command or Action	Purpose
Step 3	**ap name** *ap-name* **usb-module override** **Example:** `Device# ap name AP44d3.xy45.69a1 usb-module override`	Overrides USB status of the AP profile and considers the local AP configuration. Use the **ap name** *ap-name* **no usb-module override** command to override USB status of the AP and consider the AP profile configuration. **Note** You can configure the USB status for an AP only if you enable USB override for it.

Monitoring USB Configurations for Access Points (CLI)

- To view the inventory details of APs, use the following command:

  show ap name *ap-name* **inventory**

  The following is a sample output:

```
Device# show ap name AP500F.8059.1620  inventory
NAME: AP2800    , DESCR: Cisco Aironet 2800 Series (IEEE 802.11ac) Access Point
PID: AIR-AP2802I-D-K9  ,  VID: 01,  SN: XXX1111Y2ZZZZ2800
NAME: SanDisk    , DESCR: Cruzer Blade
PID: SanDisk  ,  SN: XXXX1110010, MaxPower: 224
```

- To view the summary of an AP module, use the following command:

  show ap module summary

  The following is a sample output:

```
Device# show ap module summary
AP Name               External Module       External Module PID   External Module
Description
-------------------------------------------------------------------------------------------
AP500F.1111.2222      Enable                SanDisk               Cruzer Blade
```

- To view the USB configuration details for each AP, use the following command:

  show ap name *ap-name* **config general**

  The following is a sample output:

```
Device# show ap name AP500F.111.2222 config general

.
.
.
USB Module Type.................................. USB Module
USB Module Status................................ Disabled
USB Module Operational State..................... Enabled
USB Override ……..... Enabled
```

- To view status of the USB module, use the following command:

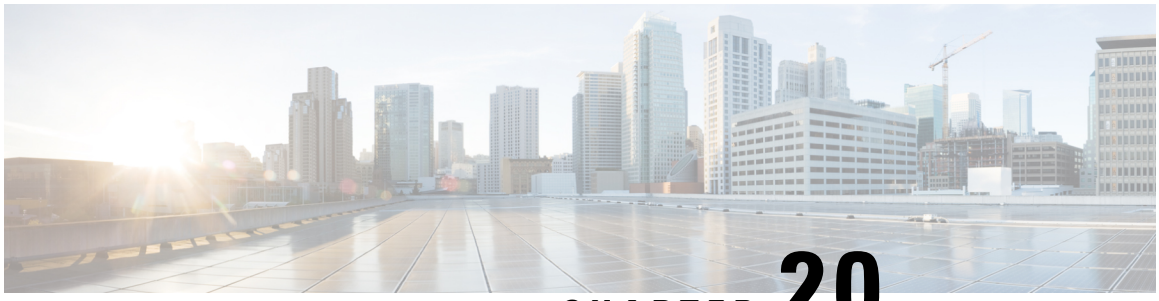
  show ap profile name *xyz* **detailed**

The following is a sample output:

```
Device# show ap profile name xyz detailed
USB Module                : ENABLED
```

PART IV

Network Management



CHAPTER 20

DHCP Option82

Information About DHCP Option 82

The embedded wireless controller can be configured to add Option 82 information to DHCP requests from clients before forwarding the requests to a DHCP server. The DHCP server can then be configured to allocate IP addresses to the wireless client based on the information present in DHCP Option 82.

DHCP provides a framework for passing configuration information to hosts on a TCP/IP network. Configuration parameters and other control information are carried in tagged data items that are stored in the Options field of the DHCP message. The data items themselves are also called options. Option 82 contains information known by the relay agent.

The Relay Agent Information option is organized as a single DHCP option that contains one or more suboptions that convey information known by the relay agent. Option 82 was designed to allow a DHCP Relay Agent to insert circuit−specific information into a request that is being forwarded to a DHCP server. This option works by setting two suboptions:

- Circuit ID
- Remote ID

The Circuit ID suboption includes information that is specific to the circuit the request came in on. This suboption is an identifier that is specific to the relay agent. Thus, the circuit that is described will vary depending on the relay agent.

The Remote ID suboption includes information on the remote host–end of the circuit. This suboption usually contains information that identifies the relay agent. In a wireless network, this would likely be a unique identifier of the wireless access point.

You can configure the following DHCP Option 82 options in a embedded wireless controller:

- DHCP Enable
- DHCP Opt82 Enable
- DHCP Opt82 Ascii

- DHCP Opt82 RID
- DHCP Opt Format
- DHCP AP MAC
- DHCP SSID
- DHCP AP ETH MAC
- DHCP AP NAME
- DHCP Site Tag
- DHCP AP Location
- DHCP VLAN ID

Note The controller includes the SSID in ASCII and the VLAN-ID in hexadecimal format within the remote-ID sub-option of option 82 in the outgoing DHCP packets to the server for the following configurations:

```
ipv4 dhcp opt82 format ssid
ipv4 dhcp opt82 format vlan-id
```

However, if *ipv4 dhcp opt82 ascii* configuration is also present, the controller adds VLAN-ID and SSID in ASCII format.

For Cisco Catalyst 9800 Series Configuration Best Practices, see the following link: https://www.cisco.com/c/en/us/products/collateral/wireless/catalyst-9800-series-wireless-controllers/guide-c07-743627.html

Configuring DHCP Option 82 Global Interface

Configuring DHCP Option 82 Globally Through Server Override (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ip dhcp-relay information option server-override** **Example:** `Device(config)# ip dhcp-relay information option server-override`	Inserts global server override and link selection suboptions.

Configuring DHCP Option 82 Globally Through Different SVIs (GUI)

Procedure

Step 1 Choose **Configuration > VLAN**.

Step 2 Choose a VLAN from the drop-down list.

The **Edit SVI** window appears.

Step 3 Click the **Advanced** tab.

Step 4 Choose an option from the **IPv4 Inbound ACL** drop-down list.

Step 5 Choose an option from the **IPv4 Outbound ACL** drop-down list.

Step 6 Choose an option from the **IPv6 Inbound ACL** drop-down list.

Step 7 Choose an option from the **IPv6 Outbound ACL** drop-down list.

Step 8 Enter an IP address in the **IPv4 Helper Address** field.

Step 9 Set the status to **Enabled** if you want to enable the **Relay Information Option** setting.

Step 10 Enter the **Subscriber ID**.

Step 11 Set the status to **Enabled** if you want to enable the **Server ID Override** setting.

Step 12 Set the status to **Enabled** if you want to enable the **Option Insert** setting.

Step 13 Choose an option from the **Source-Interface Vlan** drop-down list.

Step 14 Click **Update & Apply to Device**.

Configuring DHCP Option 82 Globally Through Different SVIs (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ip dhcp-relay source-interface vlan** *vlan-id* **Example:** `Device(config)# ip dhcp-relay source-interface vlan 74`	Sets global source interface for relayed messages.

Configuring DHCP Option 82 Format

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-name* **Example:** `Device(config)# wireless profile policy pp3`	Enables configuration for the specified profile policy.
Step 3	**shutdown** **Example:** `Device(config-wireless-policy)# shutdown`	Shuts down the profile policy.
Step 4	**vlan** *vlan-name* **Example:** `Device(config-wireless-policy)# vlan 72`	Assigns the profile policy to a VLAN.
Step 5	**session-timeout** *value-btwn-20-86400* **Example:** `Device(config-wireless-policy)# session-timeout 300`	(Optional) Sets the session timeout value in seconds. The range is between 20-86400.
Step 6	**idle-timeout** *value-btwn-15-100000* **Example:** `Device(config-wireless-policy)# idle-timeout 15`	(Optional) Sets the idle timeout value in seconds. The range is between 15-100000.
Step 7	**central switching** **Example:** `Device(config-wireless-policy)# central switching`	Enables central switching.
Step 8	**ipv4 dhcp opt82** **Example:** `Device(config-wireless-policy)# ipv4 dhcp opt82`	Enables DHCP Option 82 for the wireless clients.
Step 9	**ipv4 dhcp opt82 ascii** **Example:**	(Optional) Enables ASCII on the DHCP Option 82 feature.

	Command or Action	Purpose
	`Device(config-wireless-policy)# ipv4 dhcp opt82 ascii`	
Step 10	**ipv4 dhcp opt82 rid** **Example:** `Device(config-wireless-policy)# ipv4 dhcp opt82 rid`	(Optional) Supports the addition of Cisco 2 byte Remote ID (RID) for the DHCP Option 82 feature.
Step 11	**ipv4 dhcp opt82 format** {**ap_ethmac**|**ap_location**|**apmac**|**apname**|**policy_tag**|**ssid**|**vlan_id**} **Example:** `Device(config-wireless-policy)# ipv4 dhcp opt82 format apmac`	Enables DHCP Option 82 on the corresponding AP. For information on the various options available with the command, see Cisco Catalyst 9800 Series Wireless Controller Command Reference.
Step 12	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Enables the profile policy.

Configuring DHCP Option82 Through a VLAN Interface

Configuring DHCP Option 82 Through Option-Insert Command (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Device(config)# interface vlan 72`	Configures a VLAN ID.
Step 3	**ip dhcp relay information option-insert** **Example:** `Device(config-if)# ip dhcp relay information option-insert`	Inserts relay information in BOOTREQUEST.
Step 4	**ip address** *ip-address* **Example:**	Configures the IP address for the interface.

	Command or Action	Purpose
	`Device(config-if)# ip address 9.3.72.38 255.255.255.0`	
Step 5	**ip helper-address** *ip-address* **Example:** `Device(config-if)# ip helper-address 9.3.72.1`	Configures the destination address for UDP broadcasts.
Step 6	**[no] mop enabled** **Example:** `Device(config-if)# no mop enabled`	Disables the MOP for an interface.
Step 7	**[no] mop sysid** **Example:** `Device(config-apgroup)# [no] mop sysid`	Disables the task of sending MOP periodic system ID messages.

Configuring DHCP Option 82 Through the server-ID-override Command (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ip dhcp compatibility suboption server-override cisco** **Example:** `Device(config)# ip dhcp compatibility suboption server-override cisco`	Configures the server-id override suboption to an RFC or Cisco specific value.
Step 3	**ip dhcp compatibility suboption link-selection cisco** **Example:** `Device(config)# ip dhcp compatibility suboption link-selection cisco`	Configures the link-selection suboption to an RFC or Cisco specific value.
Step 4	**interface vlan** *vlan-id* **Example:** `Device(config)# interface vlan 72`	Configures a VLAN ID.
Step 5	**ip dhcp relay information option server-id-override** **Example:**	Inserts the server id override and link selection suboptions.

	Command or Action	Purpose
	`Device(config-if)# ip dhcp relay information option server-id-override`	
Step 6	**ip address** *ip-address* **Example:** `Device(config-if)# ip address 9.3.72.38 255.255.255.0`	Configures the IP address for the interface.
Step 7	**ip helper-address** *ip-address* **Example:** `Device(config-if)# ip helper-address 9.3.72.1`	Configures the destination address for UDP broadcasts.
Step 8	**[no] mop enabled** **Example:** `Device(config-if)# no mop enabled`	Disables MOP for an interface.
Step 9	**[no] mop sysid** **Example:** `Device(config-if)# [no] mop sysid`	Disables the task of sending MOP periodic system ID messages.

Configuring DHCP Option 82 Through a Subscriber-ID (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Device(config)# interface vlan 72`	Configures a VLAN ID.
Step 3	**ip dhcp relay information option subscriber-id** *subscriber-id* **Example:** `Device(config-if)# ip dhcp relay information option subscriber-id test10`	Inserts the subscriber identifier suboption.
Step 4	**ip address** *ip-address* **Example:**	Configures the IP address for the interface.

	Command or Action	Purpose
	`Device(config-if)# ip address 9.3.72.38 255.255.255.0`	
Step 5	**ip helper-address** *ip-address* **Example:** `Device(config-if)# ip helper-address 9.3.72.1`	Configures the destination address for UDP broadcasts.
Step 6	**[no] mop enabled** **Example:** `Device(config-if)# no mop enabled`	Disables MOP for an interface.
Step 7	**[no] mop sysid** **Example:** `Device(config-apgroup)# [no] mop sysid`	Disables the task of sending MOP periodic system ID messages.

Configuring DHCP Option 82 Through server-ID-override and subscriber-ID Commands (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Device(config)# interface vlan 72`	Configures a VLAN ID.
Step 3	**ip dhcp relay information option server-id-override** **Example:** `Device(config-if)# ip dhcp relay information option server-id-override`	Inserts server ID override and link selection suboptions.
Step 4	**ip dhcp relay information option subscriber-id** *subscriber-id* **Example:** `Device(config-if)# ip dhcp relay information option subscriber-id test10`	Inserts the subscriber identifier suboption.

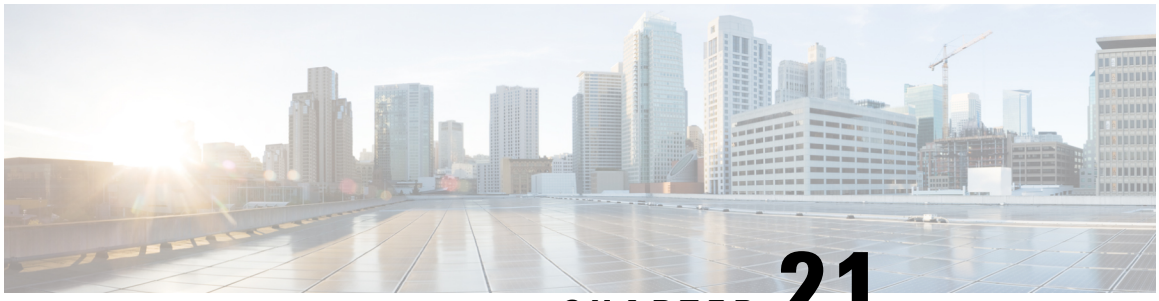
	Command or Action	Purpose
Step 5	**ip address** *ip-address* **Example:** `Device(config-if)# ip address 9.3.72.38 255.255.255.0`	Configures the IP address for the interface.
Step 6	**ip helper-address** *ip-address* **Example:** `Device(config-if)# ip helper-address 9.3.72.1`	Configures the destination address for UDP broadcasts.
Step 7	**[no] mop enabled** **Example:** `Device(config-if)# no mop enabled`	Disables the MOP for an interface.
Step 8	**[no] mop sysid** **Example:** `Device(config-apgroup)# [no] mop sysid`	Disables the task of sending MOP periodic system ID messages.

Configuring DHCP Option 82 Through Different SVIs (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface vlan** *vlan-id* **Example:** `Device(config)# interface vlan 72`	Configures a VLAN ID.
Step 3	**ip dhcp relay source-interface vlan** *vlan-id* **Example:** `Device(config-if)# ip dhcp relay source-interface vlan 74`	Configures a source interface for relayed messages on a VLAN ID.
Step 4	**ip address** *ip-address* **Example:** `Device(config-if)# ip address 9.3.72.38 255.255.255.0`	Configures the IP address for the interface.

	Command or Action	Purpose
Step 5	**ip helper-address** *ip-address* **Example:** `Device(config-if)# ip helper-address 9.3.72.1`	Configure the destination address for UDP broadcasts.
Step 6	**[no] mop enabled** **Example:** `Device(config-if)# no mop enabled`	Disables the MOP for an interface.
Step 7	**[no] mop sysid** **Example:** `Device(config-apgroup)# [no] mop sysid`	Disables the task of sending MOP periodic system ID messages.



CHAPTER 21

RADIUS Realm

Information About RADIUS Realm

The RADIUS Realm feature is associated with the domain of the user. Using this feature, a client can choose the RADIUS server through which authentication and accounting is to be processed.

When mobile clients are associated with a WLAN, RADIUS realm is received as a part of Extensible Authentication Protocol Method for UMTS Authentication and Key Agreement (EAP-AKA) identity response request in the authentication request packet. The Network Access Identifier (NAI) format (EAP-AKA) for WLAN can be specified as *username@domain.com*. The realm in the NAI format is represented after the @ symbol, which is specified as domain.com. If vendor-specific attributes are added as *test*, the NAI format is represented as test@domain.com.

The RADIUS Realm feature can be enabled and disabled on a WLAN. If Realm is enabled on a WLAN, the corresponding user should send the username in the NAI format. The embedded wireless controller sends the authentication request to the AAA server only when the realm, which is in the NAI format and is received from the client, is compiled as per the given standards. Apart from authentication, accounting requests are also required to be sent to the AAA server based on realm filtering.

Realm Support on a WLAN

Each WLAN is configured to support NAI realms. After the realm is enabled on a particular SSID, the lookup is done to match the realms received in the EAP identity response against the configured realms on the RADIUS server. If the client does not send a username with the realm, the default RADIUS server that is configured on the WLAN is used for authentication. If the realm that is received from the client does not match the configured realms on the WLAN, the client is deauthenticated and dropped.

If the RADIUS Realm feature is not enabled on a WLAN, the username that is received as part of the EAP identity request is directly used as the username and the configured RADIUS server is used for authentication and accounting. By default, the RADIUS Realm feature is disabled on WLANs.

- **Realm Match for Authentication**: In dot1x with EAP methods (similar to EAP AKA), the username is received as part of an EAP identity response. A realm is derived from the username and are matched

with the realms that are already configured in the corresponding RADIUS authentication server. If there is a match, the authentication requests are forwarded to the RADIUS server. If there is a mismatch, the client is deauthenticated.

- **Realm Match for Accounting**: A client's username is received through an access-accept message. When accounting messages are triggered, the realm is derived from the corresponding client's username and compared with the accounting realms configured on the RADIUS accounting server. If there is a match, accounting requests are forwarded to the RADIUS server. If there is a mismatch, accounting requests are dropped.

Enabling RADIUS Realm

Follow the procedure given below to enable RADIUS realm:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless aaa policy** *aaa-policy* **Example:** `Device(config)# wireless aaa policy policy-1`	Creates a new AAA policy.
Step 3	**aaa-realm enable** **Example:** `Device(config-aaa-policy)# aaa-realm enable`	Enables AAA RADIUS realm selection. **Note** Use the **no aaa-realm enable** or the **default aaa-realm enable** command to disable the RADIUS realm.

Configuring Realm to Match the RADIUS Server for Authentication and Accounting

Follow the procedure given below to configure the realm to match the RADIUS server for authentication and accounting:

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Creates a AAA authentication model.
Step 3	**aaa authorization network default group** *radius-server-group* **Example:** `Device(config)# aaa authorization network default group aaa_group_name`	Sets the authorization method.
Step 4	**aaa authentication dot1x** *realm* **group** *radius-server-group* **Example:** `Device(config)# aaa authentication dot1x cisco.com group cisco1`	Indicates that dot1x must use the realm group RADIUS server.
Step 5	**aaa authentication login** *realm* **group** *radius-server-group* **Example:** `Device(config)# aaa authentication login cisco.com group cisco1`	Defines the authentication method at login.
Step 6	**aaa accounting identity** *realm* **start-stop group** *radius-server-group* **Example:** `Device(config)# aaa accounting identity cisco.com start-stop group cisco1`	Enables accounting to send a start-record accounting notice when a client is authorized, and a stop-record at the end.

Configuring the AAA Policy for a WLAN

Follow the procedure given below to configure the AAA policy for a WLAN:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless aaa policy** *aaa-policy-name* **Example:** `Device(config)# wireless aaa policy aaa-policy-1`	Creates a new AAA policy for wireless.

	Command or Action	Purpose
Step 3	**aaa-realm enable** **Example:** `Device(config-aaa-policy)# aaa-realm enable`	Enables AAA RADIUS server selection by realm.
Step 4	**exit** **Example:** `Device(config-aaa-policy)# exit`	Returns to global configuration mode.
Step 5	**wireless profile policy** *wlan-policy-profile* **Example:** `Device(config)# wireless profile policy wlan-policy-a`	Configures a WLAN policy profile.
Step 6	**aaa-policy** *aaa-policy* **Example:** `Device(config-wireless-policy)# aaa-policy aaa-policy-1`	Maps the AAA policy.
Step 7	**accounting-list** *acct-config-realm* **Example:** `Device(config-wireless-policy)# accounting-list cisco.com`	Sets the accounting list.
Step 8	**exit** **Example:** `Device(config-wireless-policy)# exit`	Returns to global configuration mode.
Step 9	**wlan** *wlan-name wlan-id ssid* **Example:** `Device(config)# wlan wlan2 14 wlan-aaa`	Configures a WLAN.
Step 10	**security dot1x authentication-list** *auth-list-realm* **Example:** `Device(config-wlan)# security dot1x authentication-list cisco.com`	Enables the security authentication list for IEEE 802.1x.
Step 11	**exit** **Example:** `Device(config-wireless-policy)# exit`	Returns to global configuration mode.
Step 12	**wireless tag policy** *policy* **Example:** `Device(config)# wireless tag policy tag-policy-1`	Configures a policy tag.

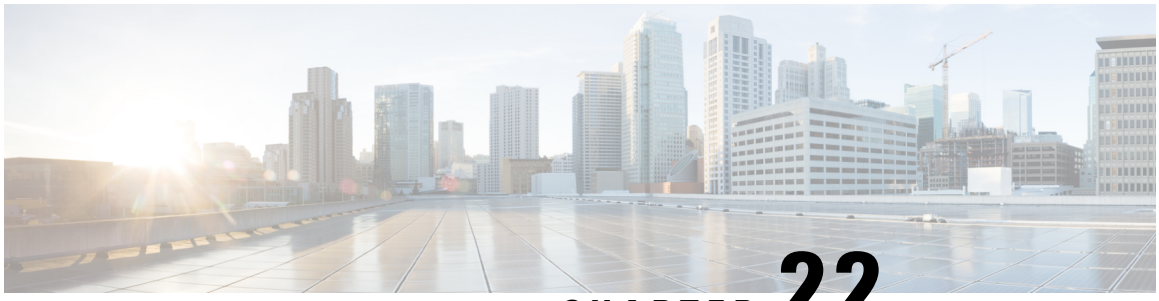
	Command or Action	Purpose
Step 13	**wlan** *wlan-name* **policy** *policy-profile* **Example:** `Device(config-policy-tag)# wlan Abc-wlan policy wlan-policy-a`	Maps a policy profile to the WLAN.
Step 14	**exit** **Example:** `Device(config-policy-tag)# exit`	Returns to global configuration mode.

Verifying the RADIUS-Realm Configuration

Use the following command to verify the RADIUS-realm configuration:

```
Device# show wireless client mac-address 14bd.61f3.6a24 detail


Client MAC Address : 14bd.61f3.6a24
Client IPv4 Address : 9.4.113.103
Client IPv6 Addresses : fe80::286e:9fe0:7fa6:8f4
Client Username : sacthoma@cisco.com
AP MAC Address : 4c77.6d79.5a00
AP Name: AP4c77.6d53.20ec
AP slot : 1
Client State : Associated
Policy Profile : name-policy-profile
Flex Profile : N/A
Wireless LAN Id : 3
Wireless LAN Name: ha_realm_WLAN_WPA2_AES_DOT1X
BSSID : 4c77.6d79.5a0f
Connected For : 26 seconds
Protocol : 802.11ac
Channel : 44
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Client CCX version : No CCX support
Re-Authentication Timeout : 1800 sec (Remaining time: 1775 sec)
Input Policy Name  : None
Input Policy State : None
Input Policy Source : None
Output Policy Name  : None
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Power Save : OFF
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 06/12/2018 19:52:35 IST
Policy Manager State: Run
```

```
NPU Fast Fast Notified : No
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 25 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : 802.1x
Encrypted Traffic Analytics : No
Management Frame Protection : No
Protected Management Frame - 802.11w : No
EAP Type : PEAP
VLAN : 113
Multicast VLAN : 0
Access VLAN : 113
Anchor VLAN : 0
WFD capable : No
Managed WFD capable : No
Cross Connection capable : No
Support Concurrent Operation : No
Session Manager:
  Interface        : capwap_9040000f
  IIF ID           : 0x9040000F
  Authorized       : TRUE
  Session timeout  : 1800
  Common Session ID: 097704090000000DF4607B3B
  Acct Session ID  : 0x00000fa2
  Aaa Server Details
  Server IP        : 9.4.23.50
  Auth Method Status List
       Method : Dot1x
              SM State         : AUTHENTICATED
              SM Bend State    : IDLE
  Local Policies:
       Service Template : wlan_svc_name-policy-profile_local (priority 254)
              Absolute-Timer   : 1800
              VLAN             : 113
  Server Policies:
  Resultant Policies:
              VLAN             : 113
              Absolute-Timer   : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
FlexConnect Data Switching : Central
FlexConnect Dhcp Status : Central
FlexConnect Authentication : Central
FlexConnect Central Association : No
.
.
.
Fabric status : Disabled
Client Scan Reports
Assisted Roaming Neighbor List
```



CHAPTER 22

Persistent SSID Broadcast

Persistent SSID Broadcast

Access Points within a mesh network work as Root Access Points (RAP) or Mesh Access Points (MAP). RAPs have wired connection to the embedded wireless controller and MAPs have wireless connection to the embedded wireless controller. This feature is applicable only to the Cisco Aironet 1542 Access Points in the Flex+Bridge mode.

This feature is about the Root Access Points (RAPs) and Mesh Access Points (MAPs) broadcasting the SSID even when the WAN connectivity is down. This is required in order to isolate the responsibility; whether the fault is with backhaul or with the access wireless network, since there can be different operators owning each part of the network.

RAPs and MAPs broadcast SSID while in standalone mode, as long as the default gateway is reachable.

Also refer Mesh Deployment Guide for Cisco Catalyst 9800 Series Wireless Controllers.

Configuring Persistent SSID Broadcast

Procedure

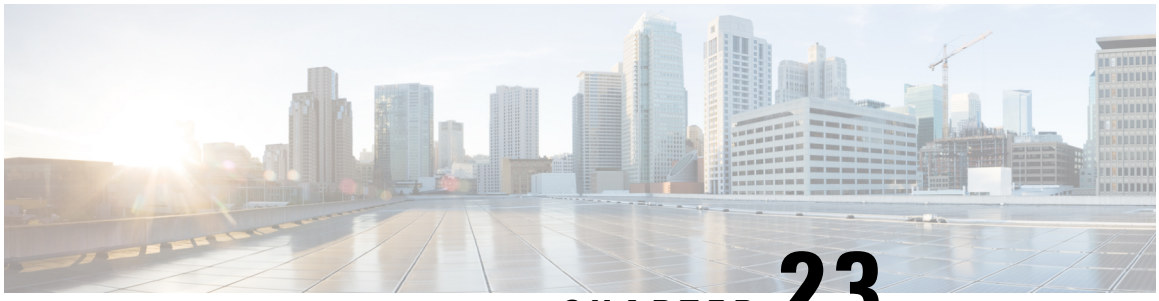
	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *ap-profile-name* **Example:** `Device(config)# ap profile ap-profile-name`	Configures the AP profile.

	Command or Action	Purpose
Step 3	**[no]ssid broadcast persistent** **Example:** `Device(config-ap-profile)# [no] ssid broadcast persistent`	The **ssid broadcast** command configures the SSID broadcast mode. The **persistent** keyword enables a persistent SSID broadcast, where the associated APs will re-join. Use the **[no]** form of the command to disable the feature. **Note** Enabling or disabling this feature causes the AP to re-join.

Verifying Persistent SSID Broadcast

To view the configuration of all Cisco APs, use the following **show** command:

```
Device#show ap config general
Cisco AP Name   : AP4C77.6DF2.D598
================================================
Office Extend Mode                              : Disabled
Persistent SSID Broadcast                       : Enabled
Remote AP Debug                                 : Disabled
```



CHAPTER 23

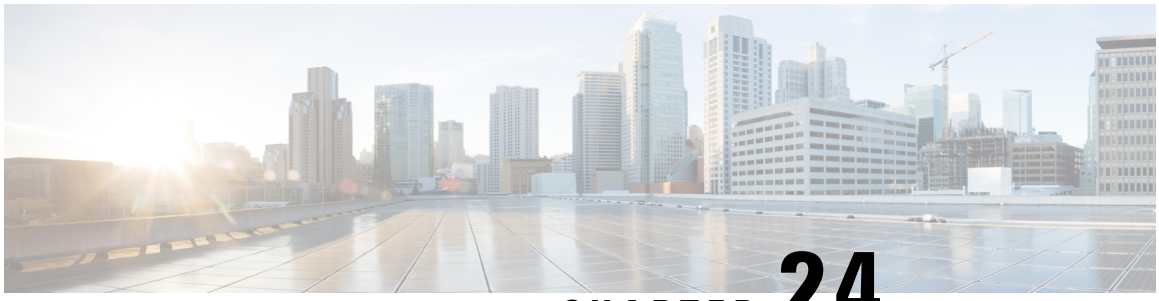
Network Monitoring

Network Monitoring

The only network monitoring supported on the Embedded Wireless Controller (EWC) is through Cisco Digital Network Architecture (DNA) Center. This is done through NETCONF using a propietary protocol for push and pull of configuration or status information.

PART V

System Management



CHAPTER 24

Network Mobility Services Protocol

Information About Network Mobility Services Protocol

Cisco Network Mobility Services Protocol (NMSP) is a secure two-way protocol that can be run over a connection-oriented (TLS) or connection-less (DTLS) transport. The wireless infrastructure runs the NMSP server and Cisco Connected Mobile Experiences (Cisco CMX) acts as an NMSP client. The embedded wireless controller supports multiple services and multiple Cisco CMXs can connect to the NMSP server to get the data for the services (location of wireless devices, probe RSSI, hyperlocation, wIPS, and so on.) over the NMSP session.

NMSP defines the intercommunication between Cisco CMX and the embedded wireless controller. Cisco CMX communicates to the embedded wireless controller over a routed IP network. Both publish-subscribe and request-reply communication models are supported. Typically, Cisco CMX establishes a subscription to receive services data from the embedded wireless controller in the form of periodic updates. The embedded wireless controller acts as a data publisher, broadcasting services data to multiple CMXs. Besides subscription, Cisco CMX can also send requests to the embedded wireless controller, causing the embedded wireless controller to send a response back.

NMSP essentially provides a way to the applications in the embedded wireless controller to talk to the outside world. The NMSP in the embedded wireless controller also provides the flexibility to change the protocol to talk to the outside world.

The following is a list of the Network Mobility Services Protocol features:

- NMSP is disabled by default.
- NMSP communicates with Cisco CMX using TCP, and uses TLS for encryption.

Note HTTPS is not supported for data transport between embedded wireless controller and Cisco CMX.

Enabling NMSP On-Premises Services

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**nmsp enable** **Example:** `Device(config)# nmsp enable`	Enables NMSP on premises services. **Note** By default, the NMSP is disabled on the embedded wireless controller.
Step 3	**end** **Example:** `Device(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

NMSP manages communication between the Cisco Connected Mobile Experiences (Cisco CMX) and the embedded wireless 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 embedded wireless controller and Cisco CMX communicate over must be open (not blocked) on any firewall that exists between the embedded wireless controller and the Cisco CMX for NMSP to function.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**end** **Example:** `Device(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, and Tags

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**location notify-threshold** {**clients** | **tags** } *threshold* **Example:** `Device(config)# location notify-threshold clients 5`	Configures the NMSP notification threshold for clients, and tags. *threshold*- RSSI threshold value in db. Valid range is from 0 to 10.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring NMSP Strong Cipher

Procedure

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

	Command or Action	Purpose
Step 2	**nmsp strong-cipher** **Example:** `Device(config)# nmsp strong-cipher`	Enable strong ciphers for NMSP server, which contains "ECDHE-RSA-AES128-GCM-SHA256:, ECDHE-ECDSA-AES128-GCM-SHA256:, AES256-SHA256:AES256-SHA:, and AES128-SHA256:AES128-SHA". Normal cipher suite contains, "ECDHE-RSA-AES128-GCM-SHA256:, ECDHE-ECDSA-AES128-GCM-SHA256:, and AES128-SHA".
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying NMSP Settings

To view the NMSP capabilities of the embedded wireless controller, use the following command:

```
Device# show nmsp capability
Service            Subservice
----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum           Aggregate Interferer, Air Quality, Interferer,
Info               Rogue, Mobile Station,
Statistics         Rogue, Tags, Mobile Station,
AP Monitor         Subscription
On Demand Services Device Info
AP Info            Subscription
```

To view the NMSP notification intervals, use the following command:

```
Device# show nmsp notification interval
NMSP Notification Intervals
---------------------------

RSSI Interval:
  Client              : 2 sec
  RFID                : 50 sec
  Rogue AP            : 2 sec
  Rogue Client        : 2 sec
  Spectrum            : 2 sec
```

To view the connection-specific statistics counters for all CMX connections, use the following command:

```
Device# show nmsp statistics connection
NMSP Connection Counters
------------------------
 CMX IP Address: 10.22.244.31, Status: Active
 State:
      Connections : 1
      Disconnections : 0
      Rx Data Frames : 13
      Tx Data Frames : 99244
      Unsupported messages : 0
```

```
Rx Message Counters:
   ID  Name                                  Count
  ---------------------------------------------
    1  Echo Request                         6076
    7  Capability Notification                 2
   13  Measurement Request                     5
   16  Information Request                     3
   20  Statistics Request                      2
   30  Service Subscribe Request               1

Tx Message Counters:
   ID  Name                                  Count
  ---------------------------------------------
    2  Echo Response                        6076
    7  Capability Notification                 1
   14  Measurement Response                   13
   15  Measurement Notification            91120
   17  Information Response                    6
   18  Information Notification             7492
   21  Statistics Response                     2
   22  Statistics Notification               305
   31  Service Subscribe Response              1
   67  AP Info Notification                  304
```

To view the common statistic counter of the embedded wireless controller's NMSP service, use the following command:

```
Device# show nmsp statistics summary
NMSP Global Counters
--------------------
Number of restarts                :

SSL Statistics
--------------------
Total amount of verifications     : 6
Verification failures             : 6
Verification success              : 0
Amount of connections created     : 8
Amount of connections closed      : 7
Total amount of accept attempts   : 8
Failures in accept                : 0
Amount of successful accepts      : 8
Amount of failed registrations    : 0

AAA Statistics
--------------------
Total amount of AAA requests      : 7
Failed to send requests           : 0
Requests sent to AAA              : 7
Responses from AAA                : 7
Responses from AAA to validate    : 7
Responses validate error          : 6
Responses validate success        : 1
```

To view the overall NMSP connections, use the following command:

```
Device# show nmsp status
NMSP Status
-----------

CMX IP Address   Active    Tx Echo Resp  Rx Echo Req   Tx Data     Rx Data     Transport
------------------------------------------------------------------------------------------
127.0.0.1        Active    6             6             1           2           TLS
```

To view all mobility services subscribed by all CMXs, use the following command:

```
Device# show nmsp subscription detail
CMX IP address 127.0.0.1:
Service            Subservice
----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription
```

To view all mobility services subscribed by a specific CMX, use the following command:

```
Device# show nmsp subscription detail <ip_addr>
CMX IP address 127.0.0.1:
Service            Subservice
----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription
```

To view the overall mobility services subscribed by all CMXs, use the following command:

```
Device# show nmsp subscription summary
Service            Subservice
----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription
```

Examples: NMSP Settings Configuration

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

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

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

```
Device# configure terminal
Device(config)# nmsp notification interval rssi clients 180
Device(config)# end
Device# show nmsp notification interval
```

Probe RSSI Location

The Probe RSSI Location feature allows the wireless embedded wireless controller and Cisco CMX to support the following:

- Load balancing
- Coverage Hole detection

- Location updates to CMX

When a wireless client is enabled, it sends probe requests to identify the wireless networks in the vicinity and also to find the received signal strength indication (RSSI) associated with the identified Service Set Identifiers (SSIDs).

The wireless client periodically performs active scanning in background even after being connected to an access point. This helps them to have an updated list of access points with best signal strength to connect. When the wireless client can no longer connect to an access point, it uses the access point list stored to connect to another access point that gives it the best signal strength. The access points in the WLAN gather these probe requests, RSSI and MAC address of the wireless clients and forwards them to the wireless embedded wireless controllers. The Cisco CMX gathers this data from the wireless embedded wireless controller and uses it to compute the updated location of the wireless client when it roams across the network.

Configuring Probe RSSI

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless probe filter** **Example:** `Device(config)# wireless probe filter`	Enables filtering of unacknowledged probe requests from AP to improve the location accuracy. Use the **no** form of the command to disable the feature. This will forward both acknowledged and unacknowledged probe requests to the embedded wireless controller.
Step 3	**wireless probe limit** *limit-value interval* **Example:** `Device(config)# wireless probe limit 10 100`	Configures the number of probe request reported to the wireless embedded wireless controller from the AP for the same client on a given interval. Use the **no** form of the command to revert to the default limit, which is 2 probes at an interval of 500 ms.
Step 4	**wireless probe locally-administered-mac** **Example:** `Device(config)# wireless probe locally-administered-mac`	Enables the reporting of probes from clients having locally administered MAC address.
Step 5	**location algorithm rssi-average** **Example:** `Device(config)# location algorithm rssi-average`	Sets the probe RSSI measurement updates to a more accurate algorithm but with more CPU overhead.

	Command or Action	Purpose
Step 6	**location algorithm simple** **Example:** `Device(config)# location algorithm simple`	(Optional) Sets the probe RSSI measurement updates to a faster algorithm with smaller CPU overhead, but less accuracy. Use the **no** form of the command to revert the algorithm type to the default one, which is *rssi-average*.
Step 7	**location expiry client** *interval* **Example:** `Device(config)# location expiry client 300`	Configures the timeout for RSSI values. The **no** form of the command sets it to a default value of 15.
Step 8	**location notify-threshold client** *threshold-db* **Example:** `Device(config)# location notify-threshold client 5`	Configures the notification threshold for clients. The **no** form of the command sets it to a default value of 0.
Step 9	**location rssi-half-life client** *time-in-seconds* **Example:** `Device(config)# location rssi-half-life client 20`	Configures half life when averaging two RSSI readings. To disable this option, set the value to 0.

What to do next

Use the **show wireless client probing** command to view each probing client (associated and probing only) by batch of 10 mac addresses.

Verifying Probe RSSI

To view the details of the AP the associated client was detected with, and with which RSSI:

```
Device# show wireless client mac-address 4.4.4 detail
****snippet of the output****
Nearby AP Statistics:
TEST_AP-1 (slot 0)
antenna 0: 0 s ago ........ -77 dBm
antenna 1: 0 s ago ........ -88 dBm
TEST_AP-5 (slot 0)
antenna 0: 0 s ago ........ -64 dBm
antenna 1: 0 s ago ........ -36 dBm
TEST_AP-6 (slot 0)
antenna 0: 0 s ago ........ -69 dBm
antenna 1: 0 s ago ........ -79 dBm
```

RFID Tag Support

The embedded wireless controller enables you to configure radio frequency identification (RFID) tag tracking. RFID tags are small wireless battery-powered tags that continuously broadcast their own signal and 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 embedded wireless controller, and the Cisco CMX. Only active RFIDs are supported. A combination of active RFID tags and wireless embedded wireless controller allows you to track the current location of equipment. *Active* tags are typically used in real-time tracking of high-value assets in *closed-loop* systems (that is,) systems in which the tags are not intended to physically leave the control premises of the tag owner or originator.

For more information on RFID tags, see the Active RFID Tags section of the *Wi-Fi Location-Based Services 4.1 Design Guide*.

General Guidelines

- Only Cisco-compliant active RFID tags are supported.
- You can verify the RFID tags on the embedded wireless controller.
- High Availability for RFID tags are supported.

Configuring RFID Tag Support

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless rfid** **Example:** `Device(config)# wireless rfid`	Enables RFID tag tracking. The default value is enabled. Use the **no** form of this command to disable RFID tag tracking.
Step 3	**wireless rfid timeout** *timeout-value* **Example:** `Device(config)# wireless rfid timeout 90`	Configures the RFID tag data timeout value to cleanup the table. The timeout value is the amount of time that the embedded wireless 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.

Verifying RFID Tag Support

To view the summary of RFID tags that are clients, use the following command:

```
Device# show wireless rfid client
```

To view the detailed information for an RFID tag, use the following command:

```
Device# show wireless rfid detail <rfid-mac-address>

RFID address  000c.cc96.0001
Vendor Cisco
Last Heard 6 seconds  ago
Packets Received 187
Bytes Received 226

Content Header
==============
  CCX Tag Version 0
  Tx power: 12
  Channel: 11
  Reg Class: 4
CCX Payload
==============
  Last Sequence Control 2735
  Payload length 221
  Payload Data Hex Dump:
  00000000  00 02 00 00 01 09 00 00  00 00 0c b8 ff ff ff 02  |................|
  00000010  07 42 03 20 00 00 0b b8  03 4b 00 00 00 00 00 00  |.B. .....K......|
  00000020  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
  00000030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
  00000040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

To view the summary information for all known RFID tags, use the following command:

```
Device# show wireless rfid summary

Total RFID entries: : 16
Total Unique RFID entries : 16
RFID ID VENDOR Closet AP RSSI Time Since Last Heard
0012.b80a.c791 Cisco 7069.5a63.0520 -31 3 minutes 30 seconds ago
0012.b80a.c953 Cisco 7069.5a63.0460 -33 4 minutes 5 seconds ago
0012.b80b.806c Cisco 7069.5a63.0520 -46 15 seconds ago
0012.b80d.e9f9 Cisco 7069.5a63.0460 -38 4 minutes 28 seconds ago
0012.b80d.ea03 Cisco 7069.5a63.0520 -43 4 minutes 29 seconds ago
0012.b80d.ea6b Cisco 7069.5a63.0460 -39 4 minutes 26 seconds ago
0012.b80d.ebe8 Cisco 7069.5a63.0520 -43 3 minutes 21 seconds ago
0012.b80d.ebeb Cisco 7069.5a63.0520 -43 4 minutes 28 seconds ago
0012.b80d.ec48 Cisco 7069.5a63.0460 -42 4 minutes 7 seconds ago
0012.b80d.ec55 Cisco 7069.5a63.0520 -41 1 minute 52 seconds ago
```

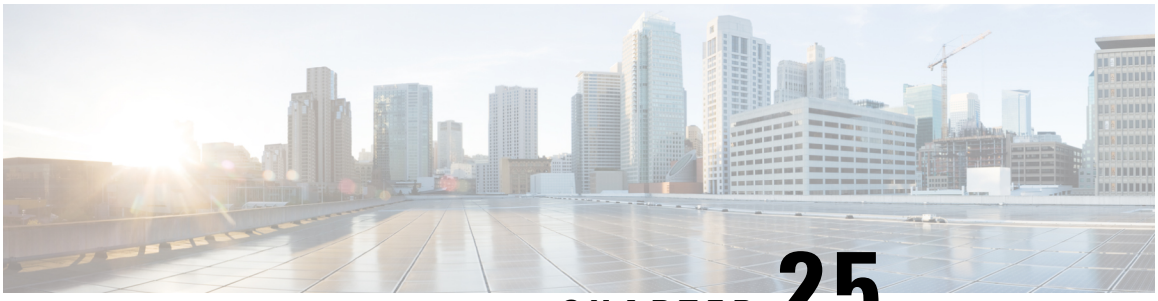
To view the location-based system RFID statistics, use the following command:

```
Device# show wireless rfid stats

RFID stats :
==============
RFID error db full : 0
RFID error invalid paylod : 0
RFID error invalid tag : 0
RFID error dot11 hdr : 0
RFID error pkt len : 0
RFID error state drop : 0
RFID total pkt received : 369
RFID populated error value : 0
RFID error insert records : 0
RFID error update records : 0
RFID total insert record : 16
RFID ccx payload error : 0
```

```
RFID total delete record : 0
RFID error exceeded ap count : 0
RFID error record remove : 0
RFID old rssi expired count: 0
RFId smallest rssi expireed count : 0
RFID total query insert : 0
RFID error invalid rssi count : 0
```

To view the NMSP notification interval, use the following command:

```
Device# show nmsp notification interval

NMSP Notification Intervals
---------------------------

RSSI Interval:
  Client              : 2 sec
  RFID                : 50 sec
  Rogue AP            : 2 sec
  Rogue Client        : 2 sec
  Spectrum            : 2 sec
```



CHAPTER 25

Application Visibility and Control

Information About Application Visibility and Control

Application Visibility and Control (AVC) is a subset of the entire Flexible NetFlow (FNF) package that can provide traffic information. The AVC feature employs a distributed approach that benefits from NBAR running on the access point (AP) or embedded wireless controller whose goal is to run deep packet inspection (DPI) and reports the results using FNF messages.

AVC enables you to perform real-time analysis and create policies to reduce network congestion, costly network link usage, and infrastructure upgrades. Traffic flows are analyzed and recognized using the NBAR2 engine. The specific flow is marked with the recognized protocol or application. This per-flow information can be used for application visibility using FNF. After the application visibility is established, a user can define control rules with policing mechanisms for a client.

Using AVC rules, you can limit the bandwidth of a particular application for all the clients joined on the WLAN. These bandwidth contracts coexist with per-client downstream rate limiting that takes precedence over the per-application rate limits.

FNF feature is supported in wireless, and relies on the NetFlow enablement on the embedded wireless controller for flex mode.

The behavior of the AVC solution changes based on the wireless deployments. The following sections describe the commonalities and differences in all scenarios:

Flex Mode

- NBAR is enabled on an AP
- AVC pushes the FNF configuration to the APs.
- Supports context transfer for roaming in AVC-FNF.
- Supports NetFlow exporter.

Prerequisites for Application Visibility and Control

- The access points should be AVC capable.
- For the control part of AVC (QoS) to work, the application visibility feature with FNF has to be configured.

Restrictions for Application Visibility and Control

- Layer 2 roaming is not supported across embedded wireless controllercontrollers.
- Multicast traffic is not supported.
- AVC is supported only on the following access points:
  - Cisco Aironet 1800 Series Access Points
  - Cisco Aironet 2700 Series Access Point
  - Cisco Aironet 2800 Series Access Point
  - Cisco Aironet 3700 Series Access Points
  - Cisco Aironet 3800 Series Access Points
  - Cisco Aironet 4800 Series Access Points
- AVC is not supported on Cisco Aironet 702W, 702I (128 M memory), and 1530 Series access points.
- Only the applications that are recognized with App visibility can be used for applying QoS control.
- Data link is not supported for NetFlow fields in AVC.
- You cannot map the same WLAN profile to both the AVC-not-enabled policy profile and the AVC-enabled policy profile.
- NBAR-based QoS policy configuration is allowed at client level and BSSID level, configured on policy profile.

When AVC is enabled, the AVC profile supports only up to 23 rules, which includes the default DSCP rule. The AVC policy will not be pushed down to the AP, if rules are more than 23.

AVC Configuration Overview

To configure AVC, follow these steps:

1. Create a flow monitor using the **record wireless avc basic** command.
2. Create a wireless policy profile.
3. Apply the flow monitor to the wireless policy profile.
4. Create a wireless policy tag.
5. Map the WLAN to the policy profile
6. Attach the policy tag to the APs.

Create a Flow Monitor

The NetFlow configuration requires a flow record, a flow monitor, and a flow exporter. This configuration should be the first step in the overall AVC configuration.

Note In Flex mode, the default values for **cache timeout active** and **cache timeout inactive** commands are not optimal for AVC. We recommend that you set both the values to 60 in the flow monitor.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor fm_avc`	Creates a flow monitor.

Step 3	**record wireless avc basic**
	Example:
	`Device(config-flow-monitor)# record wireless`
	`avc basic`
		Specifies the basic wireless AVC flow template.
		Note The **record wireless avc basic** command is same as **record wireless avc ipv4 basic** command. However, **record wireless avc ipv4 basic** command is not supported in Flex or Fabric modes. In such scenarios, use the **record wireless avc basic** command.

Configuring a Flow Monitor (GUI)

Before you begin

You must have created a flow exporter to export data from the flow monitor.

Procedure

Step 1 Choose **Configuration** > **Services** > **Application Visibility** and go to the **Flow Monitor** tab .

Step 2 In the **Monitor** area, click **Add** to add a flow monitor.

Step 3 In the **Flow Monitor** window, add a flow monitor and a description.

Step 4 Select the Flow exporter from the drop-down list to export the data from the flow monitor to a collector.

Note To export wireless netflow data, use the templates below:

- ETA (Encrypted Traffic Analysis)
- wireless avc basic
- wireless avc basic IPv6

Step 5 Click **Apply to Device** to save the configuration.

Create a Flow Exporter

You can create a flow exporter to define the export parameters for a flow. This is an optional procedure for configuring flow exporter parameters.

Note For the AVC statistics to be visible at the embedded wireless controller, you should configure a local flow exporter using the following commands:

- **flow exporter** *my_local*
- **destination local wlc**

Also, your flow monitor must use this local exporter for the statistics to be visible at the embedded wireless controller.

Procedure

	Command or Action	Purpose
Step 1	**flow exporter** *flow-export-name* **Example:**	Creates a flow monitor.

	Command or Action	Purpose
	`Device(config)# flow exporter export-test`	
Step 2	**description** *string* **Example:** `Device(config-flow-exporter)# description IPv4flow`	Describes the flow record as a maximum 63-character string.
Step 3	**Example:** `Device(config-flow-exporter)# destination local wlc`	Specifies the local WLC to which the exporter sends data.
Step 4	**end** **Example:** `Device(config-flow-exporter)# end`	Returns to privileged EXEC mode.
Step 5	**show flow exporter** **Example:** `Device# show flow exporter`	(Optional) Verifies your configuration.

Verify the Flow Exporter

To verify the flow exporter description, use the following command:

For example, to verify the flow exporter description for the flow exporter named *my-flow-exporter*, see the example below:

```
Device# show flow exporter
Flow Exporter my-flow-exporter:
  Description:              User defined
  Export protocol:          NetFlow Version 9
  Transport Configuration:
    Destination type:       Local (1)
    Destination IP address: 0.0.0.0
    Source IP address:      10.0.0.1
    Transport Protocol:     UDP
    Destination Port:       9XXX
    Source Port:            5XXXX
    DSCP:                   0x0
    TTL:                    255
    Output Features:        Not Used
```

Note A flow exporter with no destination is marked as an UNKNOWN type. The following are the two ways the exporter is marked as UNKNOWN:

1. When you configure the flow exporter using the CLI commands without a destination.
2. EWC supports a maximum of one external and one internal flow exporter. If you attempt to configure more than one flow exporter per type, this results in the destination to be rejected and the flow exporter will be considered as UNKNOWN.

Configuring a Policy Tag

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless tag policy** *policy-tag-name* **Example:** `Device(config-policy-tag)# wireless tag policy rr-xyz-policy-tag`	Configures policy tag and enters policy tag configuration mode.
Step 3	**end** **Example:** `Device(config-policy-tag)# end`	Saves the configuration and exits configuration mode and returns to privileged EXEC mode.

Attaching a Policy Profile to a WLAN Interface (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.
Step 2 On the**Manage Tags** page, click **Policy** tab.
Step 3 Click **Add** to view the **Add Policy Tag** window.
Step 4 Enter a name and description for the policy tag.
Step 5 Click **Add** to map WLAN and policy.
Step 6 Choose the WLAN profile to map with the appropriate policy profile, and click the tick icon.
Step 7 Click **Save & Apply to Device**.

Attaching a Policy Profile to a WLAN Interface (CLI)

Before you begin

- Do not attach different AVC policy profiles on the same WLAN across different policy tags.

  The following is an example of incorrect configuration:

```
wireless profile policy avc_pol1
   ipv4 flow monitor fm-avc1 input
```

```
 ipv4 flow monitor fm-avc1 output
 no shutdown
wireless profile policy avc_pol2
 ipv4 flow monitor fm-avc2 input
 ipv4 flow monitor fm-avc2 output
 no shutdown
wireless tag policy avc-tag1
 wlan wlan1 policy avc_pol1
wireless tag policy avc-tag2
 wlan wlan1 policy avc_pol2
```

This example violates the restriction stated earlier, that is, the WLAN *wlan1* is mapped to 2 policy profiles, *avc_pol1* and *avc_pol2*. This configuration is, therefore, incorrect because the WLAN *wlan1* should be mapped to either *avc_pol1* or *avc_pol2* everywhere.

- Conflicting policy profiles on the same WLAN are not supported. For example, policy profile (with and without AVC) applied to the same WLAN in different policy tags.

  The following is an example of an incorrect configuration:

```
wireless profile policy avc_pol1
   no shutdown
  wireless profile policy avc_pol2
   ipv4 flow monitor fm-avc2 input
   ipv4 flow monitor fm-avc2 output
   no shutdown
  wireless tag policy avc-tag1
   wlan wlan1 policy avc_pol1
  wireless tag policy avc-tag2
   wlan wlan1 policy avc_pol2
```

  In this example, a policy profile with and without AVC is applied to the same WLAN in different tags.

Procedure

	Command or Action	Purpose
Step 1	**wireless tag policy** *avc-tag* **Example:** `Device(config)# wireless tag policy avc-tag`	Creates a policy tag.
Step 2	**wlan** *wlan-avc* **policy** *avc-policy* **Example:** `Device(config-policy-tag)# wlan wlan_avc policy avc_pol`	Attaches a policy profile to a WLAN profile.

What to do next

- Run the **no shutdown** command on the WLAN after completing the configuration.
- If the WLAN is already in **no shutdown** mode, run the **shutdown** command, followed by **no shutdown** command.

Attaching a Policy Profile to an AP

Procedure

	Command or Action	Purpose
Step 1	**ap** *ap-ether-mac* **Example:** `Device(config)# ap 34a8.2ec7.4cf0`	Enters AP configuration mode.
Step 2	**policy-tag** *policy-tag* **Example:** `Device(config)# policy-tag avc-tag`	Specifies the policy tag that is to be attached to the access point.

Verify the AVC Configuration

Procedure

	Command or Action	Purpose
Step 1	**show avc wlan** *wlan-name* **top** *num-of-applications* **applications {aggregate | downstream | upstream}** **Example:** `Device# show avc wlan wlan_avc top 2 applications aggregate`	Displays information about top applications and users using these applications. **Note** Ensure that wireless clients are associated to the WLAN and generating traffic, and then wait for 90 seconds (to ensure the availability of statistics) before running the command.
Step 2	**show avc client** *mac* **top** *num-of-applications* **applications {aggregate | downstream | upstream}** **Example:** `Device# show avc client 9.3.4 top 3 applications aggregate`	Displays information about the top number of applications. **Note** Ensure that wireless clients are associated to the WLAN and generating traffic, and then wait for 90 seconds (to ensure the availability of statistics) before running the command.
Step 3	**show avc wlan** *wlan-name* **application** *app-name* **top** *num-of-clients* **aggregate** **Example:** `Device# show avc wlan wlan_avc application app top 4 aggregate`	Displays information about top applications and users using these applications.

	Command or Action	Purpose
Step 4	**show ap summary** **Example:** `Device# show ap summary`	Displays a summary of all the access points attached to the embedded wireless controller.
Step 5	**show ap tag summary** **Example:** `Device# show ap tag summary`	Displays a summary of all the access points with policy tags.

AVC-Based Selective Reanchoring

The AVC-Based Selective Reanchoring feature is designed to reanchor clients when they roam from one embedded wireless controller to another. Reanchoring of clients prevents the depletion of IP addresses available for new clients in Cisco WLC. The AVC profile-based statistics are used to decide whether a client must be reanchored or deferred. This is useful when a client is actively running a voice or video application defined in the AVC rules.

The reanchoring process also involves deauthentication of anchored clients. The clients get deauthenticated when they do not transmit traffic for the applications listed in the AVC rules while roaming between WLCs.

Restrictions for AVC-Based Selective Reanchoring

- This feature is supported only in local mode. FlexConnect and fabric modes are not supported.
- This feature is not supported in guest tunneling and export anchor scenarios.
- The old IP address is not released after reanchoring, until IP address' lease period ends.

Configuring the Flow Exporter

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow exporter** *name* **Example:** `Device(config)# flow exporter avc-reanchor`	Creates a flow exporter and enters flow exporter configuration mode. **Note** You can use this command to modify an existing flow exporter too.

	Command or Action	Purpose
Step 3	**destination local wlc** **Example:** `Device(config-flow-exporter)# destination local wlc`	Sets the exporter as local.

Configuring the Flow Monitor

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor fm_avc`	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. **Note** You can use this command to modify an existing flow monitor too.
Step 3	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter avc-reanchor`	Specifies the name of an exporter.
Step 4	**record wireless avc basic** **Example:** `Device(config-flow-monitor)# record wireless avc basic`	Specifies the flow record to use to define the cache.
Step 5	**cache timeout active** *value* **Example:** `Device(config-flow-monitor)# cache timeout active 60`	Sets the active flow timeout, in seconds.
Step 6	**cache timeout inactive** *value* **Example:** `Device(config-flow-monitor)# cache timeout inactive 60`	Sets the inactive flow timeout, in seconds.

Configuring the AVC Reanchoring Profile

Before you begin

- Ensure that you use the AVC-Reanchor-Class class map. All other class-map names are ignored by Selective Reanchoring.
- During boot up, the system checks for the existence of the AVC-Reanchor-Class class map. If it is not found, default protocols, for example, jabber-video, wifi-calling, and so on, are created. If AVC-Reanchor-Class class map is found, configuration changes are not made and updates to the protocols that are saved to the startup configuration persist across reboots.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**class-map** *cmap-name* **Example:** `Device(config)# class-map AVC-Reanchor-Class`	Configures the class map.
Step 3	**match any** **Example:** `Device(config-cmap)# match any`	Instructs the device to match with any of the protocols that pass through it.
Step 4	**match protocol jabber-audio** **Example:** `Device(config-cmap)# match protocol jabber-audio`	Specifies a match to the application name. You can edit the class-map configuration later, in order to add or remove protocols, for example, jabber-video, wifi-calling, and so on, if required.

Configuring the Wireless WLAN Profile Policy

Follow the procedure given below to configure the WLAN profile policy:

Procedure

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

	Command or Action	Purpose
Step 2	**wireless profile policy** *policy-name* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Configures the WLAN policy profile and enters wireless policy configuration mode.
Step 3	**shutdown** **Example:** `Device(config-wireless-policy)# shutdown`	Disables the policy profile.
Step 4	**central switching** **Example:** `Device(config-wireless-policy)# central switching`	Enables central switching.
Step 5	**ipv4 flow monitor** *monitor-name* **input** **Example:** `Device(config-wireless-policy)# ipv4 flow monitor fm_avc input`	Specifies the name of the IPv4 ingress flow monitor.
Step 6	**ipv4 flow monitor** *monitor-name* **output** **Example:** `Device(config-wireless-policy)# ipv4 flow monitor fm_avc output`	Specifies the name of the IPv4 egress flow monitor.
Step 7	**reanchor class** *class-name* **Example:** `Device(config-wireless-policy)# reanchor class AVC-Reanchor-Class`	Configure a class map with protocols for the Selective Reanchoring feature.
Step 8	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Enables the policy profile.

Verifying AVC Reanchoring

Use the following commands to verify the AVC reanchoring configuration:

```
Device# show wireless profile policy detailed avc_reanchor_policy

Policy Profile Name           : avc_reanchor_policy
Description                   :
Status                        : ENABLED
VLAN                          : 1
Wireless management interface VLAN        : 34
!
.
.
```

```
.
AVC VISIBILITY                : Enabled
Flow Monitor IPv4
  Flow Monitor Ingress Name   : fm_avc
  Flow Monitor Egress Name    : fm_avc
Flow Monitor IPv6
  Flow Monitor Ingress Name   : Not Configured
  Flow Monitor Egress Name    : Not Configured
NBAR Protocol Discovery       : Disabled
Reanchoring                   : Enabled
Classmap name for Reanchoring
  Reanchoring Classmap Name   : AVC-Reanchor-Class
!
.
.
.
  ------------------------------------------------------
```

```
Device# show platform software trace counter tag wstatsd chassis active R0 avc-stats debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_deassociated_clients 28340 1
Reanch_tracked_clients 28340 4
Reanch_deleted_clients 28340 3
```

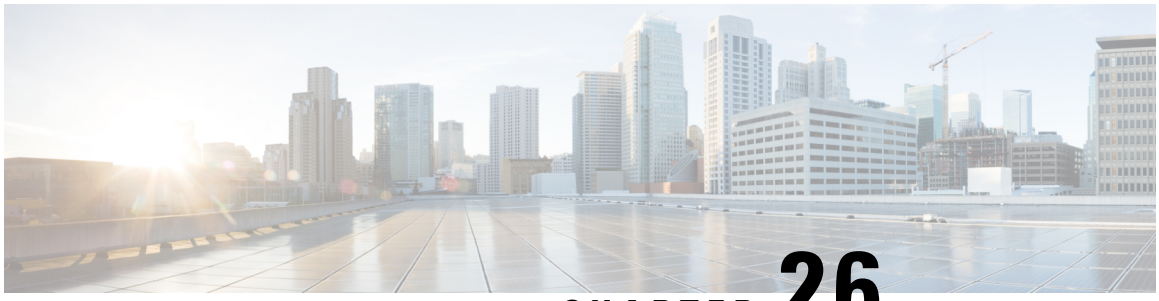
```
Device# show platform software trace counter tag wncd chassis active R0 avc-afc debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_co_ignored_clients 30063 1
Reanch_co_anchored_clients 30063 5
Reanch_co_deauthed_clients 30063 4
```

```
Device# show platform software wlavc status wncd

Event history of WNCD DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------ -------------------- -------------------------- ---- ----
06/12/2018 16:45:30.630342 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822780 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822672 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172073 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738367 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738261 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.162689 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757643 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757542 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.468749 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18857 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18717 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164304 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163877 2 :READY 1 :FSM_AFM_BIND 0 2
```

```
06/12/2018 16:35:18.593257 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:18.593152 1 :INIT 24:CREATE_FSM 0 0

AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- -------------------------- ---- ----
06/12/2018 16:45:30.664772 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822499 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822222 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.207605 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738105 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.737997 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164225 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757266 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757181 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472778 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.15413 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.15263 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164254 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163209 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163189 1 :INIT 24:CREATE_FSM 0 0

AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- -------------------------- ---- ----
06/12/2018 16:45:30.630764 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822621 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822574 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172357 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738212 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738167 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164048 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757403 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757361 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472561 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18660 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18588 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164293 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163799 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163773 1 :INIT 24:CREATE_FSM 0 0


Device# show platform software wlavc status wncmgrd

Event history of WNCMgr DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
Wlan-id : 1
```

```
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.629278 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629223 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629179 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510867 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510411 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510371 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886377 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
!

AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.664032 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.663958 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.663921 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.511151 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510624 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510608 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.810867 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807239 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807205 4 :READY 23:UNBIND_IOSD 0 0
06/12/2018 16:45:28.806734 4 :READY 3 :FSM_WLAN_DOWN 0 0
!

AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ---- ----
06/12/2018 16:45:30.629414 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629392 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629380 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510954 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510572 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510532 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886293 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807844 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807795 4 :READY 23:UNBIND_IOSD 0 0
```

```
06/12/2018 16:45:28.806990 4 :READY 3 :FSM_WLAN_DOWN 0 0
!
```



CHAPTER 26

Flexible NetFlow Exporter on Embedded Wireless Controller

Flexible NetFlow Exporter on Embedded Wireless Controller

Flexible Netflow (FnF) Exporter on Embedded Wireless Controller (EWC) is supported from Cisco IOS XE Amsterdam 17.2.1 onwards.

NetFlow is a Cisco IOS technology that provides statistics on packets flowing on the network. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides data to support 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.

FnF Exporter in EWC is supported only in the flex mode.

This feature is part of the AVC solution in EWC. For more information about AVC, refer to the *Application Visibility and Control* chapter.

AVC Configuration Limitations on EWC

- Only one local exporter (statistics collector on EWC) is supported.
- FnF supports only one per IP-type and direction in Flex mode, for Flow Monitor.
- Support of only UDP transport protocol.
- AVC cache is not supported.
- The **option** command and the command related to DP statistics are not supported on EWC.
- Support of only Wireless AVC Basic template.

- Support for only Netflow Version 9.
- IP address 0.0.0.0 is a valid destination address. However, if you use it, the Flexible NetFlow data will be discarded and not collected by any collector.

Create a Flow Exporter

The following procedure shows how to create a flow exporter in EWC:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow exporter** *flow-export-name* **Example:** `Device(config)# flow exporter export-test`	Creates a flow exporter.
Step 3	**description** *string* **Example:** `Device(config-flow-exporter)# description IPv4flow`	(Optional) Describes the flow exporter as a maximum 63-character string.
Step 4	**Example:** `Device(config-flow-exporter)# destination 10.0.1.0`	

Create a Flow Monitor

The NetFlow configuration requires a flow record, a flow monitor, and a flow exporter. This configuration should be the first step in the overall AVC configuration.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow monitor** *monitor-name* **Example:** `Device(config)# flow monitor monitor-test`	Creates a flow monitor.

	Command or Action	Purpose
Step 3	**exporter** *exporter-name* **Example:** `Device(config-flow-monitor)# exporter export-test`	Binds this flow monitor with an already defined flow exporter.
Step 4	**record wireless avc basic** **Example:** `Device(config-flow-monitor)# record wireless avc basic`	Specifies the basic wireless AVC flow template.

Configuring the Wireless WLAN Profile Policy

This configuration maps the flow-monitor or exporter constructs with wireless WLANs, thereby making APs collect FnF measurements.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-name* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Configures the WLAN policy profile and enters wireless policy configuration mode.
Step 3	**shutdown** **Example:** `Device(config-wireless-policy)# shutdown`	Disables the policy profile.
Step 4	{**ipv4** | **ipv6**} **flow monitor** *monitor-name* **input** **Example:** `Device(config-wireless-policy)# ipv4 flow monitor monitor-test input`	Specifies the name of the IPv4 or IPv6 ingress flow monitor.
Step 5	{**ipv4** | **ipv6**} **flow monitor** *monitor-name* **output** **Example:** `Device(config-wireless-policy)# ipv4 flow monitor monitor-test output`	Specifies the name of the IPv4 or IPv6 egress flow monitor.

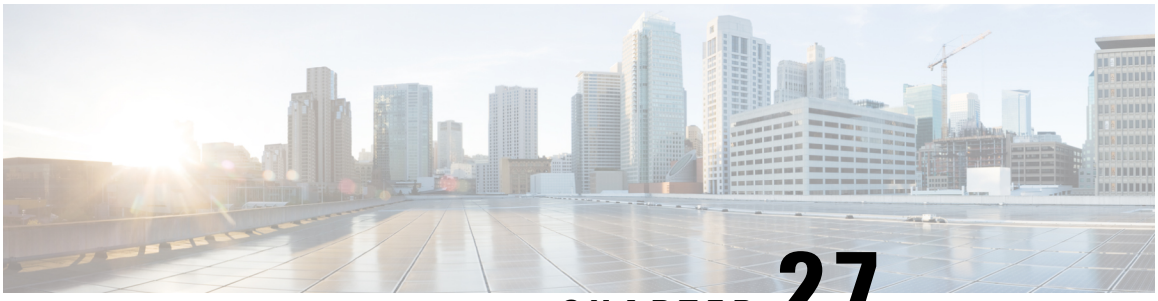
	Command or Action	Purpose
Step 6	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Enables the policy profile.

Verifying Flow Exporter in Embedded Wireless Controller

To view the flow exporter details in the Embedded Wireless Controller, use the following command:

show platform software wlavc status cp-exporter

```
show platform software wlavc status cp-exporter
AVC FNF Exporter status
IP: 10.10.1.1
connection statistics
        Sent bytes : 5672
        Sent packets : 569
        Sent records : 240
        Received packets : 800
        Received records : 564
Socket statistics
        New sockets : 3
        Closed sockets : 0
Library statistics  AVC
        cache errors : 0
        Unexpected Flow Monitor ID : 0
        Socket creation error : 0
```



CHAPTER 27

Cisco Connected Mobile Experiences Cloud

Cisco Connected Mobile Experiences (CMX) communicates with the Cisco wireless embedded wireless controller using the Network Mobility Services Protocol (NMSP), which runs over a connection-oriented (TLS) transport. This transport provides a secure 2-way connectivity and is convenient when both the embedded wireless controller and CMX are on-premise and there is direct IP connectivity between them.

Cisco CMX Cloud is a cloud-delivered version of the on-premise CMX. To access Cisco CMX Cloud services, HTTPS is used as a transport protocol.

Configuring Cisco CMX Cloud

Follow the procedure given below to configure CMX Cloud:

Before you begin

- **Configure DNS**—To resolve fully qualified domain names used by NMSP cloud-services, configure a **DNS** using the **ip name-server** *server_address* configuration command as shown in Step 2.
- **Import 3rd party root CAs**—The controller verifies the peer and the host based on the certificate that is sent by the CMX when a connection is established. However, root CAs are not preinstalled on the controller. You have to import a set of root CAs trusted by Cisco to the trustpool of the crypto PKI by using the **crypto pki trustpool import url <url>** configuration command as shown in Step 3.
- A successful registration to Cisco Spaces is required to enable **server url** and **server token** parameters configuration which is needed to complete this setup.

Procedure

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

	Command or Action	Purpose
Step 2	**ip name-server** *namesvr-ip-addr* **Example:** `Device(config)#ip name-server 10.10.10.205`	Configures the DNS on the controller to resolve the FQDN names used by the NMSP cloud-services.
Step 3	**crypto pki trustpool import url** *url* **Example:** `Device(config)#crypto pki trustpool import url http://www.cisco.com/security/pki/trs/ios.p7b`	Imports the 3rd party root CA. The controller verifies the peer using the imported certificate.
Step 4	**[no] nmsp cloud-services server url** *url* **Example:** `Device(config)# nmsp cloud-services server url https://cisco.com`	Configures the URL used for cloud services. Use the **no** form of the command to delete the server url from the configuration.
Step 5	**[no] nmsp cloud-services server token** *token* **Example:** `Device(config)# nmsp cloud-services server token test`	Configures the authentication token for the NMSP cloud service. Use the **no** form of the command to delete the server token from the configuration.
Step 6	**[no] nmsp cloud-services http-proxy** *proxy-server port* **Example:** `Device(config)# nmsp cloud-services http-proxy 10.0.0.1 10`	(Optional) Configures HTTP proxy details for the NMSP cloud service. Use the **no** form of the command to disable the use of a HTTP proxy.
Step 7	**[no] nmsp cloud-services enable** **Example:** `Device(config)# nmsp cloud-services enable`	Enables NMSP cloud services. Use the **no** form of the command to disable the feature.

Verifying Cisco CMX Cloud Configuration

Use the following commands to verify the CMX Cloud configuration.

To view the status of active NMSP connections, use the following command:

```
Device# show nmsp status

MSE IP Address   Tx Echo Resp  Rx Echo Req   Tx Data     Rx Data   Transport
------------------------------------------------------------------------------
9.9.71.78        0             0             1           1         TLS
64.103.36.133    0             0             1230        2391      HTTPs
```

To view the NMSP cloud service status, use the following command:

```
Device# show nmsp cloud-services summary

CMX Cloud-Services Status
```

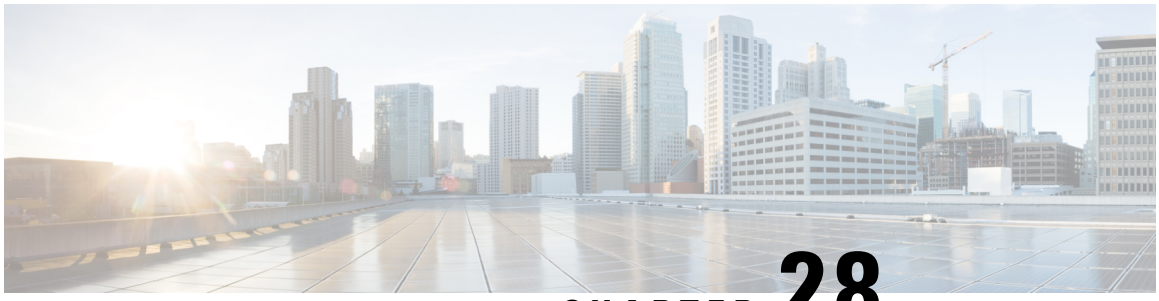
```
-------------------------

   Server:                                  https://yenth8.cmxcisco.com
   IP Address:                              64.103.36.133
   Cmx Service:                             Enabled
   Connectivity:                            https: UP
   Service Status:                          Active
   Last Request Status:                     HTTP/1.1 200 OK
   Heartbeat Status:                        OK
```

To view the NMSP cloud service statistics, use the following command:

```
Device# show nmsp cloud-services statistics

CMX Cloud-Services Statistics
-----------------------------

   Tx DataFrames:                             3213
   Rx DataFrames:                             1606
   Tx HeartBeat Req:                         31785
   Heartbeat Timeout:                            0
   Rx Subscr Req:                             2868
   Tx DataBytes:                             10069
   Rx DataBytes:                             37752
   Tx HeartBeat Fail:                            2
   Tx Data Fail:                                 0
   Tx Conn Fail:                                 0
```

To view the mobility services summary, use the following command:

```
Device# show nmsp subscription summary

Mobility Services Subscribed:
Index Server IP Services
----- --------- --------
1 209.165.200.225 RSSI, Info, Statistics, AP Monitor, AP Info
2 209.165.200.225 RSSI, Statistics, AP Info
```



CHAPTER 28

EDCA Parameters

Enhanced Distributed Channel Access 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.

This section contains the following subsections:

Configuring EDCA Parameters (GUI)

Procedure

Step 1 Choose **Configuration > Radio Configurations > Parameters**. Using this page, you can configure global parameters for 802.11a/n/ac (5 GHz) and 802.11b/g/n (2.4 GHz) radios.

Note You cannot configure or modify parameters, if the radio network is enabled. Disable the network status on the Configuration > Radio Configurations > Network page before you proceed.

Step 2 In the **EDCA Parameters** section, choose an EDCA profile from the **EDCA Profile** drop-down list. Enhanced Distributed Channel Access (EDCA) parameters are designed to provide preferential wireless channel access for voice, video, and other quality-of-service (QoS) traffic.

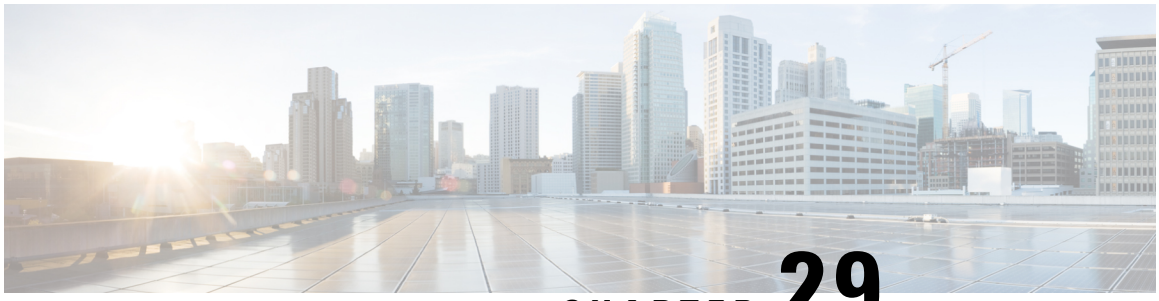
Step 3 Click **Apply**.

Configuring EDCA Parameters (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz } shutdown** **Example:** `Device(config)# ap dot11 5ghz shutdown`	Disables the radio network.
Step 3	**ap dot11 {5ghz | 24ghz} edca-parameters {custom-voice | fastlane | optimized-video-voice | optimized-voice | svp-voice | wmm-default}** **Example:** `Device(config)# ap dot11 5ghz edca-parameters optimized-voice`	Enables 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. • **fastlane**: Enables the fastlane parameters for the 802.11a or 802.11b/g network. • **optimized-video-voice**: Enables EDCA voice-optimized 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. • **wmm-default**: Enables the Wi-Fi Multimedia (WMM) default parameters for the 802.11a or 802.11b/g network. This is the default option. Choose this option when voice or video services are not deployed on your network.

	Command or Action	Purpose
Step 4	**no ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Device(config)# no ap dot11 5ghz shutdown`	Re-enables the radio network.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**show ap dot11 {5ghz | 24ghz} network** **Example:** `Device# show ap dot11 5ghz network`	Displays the current status of MAC optimization for voice.



CHAPTER 29

802.11 parameters and Band Selection

Information About Configuring Band Selection, 802.11 Bands, and Parameters

Band Select

Band select enables client radios that are capable of dual-band (2.4 and 5-GHz) operations 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 3 nonoverlapping channels. To prevent these sources of interference and improve overall network performance, configure band selection on the device.

Band select works by regulating probe responses to clients and it can be enabled on a per-WLAN basis. It makes 5-GHz channels more attractive to clients by delaying probe responses to clients on 2.4-GHz channels. In an access point, the band select table can be viewed by running the **show dot11 band-select** command. It can also be viewed by running the **show cont d0/d1 | begin Lru** command.

Band Select Algorithm

The band select algorithm affects clients that use 2.4-GHz band. Initially, when a client sends a probe request to an access point, the corresponding client probe's Active and Count values (as seen from the band select table) become 1. The algorithm functions based on the following scenarios:

- Scenario1: Client RSSI (as seen from the **show cont d0/d1** | **begin RSSI**command output) is greater than both Mid RSSI and Acceptable Client RSSI.
  - Dual-band clients: No 2.4-GHz probe responses are seen at any time; 5-GHz probe responses are seen for all 5-GHz probe requests.
  - Single-band (2.4-GHz) clients: 2.4-GHz probe responses are seen only after the probe suppression cycle.

- After the client's probe count reaches the configured probe cycle count, the algorithm waits for the Age Out Suppression time and then marks the client probe's Active value as 0. Then, the algorithm is restarted.

- Scenario2: Client RSSI (as seen from **show cont d0/d1** | **begin RSSI**) lies between Mid-RSSI and Acceptable Client RSSI.
  - All 2.4-GHz and 5-GHz probe requests are responded to without any restrictions.
  - This scenario is similar to the band select disabled.

Note The client RSSI value (as seen in the **sh cont d0** | **begin RSSI** command output) is the average of the client packets received, and the Mid RSSI feature is the instantaneous RSSI value of the probe packets. As a result, the client RSSI is seen as weaker than the configured Mid RSSI value (7-dB delta). The 802.11b probes from the client are suppressed to push the client to associate with the 802.11a band.

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.

This section contains the following subsections:

802.11n Parameters

This section provides instructions for managing 802.11n 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 all the 802.11n access points for the WLANs using WMM with no Layer 2 encryption or with WPA2/AES encryption enabled.

Note To disable MCS rates for 802.11n, 802.11ac and 802.11ax, ensure that at least one MCS rate is enabled. To disable 802.11n on the controller to force APs to use only legacy 802.11a/b/g rates, first disable 802.11ax and 802.11ac on the controller for a particular band. Irrespective of the APs mapped to a Custom-RF-Profile, disabling 802.11n globally on the controller applies to all the APs.

802.11h Parameters

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

Restrictions for Band Selection, 802.11 Bands, and Parameters

- Band selection-enabled WLANs do not support time-sensitive applications such as voice and video because of roaming delays.

- Band selection is supported only on Cisco Wave 2 and 802.11ax APs.

  For more information about support on specific APs, see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html.

- Band selection operates only on APs that are connected to a controller. A FlexConnect AP without a controller connection does not perform band selection after a reboot.
- The band-selection algorithm directs dual-band clients only from the 2.4-GHz radio to the 5-GHz radio of the same AP, and it only runs on an AP when both the 2.4-GHz and 5-GHz radios are up and running.
- 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.

How to Configure 802.11 Bands and Parameters

Configuring Band Selection (GUI)

Before you begin

Ensure that you have configured an AP Join Profile prior to configuring the primary and backup controllers.

Procedure

Step 1 Choose **Configuration** > **Wireless Advanced** > **Band Select**.

Step 2 In the **Cycle Count** field, enter a value between 1 and 10. The cycle count sets the number of suppression cycles for a new client. The default cycle count is 2.

Step 3 In the **Cycle Threshold (milliseconds)** field, enter a value between 1 and 1000 milliseconds for the scan cycle period threshold. This setting determines the time threshold during which new probe requests from a client come from a new scanning cycle. The default cycle threshold is 200 milliseconds.

Step 4 In the **Age Out Suppression (seconds)** field, enter a value between 10 and 200 seconds. Age-out suppression sets the expiration time for pruning previously known 802.11b/g/n clients. The default value is 20 seconds. After this time elapses, clients become new and are subject to probe response suppression.

Step 5 In the **Age Out Dual Band (seconds)** field, enter a value between 10 and 300 seconds. The age-out period sets the expiration time for pruning previously known dual-band clients. The default value is 50 seconds. After this time elapses, clients become new and are subject to probe response suppression.

Step 6 In the Client RSSI (dbm) field, enter a value between -90 to -20. This is the average of the client packets received.

Step 7 In the Client Mid RSSI (dbm) field, enter a value between -90 to -20. This the instantaneous RSSI value of the probe packets.

Step 8 On the **AP Join Profile** page, click the AP Join Profile name.

Step 9 Click **Apply**.

Configuring Band Selection (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless client band-select cycle-count** *cycle_count* **Example:** `Device(config)# wireless client band-select cycle-count 3`	Sets the probe cycle count for band select. Valid range is between 1 and 10.
Step 3	**wireless client band-select cycle-threshold** *milliseconds* **Example:** `Device(config)# wireless client band-select cycle-threshold 5000`	Sets the time threshold for a new scanning cycle period. Valid range is between 1 and 1000.
Step 4	**wireless client band-select expire suppression** *seconds* **Example:** `Device(config)# wireless client band-select expire suppression 100`	Sets the suppression expire to the band select. Valid range is between 10 and 200.
Step 5	**wireless client band-select expire dual-band** *seconds* **Example:** `Device(config)# wireless client band-select expire dual-band 100`	Sets the dual band expire. Valid range is between 10 and 300.
Step 6	**wireless client band-select client-rssi** *client_rssi* **Example:** `Device(config)# wireless client band-select client-rssi 40`	Sets the client RSSI threshold. Valid range is between 20 and 90.

Step 7	**wlan** *wlan_profile_name wlan_ID SSID_network_name* **band-select**
	Example:
	`Device(config)# wlan wlan1 25 ssid12`
	`Device(config-wlan)# band-select`
		Configures band selection on specific WLANs. Valid range is between 1 and 512. You can enter up to 32 alphanumeric characters for *SSID_network_name* parameter.

Configuring the 802.11 Bands (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **Network**.

Step 2 Click either **5 GHz Band** or **2.4 GHz Band**.

Step 3 Uncheck the **Network Status** check box to disable the network in order to be able to configure the network parameters.

Step 4 In the **Beacon Interval** field, enter the rate at which the SSID is broadcast by the APs, from 100 to 600 milliseconds. The default is 100 milliseconds.

Step 5 For 802.11b/g/n (2.4-GHz) radios, to enable short preamble on the radio, check the **Short Preamble** check box. A short preamble improves throughput performance.

Step 6 In the **Fragmentation Threshold (in bytes)** field, enter a value between 256 to 2346 bytes. Packets larger than the size you specify here will be fragmented.

Step 7 Check the **DTPC Support** check box to advertise the transmit power level of the radio in the beacons and the probe responses. 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. You cannot configure a power constraint value on your 802.11a/n/ac (5-GHz) radio network if the **DTPC Support** check box is checked.

Step 8 Click **Apply**.

Step 9 In the **CCX Location Measurement** section, check the **Mode** check box to globally enable CCX radio management for the network. This parameter causes the APs connected to this device to issue broadcast radio measurement requests to clients running CCX v2 or later releases.

Step 10 In the **Interval** field, enter a value to specify how often the APs must issue broadcast radio measurement requests.

Step 11 Click **Apply**.

Step 12 In the **Data Rates** section, choose a value to specify the rates at which data can be transmitted between the access point and the client:

- Mandatory: Clients must support this data rate in order to associate to an access point on the controller embedded wireless controller.
- Supported: Any associated clients that support this data rate may communicate with the access point using that rate.
- Disabled: The clients specify the data rates used for communication.

Step 13 Click **Apply**.

Step 14 Save the configuration.

Configuring the 802.11 Bands (CLI)

Follow the procedure given below to configure 802.11 bands and parameters:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz shutdown** **Example:** `Device(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:** `Device(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:** `Device(config)# ap dot11 5ghz beaconperiod 500`	Specifies the rate at which the SSID is broadcast by the corresponding 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:** `Device(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	**[no] ap dot11** {**5ghz** | **24ghz** } **dtpc**
	Example:
	`Device(config)# ap dot11 5ghz dtpc`
	`Device(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-level and power-level information from the access points and adjust their settings automatically. For example, a client device used primarily in Japan can rely on DTPC to adjust its channel and power settings automatically when it travels to Italy and joins a network there.
		The **no** form of the command disables the DTPC setting.

Step 7	**wireless client association limit** *number* **interval** *milliseconds*	Specifies the maximum allowed clients that can be configured.

	Command or Action	Purpose
	Example: `Device(config)# wireless client association limit 50 interval 1000`	You can configure the maximum number of association requests on a single access point slot at a given interval. The range of association limit that you can configure is from 1 to 100. The association request limit interval is measured between 100 to 10000 milliseconds.
Step 8	**ap dot11 {5ghz | 24ghz} rate** *rate* {**disable** | **mandatory** | **supported**} **Example:** `Device(config)# ap dot11 5ghz rate 36 mandatory`	Specifies the rate at which data can be transmitted between the controller embedded wireless controller and the client. • **disable**: 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 embedded wireless controller. • **supported**: Any associated clients that support this data rate can communicate with the access point using that rate. However, the clients are not required 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:** `Device(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:** `Device(config)# no ap dot11 24ghz shutdown`	Enables the 802.11b band. **Note** The default value is enabled.
Step 11	**ap dot11 24ghz dot11g** **Example:** `Device(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.

	Command or Action	Purpose
Step 12	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring a Band-Select RF Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Advanced**.

Step 2 In the **Band Select** tab, enter a value between 1 and 10 in the **Cycle Count** field. The cycle count sets the number of suppression cycles for a new client. The default cycle count is 2.

Step 3 In the **Cycle Threshold** field, enter a value between 1 and 1000 milliseconds for the scan cycle period threshold. This setting determines the time threshold during which new probe requests from a client come from a new scanning cycle. The default cycle threshold is 200 milliseconds.

Step 4 In the **Age Out Suppression** field, enter a value between 10 and 200 seconds. Age-out suppression sets the expiration time for pruning previously known 802.11b/g/n clients. The default value is 20 seconds. After this time elapses, clients become new and are subject to probe response suppression.

Step 5 In the **Age Out Dual Band** field, enter a value between 10 and 300 seconds. The age-out period sets the expiration time for pruning previously known dual-band clients. The default value is 50 seconds. After this time elapses, clients become new and are subject to probe response suppression.

Step 6 In the **Client RSSI** field, enter a value between -90 dBm and -20 dBm. This is the minimum RSSI for a client to respond to a probe.

Step 7 In the **Client Mid RSSI** field, enter a value between –20 dBm and –90 dBm. This parameter sets the mid-RSSI, whose value can be used for toggling 2.4 GHz probe suppression based on the RSSI value.

Step 8 Click **Apply**.

Configuring 802.11n Parameters (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **RF**.

Step 2 Click **Add** to view the **Add RF Profile** window.

Step 3 In the **802.11** tab, proceed as follows:

a) Choose the required operational rates.

b) Select the required **802.11n MCS Rates** by checking the corresponding check boxes.

Step 4 Click **Save & Apply to Device**.

Configuring 802.11n Parameters (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 {5ghz | 24ghz} dot11n** **Example:** `Device(config)# ap dot11 5ghz dot11n`	Enables 802.11n support on the network. The **no** form of this command disables the 802.11n support on the network.
Step 3	**ap dot11 {5ghz | 24ghz} dot11n mcs tx** *rtu* **Example:** `Device(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. *rtu*-The valid range is between 0 and 23. The **no** form of this command disables the MCS rates that are configured.

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

Step 5	**ap dot11 {5ghz | 24ghz} shutdown** **Example:** `Device(config)# ap dot11 5ghz shutdown`	Disables the network.
Step 6	**{ap | no ap} dot11 {5ghz | 24 ghz} dot11n a-mpdu tx priority {all | 0-7}** **Example:** `Device(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 list defines the priority levels (0-7) assigned per traffic type. • 0—Best effort

	Command or Action	Purpose
		• 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 the parameters to configure all 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, 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:** `Device(config)# no ap dot11 5ghz shutdown`	Re-enables the network.
Step 8	**ap dot11 {5ghz | 24ghz} dot11n guard-interval {any | long}** **Example:** `Device(config)# ap dot11 5ghz dot11n guard-interval long`	Configures the guard interval for the network.

	Command or Action	Purpose
Step 9	**ap dot11 {5ghz | 24ghz} dot11n rifs rx** **Example:** `Device(config)# ap dot11 5ghz dot11n rifs rx`	Configures the Reduced Interframe Space (RIFS) for the network.
Step 10	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring 802.11h Parameters (CLI)

Procedure

	Command or Action	Purpose
Step 1	**ap dot11 5ghz shutdown** **Example:** `Device(config)# ap dot11 5ghz shutdown`	Disables the 802.11 network.
Step 2	**{ap | no ap} dot11 5ghz channelswitch mode** *switch_mode* **Example:** `Device(config)# ap dot11 5ghz channelswitch mode 0`	Enables or disables the access point to announce when it is switching to a new channel. *switch_mode*--Enter 0 or 1 to specify whether transmissions are restricted until the actual channel switch (**0**) or are not restricted (**1**). The default value is disabled.
Step 3	**ap dot11 5ghz power-constraint** *value* **Example:** `Device(config)# ap dot11 5ghz power-constraint 200`	Configures the 802.11h power constraint value in dB. The valid range is from 0 to 255. The default value is 3.
Step 4	**no ap dot11 5ghz shutdown** **Example:** `Device(config)# no ap dot11 5ghz shutdown`	Re-enables the 802.11a network.

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

Verifying Configuration Settings Using Band Selection and 802.11 Bands Commands

The following commands can be used to verify band selection, 802.11 bands, and parameters on the embedded wireless controller.

Table 15: Monitoring Configuration Settings Using Band Selection and 802.11 Band Commands

Command	Purpose
show ap dot11 5ghz network	Displays 802.11a band network parameters, 802.11a operational rates, 802.11n MCS settings, and 802.11n status information.
show ap dot11 24ghz network	Displays 802.11b band 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 the 5-GHz Band

```
Device# 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 the 2.4-GHz Band

```
Device# 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

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

Example: Verifying the Band-Selection Settings

The following example displays a band-select configuration:

```
Device# 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
Client Mid 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:

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

This example shows how to set the suppression expiry time to the band select:

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

This example shows how to set the dual-band expiry time for the band select:

```
Device# configure terminal
Device(config)# wireless client band-select expire dual-band 100
Device(config)# end
```

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

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

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

```
Device# configure terminal
Device(config)# wlan wlan1 25 ssid12
Device(config-wlan)# band-select
Device(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:

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

Examples: 802.11n Configuration

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

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

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

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

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

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

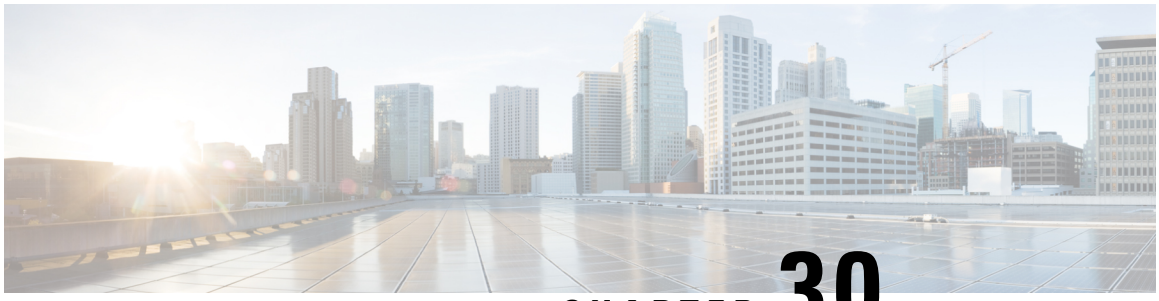
Examples: 802.11h Configuration

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

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

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

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



CHAPTER 30

Image Download

Information About Image Download

Software updates ensure that all the access points in the Cisco Embedded Wireless Controller network are running the latest software. The software update or image download can be performed using both the GUI and the CLI.

A typical Cisco Embedded Wireless Controller network contains the following components:

- Cisco Catalyst APs acting as controller (embedded wireless controller)
- Cisco Embedded Wireless Controller-capable APs (Other Cisco Catalyst series APs that participate in the Virtual Router Redundancy Protocol (VRRP)-based election process)
- Subordinate APs (Cisco Catalyst Series or Cisco Aironet Series Wave 2 APs)
- External TFTP and SFTP server.

Note For best user experience when using the GUI, view the browser at 100% resolution. The lines may break if the resolution is greater than 100%.

Updates to the AP Image Predownload Status (GUI)

From Cisco IOS XE Amsterdam, Release 17.3.1 onwards, during an access point (AP) image download, the Cisco Embedded Wireless Controller on Catalyst Access Points calculates the current percentage of the download and the estimated completion time of the download. (You can view these values in the CLI output by running the **show wireless ewc-ap ap image predownload status** command.)

To access the **Software Upgrage** window, from the Cisco Embedded Wireless Controller on Catalyst Access Points home page, choose **Administration** > **Software Management** > **Software Upgrade**.

The **Software Update Status** section in the GUI displays the update status bar that shows the progress of a software update, such as, **Initiate**, **Controller Image Download**, **AP Image Download**, **Network Upgrade**, **Activate**, and **Reload**.

To view the logs, click the **Show Install Logs** link.

The **Status** field displays the current status of the upgrade and indicates further action, if any, that you should perform.

The other details displayed in the window are - **Total Number of APs**, **Initiated**, **Predownloading AP Image**, **Predownloading Controller Image**, **Completed Predownloading AP Image**, **Completed Predownloading Controller Image**, **Failed to Predownload AP Image**, **Failed to Predownload Controller Image**.

The currently active AP, the AP on standby, and the preferred active AP are also displayed.

Image Download Scenarios

In a Cisco Embedded Wireless Controller network, image download from the embedded wireless controller to the subordinate AP takes place in the following scenarios:

- During AP join
- During network software upgrade (pre-download)

Image Download During AP Join

APs with older software trying to join the Cisco Embedded Wireless Controller network are automatically upgraded to match the latest software version on the embedded wireless controller. The embedded wireless controller compares the software version on the new AP with that on itself. If there is a mismatch, the AP requests the controller for a software upgrade and image download is triggered. The embedded wireless controller facilitates the transfer of the latest software from an external TFTP server or SFTP server, to the new AP.

Depending on the new AP joining the network, there are two image downloads that take place:

- **AP software image download:** This applies to all new APs joining the Cisco Embedded Wireless Controller.
- **Controller software image download:** This applies only to Cisco Catalyst series APs, capable of becoming a controller, trying to join the Cisco Embedded Wireless Controller network.

AP Software Image Download

Any Cisco Catalyst Series AP or Cisco Aironet Series Wave 2 AP can only join an embedded wireless controller if its AP software image version matches that of the controller.

During the AP join process, the embedded wireless controller first checks the AP software image version on the new AP and if it does not match what is on the controller, the latest AP software is downloaded from the controller to the new AP. Once the AP software image on the new AP is upgraded to match the version that is on the embedded wireless controller in the network, the new AP reloads. Once the new AP is back up with the upgraded AP software image, it joins the embedded wireless controller.

Controller Software Image Download

If the new AP joining the network is a CiscoCatalyst Series AP capable of becoming an embedded wireless controller, first the controller checks the AP software image on the new AP and if outdated, it is upgraded to

match the AP software version on the controller. The AP then reloads with the new AP software image and joins theembedded wireless controller in the network.

Next, the embedded wireless controller does a similar check to compare the controller software version on the embedded wireless controller-capable AP. Similar to the AP software upgrade, if there is a mismatch, the controller software on this CiscoCatalyst Series AP is also upgraded to the latest version on the embedded wireless controller. The AP reloads again, this time with the upgraded controller software image.

Efficient AP Join

If the Cisco Embedded Wireless Controllerr network contains an AP of the same image type as the newly joining AP, then the new AP downloads the AP software image from this AP. For example, if a CiscoCatalyst 9130AX Series AP is newly joining the Cisco Embedded Wireless Controller network and another CiscoCatalyst 9130AX Series AP already exists in the network, then the new AP gets its AP software image from the already joined AP.

This method, known as efficient AP join, enables homogenous APs to get the software locally (within the Cisco Embedded Wireless Controller network) rather than downloading it from an external server. This improves software download efficiency.

The first AP of a series that joins the network and downloads the software from the embedded wireless controller is called a primary image. The other APs of the same series are known as image subordinates.

Network Software Upgrade (Pre-Download)

In the pre-download scenario, image download in the Cisco Embedded Wireless Controller network occurs to upgrade the software on all the APs from one software version to another. However, these APs continue to serve existing as well as new clients and there is no network disruption.

For pre-download, all the APs should be connected to the embedded wireless controller in a stable join state. Once image download is initiated during pre-download, new APs are not allowed to join the embedded wireless controller.

Efficient AP Upgrade

In this method, the first AP of an AP series to get the image from the embedded wireless controller becomes the primary image. The remaining APs of the same AP series, the image subordinates, then download the software image locally from this primary image. This method is also known as efficient AP upgrade.

Methods Supported for Image Download

In a Cisco Embedded Wireless Controller network, there are four ways in which the software image can be downloaded from the embedded wireless controller. These methods are based on the location from where the controller transfers the software image to the subordinate AP:

- From an external TFTP server
- From an external SFTP server
- From the desktop (via HTTP)

TFTP Image Download Method

In the TFTP method, the AP and controller software images are stored on a TFTP server. To download the software images from the TFTP server, you need to specify the IP address of the TFTP server and the path to the software image bundle on the TFTP server.

The TFTP image download method can be triggered using both the GUI and CLI.

SFTP Image Download Method

In the SFTP method, the AP and controller software images are stored on an SFTP server. To download the software images from the SFTP server, in addition to the IP address of the SFTP server and the software image bundle path, you need to specify the SFTP server credentials.

The SFTP image download method also can be triggered using both the GUI and CLI.

Desktop (HTTP) Image Download Method

Image download through desktop (HTTP) is applicable only in the network software upgrade (pre-download) scenario.

For the desktop (HTTP) method, download the software image bundle for the Cisco Embedded Wireless Controller to your computer or laptop desktop. This downloaded bundle contains the AP and controller software images which need to be extracted to the computer or laptop desktop before they can be uploaded to the embedded wireless controller.

Note that the desktop (HTTP) method works only for a homogenous network. A homogenous Cisco Embedded Wireless Controller network is one which contains APs that have the same AP software image type. For example, the Cisco Catalyst 9115AX series AP and the Cisco Catalyst 9120AX series AP use the ap1g7 AP software image file. So, the Cisco Embedded Wireless Controller network in this example containing Cisco Catalyst 9115AX series and 9120AX series APs is a homogenous network.

The embedded wireless controller CLI can only be used to set the mode for image download as desktop (HTTP). The Cisco Embedded Wireless Controller GUI has to be used to configure and trigger network software upgrade (pre-download) using the desktop (HTTP) image download method.

Prerequisites for Image Download

- Connectivity to an external (TFTP or SFTP) server is required for image download during AP join in a Cisco Embedded Wireless Controller network.
- Connectivity to a PC or laptop is required for image download during network software upgrade in a Cisco Embedded Wireless Controller network.
- All APs should be connected to the embedded wireless controller for image download in the network software upgrade (pre-download) scenario.
- For image upgrade, you must not configure a preferred-master. If you configure a preferred-master, ensure that it points to the currently active AP, which is displayed in the **show wireless ewc-ap redundancy summary** command.

  If a different AP is configured as the preferred-master, the upgrade process will not take place in the **install activate** step. If the upgrade does not take place, you should either remove the preferred-master

configuration, or re-configure the preferred-master to match the AP that is currently active, and then run the **install activate** command, again.

•

Configuring Image Download Profile

You need to configure the image download profile for both the AP join image download and pre-download scenarios. The only profile supported is *default*. In a Cisco Embedded Wireless Controller network, only one site tag is supported, the *default-site-tag*. The *default* image download profile is attached to the *default-site-tag*.

Note When an AP of a different type tries to join a homogenous network that had earlier used the HTTP mode for image upgrade, the AP join fails. To avoid this failure, you must update the **image-download-mode** to **tftp** in the **wireless profile image-download default** configuration step.

Configuring TFTP Image Download (GUI)

Procedure

Step 1 Choose **Administration** > **Software Management**.

Step 2 On the **Software Management** page, under the **Software Upgrade** tab, select the **Mode** as TFTP.

Step 3 In the **Image Server** field, enter the TFTP server IP address.

Step 4 In the **Image Path** field, enter the absolute or relative path to the software image bundle.

Step 5 Choose one of the following:

- **Save:** Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
- **Save & Download:** Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
- **Activate:** Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
- **Cancel:** Choose this option to cancel any changes made to the image download profile.

Option	Description
Save	Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
Save & Download	Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.

Option	Description
Activate	Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
Cancel	Choose this option to cancel any changes made to the image download profile.

Configuring TFTP Image Download (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile image-download default** **Example:** `Device (config)# wireless profile image-download default`	Configures the default AP profile.
Step 3	**image-download-mode tftp** **Example:** `Device(config-wireless-image-download-profile)# image-download-mode tftp`	Configure image download using TFTP.
Step 4	**tftp-image-server** *server-ip* **Example:** `Device(config-wireless-image-download-profile-tftp)# tftp-image-server 10.1.1.1`	Configure the TFTP server for image download by specifying the IPv4 or IPv6 *server-ip* address.
Step 5	**tftp-image-path** *server-path* **Example:** `Device(config-wireless-image-download-profile-tftp)# tftp-image-path /download/object/stream/images/ap-images`	Configure the absolute or relative path to the software image on the TFTP server.
Step 6	**end** **Example:** `Device(config-wireless-image-download-profile-tftp)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring SFTP Image Download (GUI)

Procedure

Step 1 Choose **Administration** > **Software Management**.

Step 2 On the **Software Management** page, under the **Software Upgrade** tab, select the **Mode** as SFTP.

The SFTP port is not configurable and is fixed at 22.

Step 3 In the **Image Server** field, enter the SFTP server IP address.

Step 4 In the **Image Path** field, enter the path to the software image bundle.

Step 5 In the **User Name** field, enter the SFTP server username.

Step 6 Choose the appropriate **Password Type** from Unencrypted or AES Encrypted.

Step 7 In the **Password** field, enter the SFTP server password.

Step 8 Choose one of the following:

Option	Description
Save	Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
Save & Download	Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
Activate	Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
Cancel	Choose this option to cancel any changes made to the image download profile.

Configuring SFTP Image Download (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile image-download default** **Example:** `Device (config)# wireless profile image-download default`	Configures the default AP profile.

	Command or Action	Purpose
Step 3	**image-download-mode sftp** **Example:** `Device(config-wireless-image-download-profile)# image-download-mode sftp`	Configure image download using SFTP.
Step 4	**sftp-image-server** *server-ip* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-image-server 10.1.1.1`	Configure the SFTP server for image download by specifying the IPv4 or IPv6 *server-ip* address.
Step 5	**sftp-image-path** *server-path* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-image-path /download/object/stream/images/ap-images`	Configure the path to the software image on the SFTP server.
Step 6	**sftp-username** *username* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-username test`	Specify the username to log in to the SFTP server for image download.
Step 7	**sftp-password** {**0** | **8**} *password* **Example:** `Device(config-wireless-image-download-profile-sftp)# sftp-password 0 password1`	Specify the password associated with the above username to download the image from the SFTP server. You need to re-enter the password to confirm the entry. To configure an AES encrypted password, specify `8`, else specify `0` to configure an unencrypted password.
Step 8	**end** **Example:** `Device(config-wireless-image-download-profile-tftp)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring CCO Mode for Software Upgrade (GUI)

Before you begin

The CCO account must have a physical address entered at the CCO Profile Manager. The account must have EULA and K9 acknowledged. For more information about creating a CCO account, refer to https://www.cisco.com/c/en/us/about/help/registration-benefits-help.html.

Procedure

Step 1 Choose **Administration** > **Software Management**.

Step 2 On the **Software Management** page, under the **Software Upgrade** tab, select the **Mode** as CCO.

Step 3 In the **User Name** field, enter the CCO username.

Step 4 In the **Password** field, enter the password to access the CCO server.

Step 5 Choose the appropriate **Password Type** from Unencrypted or AES Encrypted.

Step 6 Choose either Enabled or Disabled from the **Automatically Check for Updates** field. If you enable this option, the system automatically checks for software updates.

The interval is for 30 days. After the interval expires, the controller automatically checks and updates for the latest or recommend software version information in the controller configuration.

Step 7 In the **Software Check** field, click the **Check now** button and retrieve up-to-date information about the **Latest software release** (the latest version available on the CCO website) and the **Recommended software release** (the recommended software version for the currently running software) version numbers.

Step 8 The **Last CCO Response** field displays the error messages encountered when configuring the CCO image download method. For example, if you have entered a wrong username and password, the following error message is displayed: `HTTP 400 Error: 400 Client Error: Bad Request for url: https://cloudsso.cisco.com/as/token.oauth2 Please check your username/password and try again.` For more information about the **Last CCO Response** error messages, refer to Troubleshooting - CCO Image Download Error Messages, on page 340.

Step 9 From the **Version** drop-down list, choose either **Recommended** or **Latest**. After fetching the latest and the recommended software versions, you can choose the version to upgrade.

Step 10 Choose one of the following:

Option	Description
Save	Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
Save & Download	Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
Activate	Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
Cancel	Choose this option to cancel any changes made to the image download profile.

Configuring CCO Image Download (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile image-download default** **Example:** `Device (config)# wireless profile image-download default`	Configures the default AP profile.
Step 3	**image-download-mode cco** **Example:** `Device(config-wireless-image-download-profile)# image-download-mode cco`	Configure image download using CCO.
Step 4	**cco-username** *username* **Example:** `Device(config-wireless-image-download-profile-cco)# cco-username username`	Specify the username to log in to the CCO server for image download.
Step 5	**cco-password** {**0** | **8**} *password* **Example:** `Device(config-wireless-image-download-profile-cco)# cco-password 0 password1`	Specify the password associated with the above username to download the image from the CCO server. You need to re-enter the password to confirm the entry. To configure an AES encrypted password, specify `8`, else specify `0` to configure an unencrypted password.
Step 6	**cco-version** {**latest** | **suggested**} **Example:** `Device(config-wireless-image-download-profile-cco)# cco-version latest`	Specify the **latest** or the **suggested** version to be downloaded from the CCO server. By default the **suggested** version is downloaded.
Step 7	**cco-auto-check** **Example:** `Device(config-wireless-image-download-profile-cco)# cco-auto-check`	Enables or disables automatic check of new software versions at CCO every 30 days. This is applicable to Image Upgrade or Predownload only. By default, **cco-auto-check** is enabled. To disable the command use the **no** form of the command.

	Command or Action	Purpose
Step 8	**end** **Example:** `Device(config-wireless-image-download-profile-cco)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 9	**wireless ewc-ap predownload poll-cco** **Example:** `Device# wireless ewc-ap predownload poll-cco`	Polls the CCO server to check for the latest software version.
Step 10	**clear ap predownload statistics** **Example:** `Device# clear ap predownload statistics`	Clears the AP predownload statistics.
Step 11	**install remove profile default** **Example:** `Device# install remove profile default`	Removes the image download profile. Choose **Y** to remove the profile or choose **N** to cancel.
Step 12	**install add profile default** **Example:** `Device# clear ap predownload statistics`	Downloads the controller and AP software image from the embedded wireless controller. The controller image is sent to all Cisco Embedded Wireless Controller-capable APs. The AP image is downloaded to all APs sharing the same image type
Step 13	**install activate** **Example:** `Device# install activate`	Activates the network after upgrade. All the subordinate APs get the new AP image and reboot. Once all APs are rebooted, the embedded wireless controller also reboots. **Note** The network can also be activated if the controller image is downloaded but all APs have not received the AP image via predownload. **Important** If the network is activated during partial predownload success, and a Cisco Embedded Wireless Controller-capable AP with old controller software becomes the controller, then the network will not get upgraded to the new image.

	Command or Action	Purpose
Step 14	**install commit** **Example:** `Device# install commit`	Commits the current software image once the embedded wireless controller comes up after rebooting. **Note** While upgrading, you must not use the **add**, **active**, **commit** keywords in a single command, as the activation process fails.

Troubleshooting - CCO Image Download Error Messages

Following are the expected error messages and the causes, which will be displayed at the **Last CCO Response** field:

DNS resolution or connectivity issue

Connection Error: HTTPSConnectionPool(host='cloudsso.cisco.com', port=443): Max retries exceeded with url: /as/token.oauth2 (Caused by NewConnectionError('<urllib3.connection.VerifiedHTTPSConnection object at 0xf6170250>: Failed to establish a new connection: [Errno -3] Temporary failure in name resolution',))

CCO username/password error

HTTP 400 Error: 400 Client Error: Bad Request for url: https://cloudsso.cisco.com/as/token.oauth2 Please check your username/password and try again

Address missing exception

Thank you for registering with Cisco.com. In order to consume software or services we require your full address. Please follow this link to return to profile manager to complete your profile.

EULA form missing exception

Eula form have not been accepted or rejected to continue download. Please go tohttps://software.cisco.com/download/eula.

K9 form missing exception

K9 form have not been accepted or rejected to continue download. Please go to https://software.cisco.com/download/k9

Configuring Desktop (HTTP) Image Download (GUI)

- Image download using desktop (HTTP) is only enabled in a homogeneous network, that is a network containing APs that have the same image type.
- Image download using desktop (HTTP) can only be configured from the GUI.
- The CLI can only be used to set the image download mode to desktop (HTTP).

Procedure

Step 1 Choose **Administration** > **Software Management**.

Step 2 On the **Software Management** page, under the **Software Upgrade** tab, select the **Mode** as Desktop (HTTP).

Step 3 In the **Controller Image** field, navigate to the embedded wireless controller software image on your computer or laptop desktop.

Step 4 In the **AP Image** field, navigate to the AP software image on your computer or laptop desktop.

The GUI displays the name of the AP image to be used. Depending on the AP model, the name of the AP image varies.

Step 5 Choose one of the following:

Option	Description
Save	Choose this option to save the image download profile and enable image download for new APs joining the Cisco Embedded Wireless Controller network.
Save & Download	Choose this option to save the configuration and enable network software upgrade (pre-download). The image download profile is saved (even if no change is made to the configuration) and the latest image is downloaded in the background. This allows the APs to continue serving the clients.
Activate	Choose this option to enable the APs in the network to swap to the latest image and reboot. The Cisco Embedded Wireless Controller network is activated once the APs come up with the new image file.
Cancel	Choose this option to cancel any changes made to the image download profile.

Initiating Pre-Download (CLI)

Procedure

	Command or Action	Purpose
Step 1	**wireless ewc-ap predownload poll-cco**	Check the latest and recommended version for image upgrade.
Step 2	**clear ap predownload statistics**	Clear AP predownload statistics.
Step 3	**install remove profile default**	Remove the image download profile. Choose **Y** to remove the profile or choose **N** to cancel.
Step 4	**install add profile default**	Download the controller and AP software image from the embedded wireless controller. The controller image is sent to all Cisco Embedded Wireless Controller-capable APs.

	Command or Action	Purpose
		The AP image is downloaded to all APs sharing the same image type.
Step 5	**show wireless ewc-ap predownload status**	Monitor the overall software download status. The download is successful when the status message is `Controller Image Predownload to EWC Capable APs Complete`.
Step 6	**install activate**	Activate the network after upgrade. All the subordinate APs get the new AP image and reboot. Once all APs are rebooted, the embedded wireless controller also reboots. **Note** The network can also be activated if the controller image is downloaded but all APs have not received the AP image via predownload. **Important** If the network is activated during partial predownload success, and a Cisco Embedded Wireless Controller-capable AP with old controller software becomes the controller, then the network will not get upgraded to the new image.
Step 7	**show install summary**	Verify the current image status after rebooting. If the status is `Activated and Uncommitted`, proceed to Step 7, else wait.
Step 8	**install commit**	Commits the current software image once the embedded wireless controller comes up after rebooting. **Note** While upgrading, you must not use the **add**, **active**, **commit** keywords in a single command, as the activation process fails.

During the image upgrade process, the image predownload status is shown in various stages such as Controller Image Download In Progress, AP Image Predownload in Progress, Controller Image Predownload to EWC Capable APs In Progress, and so on. Sometimes, the image upgrade might fail due to various reasons. In such a case, you can either continue with the **install activate** operation or cancel it, based on the output of the **show wireless ewc-ap ap image predownload status** command, which displays the individual predownload status for each AP.

Verifying Image Download

To monitor the overall progress of the software download process during predownload, run the following command.

```
Device# show wireless ewc-ap predownload status
```

The following are the various status messages indicating the status of the predownload operation. These are displayed when you run the **show wireless ewc-ap predownload status** command:

- None
- Controller Image Download Initiated
- Controller Image Download In Progress
- Controller Image Download Complete
- Controller Image Download Failed
- AP Image Predownload Initiated
- AP Image Predownload In Progress
- AP Image Predownload Complete
- AP Image Predownload Unsupported
- AP Image Predownload Failed
- Controller Image Predownload to EWC Capable APs In Progress
- Controller Image Predownload to EWC Capable APs Complete
- Controller Image Predownload to EWC Capable APs Failed
- Image Activation Succeeded
- Image Activation Failed
- Invalid State

To view the AP image predownload statistics, run the following command:

```
Device# show wireless ewc-ap ap image predownload status
Total number of APs                          : 5
Total number of EWC capable APs              : 4
Number of APs
        Initiated                            : 0
        Predownloading AP image              : 0
        Predownloading Controller image      : 1
        Completed predownloading AP          : 5
        Completed predownloading Controller  : 0
        Failed to Predownload AP             : 0
        Failed to Predownload Controller     : 0
AP Name               Primary Image (AP/Controller)          Backup Image (AP/Controller)
          Predownload Status         Predownload Version                    AP Image
Role    Retries  AP image       Controller image

                                                                            Type
```

```
                ETA/Percent    ETA/Percent
-----------------------------------------------------------------------------------------------------------------
APXXXX.9XXX.8FXX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /17.2.02.0.XXXX
         Complete                   17.2.2.2       /17.2.02.0.2XXX          ap1g7       Slave
   0        00:00:00/100%  00:00:00/  0%
APXXXX.5XXX.71XX       17.3.0.85      /                       17.2.2.2       /
          Complete                    17.2.2.2       /                        ap1g5
Master  0        00:00:00/100%  00:00:00/  0%
APXXXX.8XXX.59XX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /17.2.02.0.XXXX
         Complete                   17.2.2.2       /                        ap1g7       Slave
   0        00:00:00/100%  00:00:00/  0%
APXXXX.8XXX.5AXX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /17.3.01.0.XXX
          Controller Predownloading  17.2.2.2       /                        ap1g7
Master  0        00:00:00/100%  00:00:00/  0%
APXXXX.8XXX.5BXX       17.3.0.85      /17.3.01.0.XXXX         17.2.2.2       /
          Complete                    17.2.2.2       /                        ap1g7
Slave   0        00:00:00/100%  00:00:00/  0%
```

To view details of the AP acting as the primary image , use the following command:

```
Device# show wireless ewc-ap image-master
Image Master List
Image Name: ap1g7
-----------------------------------------------------------------------------------------------------------------
Master AP MAC                AP                          AP                 Controller
         Controller
                Predownload In Progress   Predownload Complete     Predownload In Progress
  Predownload Complete
-----------------------------------------------------------------------------------------------------------------
c0XX.eXXX.90XX    No                          No                       No
   Yes
Image Name: ap1g5
-----------------------------------------------------------------------------------------------------------------
Master AP MAC                AP                          AP                 Controller
         Controller
                Predownload In Progress   Predownload Complete     Predownload In Progress
  Predownload Complete
-----------------------------------------------------------------------------------------------------------------
70XX.1XXX.4bXX    No                          No                       No
   Yes
```

To check the image download status on all the APs, run the following command:

```
Device# show ap image
```

To check AP status during image download, run the following command:

```
Device# show ap summary
```

To monitor efficient AP join status, run the following command:

```
Device# show ap master list
```

To view the details of the last AP image download attempt, run the following command:

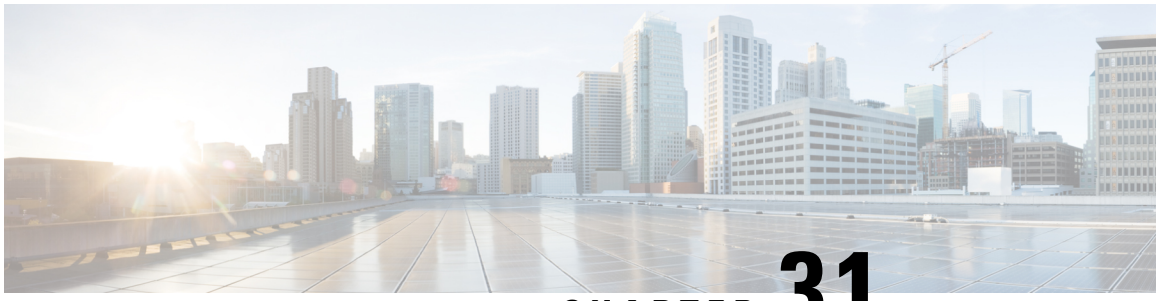
```
Device# show wireless stats ap image-download
```

To check the current status of the upgraded image, run the following command:

```
Device# show install summary
```

To check the download status from external servers (TFTP or SFTP), run the following command:

```
Device# show install log
```



CHAPTER 31

Conditional Debug and Radioactive Tracing

Introduction to Conditional Debugging

The Conditional Debugging feature allows you to selectively enable debugging and logging for specific features based on the set of conditions you define. This feature is useful in systems where a large number of features are supported.

The Conditional debug allows granular debugging in a network that is operating at a large scale with a large number of features. It allows you to observe detailed debugs for granular instances within the system. This is very useful when we need to debug only a particular session among thousands of sessions. It is also possible to specify multiple conditions.

A condition refers to a feature or identity, where identity could be an interface, IP Address, or a MAC address and so on.

This is in contrast to the general debug command, that produces its output without discriminating on the feature objects that are being processed. General debug command consumes a lot of system resources and impacts the system performance.

Introduction to Radioactive Tracing

Radioactive tracing (RA) provides the ability to stitch together a chain of execution for operations of interest across the system, at an increased verbosity level. This provides a way to conditionally print debug information (up to DEBUG Level or a specified level) across threads, processes and function calls.

Note

- The radioactive tracing supports First-Hop Security (FHS).
- The radioactive tracing filter does not work, if the certificate is not valid.
- For effective debugging of issues on mesh features, ensure that you add both Ethernet and Radio MAC address as conditional MAC for RA tracing, while collecting logs.
- To enable debug for wireless IPs, use the **debug platform condition feature wireless ip** *ip-address* command.

Conditional Debugging and Radioactive Tracing

Radioactive Tracing when coupled with Conditional Debugging, enable us to have a single debug CLI to debug all execution contexts related to the condition. This can be done without being aware of the various control flow processes of the feature within the box and without having to issue debugs at these processes individually.

Note Use the **clear platform condition all** command to remove the debug conditions applied to the platform.

Location of Tracefiles

By default the tracefile logs will be generated for each process and saved into either the **/tmp/rp/trace** or **/tmp/fp/trace** directory. In this temp directory, the trace logs are written to files, which are of 1 MB size each. You can verify these logs (per-process) using the **show platform software trace message** *process_name* **chassis active R0** command. The directory can hold up to a maximum of 25 such files for a given process. When a tracefile in the **/tmp** directory reaches its 1MB limit or whatever size was configured for it during the boot time, it is rotated out to an archive location in the **/crashinfo** partition under **tracelogs** directory.

The **/tmp** directory holds only a single tracefile for a given process. Once the file reaches its file size limit it is rotated out to **/crashinfo/tracelogs**. In the archive directory, up to 25 files are accumulated, after which the oldest one is replaced by the newly rotated file from **/tmp**. File size is process dependent and some processes uses larger file sizes (upto 10MB). Similarly, the number of files in the **tracelogs** directory is also decided by the process. For example, WNCD process uses a limit of 400 files per instance, depending on the platform.

The tracefiles in the crashinfo directory are located in the following formats:

1. Process-name_Process-ID_running-counter.timestamp.gz

   Example: IOSRP_R0-0.bin_0.14239.20151101234827.gz

2. Process-name_pmanlog_Process-ID_running-counter.timestamp.bin.gz

   Example: wncmgrd_R0-0.27958_1.20180902081532.bin.gz

Configuring Conditional Debugging (GUI)

Procedure

Step 1 Choose **Troubleshooting** > **Radioactive Trace**.

Step 2 Click **Add**.

Step 3 Enter the **MAC/IP Address**.

Step 4 Click **Apply to Device**.

Step 5 Click **Start** to start or **Stop** to stop the conditional debug.

Step 6 Click **Generate** to create a radioactive trace log.

Step 7 Click the radio button to set the time interval.

Step 8 Click the **Download Logs** icon that is displayed next to the trace file name, to download the logs to your local folder.

Step 9 Click the **View Logs** icon that is displayed next to the trace file name, to view the log files on the GUI page. Click **Load More** to view more lines of the log file.

Step 10 Click **Apply to Device**.

Configuring Conditional Debugging

Follow the procedure given below to configure conditional debugging:

Procedure

	Command or Action	Purpose
Step 1	**debug platform condition feature wireless mac** {*mac-address*} **Example:** `Device# debug platform condition feature wireless mac b838.61a1.5433`	Configures conditional debugging for a feature using the specified MAC address. **Note** This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.
Step 2	**debug platform condition start** **Example:** `Device# debug platform condition start`	Starts conditional debugging (this will start radioactive tracing if there is a match on one of the conditions above). **Note** This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.
Step 3	**show platform condition** OR **show debug** **Example:**	Displays the current conditions set.

	Command or Action	Purpose
	`Device# show platform condition` `Device# show debug`	
Step 4	**debug platform condition stop** **Example:** `Device# debug platform condition stop`	Stops conditional debugging (this will stop radioactive tracing). **Note** This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.
Step 5	**show logging profile wireless** [**counter** | **[last]**{*x days/hours*} | **filter mac**{<*mac address*>} **[to-file]**{<*destination*>} **Example:** `Device# show logging profile wireless start last 20 minutes to-file bootflash:logs.txt`	Displays the logs from the latest wireless profile. **Note** You can use either the *show logging profile wireless* command or *show logging process* command to collect the logs.
Step 6	**show logging process** <*process name*> **Example:** `Device# show logging process wncd to-file flash:wncd.txt`	Displays the logs collection specific to the process.
Step 7	**clear platform condition all** **Example:** `Device# clear platform condition all`	Clears all conditions.

What to do next

Note The command **request platform software trace filter-binary wireless** {*mac-address*} generates 3 flash files:

- *collated_log_<.date..>*
- *mac_log <..date..>*
- *mac_database .. file*

Of these, *mac_log <..date..>* is the most important file, as it gives the messages for the MAC address we are debugging. The command **show platform software trace filter-binary** also generates the same flash files, and also prints the mac_log on the screen.

Recommended Workflow for Trace files

1. To request the tracelogs for a specific time period.

EXAMPLE 1 day.

Use the command:

Device#**show logging process wncd to-file flash:wncd.txt**

2. The system generates a text file of the tracelogs in the location /flash:
3. Copy the file off the device. By copying the file, the tracelogs can be used to work offline. For more details on copying files, see section below.
4. Delete the tracelog file (.txt) file from /flash: location. This will ensure enough space on the device for other operations.

Copying Tracefiles Off the Box

An example of the tracefile is shown below:

```
Device# dir flash:/tracelogs
Directory of crashinfo:/tracelogs/

50664 -rwx 760 Sep 22 2015 11:12:21 +00:00 plogd_F0-0.bin_0.gz
50603 -rwx 991 Sep 22 2015 11:12:08 +00:00 fed_pmanlog_F0-0.bin_0.9558.20150922111208.gz
50610 -rw- 11 Nov 2 2015 00:15:59 +00:00 timestamp
50611 -rwx 1443 Sep 22 2015 11:11:31 +00:00
auto_upgrade_client_sh_pmanlog_R0-.bin_0.3817.20150922111130.gz
50669 -rwx 589 Sep 30 2015 03:59:04 +00:00 cfgwr-8021_R0-0.bin_0.gz
50612 -rwx 1136 Sep 22 2015 11:11:46 +00:00 reflector_803_R0-0.bin_0.1312.20150922111116.gz
50794 -rwx 4239 Nov 2 2015 00:04:32 +00:00 IOSRP_R0-0.bin_0.14239.20151101234827.gz
50615 -rwx 131072 Nov 2 2015 00:19:59 +00:00 linux_iosd_image_pmanlog_R0-0.bin_0
```

The trace files can be copied using one of the various options shown below:

```
Device# copy flash:/tracelogs ?
  crashinfo: Copy to crashinfo: file system
  flash: Copy to flash: file system
  ftp: Copy to ftp: file system
  http: Copy to http: file system
  https: Copy to https: file system
  null: Copy to null: file system
  nvram: Copy to nvram: file system
  rcp: Copy to rcp: file system
  running-config Update (merge with) current system configuration
  scp: Copy to scp: file system
  startup-config Copy to startup configuration
  syslog: Copy to syslog: file system
  system: Copy to system: file system
  tftp: Copy to tftp: file system
  tmpsys: Copy to tmpsys: file system
```

The general syntax for copying onto a TFTP server is as follows:

```
Device# copy source: tftp:
Device# copy crashinfo:/tracelogs/IOSRP_R0-0.bin_0.14239.20151101234827.gz tftp:
Address or name of remote host []? 2.2.2.2
```

```
Destination filename [IOSRP_R0-0.bin_0.14239.20151101234827.gz]?
```

Note It is important to clear the generated report or archive files off the switch in order to have flash space available for tracelog and other purposes.

Configuration Examples for Conditional Debugging

The following is an output example of the *show platform condition* command.

```
Device# show platform condition
Conditional Debug Global State: Stop
Conditions Direction
------------------------------------------------------------------------------------------|---------
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
----------------------|----------------------|--------------------------------
Device#
```

The following is an output example of the *show debug* command.

```
Device# show debug
IOSXE Conditional Debug Configs:
Conditional Debug Global State: Start
Conditions Direction
------------------------------------------------------------------------------------------|---------
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
----------------------|----------------------|--------------------------------
Packet Infra debugs:
Ip Address Port
------------------------------------------------------|----------
Device#
```

Verifying Conditional Debugging

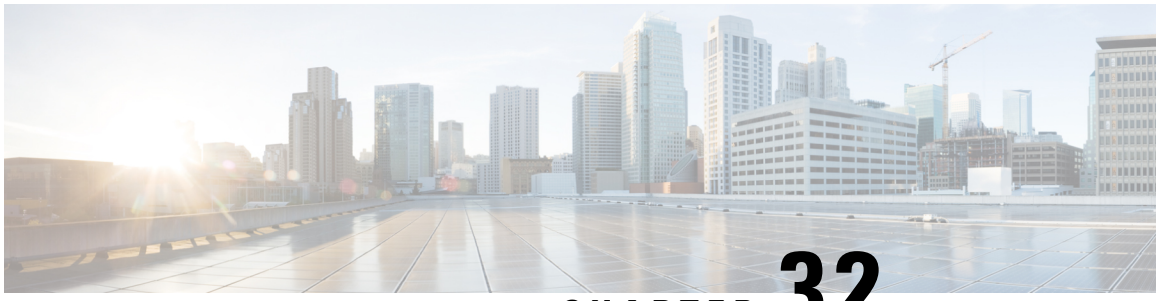
The table shown below lists the various commands that can be used to verify conditional debugging:

Command	Purpose
show platform condition	Displays the current conditions set.
show debug	Displays the current debug conditions set.
show platform software trace filter-binary	Displays logs merged from the latest tracefile.
request platform software trace filter-binary	Displays historical logs of merged tracefiles on the system.

Example: Verifying Radioactive Tracing Log for SISF

The following is an output example of the *show platform software trace message ios chassis active R0 | inc sisf* command.

```
Device# show platform software trace message ios chassis active R0 | inc sisf

2017/10/26 13:46:22.104 {IOSRP_R0-0}{1}: [parser]: [5437]:  UUID: 0, ra: 0 (note):  CMD:
'show platform software trace message ios switch active R0 | inc sisf' 13:46:22 UTC Thu Oct
 26 2017
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
   FF8E802918 semaphore system unlocked
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Unlocking, count is now 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
   FF8E802918 semaphore system unlocked
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Unlocking, count is now 1
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Setting State to 2
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Start timer 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc  Timer value/granularity for 0 :299998/1000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Updated Mac Timer : 299998
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Before Timer :  350000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Timer 0, default value is 350000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
   Allocating timer wheel for 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc No timer running
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Granularity for timer MAC_T1  is 1000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060, ra: 7 (debug):
  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Current State :MAC-STALE, Req Timer : MAC_T1  Current Timer
 MAC_T1
```



CHAPTER 32

Aggressive Client Load Balancing

Information About Aggressive Client Load Balancing

The Aggressive Client Load Balancing feature allows lightweight access points to load balance wireless clients across access points.

When a wireless client attempts to associate to a lightweight access point, the associated response packets are sent to a client with an 802.11 response packet including status code 17. This code 17 indicates that the corresponding AP is busy. The AP does not respond with the response 'success' if the AP threshold is not met, and with code 17 (AP busy) if the AP utilization threshold is exceeded, and another less busy AP hears the client request.

For example, if the number of clients on AP1 is more than the number of clients on AP2 and the load-balancing window, then AP1 is considered to be busier than AP2. When a client attempts to associate to AP1, the client 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 embedded wireless controller to deny client associations up to 10 times (if a client attempts to associate 11 times, it will 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.

Note A voice client does not authenticate when delay is configured to more than 300 ms. To avoid this, configure a central-authentication, local-switching WLAN with Cisco Centralized Key Management (CCKM), configure a pagent router between an AP and WLC with a delay of 600 ms (300 ms UP and 300 ms DOWN), and try associating the voice client.

Note For a FlexConnect AP, the association is locally handled. The load-balancing decisions are taken at the controller. A FlexConnect AP sends an initial response to the client before knowing the result of the calculations in the controller. Load-balancing does not take effect when the FlexConnect AP is in standalone mode.

A FlexConnect AP does not send (re)association response with status 17 for load balancing the way local-mode APs do; instead, it first sends (re)association with status 0 (success) and then deauth with reason 5.

Enabling Aggressive Client Load Balancing (GUI)

Procedure

Step 1 Choose **Configuration > Wireless > WLANs > Wireless Networks**.

Step 2 Select a **WLAN** to view the **Edit WLAN** window.

Step 3 Click **Advanced** tab.

Step 4 Select the **Load Balance** check box to enable the feature.

Step 5 Click **Update & Apply to Device**.

Configuring Aggressive Client Load Balancing (GUI)

Procedure

Step 1 Choose **Configuration > Wireless > Advanced**.

The **Load Balancing** window is displayed.

Step 2 In the **Aggressive Load Balancing Window (clients)** field, enter the number of clients for the aggressive load balancing client window.

Step 3 In the **Aggressive Load Balancing Denial Count** field, enter the load balancing denial count.

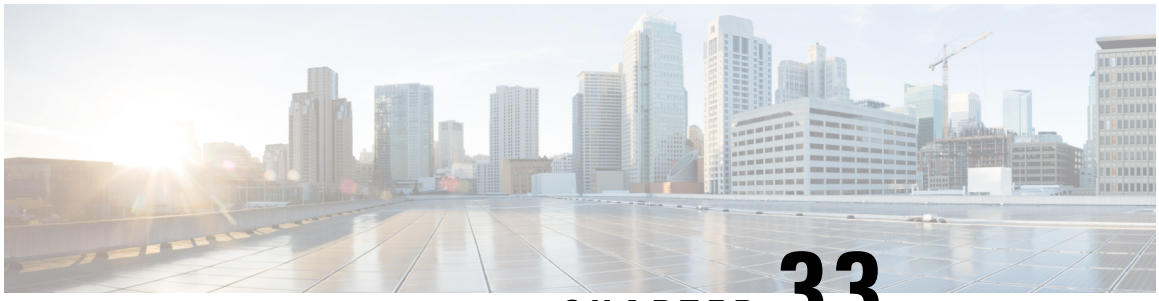
Step 4 Click **Apply**.

Configuring Aggressive Client Load Balancing (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wlan** *wlan-name* **Example:** `Device(config)# wlan test-wlan`	Specifies the WLAN name.
Step 4	**shutdown** **Example:** `Device(config-wlan)# shutdown`	Disables the WLAN.
Step 5	**load-balance** **Example:** `Device(config-wlan)# load-balance`	Configures a guest embedded wireless controller as mobility controller, in order to enable client load balance to a particular WLAN. Configure the WLAN security settings as the WLAN requirements.
Step 6	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables WLAN.
Step 7	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 8	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 9	**ap dot11** {**24ghz** | **5ghz**} **load-balancingdenial** *count* **Example:** `Device(config)# ap dot11 5ghz load-balancing denial 10`	Configures the load balancing denial count.

	Command or Action	Purpose
Step 10	**ap dot11** {**24ghz** | **5ghz**} **load-balancingwindow** *clients* **Example:** `Device(config)# ap dot11 5ghz load-balancing denial 10`	Configures the number of clients for the aggressive load balancing client window.
Step 11	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.
Step 12	**show running-config | section** *wlan-name* **Example:** `Device# show running-config | section test-wlan`	Displays a filtered section of the current configuration.



CHAPTER 33

Accounting Identity List

Configuring Accounting Identity List (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA**.

Step 2 In the **AAA Method List** tab, go to the **Accounting** section, and click **Add**.

Step 3 In the **Quick Setup: AAA Accounting** window that is displayed, enter a name for your method list.

Step 4 Choose the type of authentication as identity, in the **Type** drop-down list.

Step 5 Choose the server groups you want to use to authenticate access to your network, from the **Available Server Groups** list and click > icon to move them to the **Assigned Server Groups** list.

Step 6 Click **Save & Apply to Device**.

Configuring Accounting Identity List (CLI)

Accounting is the process of logging the user actions and keeping track of their network usage. 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.

Follow the procedure given below to configure accounting identity list.

Before you begin

Configure the RADIUS server and AAA group server.

Procedure

	Command or Action	Purpose
Step 1	**aaa accounting identity** *named-list* **start-stop group** *server-group-name* **Example:** `Device(config)# aaa accounting identity user1 start-stop group aaa-test`	Enables accounting to send a start-record accounting notice when a client is authorized and a stop-record at the end. **Note** You can also use the default list, instead of a named list.

Whenever there is a change in the client attribute, for example, change in IP address, client roaming, and so on, an accounting interim update is sent to the RADIUS server.

Configuring Client Accounting (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click the **Policy Profile Name** and in the **Edit Policy Profile** window, go to the **Advanced** tab.

Step 3 From the **Accounting List** drop-down, select the appropriate accounting list for this policy profile. This will ensure that the policy profile undergoes that type of accounting you want to perform, before allowing it access to the network.

Step 4 Click **Save & Apply to Device**.

Configuring Client Accounting (CLI)

Follow the procedure given below to configure client accounting.

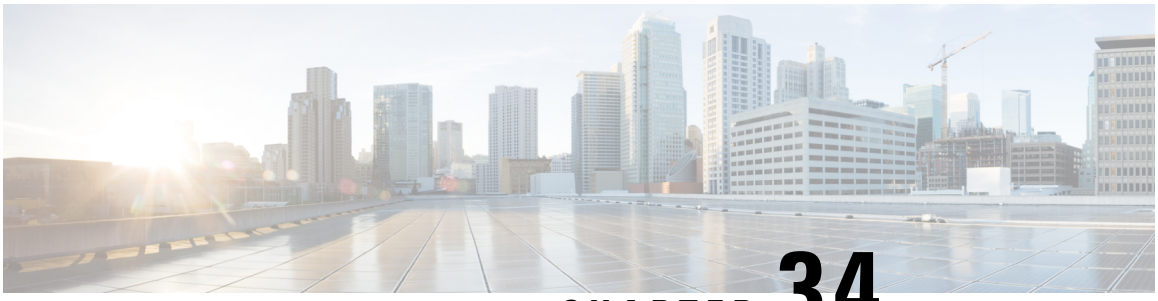
Before you begin

Ensure that RADIUS accounting is configured.

Procedure

	Command or Action	Purpose
Step 1	**wireless profile policy** *profile-policy* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Configures WLAN policy profile and enters wireless policy configuration mode.
Step 2	**shutdown** **Example:**	Disables the policy profile.

	Command or Action	Purpose
	`Device(config-wireless-policy)# shutdown`	
Step 3	**accounting-list** *list-name* **Example:** `Device(config-wireless-policy)# accounting-list user1`	Sets the accounting list.
Step 4	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Enables the policy profile.



CHAPTER 34

Volume Metering

The Volume Metering feature allows you to configure the interval at which an access point (AP) updates client accounting statistics to the embedded wireless controller and in turn to the RADIUS server. Currently, the report is sent from an AP to the controller every 90 seconds. With this feature, you can configure the time from 5 to 90 seconds. This helps reduce the delay in accounting data usage by a device.

• Configuring Volume Metering, on page 361

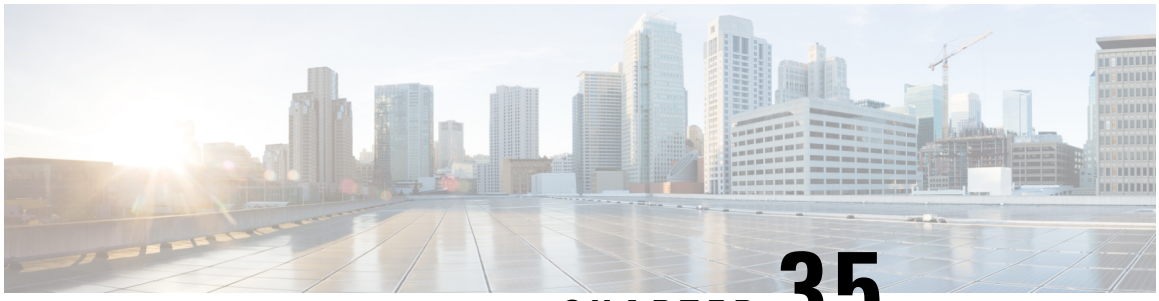
Configuring Volume Metering

Follow the procedure given below to configure volume metering:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *profile-name* **Example:** `Device(config)# ap profile yy-ap-profile`	Configures an AP profile and enters ap profile configuration mode.
Step 3	**dot11 24ghz reporting-interval** *reporting-interval* **Example:** `Device(config-ap-profile)# dot11 24ghz reporting-interval 60`	Configures the dot11 parameters.
Step 4	**dot11 5ghz reporting-interval** *reporting-interval* **Example:** `Device(config-ap-profile)# dot11 5ghz reporting-interval 60`	Configures the dot11 parameters.

	Command or Action	Purpose
Step 5	**exit** **Example:** `Device(config-ap-profile)# exit`	Returns to global configuration mode.
Step 6	**aaa accounting update periodic** *interval-in-minutes* **Example:** `Device(config)# aaa accounting update periodic 75`	Sets the time interval (in minutes) at which the embedded wireless controller sends interim accounting updates of the client to the RADIUS server.
Step 7	**exit** **Example:** `Device(config)# exit`	Exits configuration mode and returns to privileged EXEC mode.



CHAPTER 35

Enabling Syslog Messages in Access Points and Controller for Syslog Server

Information About Enabling Syslog Messages in Access Points and Embedded Wireless Controller for Syslog Server

The Syslog server on access points and embedded wireless controller has many levels and facilities.

The following are the Syslog levels:

- Emergencies
- Alerts
- Critical
- Errors
- Warnings
- Notifications
- Informational
- Debugging

The following options are available for the Syslog facility:

- auth—Authorization system.
- cron—Cron/ at facility.
- daemon—System daemons.

- kern—Kernel.
- local0—Local use.
- local1—Local use.
- local2—Local use.
- local3—Local use.
- local4—Local use.
- local5—Local use.
- local6—Local use.
- local7—Local use.
- lpr—Line printer system.
- mail—Mail system.
- news—USENET news.
- sys10—System use.
- sys11—System use.
- sys12—System use.
- sys13—System use.
- sys14—System use.
- sys9—System use.
- syslog—Syslog itself.
- user—User process.
- uucp—Unix-to-Unix copy system.

Configuring Syslog Server for an AP Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *ap-profile* **Example:** `Device(config)# ap profile xyz-ap-profile`	Configures an AP profile and enters the AP profile configuration mode.

	Command or Action	Purpose
Step 3	**syslog facility** **Example:** `Device(config-ap-profile)# syslog facility`	Configures the facility parameter for Syslog messages.
Step 4	**syslog host** *ip-address* **Example:** `Device(config-ap-profile)# syslog host 9.3.72.1`	Configures the Syslog server IP address and parameters.
Step 5	**syslog level** {**alerts** | **critical** | **debugging** | **emergencies** | **errors** | **informational** | **notifications** | **warnings**} **Example:** `Device(config-ap-profile)# syslog level`	Configures the Syslog server logging level. The following are the Syslog server logging levels: • **emergencies**—Signifies severity 0. Implies that the system is not usable. • **alerts**—Signifies severity 1. Implies that an immediate action is required. • **critical**—Signifies severity 2. Implies critical conditions. • **errors**—Signifies severity 3. Implies error conditions. • **warnings**—Signifies severity 4. Implies warning conditions. • **notifications**—Signifies severity 5. Implies normal but significant conditions. • **informational**—Signifies severity 6. Implies informational messages. • **debugging**—Signifies severity 7. Implies debugging messages. **Note** To know the number of Syslog levels supported, you need to select a Syslog level. Once a Syslog level is selected, all the levels below it are also enabled. If you enable *critical* Syslog level then all levels below it are also enabled. So, all three of them, namely, *critical*, *alerts*, and *emergencies* are enabled.

	Command or Action	Purpose
Step 6	**end** **Example:** `Device(config-ap-profile)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Syslog Server for the Controller (GUI)

Procedure

Step 1 Choose **Troubleshooting** > **Logs**.

Step 2 Click **Manage Syslog Servers** button.

Step 3 In **Log Level Settings**, from the **Syslog** drop-down list, choose a security level.

Step 4 From the **Message Console** drop-down list, choose a logging level.

Step 5 In **Message Buffer Configuration**, from the **Level** drop-down list, choose a server logging level.

Step 6 In **IP Configuration** settings, click **Add**.

Step 7 Choose the **Server Type**, from the **IPv4 / IPv6** or **FQDN** option.

Step 8 For Server Type **IPv4 / IPv6**, enter the **IPv4 / IPv6 Server Address**. For Server Type **FQDN**, enter the **Host Name**, choose the IP type and the appropriate **VRF Name** from the drop-down lists.

To delete a syslog server, click 'x' next to the appropriate server entry, under the **Remove** column.

Note When creating a host name, spaces are not allowed.

Step 9 Click **Apply to Device**.

Note When you click on **Apply to Device**, the changes are configured. If you click on **Cancel**, the configurations are discarded.

Configuring Syslog Server for the Embedded Wireless Controller

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**logging host** {*hostname* | *ipv6*} **Example:**	Enables Syslog server IP address and parameters.

	Command or Action	Purpose
	`Device(config)# logging host 124.3.52.62`	
Step 3	**logging facility** {**auth** | **cron** | **daemon** | **kern** | **local0** | **local1** | **local2** | **local3** | **local4** | **local5** | **local6** | **local7** | **lpr** | **mail** | **news** | **sys10** | **sys11** | **sys12** | **sys13** | **sys14** | **sys9** | **syslog** | **user** | **uucp**} **Example:** `Device(config)# logging facility syslog`	Enables facility parameter for the Syslog messages. You can enable the following facility parameter for the Syslog messages: • **auth**—Authorization system. • **cron**—Cron facility. • **daemon**—System daemons. • **kern**—Kernel. • **local0** to **local7**—Local use. • **lpr**—Line printer system. • **mail**—Mail system. • **news**—USENET news. • **sys10** to **sys14** and **sys9**—System use. • **syslog**—Syslog itself. • **user**—User process. • **uucp**—Unix-to-Unix copy system.
Step 4	**logging trap** {*severity-level* | **alerts** | **critical** | **debugging** | **emergencies** | **errors** | **informational** | **notifications** | **warnings**} **Example:** `Device(config)# logging trap 2`	Enables Syslog server logging level. *severity-level*- Refers to the logging severity level. The valid range is from 0 to 7. The following are the Syslog server logging levels: • **emergencies**—Signifies severity 0. Implies that the system is not usable. • **alerts**—Signifies severity 1. Implies that an immediate action is required. • **critical**—Signifies severity 2. Implies critical conditions. • **errors**—Signifies severity 3. Implies error conditions. • **warnings**—Signifies severity 4. Implies warning conditions. • **notifications**—Signifies severity 5. Implies normal but significant conditions.

	Command or Action	Purpose
		• **informational**—Signifies severity 6. Implies informational messages. • **debugging**—Signifies severity 7. Implies debugging messages. **Note** To know the number of Syslog levels supported, you need to select a Syslog level. Once a Syslog level is selected, all the levels below it are also enabled. If you enable *critical* Syslog level then all levels below it are also enabled. So, all three of them, namely, *critical*, *alerts*, and *emergencies* are enabled.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Syslog Server Configurations

Verifying Global Syslog Server Settings for all Access Points

To view the global Syslog server settings for all access points that joins the controller, use the following command:

```
Device# show ap config general
Cisco AP Name : APA0F8.4984.5E48
================================================

Cisco AP Identifier : a0f8.4985.d360
Country Code : IN
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-DN
AP Country Code : IN - India
AP Regulatory Domain
Slot 0 : -A
Slot 1 : -D
MAC Address : a0f8.4984.5e48
IP Address Configuration : DHCP
IP Address : 9.4.172.111
IP Netmask : 255.255.255.0
Gateway IP Address : 9.4.172.1
Fallback IP Address Being Used :
Domain :
Name Server :
CAPWAP Path MTU : 1485
Telnet State : Disabled
SSH State : Disabled
Jumbo MTU Status : Disabled
Cisco AP Location : default location
```

```
Site Tag Name : ST1
RF Tag Name : default-rf-tag
Policy Tag Name : PT3
AP join Profile : default-ap-profile
Primary Cisco Controller Name : WLC2
Primary Cisco Controller IP Address : 9.4.172.31
Secondary Cisco Controller Name : Not Configured
Secondary Cisco Controller IP Address : 0.0.0.0
Tertiary Cisco Controller Name : Not Configured
Tertiary Cisco Controller IP Address : 0.0.0.0
Administrative State : Enabled
Operation State : Registered
AP Certificate type : Manufacturer Installed Certificate
AP Mode : Local
AP VLAN tagging state : Disabled
AP VLAN tag : 0
CAPWAP Preferred mode : Not Configured
AP Submode : Not Configured
Office Extend Mode : Disabled
Remote AP Debug : Disabled
Logging Trap Severity Level : notification
Software Version : 16.10.1.24
Boot Version : 1.1.2.4
Mini IOS Version : 0.0.0.0
Stats Reporting Period : 180
LED State : Enabled
PoE Pre-Standard Switch : Disabled
PoE Power Injector MAC Address : Disabled
Power Type/Mode : PoE/Full Power (normal mode)
Number of Slots : 3
AP Model : AIR-AP1852I-D-K9
IOS Version : 16.10.1.24
Reset Button : Disabled
AP Serial Number : KWC212904UB
Management Frame Protection Validation : Disabled
AP User Mode : Automatic
AP User Name : Not Configured
AP 802.1X User Mode : Global
AP 802.1X User Name : Not Configured
Cisco AP System Logging Host : 9.4.172.116
AP Up Time : 11 days 1 hour 15 minutes 52 seconds
AP CAPWAP Up Time : 6 days 3 hours 11 minutes 6 seconds
Join Date and Time : 09/05/2018 04:18:52
Join Taken Time : 3 minutes 1 second
Join Priority : 1
Ethernet Port Duplex : Auto
Ethernet Port Speed : Auto
AP Link Latency : Disable
AP Lag Configuration Status : Disabled
AP Lag Operational Status : Disabled
Lag Support for AP : Yes
Rogue Detection : Enabled
Rogue Containment auto-rate : Disabled
Rogue Containment of standalone flexconnect APs : Disabled
Rogue Detection Report Interval : 10
Rogue AP minimum RSSI : -90
Rogue AP minimum transient time : 0
AP TCP MSS Adjust : Enabled
AP TCP MSS Size : 1250
AP IPv6 TCP MSS Adjust : Enabled
AP IPv6 TCP MSS Size : 1250
Hyperlocation Admin Status : Disabled
Retransmit count : 5
Retransmit interval : 3
```

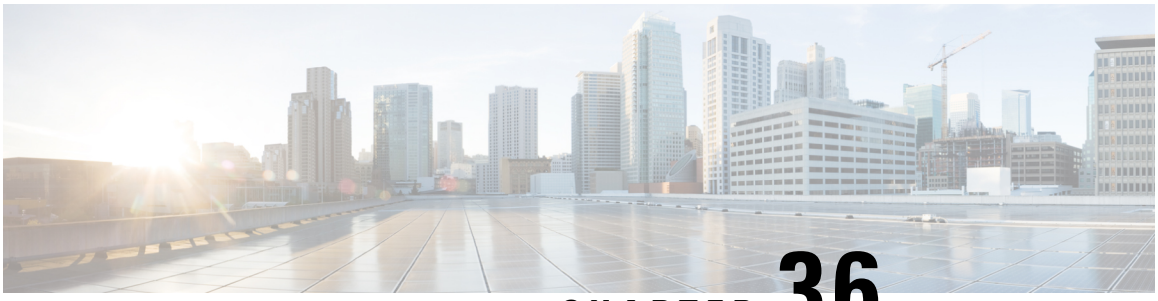
```
Fabric status : Disabled
FIPS status : Disabled
WLANCC status : Disabled
USB Module Type : USB Module
USB Module State : Enabled
USB Operational State : Disabled
USB Override : Disabled
Lawful-Interception Admin status : Disabled
Lawful-Interception Oper status : Disabled
```

Verifying Syslog Server Settings for a Specific Access Point

To view the Syslog server settings for a specific access point, use the following command:

```
Device# show ap name <ap-name> config general
show ap name APA0F8.4984.5E48 config general
Cisco AP Name : APA0F8.4984.5E48
================================================

Cisco AP Identifier : a0f8.4985.d360
Country Code : IN
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-DN
AP Country Code : IN - India
AP Regulatory Domain
Slot 0 : -A
Slot 1 : -D
MAC Address : a0f8.4984.5e48
IP Address Configuration : DHCP
IP Address : 9.4.172.111
IP Netmask : 255.255.255.0
Gateway IP Address : 9.4.172.1
Fallback IP Address Being Used :
Domain :
Name Server :
CAPWAP Path MTU : 1485
Telnet State : Disabled
SSH State : Disabled
Jumbo MTU Status : Disabled
Cisco AP Location : default location
Site Tag Name : ST1
RF Tag Name : default-rf-tag
Policy Tag Name : PT3
AP join Profile : default-ap-profile
Primary Cisco Controller Name : WLC2
Primary Cisco Controller IP Address : 9.4.172.31
Secondary Cisco Controller Name : Not Configured
Secondary Cisco Controller IP Address : 0.0.0.0
Tertiary Cisco Controller Name : Not Configured
Tertiary Cisco Controller IP Address : 0.0.0.0
Administrative State : Enabled
Operation State : Registered
AP Certificate type : Manufacturer Installed Certificate
AP Mode : Local
AP VLAN tagging state : Disabled
AP VLAN tag : 0
CAPWAP Preferred mode : Not Configured
AP Submode : Not Configured
Office Extend Mode : Disabled
Remote AP Debug : Disabled
Logging Trap Severity Level : notification
Software Version : 16.10.1.24
Boot Version : 1.1.2.4
Mini IOS Version : 0.0.0.0
Stats Reporting Period : 180
```

```
LED State : Enabled
PoE Pre-Standard Switch : Disabled
PoE Power Injector MAC Address : Disabled
Power Type/Mode : PoE/Full Power (normal mode)
Number of Slots : 3
AP Model : AIR-AP1852I-D-K9
IOS Version : 16.10.1.24
Reset Button : Disabled
AP Serial Number : KWC212904UB
Management Frame Protection Validation : Disabled
AP User Mode : Automatic
AP User Name : Not Configured
AP 802.1X User Mode : Global
AP 802.1X User Name : Not Configured
Cisco AP System Logging Host : 9.4.172.116
AP Up Time : 11 days 1 hour 15 minutes 52 seconds
AP CAPWAP Up Time : 6 days 3 hours 11 minutes 6 seconds
Join Date and Time : 09/05/2018 04:18:52
Join Taken Time : 3 minutes 1 second
Join Priority : 1
Ethernet Port Duplex : Auto
Ethernet Port Speed : Auto
AP Link Latency : Disable
AP Lag Configuration Status : Disabled
AP Lag Operational Status : Disabled
Lag Support for AP : Yes
Rogue Detection : Enabled
Rogue Containment auto-rate : Disabled
Rogue Containment of standalone flexconnect APs : Disabled
Rogue Detection Report Interval : 10
Rogue AP minimum RSSI : -90
Rogue AP minimum transient time : 0
AP TCP MSS Adjust : Enabled
AP TCP MSS Size : 1250
AP IPv6 TCP MSS Adjust : Enabled
AP IPv6 TCP MSS Size : 1250
Hyperlocation Admin Status : Disabled
Retransmit count : 5
Retransmit interval : 3
Fabric status : Disabled
FIPS status : Disabled
WLANCC status : Disabled
USB Module Type : USB Module
USB Module State : Enabled
USB Operational State : Disabled
USB Override : Disabled
Lawful-Interception Admin status : Disabled
Lawful-Interception Oper status : Disabled
```



CHAPTER 36

Software Maintenance Upgrade

Introduction to Software Maintenance Upgrade

The Software Maintenance Upgrade (SMU) is a package that can be installed on a system to provide a patch fix or a security resolution to a released image. A SMU package is provided for each release and is specific to the corresponding platform.

A SMU provides a significant benefit over classic Cisco IOS software because it allows you to address the network issue quickly while reducing the time and scope of the testing required. The Cisco IOS XE platform internally validates the SMU compatibility and does not allow you to install noncompatible SMUs.

All the SMUs are integrated into the subsequent Cisco IOS XE software maintenance releases. A SMU is an independent and self-sufficient package and does not have any prerequisites or dependencies. You can choose which SMUs to install or uninstall in any order.

Note SMUs are supported only on Extended Maintenance releases and for the full lifecycle of the underlying software release.

Note You can activate the file used in the **install add file** command only from the filesystems of the active device. You cannot use the file from the standby or member filesystems; the **install add file** command will fail in such instances.

Note When the SMU file is deleted and a reboot is performed, the device may display the following error message:

```
--- Starting SMU Add operation ---
Performing SMU_ADD on all members
    FAILED: Improper State./bootflash/<previously-installed-smu-filename>.smu.bin not
present. Please restore file for stability.
Checking status of SMU_ADD on [1/R0]
SMU_ADD: Passed on []. Failed on [1/R0]
Finished SMU Add operation
FAILED: add_activate_commit /bootflash/<tobeinstalled-wlc-smu-filename>.smu.bin Wed Aug 02
 08:30:18 UTC 2023.
```

This error occurs because the previous SMU file was not properly removed from the controller. It may lead to functional errors, such as the inability to install new SMU or APSP files.

We recommend that you use the install remove file command to remove previous instances of APSP or SMU files from the bootflash.

SMU infrastructure can be used to meet the following requirements in the wireless context:

- Controller SMU: Embedded Wireless Controller bug fixes or Cisco Product Security Incident Response information (PSIRT).
- AP bug fixes, PSIRTs, or minor features which do not require any embedded wireless controller changes.
- APDP: Support for new AP models without introduction of new hardware or software capabilities.

Note The **show ap image** command displays cumulative statistics regarding the AP images in the controller. We recommend that you clear the statistics using the **clear ap predownload statistics** command, before using the **show ap image** command, to ensure that correct data is displayed.

SMU Workflow

The SMU process should be initiated with a request to the SMU committee. Contact your customer support to raise an SMU request. During the release, the SMU package is posted on the Cisco Software Download page and can be downloaded and installed.

SMU Package

An SMU package contains the metadata and fix for the reported issue the SMU is requested for.

SMU Reload

The SMU type describes the effect to a system after installing the SMU. SMUs can be non-traffic affecting or can result in device restart, reload, or switchover.

Controller hot patching support allows SMU to be effective immediately after activation without reloading the system. Other controller SMUs require a cold reload of the system during activation. A cold reload is the complete reload of the operating system. This action affects the traffic flow for the duration of the reload (~5 min currently). This reload ensures that all processes are started with the correct libraries and files that are installed as part of the SMU.

After the SMU is committed, the activation changes are persistent across reloads.

Overview of Controller SMUs

The following table describes the SMU types supported in the Cisco Embedded Wireless Controller:

Table 16: Supported SMU Types in the Embedded Wireless Controller

Package Type	Use Case	SMU Type	Supported on EWC
Controller SMU - Cold Patch	Replace impacted binaries, libraries, or subpackages.	Reload	Limited support (Patch size < 20 MB). No support for IOSD.
Controller SMU - Hot Patch	Replace impacted functions.	Nonreload	Yes
APSP	AP fix by replacing the AP image (does not impact the AP running the active controller).	Nonreload	Yes
APSP	AP fix by replacing the AP image (impacts the AP that is running the active controller).	Reload	Yes (EWC specific variation)
APDP	New AP model support without upgrading the controller.	Nonreload	Yes

Managing Controller Hot or Cold SMU Package

Procedure

	Command or Action	Purpose
Step 1	**install add file tftp:**`//<server-ip>/<path>/<smu-filename>` **Example:** `Device# install add file tftp://<server-ip>/<path>/<smu-filename>`	The `install add` command copies the file from the external server to the backup_image directory on the embedded wireless controller.
Step 2	**install activate file backup_image:** *smu-filename* **Example:** `Device# install activate file backup_image:<smu-filename>`	This command is used to activate the patch. The `install activate` causes the controller reload only for a cold patch. There is no reload for a hot patch.

	Command or Action	Purpose
Step 3	**install auto-abort-timer stop** **Example:** `Device# install auto-abort-timer stop`	(Optional) Stops the auto cancel timer in case of activated or deactivated SMUs.
Step 4	**install commit** **Example:** `Device# install commit`	Commits the activation changes to be persistent across reloads. The commit can be done after activation while the system is up, or after the first reload. If a patch is activated and not committed, the auto cancel timer automatically cancels the activation of the patch in six hours .
Step 5	**show install rollback** **Example:** `Device# show install rollback`	Displays the list of rollback IDs that are available.
Step 6	**install rollback to** {**base** | **committed** | **id** | **label** } *specific-rollback-point* **Example:** `Device# install rollback to base`	Rolls back a committed patch. The committed patch can be deactivated and the commit for deactivation can be done using the single install rollback command.
Step 7	**install deactivate file backup_image:** *smu-filename* **Example:** `Device# install deactivate file backup_image:<Smu-Filename>`	Deactivates a comitted patch. The `install deactivate` command causes the reload of the controller in case of a cold patch. There is no reload of the controller in case of a hot patch.
Step 8	**install auto-abort-timer stop** **Example:** `Device# install auto-abort-timer stop`	(Optional) Stops the auto cancel timer in case of activated or deactivated SMUs.
Step 9	**install commit** **Example:** `Device# install commit`	Commits the deactivation changes to be persistent across reloads.
Step 10	**install remove file backup_image:** *smu-filename* **Example:** `Device# install remove file backup_image:<smu-filename>`	Removes a patch that is in the inactive state. This command also removes the file physically from `backup-image:`
Step 11	**install abort** **Example:** `Device# install abort`	Cancels the upgrade by resetting the APs in rolling fashion.
Step 12	**show install summary**	Displays information about the active package.

	Command or Action	Purpose
	Example: `Device# show install summary`	The output of this command varies based on the packages, and the package states that are installed.
Step 13	**show install package backup_image:** *smu-filename* **Example:** `Device# show install package backup-image: <smu_filename>`	Displays information about the SMU package.

Creating SMU Files (GUI)

Follow the steps given below to create SMU files:

Procedure

Step 1 Choose **Administration** > **Software Management** > **Software Maintenance Upgrade (SMU)**.

Step 2 Click **Add**.
A dialog box is displayed.

Step 3 From the **Transport Type** drop-down list,

- **TFTP**: Specify the **Server IP Address (IPv4/IPv6)**, **File Path**, **File Name**, and **File System**.
- **SFTP**: Specify the **Server IP Address (IPv4/IPv6)**, **Port Number** (Default port number is 22), SFTP username and password, **File Path**, **File Name**, and **File System**.
- **FTP**: Specify the **Server IP Address (IPv4/IPv6)**, **Port Number** (Default port number is 22), FTP username and password, **File Path**, **File Name**, and **File System**.
- **Device**: Specify the **File System** and **File path**.
- **My Desktop**: Specify the **File System** and **Source File Path** .

Step 4 Click **Add File**.

Configuration Examples for SMU

The following is sample of the SMU configuration:

```
Device# install add file
tftp://10.1.1.2/auto/tftpboot/user1/ewc/ewc-apsp1.bin
install_add: START Tue Jun 4 15:08:26 UTC 2019
Downloading file tftp://10.1.1.2/auto/tftpboot/user1/ewc/ewc-smu.bin
Finished downloading file tftp://10.1.1.2/auto/tftpboot/user1/ewc/ewc-smu.bin to
backup_image:ewc-smu.bin
install_add: Adding SMU
install_add: Checking whether new add is allowed ....
install_add: ap image predownload is allowed.

--- Starting initial file syncing ---
Info: Finished copying backup_image: ewc-smu.bin to the selected chassis
```

```
Finished initial file syncing

--- Starting SMU Add operation ---
Performing SMU_ADD on all members
[1] SMU_ADD package(s) on chassis 1
MEWLC response success sync_successCumulative SMU Size: 24 KB
Cumulative size of all SMU's will not exceed 20000 KB
Available Memory in /backup_image is 251480 KB
Available memory 251480 KB is greater than available memory required 2000 KB
[1] Finished SMU_ADD on chassis 1
Checking status of SMU_ADD on [1]
SMU_ADD: Passed on [1]
Finished SMU Add operation

SUCCESS: install_add
```

Device# install activate file backup_image:ewc-apsp1.bin

```
install_activate: START Tue Jun 4 15:18:58 UTC 2019
install_activate: Activating SMU
Cumulative SMU Size: 24 KB
Cumulative size of all SMU's will not exceed 20000 KB
Available Memory in /backup_image is 250984 KB
Available memory 250984 KB is greater than available memory required 2000 KB
MEWLC response success sync_successExecuting pre scripts....
Executing pre sripts done.

--- Starting SMU Activate operation ---
Performing SMU_ACTIVATE on all members
ls: cannot access '/tmp/sw/fp/*/*/*/mount/.pkginfo': No such file or directory
ls: cannot access '/tmp/sw/fp/*/*/*/mount/.pkginfo': No such file or directory
[1] SMU_ACTIVATE package(s) on chassis 1
valid
install_activate: FP fp error skipping. Platform to fix this in Fru List
[1] Finished SMU_ACTIVATE on chassis 1
Checking status of SMU_ACTIVATE on [1]
SMU_ACTIVATE: Passed on [1]
Finished SMU Activate operation

Executing post scripts....
Executing post scripts done.
Executing post scripts....
Executing post scripts done.
SUCCESS: install_activate /backup_image/ewc-apsp1.bin
```

Device#install commit

```
install_commit: START Tue Jun 4 16:15:25 UTC 2019
install_commit: Committing SMU
Executing pre scripts....
install_commit:
Executing pre sripts done.
--- Starting SMU Commit operation ---
Performing SMU_COMMIT on all members
ls: cannot access '/tmp/sw/fp/*/*/*/mount/.pkginfo': No such file or directory
ls: cannot access '/tmp/sw/fp/*/*/*/mount/.pkginfo': No such file or directory
[1] SMU_COMMIT package(s) on chassis 1
valid
[1] Finished SMU_COMMIT on chassis 1
Checking status of SMU_COMMIT on [1]
SMU_COMMIT: Passed on [1]
Finished SMU Commit operation

Waiting for the platform to set the SMU sync timerSMU sync status is sync_successSMU sync
to AP's success
/tmp/rp/chasfs/wireless/wlc_notify
SUCCESS: install_commit /backup_image/ewc-apsp1.bin
```

```
Device#install rollback to base
install_rollback: START Tue Jun 4 16:42:24 UTC 2019
install_rollback: Rolling back SMU
Executing pre scripts....
install_rollback:
Executing pre sripts done.

--- Starting SMU Rollback operation ---
Performing SMU_ROLLBACK on all members
ls: cannot access '/tmp/sw/fp/*/*/*/mount/.pkginfo': No such file or directory
ls: cannot access '/tmp/sw/fp/*/*/*/mount/.pkginfo': No such file or directory
[1] SMU_ROLLBACK package(s) on chassis 1
[1] Finished SMU_ROLLBACK on chassis 1
Checking status of SMU_ROLLBACK on [1]
SMU_ROLLBACK: Passed on [1]
Finished SMU Rollback operation

Executing post scripts....
Executing post scripts done.
Waiting for the platform to set the SMU sync timerSMU sync status is sync_successSMU sync
to AP's success
/tmp/rp/chasfs/wireless/wlc_notifyExecuting post scripts....
Executing post scripts done.
SUCCESS: install_rollback /backup_image/ewc-apsp1.bin Tue Jun 4 16:43:01 UTC 2019
```

```
Device# install deactivate file backup_image: ewc-apsp1.bin

install remove file backup_image:ewc-apsp1.bin
```

```
Device#show install sum
[ Chassis 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type St Filename/Version
--------------------------------------------------------------------------------
APSP C backup_image:ewc-apsp1.bin
IMG C 17.1.1.0.69043

--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

Rolling AP Upgrade

Rolling AP upgrade is a method of upgrading the APs in a staggered manner such that some APs are always up in the network and provide seamless coverage to clients, while the other APs are selected to be upgraded.

Note The AP images should be downloaded before the rolling upgrade is triggered, so that all the APs that are to be upgraded have the new image version.

Rolling AP Upgrade Process

Rolling AP upgrade is done on a per controller basis. The number of APs to be upgraded at a given time, is the percentage of the total number of APs that are connected to the controller. The percentage is capped at a

user configured value. The default percentage is 15. The non-client APs will be upgraded before the actual upgrade of APs begin.

The upgrade process is as follows:

1. **Candidate AP Set Selection**

   In this stage, a set of AP candidates are selected based on neighbouring AP information. For example, if you identify an AP for upgrade, a certain number (N) of its neighbours are excluded from candidate selection. The N values are generated in the following manner:

   If the user configurable capped percentage is 25%, then N=6 (Expected number of iterations =5)

   If the user configurable capped percentage is 15%, then N=12 (Expected number of iterations=12)

   If the user configurable capped percentage is 5%, then N=24 (Expected number of iterations =22)

   If the candidates cannot be selected using the neighbouring AP information, select candidates from indirect neighbours. If you still are not able to select candidates, the AP will be upgraded successfully without any failure.

Note After the candidates are selected, if the number of candidates are more than the configured percentage value, the extra candidates are removed to maintain the percentage cap.

2. **Client Steering**

   Clients that are connected to the candidate APs are steered to APs that are not there in the candidate AP list, prior to rebooting the candidate APs. The AP sends out a request to each of its associated clients with a list of APs that are best suited for them. This does not include the candidate APs. The candidate APs are marked as unavailable for neighbour lists. Later, the markings are reset in the AP rejoin and reload process.

3. **AP Rejoin and Reload Process**

   After the client steering process, if the clients are still connected to the candidate AP, the clients are sent a de-authorization and the AP is reloaded and comes up with a new image. A three-minute timer is set for the APs to rejoin. When this timer expires, all the candidates are checked and marked if they have either joined the controller or the mobility peer. If 90% of the candidate APs have joined, the iteration is concluded; if not, the timer is extended to three more minutes. The same check is repeated after three minutes. After checking thrice, the iteration ends and the next iteration begins. Each iteration may last for about 10 minutes.

   For rolling AP upgrade, there is only one configuration that is required. It is the number of APs to be upgraded at a time, as a percentage of the total number of APs in the network.

   Default value will be 15.

```
Device (config)#ap upgrade staggered <25 | 15 | 5>
```

Verifying AP Upgrade on the Controller

Use the following **show** command to verify the AP upgrade on the controller:

```
Device# show ap upgrade
AP upgrade is in progress
```

```
From version: 17.1.0.6
To version: 17.1.0.99

Started at: 06/04/2019 15:19:32 UTC
Configured percentage: 15
Percentage complete: 0
Expected time of completion: 06/04/2019 16:39:32 UTC

Progress Report
---------------
Iterations
----------
Iteration Start time End time AP count
-------------------------------------------------------------------------------------------------------
0 06/04/2019 15:19:33 UTC 06/04/2019 15:19:33 UTC 1
1 06/04/2019 15:19:33 UTC ONGOING 1

Upgraded
--------
Number of APs: 1
AP Name Ethernet MAC Iteration Status Site
-------------------------------------------------------------------------------------------------------
AP7069.5A74.7604 7069.5a78.5580 0 Not Impacted default-site-tag

In Progress
-----------
Number of APs: 1
AP Name Ethernet MAC
-----------------------------------------------
APB4DE.3169.7842 4c77.6dc4.a220

Remaining
---------
Number of APs: 0

AP Name Ethernet MAC
-----------------------------------------------

APs not handled by Rolling AP Upgrade
-------------------------------------
AP Name Ethernet MAC Status Reason for not handling by Rolling AP Upgrade
```

AP Device Pack (APDP) and AP Service Pack (APSP)

APSP and APDP

AP Service Pack (APSP) - APSP rolls out fixes to AP images for one or more AP models. Pre-download the AP images and activate (through rolling upgrade) these images to a subset of AP models.

- Patched APs run a different CAPWAP version than the rest of the APs. For e.g. 17.1.0.100 and 17.1.0.0.
- Per site APSP rollout is not supported. In embedded wireless controller APSP all APs must be in a single default site.

AP Device Pack (APDP) -

Currently, when a new AP hardware model is introduced, those get shipped along with the corresponding embedded wireless controller related major software version. Then you need to wait for the release of a

corresponding embedded wireless controller version relative to the new AP model and upgrade the entire network.

APDP allows you introduce the new AP model into your wireless network using the SMU infrastructure without the need to upgrade to the new embedded wireless controller version.

AP Image Changes -

When new AP models are introduced, there may or may not be corresponding new AP images. This means that AP images are mapped to the AP model families. If a new AP model belongs to an existing AP model family then you will have existing AP image entries (Example: ap3g3, ap1g5, and so on). For instance, if an AP model belongs to either ap3g3 or ap1g5, the respective image file is bundled with APDP SMU zip file. The corresponding metadata file is updated with the new AP model capability information including the AP image that it requires.

If a new AP model belongs to a new AP model family, a new image file would be bundled in the APDP SMU zip file. The corresponding metadata file is updated with the new AP model capability information including the AP image that it requires.

Information about APSP and APDP

SMU AP images are not part of the SMU binary, and the AP images are hosted outside the contoller.

- Only SFTP and TFTP methods are supported for SMU AP image download.
- HTTP, HTTP, and CCO methods are not supported for APSP or APDP.

A SMU package contains the metadata that carry AP model and its capability related details.

Note All the zipped files are required in order to successfully proceed with the upgrade. All the contained files in the zip folder are made accessible through the download method.

Following are the pre-requisites for TFTP/SFTP software upgrade:

- A TFTP/SFTP server is reachable from the management IP address of the embedded wireless controller.
- The upgrade bundle with the AP images (ap1g6, ap1g6a, ap1g7, ap3g3, and so on) and the controller image (C9800-AP-iosxe-wlc.bin) that is downloaded from the website is unzipped and copied onto the TFTP/SFTP server.

Managing APSP and APDP

AP images are hosted outside the wireless controller. In the embedded wireless controller, only TFTP or SFTP is supported for SMU AP image download.

Configuring the APSP and APDP Files (GUI)

Follow the steps given below to add APSP or APDP files:

Procedure

Step 1 Choose **Administration** > **Software Management** > **AP Service Package (APSP)** or **AP Device Package (APDP)**.
The **Add an AP Device Package** or **Add an AP Service Package** window is displayed.

Step 2 From the **Transport Type** drop-down list,

- **TFTP**: Specify the **Server IP Address (IPv4/IPv6)**, **File Path**, **File Name**, and **File System**.
- **SFTP**: Specify the **Server IP Address (IPv4/IPv6)**, **Port Number** (Default port number is 22), SFTP username and password, **File Path**, **File Name**, and **File System**.

Step 3 Click **Add File**.

Configuring the TFTP Server Directory

To set up the TFTP server directory, complete the following steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device#configure terminal`	Enter the configuration mode.
Step 2	**wireless profile image-download default** **Example:** `Device(config)#wireless profile image-download default`	Configures EWC-AP image download parameters. Use only default as the image download profile name.
Step 3	**image-download-mode** { **tftp** | **sftp** } **Example:** `Device(config-wireless-image-download-profile)#image-download-mode tftp`	Configures image download using TFTP.
Step 4	**tftp-image-path** *tftp-image-path* **Example:** `Device(config-wireless-image-download-profile-tftp)#tftp-image-path /tftpboot/cisco/ewc/`	Configures the TFTP server root directory for the AP images.
Step 5	**tftp-image-server** { **A.B.C.D** | **X:X:X:X::X** } **Example:** `Device(config-wireless-image-download-profile-tftp)#tftp-image-server 5.5.5.5`	Configures the TFTP server address.

What to do next

- Set up the remote server directory: When you receive the complete bundle in a zip file, copy the zip file to a root directory, for example, `/tftpboot/user/ewc`. Example of the complete bundle - `/tftpboot/user/ewc/17.1.zip`.
- Unzip the file. The following are the examples of the files that will be present in the root directory: `ap3g3, ap1g4, C9800-AP-iosxe-wlc.bin`, and so on.

Note When there is an issue and you want to patch an APSP SMU based on the 17.1 patch file `C9800_AP.17_1.22.CSCvr11111.apsp.zip` is pasted in the same root folder, that is, `/tftproot/user/ewc/C9800_AP.17_1.22.CSCvr11111.apsp.zip`. When you unzip the file, a sub-directory, for example, `/tftpboot/user/ewc/17_1.22.CSCvr11111/` is created automatically. The AP images (for example, `ap3g3`) and SMU binary (`apsp_CSCvr11111.bin`) are present in that sub-directory.

Configuring the SFTP Server Directory

To set up the SFTP server directory, complete the following steps:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device#configure terminal`	Enter the configuration mode.
Step 2	**wireless profile image-download default** **Example:** `Device(config)#wireless profile image-download default`	Configures EWC-AP image download parameters. Use only default as the image download profile name.
Step 3	**image-download-mode** {**tftp** | **sftp**} **Example:** `Device(config-wireless-image-download-profile)#image-download-mode sftp`	Configures image download using SFTP.
Step 4	**sftp-image-path** *sftp-image-path* **Example:** `Device(config-wireless-image-download-profile-sftp)#sftp-image-path /sftpboot/cisco/ewc/`	Configures the SFTP server root directory for the AP images.
Step 5	**sftp-image-server** {**A.B.C.D** | **X:X:X:X::X**} **Example:** `Device(config-wireless-image-download-profile-sftp)#sftp-image-server 5.5.5.5`	Configures the SFTP server address.

	Command or Action	Purpose
Step 6	**sftp-password** {**0** | **8**} *password re-enter password* **Example:** `Device(config-wireless-image-download-profile-sftp)#sftp-password 0 admin`	Configures the SFTP password.
Step 7	**sftp-username** *username* **Example:** `Device(config-wireless-image-download-profile-sftp)#sftp-username admin`	Configures the SFTP username.

What to do next

- Set up the remote server directory: When you receive the complete bundle in a zip file, copy the zip file to a root directory, for example, `/sftpboot/user/ewc`. Example of the complete bundle - `/sftpboot/user/ewc/17.1.zip`.
- Unzip the file. The following are the examples of the files that will be present in the root directory: `ap3g3, ap1g4, C9800-AP-iosxe-wlc.bin`, and so on.

Note When there is an issue and you want to patch an APSP SMU based on the 17.1 patch file `C9800_AP.17_1.22.CSCvr11111.apsp.zip` is pasted in the same root folder, that is, `/sftproot/user/ewc/C9800_AP.17_1.22.CSCvr11111.apsp.zip`. When you unzip the file, a sub-directory, for example, `/sftpboot/user/ewc/17_1.22.CSCvr11111/` is created automatically, and the AP images (for example, `ap3g3`) and SMU binary (`apsp_CSCvr11111.bin`) are present in that sub-directory.

Positive Workflow - APSP and APDP

Procedure

	Command or Action	Purpose

Step 1	**install add file** {**tftp:** | **sftp:** | **backup_image:**} `apsp.bin`
	Example:
	TFTP and Backup Image -
	`Device# install add file tftp://server_path/a.b.c/tftpboot/user/ewc/17_1.22.CSCvr11111/apsp_CSCvr11111.bin`
	`Device#install add file backup-image:apsp_CSCvr11111.bin`
		The `install add` command copies the file from the external server to the backup_image directory on the embedded wireless controller.

Step 2	**ap image predownload** **Example:** `Device# ap image predownload`	This command is optional. The command predownloads the AP image. If the predownload has started, ensure that it completes before step 3 is initiated.

	Command or Action	Purpose
Step 3	**install activate file backup-image:** *apsp.bin* **Example:** `Device# install activate file backup-image:apsp.bin`	This command starts the rolling AP upgrade. **Note** For APDP, after activate, the EWC Controller allows APs of the new AP model to join, and get the newly installed SMU AP image.
Step 4	**install commit** **Example:** `Device# install commit`	Commits the activation changes to be persistent across reloads. The commit can be done after activation while the system is up, or after one reload. If a patch is activated and not committed, the auto abort timer automatically cancels the activation of the patch in six hours .

Rollback and Cancel

One-Shot Rollback

Procedure

	Command or Action	Purpose
Step 1	**show install rollback** **Example:** `Device# show install rollback`	Displays the possible rollback points.
Step 2	**install rollback to** {**base** | **committed** | **id** | **label** } *specific-rollback-point* **Example:** `Device# install rollback to base`	This command triggers the Rolling AP upgrade. Rolling upgrade works for all APs that have the required image. Rest of the APs are rebooted together. Rolls back a committed patch. The committed patch can be deactivated and the commit for deactivation can be done using the single install rollback command.

Multi-Step Rollback

Procedure

	Command or Action	Purpose
Step 1	**show install profile** **Example:** `Device# show install profile`	The `show install profile` command displays the profiles corresponding to the rollback points.

	Command or Action	Purpose
Step 2	**install add profile** *profile-rollback-point* **Example:** `Device# install add profile profile-rollback-point`	This command prepares the wireless module for the predownload step corresponding to the rollback point.
Step 3	**install rollback to** {**base** | **committed** | **id** | **label** } *specific-rollback-point* **Example:** `Device# install rollback to base`	This command triggers the Rolling AP upgrade. Rolling upgrade works for all APs that have the required image. Rest of the APs are rebooted together. Rolls back a committed patch. The committed patch can be deactivated and the commit for deactivation can be done using the single install rollback command.

One-Shot Cancel

The following command is used for the One-Shot manual cancel:

Procedure

- **install abort**

  Example:

  ```
  Device# install abort
  ```

  This command triggers rolling AP upgrade. Cancel is allowed only if commit is not yet completed. With One-Shot Cancel there is no predownload step. Rolling AP upgrade works for all APs which have the required image. Rest are rebooted together.

Automatic Timer-Based One-Shot Cancel

After activation, a default 6-hour cancel timer is started. The cancel timer can be set to a different value when the **activate** command is issued, through the **auto-abort-timer** parameter. When the cancel timer expires, cancellation is performed the same way as the manual cancellation.

Configuring Rollback (GUI)

Follow the steps given below to configure rollback for APSP and APDP:

Procedure

Step 1 Choose **Administration** > **Software Management** .

Step 2 Select either **AP Service Pack (APSP)** or **AP Device Pack (APDP)**.

Step 3 From the **Rollback to** drop-down list, choose the Rollback type as *Base* or *Committed*.

Step 4 Click **Submit**.

Verifying APDP on the Embedded Wireless Controller

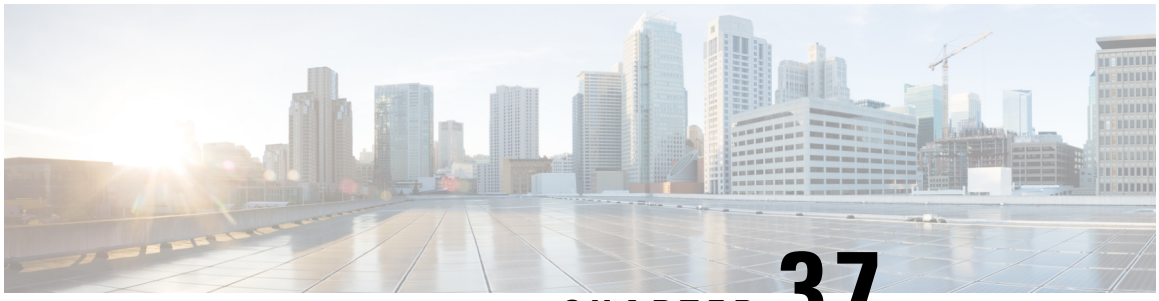
To verify the status of APDP packages on the embedded wireless controller, use the following command:

```
Device# show install summary

[ Chassis 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
APDP  I    bootflash:apdp_CSCvp12345.bin
IMG   C    17.1.0.0
--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

Note The output of this command varies based on the packages, and the package states that are installed.

PART VI

Security



CHAPTER 37

IPv4 ACLs

Information about Network Security with ACLs

This chapter describes how to configure network security on the switch by using access control lists (ACLs), which in commands and tables are also referred to as access lists.

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 controller and permit or deny packets crossing specified interfaces. 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 controller accepts or rejects the packets. Because the controller stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the controller rejects the packet. If there are no restrictions, the controller forwards the packet; otherwise, the controller drops the packet. The controller can use ACLs on all packets it forwards. There is implcit any host deny deny rule.

You configure access lists on a controller 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.

Note EWC does not support ACL on the Gi0 port as EWC does not support interface or port ACLs.

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.

Note The maximum number of ACEs that can be applied under an access policy (ACL) for central switching is 256 ACEs. The maximum number of ACEs applicable for Flex Mode or Local Switching is 64 ACEs.

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.

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.

Note For TCP ACEs with L4 Ops, the fragmented packets will be dropped per RFC 1858.

- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

ACEs and Fragmented and Unfragmented Traffic Examples

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Device(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Device(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Device(config)# access-list 102 permit tcp any host 10.1.1.2
```

```
Device(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.

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.

Note Only extended ACLs are supported while the standard ACLs are not supported.

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 following ACL-related features are not supported:

- Non-IP protocol ACLs
- IP accounting
- Reflexive ACLs, URL Redirect ACLs and Dynamic ACLs are not supported.

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating.

This lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 17: 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 (virtual teletype (VTY) lines), or to interfaces.

Numbered Extended IPv4 ACLs

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Some protocols also have specific parameters and keywords that apply to that protocol.

You can define an extended TCP, UDP, ICMP, IGMP, or other IP ACL. The switch also supports these IP protocols:

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, at times, not all commands that use IP access lists accept a named access list.

Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99 and . The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines before configuring named ACLs:

- Numbered ACLs are also available.
- A standard ACL and an extended ACL cannot have the same name.

ACL Logging

The controller 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.

Note The logging facility might drop some logging message packets if there are too many to be handled or if there is more than one logging message to be handled in 1 second. This behavior prevents the router from crashing due to too many logging packets. Therefore, the logging facility should not be used as a billing tool or an accurate source of the number of matches to an access list.

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.

The ACL scale for controllers is as follows:

- Cisco Catalyst 9800-40 Wireless Controller, Cisco Catalyst 9800-L Wireless Controller, Cisco Catalyst 9800-CL Wireless Controller (small and medium) support 128 ACLs with 128 Access List Entries (ACEs).
- Cisco Catalyst 9800-80 Wireless Controller and Cisco Catalyst 9800-CL Wireless Controller (large) support 256 ACLs and 256 ACEs.
- FlexConnect and Fabric mode APs support 96 ACLs.

Note If an ACL configuration cannot be implemented in the hardware due to an out-of-resource condition on the controller, then only the traffic in that VLAN arriving on that controller is affected.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

IPv4 ACL Interface Considerations

For inbound ACLs, after receiving a packet, the controller checks the packet against the ACL. If the ACL permits the packet, the controller continues to process the packet. If the ACL rejects the packet, the controller discards the packet.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the controller checks the packet against the ACL. If the ACL permits the packet, the controller sends the packet. If the ACL rejects the packet, the controller discards the packet.

If an undefined ACL has nothing listed in it, it is an empty access list.

Restrictions for Configuring IPv4 Access Control Lists

The following are restrictions for configuring network security with ACLs:

General Network Security

The following are restrictions for configuring network security with ACLs:

- A standard ACL and an extended ACL cannot have the same name.
- Though visible in the command-line help strings, **AppleTalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.
- DNS traffic is permitted by default with or without ACL entries for clients that are awaiting web authentication.

IP Access List Entry Sequence Numbering

- This feature does not support dynamic, reflexive, or firewall access lists.

How to Configure ACLs

Configuring IPv4 ACLs (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 Click **Add**.

Step 3 In the **Add ACL Setup** dialog box, enter the following parameters.

- **ACL Name:** Enter the name for the ACL.
- **ACL Type:** IPv4 Standard.
- **Sequence:** Enter the sequence number.
- **Action:** Choose **Permit** or **Deny** the packet flow from the drop-down list.
- **Source Type:** Choose **any**, **Host** or **Network** from which the packet is sent.
- **Log:** Enable or disable logging.

Step 4 Click **Add**.

Step 5 Add the rest of the rules and click **Apply to Device**.

Configuring IPv4 ACLs

Follow the procedure given below to use IP ACLs on the switch:

Procedure

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..

Creating a Numbered Standard ACL (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 On the **ACL** page, click **Add**.

Step 3 In the **Add ACL Setup** window, enter the following parameters.

- **ACL Name:** Enter the name for the ACL.
- **ACL Type:** IPv4 Standard.
- **Sequence:** Enter the sequence number.
- **Action:** Choose **Permit** or **Deny** access from the drop-down list.
- **Source Type:** Choose **any**, **Host** or **Network**
- **Log:** Enable or disable logging, this is limited to ACLs associated to Layer 3 interface only.

Step 4 Click **Add**.

Step 5 Click **Save & Apply to Device**.

Creating a Numbered Standard ACL (CLI)

Follow the procedure given below to create a numbered standard ACL:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard*] **Example:** `Device(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.

	Command or Action	Purpose
		• The keyword **any** as an abbreviation for *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword **host** as an abbreviation for source and *source-wildcard* of *source* 0.0.0.0. (Optional) The *source-wildcard* applies wildcard bits to the source. **Note** Logging is supported only on ACLs attached to Layer 3 interfaces.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Creating a Numbered Extended ACL (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 On the **ACL** page, click **Add**.

Step 3 In the **Add ACL Setup** window, enter the following parameters.

- **ACL Name:** Enter the name for the ACL.
- **ACL Type:** IPv4 Extended.
- **Sequence:** Enter the sequence number.

- **Action:** Choose **Permit** or **Deny** the packet flow from the drop-down list.
- **Source Type:** Choose **any**, **Host** or **Network** from which the packet is sent.
- **Destination Type:** Choose **any**, **Host** or **Network** to which the packet is sent.
- **Protocol:** Choose a protocol from the drop-down list.
- **Log:** Enable or disable logging.
- **DSCP:** Enter to match packets with the DSCP value

Step 4 Click **Add**.

Step 5 Click **Save & Apply to Device**.

Creating a Numbered Extended ACL (CLI)

Follow the procedure given below to create a numbered extended ACL:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Device(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**. **Note** This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see the following steps.

	Command or Action	Purpose
		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). • **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.

	Command or Action	Purpose
		Note Your embedded controller must support the ability to: • Mark DCSP • Mark UP • Map DSCP and UP For more information on **DSCP-to-UP Mapping**, see: https://tools.ietf.org/html/draft-ietf-tsvwg-ieee-802-11-01 **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*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*] **Example:** `Device(config)# access-list 101 permit tcp any any eq 500`	Defines an extended TCP access list and the access conditions. The parameters are the same as those described for an extended IPv4 ACL, with these exceptions: (Optional) Enter an *operator* and *port* to compare source (if positioned after *source source-wildcard*) or destination (if positioned after *destination destination-wildcard*) port. Possible operators include **eq** (equal), **gt** (greater than), **lt** (less than), **neq** (not equal), and **range** (inclusive range). Operators require a port number (range requires two port numbers separated by a space). Enter the *port* number as a decimal number (from 0 to 65535) or the name of a TCP port. Use only TCP port numbers or names when filtering TCP. The other optional keywords have these meanings: • *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).

	Command or Action	Purpose
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**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Device(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** 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**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Device(config)# access-list 101 permit icmp any any 200`	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. • *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**] [**time-range** *time-range-name*] [**dscp** *dscp*] **Example:** `Device(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:** `Device(config)# end`	Returns to privileged EXEC mode.

Creating Named Standard ACLs (GUI)

Procedure

Step 1 Click **Configuration** > **Security** > **ACL**.

Step 2 Click **Add** to create a new ACL setup.

Step 3 In the **Add ACL Setup** window, enter the following parameters.

- **ACL Name:** Enter the name for the ACL
- **ACL Type:** IPv4 Standard
- **Sequence:** The valid range is between 1 and 99 or 1300 and 1999
- **Action:** Choose **Permit** or **Deny** access from the drop-down list.
- **Source Type:** Choose **any**, **Host** or **Network**
- **Log:** Enable or disable logging, this is limited to ACLs associated to Layer 3 interface only.

Step 4 Click **Add** to add the rule.

Step 5 Click **Save & Apply to Device**.

Creating Named Standard ACLs

Follow the procedure given below to create a standard ACL using names:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip access-list standard** *name* **Example:** `Device(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.

	Command or Action	Purpose

Step 4	Use one of the following:
	• **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
	• **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
	Example:
	`Device(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255`
	or
	`Device(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0`
		In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped.
		• **host** *source*—A source and source wildcard of *source* 0.0.0.0.
		• **any**—A source and source wildcard of 0.0.0.0 255.255.255.255.

Step 5	**end** **Example:** `Device(config-std-nacl)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Creating Extended Named ACLs (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **ACL**.

Step 2 Click **Add**.

Step 3 In the **Add ACL Setup** window, enter the following parameters.

- **ACL Name:** Enter the name for the ACL.

- **ACL Type:** IPv4 Extended.
- **Sequence:** Enter the sequence number.
- **Action:** Choose **Permit** or **Deny** the packet flow from the drop-down list.
- **Source Type:** Choose **any**, **Host** or **Network** from which the packet is sent.
- **Destination Type:** Choose **any**, **Host** or **Network** to which the packet is sent.
- **Protocol:** Choose a protocol from the drop-down list.
- **Log:** Enable or disable logging.
- **DSCP:** Enter to match packets with the DSCP value

Step 4 Click **Add**.

Step 5 Add the rest of the rules and click **Apply to Device**.

Creating Extended Named ACLs

Follow the procedure given below to create an extended ACL using names:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ip access-list extended** *name* **Example:** `Device(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 4	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*]	In access-list configuration mode, specify the conditions allowed or denied. Use the **log** keyword to get access list logging messages, including violations.

	Command or Action	Purpose
	Example: `Device(config-ext-nacl)# permit 0 any any`	• **host** *source*—A source and source wildcard of *source* 0.0.0.0. • **host** *destintation*—A destination and destination wildcard of *destination* 0.0.0.0. • **any**—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 5	**end** **Example:** `Device(config-ext-nacl)# end`	Returns to privileged EXEC mode.
Step 6	**show running-config** **Example:** `Device# show running-config`	Verifies your entries.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

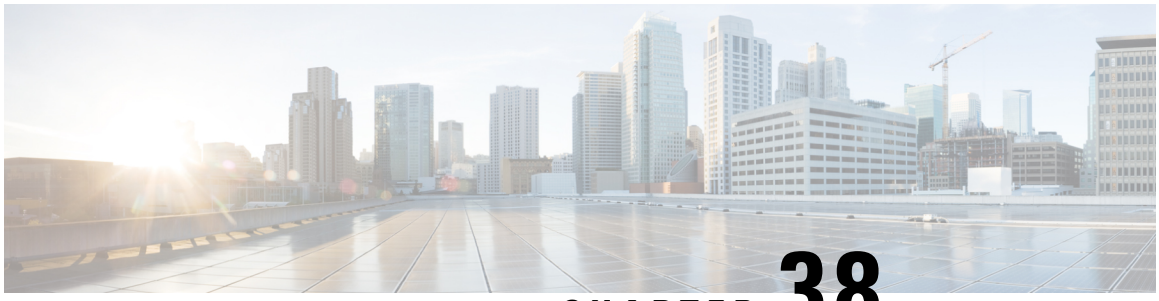
When you are creating extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL.

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to do next

After creating a named ACL, you can apply it to interfaces or to VLANs.



CHAPTER 38

DNS-Based Access Control Lists

Information About DNS-Based Access Control Lists

The DNS-based ACLs are used for wireless client devices. When using these devices, you can set pre-authentication ACLs on the embedded wireless controller to determine the data requests that are allowed or blocked.

To enable DNS-based ACLs on the embedded wireless controller, you need to configure the allowed URLs or denied URLs for the ACLs. The URLs need to be pre-configured on the ACL.

With DNS-based ACLs, the client when in registration phase is allowed to connect to the configured URLs. The embedded wireless controller is configured with the ACL name that is returned by the AAA server. If the ACL name is returned by the AAA server, then the ACL is applied to the client for web-redirection.

At the client authentication phase, the AAA server returns the pre-authentication ACL (url-redirect-acl, which is the attribute name given to the AAA server). The DNS snooping is performed on the AP for each client until the registration is complete and the client is in SUPPLICANT PROVISIONING state. When the ACL configured with the URLs is received on the embedded wireless controller, the CAPWAP payload is sent to the AP enabling DNS snooping for the URLs to be snooped.

With URL snooping in place, the AP learns the IP address of the resolved domain name in the DNS response. If the domain name matches the configured URL, then the DNS response is parsed for the IP address. The AP adds the IP address to the allowed list of IP addresses and thus the client can access the URLs configured.

During pre-authentication or post-authentication, DNS ACL is applied to the client in the access point. If the client roams from one AP to another AP, the DNS learned IP addresses on the old AP is valid on the new AP as well.

This feature supports:

- A maximum of 32 URL lists.
- A maximum of 32 URLs per URL list.
- Up to 30 IP addresses per URL.

- A maximum of 16 URL lists with wild-cards.
- A maximum of 10 URLs per wild-card URL.

Note When configuring wild-card based URLs, generic wild-card URLs are not allowed; wild-cards cannot be present between the domain name; multiple wild-cards are not allowed in a URL. Wild-card specification in a URL can only be at a third-degree level or a higher level.

Note Conflicting or invalid configurations are not allowed. The same URL cannot have different actions. For example, Deny and Allow cannot be configured on www.yahoo.com.

Note URL filter needs to be attached to a policy profile in case of the local mode. In the flex mode, the URL filter is attached to the flex profile and it is not need to be attached to a policy profile.

Note DNS based URLs work with active DNS query from the client. Hence, for URL filtering, the DNS should be setup correctly.

Note URL filter takes precedence over punt or redirect ACL, and over custom or static pre-auth ACL.s

FlexConnect in Embedded Wireless Controller

FlexConnect is a wireless solution for branch office and remote office deployments. It enables customers to configure and control access points in a branch or remote office from the corporate office through a wide area network (WAN) link without deploying a embedded wireless controller in each branch office.

The FlexConnect access points can switch client data traffic locally while carrying the authentication centrally. Also, FlexConnect APs perform client authentication locally when their connection to the controller is lost. When they are connected back to the controller, they can also send authentication/policy details back to the embedded wireless controller.

The embedded wireless controller network comprises of at least one 802.11ax Wave 2 Cisco Aironet Series access point (AP) with a software-based embedded wireless controller managing other APs in the network. The AP acting as the embedded wireless controller is referred to as the primary AP while the other APs in the network, which are managed by this primary AP, are referred to as subordinate APs. In addition to acting as an embedded wireless controller, the primary AP also operates as an AP to serve clients along with the subordinate APs.

Pre-Auth DNS ACL feature is also known as Walled Garden feature. The walled garden is a list of web sites or domains that you can visit without being authenticated. DNS snooping is performed on the AP for each client and configured rule is applied to client traffic after matching the Source or Destination IP.

Roaming

During Roaming, the support clients roam from one AP to the other using the existing roaming support. DNS ACLs are retained at the target AP even after roaming. For Roaming with DNS Pre-Auth ACL and Post-Auth ACL, the target AP learns the client-resolved IP from the serving AP.

Restrictions on DNS-Based Access Control Lists

The restriction for DNS-based ACLs is as follows:

- Only supported for FlexConnect local switching APs with Central Authorization.
- Post-Auth DNS based ACL is not supported for FlexConnect with local Authorization when AP is in FlexConnect local switching mode.
- Fully qualified domain name (FQDN) or DNS based ACLs are not supported on Cisco Wave 1 Access Points.
- The URL filter considers only the first 20 URLs, though you can add more.
- The URL filter employs regular regex patterns and permits wildcard characters only at the beginning or at the end of an URL.
- The URL ACLs are defined and added to the FlexConnect policy profile in which they associate with a WLAN. The URL ACL creation follows a similar mechanism as that of local mode URL ACLs.
- In FlexConnect mode, the URL domain ACL works only if they are connected to a FlexConnect policy profile.
- The ACL can be attached to a WLAN by associating a policy profile with a WLAN or local policies. However, you can override it using "url-redirect-acl".
- For the Cisco AV pair received from ISE, the policy that needs to be applied for a particular client is pushed as part of ADD MOBILE

  message.
- When an AP joins or when an existing URL ACL is modified and applied on FlexConnect profile, the ACL definition along with mapped URL filter list is pushed to the AP.
- The AP stores the URL ACL definition with mapped ACL name and snoops the DNS packets for learning the first IP address for each URL in the ACL. When the AP learns the IP addresses, it updates the controller of the URL and IP bindings. The controller records this information in the client database for future use.
- When a client roams to another AP during the pre-authentication state, the learned IP addresses are pushed to a new AP. Otherwise, these learned IP addresses are purged when a client moves to a post-authentication state or when the TTL for the learned IP address expires.

Flex Mode

Configuring the URL Filter List (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile flex** *custom-flex-profile* **Example:** `Device(config)# wireless profile flex custom-flex-profile`	Configures a wireless flex profile and enters wireless flex profile configuration mode.
Step 3	**acl-policy** *acl-policy-name* **Example:** `Device(config-wireless-flex-profile)#acl-policy acl-policy-name`	Configures the ACL policy description
Step 4	**urlfilter list** *url-filterlist-name* **Example:** `Device(config-wireless-flex-profile-acl)# urlfilter list url-filterlist-name`	Configures and applies the name of the URL filter list to the flex profile. This is the Flex URL filter configuration command for ACL binding.

Configuring the URL Filter List (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **URL Filters**.
The **URL Filters** page is displayed.

Step 2 Click the **Add** button.
The **Add URL Filters** window is displayed.

Step 3 From the **Type** drop-down list, choose either **PRE-AUTH** or **POST-AUTH**.

a) **POST-AUTH**: Specify the **Redirect Servers** for **IPv4** and **IPv6**.

Step 4 Use the slider to **Permit** or **Deny** the **Action**.

Step 5 Specify the URLs in the **URLs** field. Enter every URL on a new line.

Step 6 Click **Apply to Device**.

Applying Custom Pre-Auth DNS ACL on WLAN

For pre-auth, this configuration should be on a web-auth WLAN.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan***wlan-name wlan-id ssid-name* **Example:** `Device(config)# wlan wlan-name wlan-id ssid-name`	Enters the WLAN configuration sub-mode. 1. wlan-name — Enter the profile name. The range is from 1 to 32 alphanumeric characters. 2. wlan-id—Enter the WLANID. The range is from 1 to 512. 3. SSID-name—Enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID. If you have already configured WLAN, enter wlan wlan-name command.
Step 3	**ip access-group web** *access-list-name* **Example:** `Device(config-wlan)#ip access-group web preauth-acl-wlan`	Maps the ACL to the web auth WLAN. access-list-name is the IPv4 ACL name or ID.

Applying Custom Post-Auth DNS ACL on Policy Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**Wireless profile policy** *profile-name* **Example:** `Device(config)# wireless profile policy custom-policy-profile`	Creates policy profile for the WLAN.
Step 3	{**ipv4** | **ipv6**} **acl** *post-acl-name* **Example:** `Device(config-wireless-policy)#ipv4 acl post-acl`	Creates ACL configuration for wireless IPv4 or IPv6 configuration.

Configuring ISE for Central Web Authentication (GUI)

Perform the following steps to configure ISE for Central Web Authentication.

Procedure

Step 1 Login to the Cisco Identity Services Engine (ISE).

Step 2 Click **Policy** and then click **Policy Elements**.

Step 3 Click **Results**.

Step 4 Expand **Authorization** and click **Authorization Profiles**.

Step 5 Click **Add** to create a new authorization profile for URL filter.

Step 6 Enter a name for the profile in the **Name** field. For example, CentralWebauth.

Step 7 Choose **ACCESS_ACCEPT** option from the **Access Type** drop-down list.

Step 8 Alternatively, in the **Common Tasks** section, check **Web Redirection**..

Step 9 Choose the **Centralized Web Auth** option from the drop-down list.

Step 10 Specify the ACL and choose the ACL value from the drop-down list.

Step 11 In the **Advanced Attributes Setting** section, choose **Cisco:cisco-av-pair** from the drop-down list.

Note Multiple ACL can be applied on the controller based on priority. In L2 Auth + webauth multi-auth scenario, if the ISE returns ACL during L2 Auth then ISE ACL takes precedence over the default webauth redirect ACL. This leads to traffic running in webauth pending state, if ISE ACL has permit rule. To avoid this scenario, you need to set the precedence for L2 Auth ISE returned ACL. The default webauth redirect ACL priority is 100. To avoid traffic issue, you need to configure the redirect ACL priority above 100 for ACL returned by ISE.

Step 12 Enter the following one by one and click (+) icon after each of them:

- url-redirect-acl=<sample_name>
- url-redirect=<sample_redirect_URL>

  For example,

```
Cisco:cisco-av-pair = priv-lvl=15
Cisco:cisco-av-pair = url-redirect-acl=ACL-REDIRECT2
Cisco:cisco-av-pair = url-redirect=
https://9.10.8.247:port/portal/gateway?
sessionId=SessionIdValue&portal=0ce17ad0-6d90-11e5-978e-005056bf2f0a&daysToExpiry=value&action=cwa
```

Step 13 Verify contents in the **Attributes Details** section and click **Save**.

Viewing DNS-Based Access Control Lists

To view the URL Lists, use the following command:

```
Device #show wireless urlacl-enhanced summary
URL-List
--------------------------------
```

```
urllist_ut
urllist_max1
urllist_max2
urllist_max3
urllist_max4
urllist_max5
```

To view the details of a particular URL List, use the following command:

```
Device#show wireless urlacl-enhanced details urllist_ut
List Name..................... : urllist_ut
Configured List of URLs
URL                  Preference Action Validity Invalidated URL
------------------------------------------------------------------------------
url1.dns.com         1                PERMIT         VALID 0
url2.dns.com         2                DENY           VALID 0
url3.dns.com         3                PERMIT         VALID 0
url4.dns.com         4                DENY           VALID 0
url11.dns.com        6                DENY           VALID 0
url12.dns.com        7                PERMIT         VALID 0
url13.dns.com        8                DENY           VALID 0
www.example.com      14               PERMIT         VALID 0
```

To view the flex profile details, use the following command:

```
Device# sh wireless profile flex detailed custom-flex-profile
Flex Profile Name : custom-flex-profile
Description : custom flex profile
Local Auth :
        AP:
                Radius Enable            : ENABLED
                PEAP                     : DISABLED
                LEAP                     : DISABLED
                TLS                      : DISABLED
                EAP fast profile         : Not Configured
                User List                : Not Configured
        RADIUS:
                RADIUS server group name : Not Configured
        Fallback Radio shut              : DISABLED
        ARP caching                      : ENABLED
        Efficient Image Upgrade          : ENABLED
        OfficeExtend AP                  : DISABLED
        Join min latency                 : DISABLED
        Policy ACL :
                ACL Name                          URL Filter List
        Name                 Central Webauth
        ---------------------------------------------------------------
        post-acl               urllist_ut        DISABLED
        pre_v4             urllist_pre_cwa   DISABLED
        ACL-REDIRECTTTTTTT2     urllist_ut        DISABLED
        VLAN Name - VLAN ID mapping       : Not Configured
```

To view client details, use the following command:

```
Device#sh wireless client mac-address <Mac-address> detail
```

Verifying the Access Point

To view the ACL configuration on the AP, use the following command:

```
Device# show ip access-lists
Extended IP access list pre_v4
        1 permit udp any range 0 65535 any eq 53
        2 permit tcp any range 0 65535 any eq 53
        3 permit udp any dhcp_server any range 0 65535
```

```
        4 permit udp any range 0 65535 any eq 68
        5 permit udp any dhcp_client any range 0 65535
        6 deny ip any any
```

To view the URL List configuration, use the following command:

```
Device#show flexconnect url-acl
ACL-NAME         ACTION         URL-LIST
pre_v4
                 allow          test.dns.com
                 allow          url2.dns.com
                 allow          url3.dns.com
                 allow          url10.dns.com
                 allow          url11.dns.com
                 allow          www.cwapre.com
                 allow          www.google.com
                 allow          oldconfig.dns.com
                 allow          *.cisco.com
```

To view pre-auth client configuration, use the following command:

```
Device# show client access-lists pre-auth all C0:C1:C0:70:58:2F
Pre-Auth URL ACLs for Client: C0:C1:C0:70:58:2F
IPv4 ACL: pre_v4
IPv6 ACL:
ACTION         URL-LIST
allow          url1.dns.com
deny           url2.dns.com
allow          url3.dns.com
deny           url4.dns.com
allow          www.example.com
deny           url11.dns.com
allow          url12.dns.com
deny           url13.dns.com

Resolved IPs for Client: C0:C1:C0:70:58:2F
HIT-COUNT         URL         ACTION         IP-LIST
post-acl
        rule 0:   allow true
No IPv6 ACL found
```

To view post-auth client configuration, use the following command:

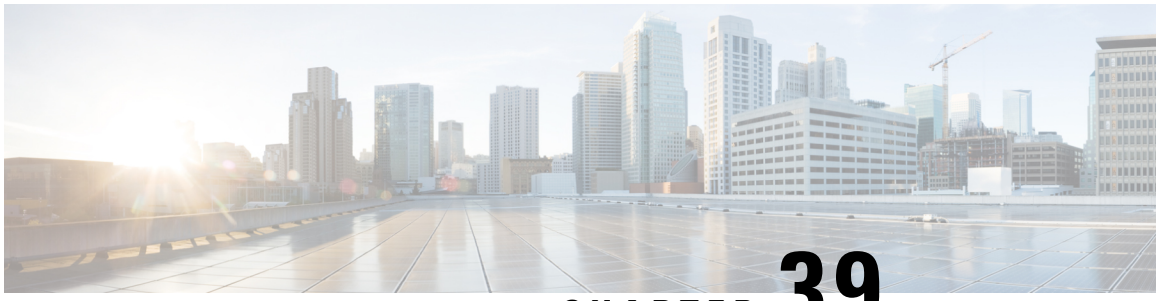
```
Device# show client access-lists post-auth all C0:C1:C0:70:58:2F
Post-Auth URL ACLs for Client: C0:C1:C0:70:58:2F
IPv4 ACL: post-acl
IPv6 ACL:
ACTION         URL-LIST
allow          url1.dns.com
deny           url2.dns.com
allow          url3.dns.com
deny           url4.dns.com
allow          www.example.com
deny           url11.dns.com
allow          url12.dns.com
deny           url13.dns.com

Resolved IPs for Client: C0:C1:C0:70:58:2F
HIT-COUNT         URL         ACTION         IP-LIST
post-acl
        rule 0: allow true
No IPv6 ACL found
```

To view the IPs learnt in pre-auth, use the following command:

```
Device#show client access-lists pre-auth all 60:14:B3:AA:C6:FB
Pre-Auth URL ACLs for Client: 60:14:B3:AA:C6:FB
IPv4 ACL: acl_1
IPv6 ACL:
ACTION         URL-LIST
allow          url1.dns.com
deny           url2.dns.com

Resolved IPs for Client: 60:14:B3:AA:C5:FB
HIT-COUNT         URL                 ACTION         IP-LIST
10                url1.dns.com        allow          9.10.8.1
```

To view the IPs learnt in post-auth, use the following command:

```
Device#show client access-lists post-auth all 60:14:B3:AA:C6:FB
Post-Auth URL ACLs for Client: 60:14:B3:AA:C5:FB
IPv4 ACL: post_acl
IPv6 ACL:
ACTION         URL-LIST
deny           url1.dns.com
allow          url2.dns.com

Resolved IPs for Client: 60:14:B3:AA:C5:FB
HIT-COUNT         URL                 ACTION         IP-LIST
16                url2.dns.com        allow          9.10.9.1
postauth_acl
          rule 0: allow true
```



CHAPTER 39

Allowed List of Specific URLs

Allowed List of Specific URLs

This feature helps you to add specific URLs to allowed list on the embedded wireless controller or the AP so that those specific URLs are available for use, even when there is no connectivity to the internet. You can add URLs to allowed list for web authentication of captive portal and walled garden. Authentication is not required to access the allowed list of URLs. When you try to access sites that are not in allowed list, you are redirected to the Login page.

Adding URL to Allowed List

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**urlfilter list** *<urlfilter-name>* **Example:** `Device(config)# urlfilter list url-allowedlist-nbn`	Configures the URL filter profile.
Step 3	**action** [**deny** | **permit**] **Example:** `Device(config-urlfilter-params)# action permit`	Configures the list as allowed list. The **permit** command configures the list as allowed list and the **deny** command configures the list as blocked list.

	Command or Action	Purpose
Step 4	{**redirect-server-ipv4** | **redirect-server-ipv6**} **Example:** `Device(config-urlfilter-params)# redirect-server-ipv4 X.X.X.X`	Configures the IP address of the redirect servers to which the user requests will be redirected in case of denied requests.
Step 5	**url** *url-to-be-allowed* **Example:** `Device(config-urlfilter-params)# url www.cisco.com`	Configures the URL to be allowed.

Note The controller uses two IP addresses and the mechanism only allows for one portal IP to be allowed. To allow pre-authentication access to more HTTP ressources, you need to use URL filters which will dynamically makes holes in the intercept (redirect) and security (preauth) ACLs for the IPs related to the website whose URL you enter in the URL filter. DNS requests will be dynamically snooped for the controller to learn the IP address of those URLs and add it to the ACLs dynamically.

Note **redirect-server-ipv4** and **redirect-server-ipv6** is applicable only in the local mode, specifically in post-authentication. For any further tracking or displaying any warning messages, the denied user request is redirected to the configured server.

But the **redirect-server-ipv4** and **redirect-server-ipv6** configurations do not apply to pre-authentication scenario as you will be redirected to the controller for the redirect login URL for any denied access.

You can associate the allowed URL with the ACL policy in flex profile.

Example

Associating the allowed URL with the ACL policy in flex profile:

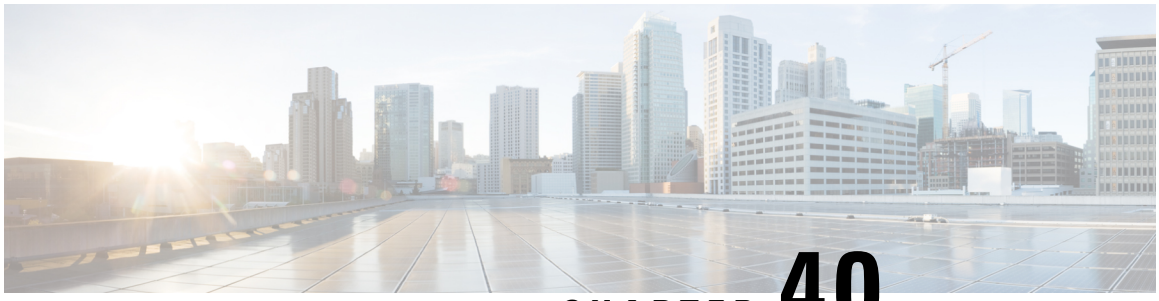
```
Device(config)# wireless profile flex default-flex-profile
Device(config-wireless-flex-profile)# acl-policy user_v4_acl
Device(config-wireless-flex-profile-acl)# urlfilter list url_allowedlist_nbn
Device(config-wireless-flex-profile-acl)# exit
Device(config-wireless-flex-profile)# description "default flex profile"

Device(config)# urlfilter enhanced-list urllist_pre_cwa
Device(config-urlfilter-enhanced-params)# url url1.dns.com preference 1 action permit
Device(config-urlfilter-enhanced-params)# url url2.dns.com preference 2 action deny
Device(config-urlfilter-enhanced-params)# url url3.dns.com preference 3 action permit

Device(config)# wlan wlan5 5 wlan5
Device(config-wlan)#ip access-group web user_v4_acl
Device(config-wlan)#no security wpa
Device(config-wlan)#no security wpa
Device(config-wlan)#no security wpa wpa2 ciphers aes
Device(config-wlan)#no security wpa akm dot1x
Device(config-wlan)#security web-auth
Device(config-wlan)#security web-auth authentication-list default
```

```
Device(config-wlan)#security web-auth parameter-map global
Device(config-wlan)#no shutdown
```

Verifying URLs on the Allowed List

To verify the summary and the details of the URLs on the allowed list, use the following **show** commands:

```
Device# show wireless urlfilter summary
Black-list    - DENY
White-list    - PERMIT
Filter-Type   - Specific to Local Mode

URL-List                        ID  Filter-Type  Action   Redirect-ipv4  Redirect-ipv6
-------------------------------------------------------------------------------------------------
url-whitelist                   1    PRE-AUTH     PERMIT   1.1.1.1

Device#

Device# show wireless urlfilter details url-whitelist
List Name..................... : url-whitelist
Filter ID..................... : : 1
Filter Type................... : PRE-AUTH
Action........................ : PERMIT
Redirect server ipv4...... : 1.1.1.1
Redirect server ipv6...... :
Configured List of URLs
   URL....................... : www.cisco.com
```



CHAPTER 40

Web-Based Authentication

This chapter describes how to configure web-based authentication on the device. It contains these sections:

Authentication Overview

Web authentication is a Layer 3 security solution designed for providing easy and secure guest access to hosts on WLAN with open authentication or appropriate layer 2 security methods. Web authentication allows users to get authenticated through a web browser on a wireless client, with minimal configuration on the client side. It allows users to associate with an open SSID without having to set up a user profile. The host receives an IP address and DNS information from the DHCP server, however cannot access any of the network resources until they authenticate successfully. When the host connects to the guest network, the WLC redirects the host to an authentication web page where the user needs to enter valid credentials. The credentials are authenticated by the WLC or an external authentication server and if authenticated successfully is given full access to the network. Hosts can also be given limited access to particular network resources before authentication for which the pre-authentication ACL functionality needs to be configured.

The following are the different types of web authentication methods:

- Local Web Authentication (LWA): Configured as Layer 3 security on the controller, the web authentication page and the pre-authentication ACL are locally configured on the controller. The controller intercepts htttp(s) traffic and redirects the client to the internal web page for authentication. The credentials entered by the client on the login page is authenticated by the controller locally or through a RADIUS or LDAP server.

- External Web Authentication (EWA): Configured as Layer 3 security on the controller, the controller intercepts htttp(s) traffic and redirects the client to the login page hosted on the external web server. The credentials entered by the client on the login page is authenticated by the controller locally or through a RADIUS or LDAP server. The pre-authentication ACL is configured statically on the controller.

- Central Web Authentication (CWA): Configured mostly as Layer 2 security on the controller, the redirection URL and the pre-authentication ACL reside on ISE and are pushed during layer 2 authentication to the controller. The controller redirects all web traffic from the client to the ISE login page. ISE validates the credentials entered by the client through HTTPS and authenticates the user.

Use the authentication feature, known as web authentication proxy, to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.

When a client initiates an HTTP session, authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

If authentication succeeds, authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server.

If authentication fails, 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, authentication forwards a Login-Expired HTML page to the host, and the user is .

Note The traceback that you receive when webauth client tries to do authentication does not have any performance or behavioral impact. It happens rarely when the context for which FFM replied back to EPM for ACL application is already dequeued (possibly due to timer expiry) and the session becomes ‘unauthorized’.

Note When command authorization is enabled as a part of AAA Authorization configuration through TACACS and the corresponding method list is not configured as a part of the HTTP configuration, WebUI pages will not load any data. However, some wireless feature pages may work as they are privilege based and not command based.

Based on where the web pages are hosted, the local web authentication can be categorized as follows:

- *Internal*—The internal default HTML pages (Login, Success, Fail, and Expire) in the embedded wireless controller are used during the local web authentication.
- *Customized*—The customized web pages (Login, Success, Fail, and Expire) are downloaded onto the embedded wireless controller and used during the local web authentication.
- *External*—The customized web pages are hosted on the external web server instead of using the in-built or custom web pages.

Based on the various web authentication pages, the types of web authentication are as follows:

- *Webauth*—This is a basic web authentication. Herein, the embedded wireless controller presents a policy page with the user name and password. You need to enter the correct credentials to access the network.
- *Consent or web-passthrough*—Herein, the controller presents a policy page with the Accept or Deny buttons. You need to click the Accept button to access the network.
- *Webconsent*—This is a combination of webauth and consent web authentication types. Herein, the embedded wireless controller presents a policy page with Accept or Deny buttons along with user name or password. You need to enter the correct credentials and click the Accept button to access the network.

Note

- You can view the webauth parameter-map information using the **show running-config** command output.
- The wireless Web-Authentication feature does not support the bypass type.
- Change in web authentication parameter map redirect login URL does not occur until a AP rejoin happens. You must enable and disable the WLAN to apply the new URL redirection.

Note We recommend that you follow the Cisco guidelines to create a customized web authentication login page. If you have upgraded to the latest versions of Google Chrome or Mozilla Firefox browsers, ensure that your webauth bundle has the following line in the *login.html* file:

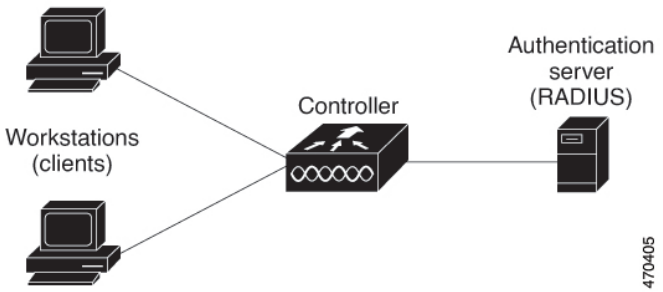
```
<body onload="loadAction();">
```

Device Roles

With local web authentication, the devices in the network have these specific roles:

- *Client*—The device (workstation) that requests access to the network and the controller and responds to requests from the controller. 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 controller that the client is authorized to access the network and the controller services or that the client is denied.
- *Controller*—Controls the physical access to the network based on the authentication status of the client. The controller acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client.

Figure 14: Local Web Authentication Device Roles



Authentication Process

When the page is hosted on the controller, the controller uses its virtual IP (a non-routable IP like 192.0.2.1 typically) to serve the request. If the page is hosted externally, the web redirection sends the client first to the virtual IP, which then sends the user again to the external login page while it adds arguments to the URL,

such as the location of the virtual IP. Even when the page is hosted externally, the user submits its credentials to the virtual IP.

When you enable local web authentication, these events occur:

- The user initiates an HTTP session.
- The HTTP traffic is intercepted, and authorization is initiated. The controller sends the login page to the user. The user enters a username and password, and the controller sends the entries to the authentication server.
- If the authentication succeeds, the controller 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 controller sends the login fail page. The user retries the login. If the maximum number of attempts fails, the controller 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 authentication server is not available, after the web authentication retries, the client moves to the excluded state and the client receives an Authentication Server is Unavailable page.
- The controller 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.
- Web authentication sessions can not apply new VLAN as part of the authorization policy, as the client already has been assigned an IP address and you will not be able to change the IP address in the client, in case the VLAN changes.
- If the terminate action is default, the session is dismantled, and the applied policy is removed.

Note Do not use semicolons (;) while configuring username for GUI access.

Local Web Authentication Banner

With Web Authentication, you can create a default and customized web-browser banners that appears when you log in to the controller.

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 as follows:

- Use the following global configuration command:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner ?
file <file-name>
text <Banner text>
title <Banner title>
```

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 15: Authentication Successful Banner

The banner can be customized as follows:

- Add a message, such as switch, router, or company name to the banner:
  - New-style mode—Use the following global configuration command:

    parameter-map type webauth global

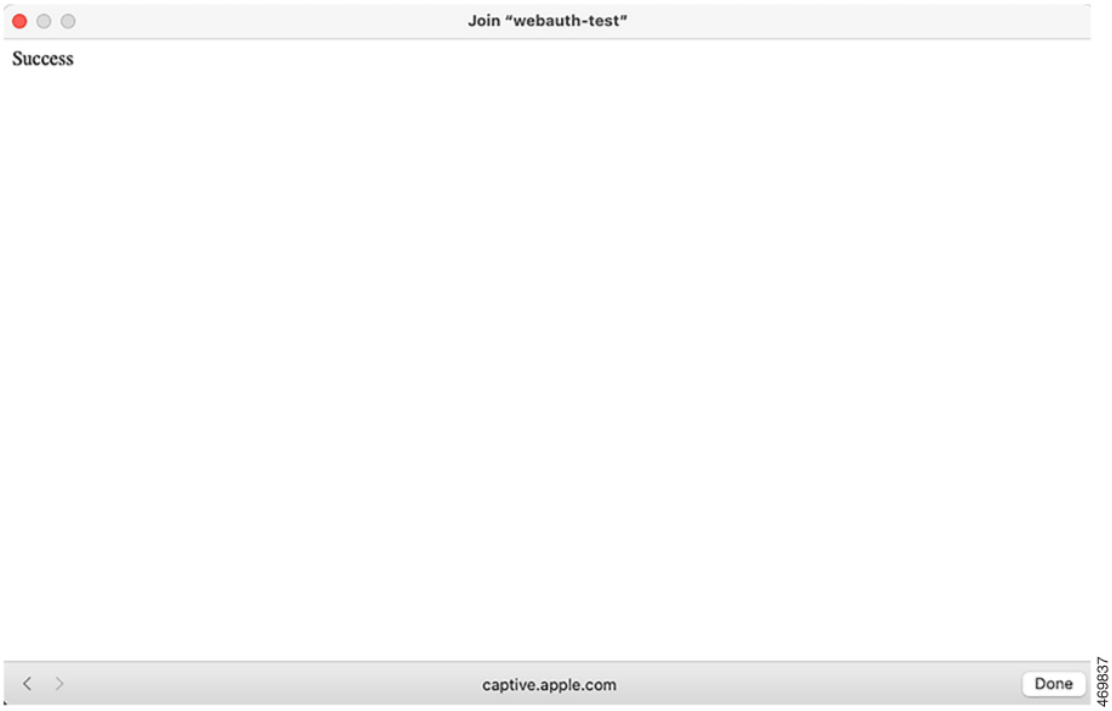
    banner text <text>

- Add a logo or text file to the banner:
  - New-style mode—Use the following global configuration command:

    parameter-map type webauth global

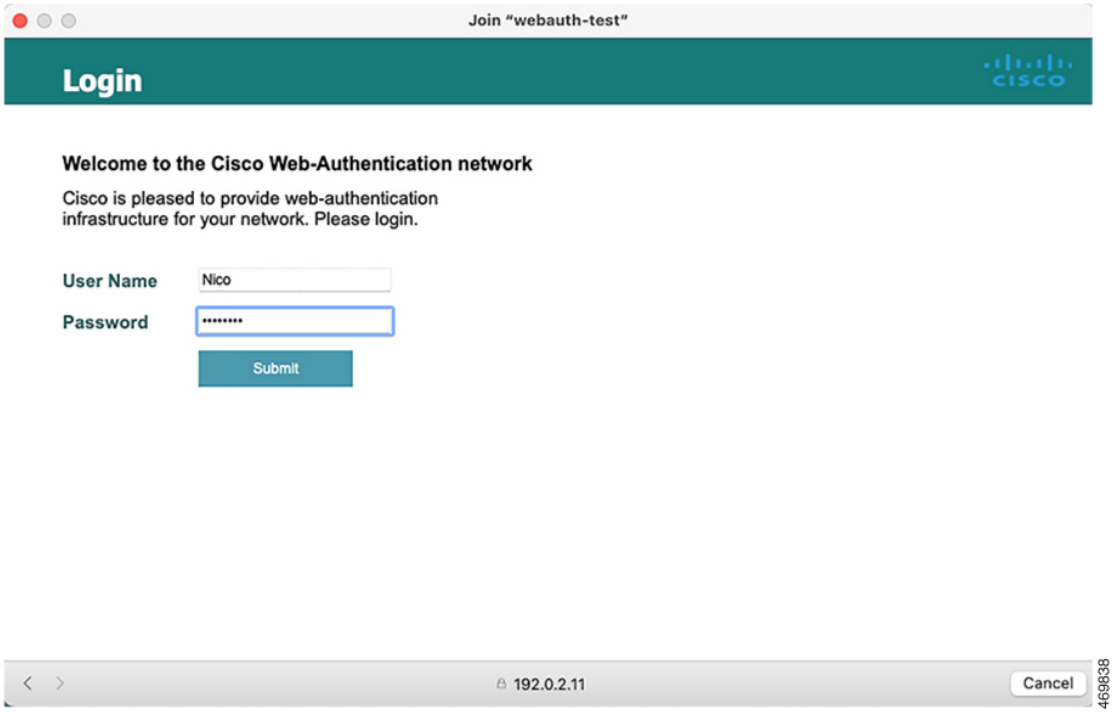
    banner file <filepath>

Figure 16: 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 17: Login Screen With No Banner



Customized Local Web Authentication

During the local web authentication process, the switch's 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

Note Virtual IP address is mandatory to configure custom web authentication.

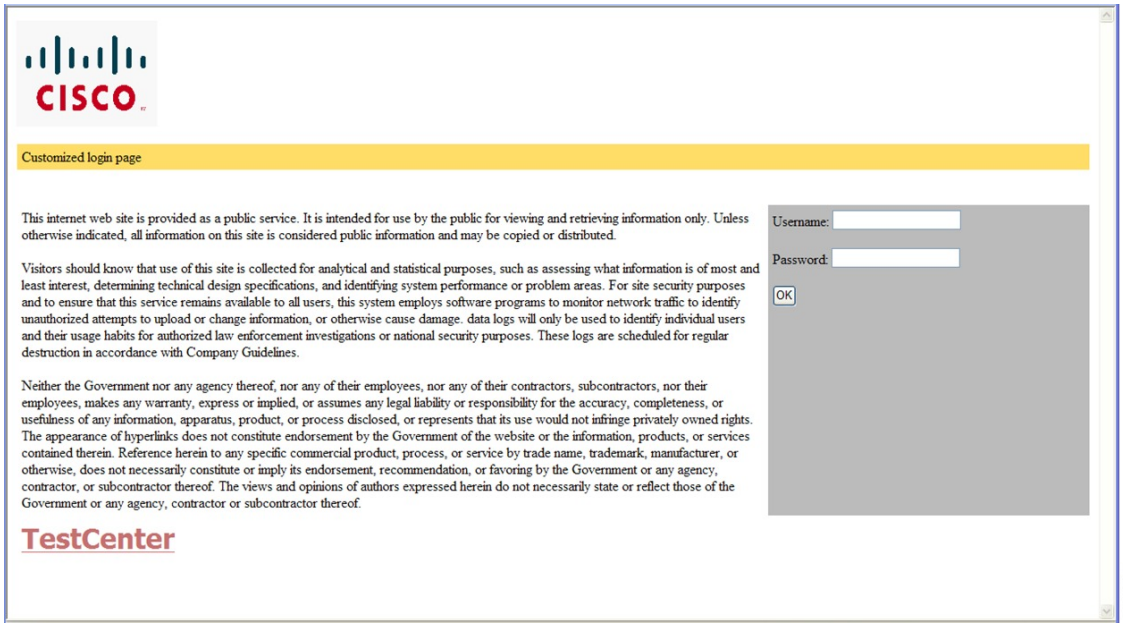
Guidelines

- You can substitute your own HTML pages for the default internal HTML pages.
- You can use a logo or specify text in the *login*, *success*, *failure*, and *expire* web pages.
- On the banner page, you can specify text in the login page.
- The pages are in HTML.

- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, http://www.cisco.com). An incomplete URL might cause *page not found* 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 custom page samples in the webauth bundle are provided with the image and the details of what you can and cannot change.
- 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.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the active switch or a member switch).
- You must configure all four pages.
- 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 are 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 18: Customizable Authentication Page



Redirection URL for Successful Login Guidelines

When configuring a redirection URL for successful login, consider these guidelines:

- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.
- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used
- To remove the specification of a redirection URL, use the **no** form of the command.
- If the redirection URL is required after the web-based authentication client is successfully authenticated, then the URL string must start with a valid URL (for example, http://) followed by the URL information. If only the URL is given without http://, then the redirection URL on successful authentication might cause page not found or similar errors on a web browser.

How to Configure Local Web Authentication

Configuring Default Local Web Authentication

The following table shows the default configurations required for local web authentication.

Table 18: Default Local Web Authentication Configuration

Feature	Default Setting
AAA	Disabled
RADIUS server • IP address • UDP authentication port • Key	• None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Disabled

Configuring AAA Authentication (GUI)

Note The WebUI does not support the ipv6 radius source-interface under AAA radius server group configuration.

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA**.

Step 2 In the **Authentication** section, click **Add**.

Step 3 In the **Quick Setup: AAA Authentication** window that is displayed, enter a name for your method list.

Step 4 Choose the type of authentication you want to perform before allowing access to the network, in the **Type** drop-down list.

Step 5 Choose if you want to assign a group of servers as your access server, or if you want to use a local server to authenticate access, from the **Group** Type drop-down list.

Step 6 To configure a local server to act as a fallback method when servers in the group are unavailable, check the **Fallback** to local check box.

Step 7 Choose the server groups you want to use to authenticate access to your network, from the **Available Server Groups** list and click > icon to move them to the **Assigned Server Groups** list.

Step 8 Click **Save & Apply to Device**.

Configuring AAA Authentication (CLI)

Procedure

	Command or Action	Purpose
Step 1	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables AAA functionality.
Step 2	**aaa authentication login** {**default** | *named_authentication_list*} **group** *AAA_group_name* **Example:** `Device(config)# aaa authentication login default group group1`	Defines the list of authentication methods at login. **named_authentication_list** refers to any name that is not greater than 31 characters. **AAA_group_name** refers to the server group name. You need to define the server-group **server_name** at the beginning itself.
Step 3	**aaa authorization network** {**default** | **named**} **group** *AAA_group_name* **Example:** `Device(config)# aaa authorization network default group group1`	Creates an authorization method list for web-based authorization.
Step 4	**tacacs-server host** {*hostname* | *ip_address*} **Example:** `Device(config)# tacacs-server host 10.1.1.1`	Specifies a AAA server.

Configuring the HTTP/HTTPS Server (GUI)

Procedure

Step 1 Choose **Administration** > **Management** > **HTTP/HTTPS/Netconf**.

Step 2 In the **HTTP/HTTPS Access Configuration** section, enable HTTP Access and enter the port that will listen for HTTP requests. The default port is 80. Valid values are 80, and ports between 1025 and 65535.

Step 3 Enable **HTTPS Access** on the device and enter the designated port to listen for HTTPS requests. The default port is 1025. Valid values are 443, and ports between 1025 and 65535. 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.

Step 4 Choose the **Personal Identity Verification** as enabled or disabled.

Step 5 In the **HTTP Trust Point Configuration** section, enable **Enable Trust Point** to use Certificate Authority servers as trustpoints.

Step 6 From the **Trust Points** drop-down list, choose a trust point.

Step 7 In the **Timeout Policy Configuration** section, enter the HTTP timeout policy in seconds. Valid values can range from 1 to 600 seconds.

Step 8 Enter the number of minutes of inactivity allowed before the session times out. Valid values can range from 180 to 1200 seconds.

Step 9 Enter the server life time in seconds. Valid values can range from 1 to 86400 seconds.

Step 10 Enter the maximum number of requests the device can accept. Valid values range from 1 to 86400 requests.

Step 11 Save the configuration.

Configuring the HTTP Server (CLI)

To use local web authentication, you must enable the HTTP server within the device. You can enable the server for either HTTP or HTTPS.

Note The Apple psuedo-browser will not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Follow the procedure given below to enable the server for either HTTP or HTTPS:

Procedure

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

	Command or Action	Purpose
Step 2	**ip http server** **Example:** `Device(config)# ip http server`	Enables the HTTP server. The local web authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 3	**ip http secure-server** **Example:** `Device(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:** `Device(config)# end`	Exits configuration mode.

Creating a Parameter Map (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 Click **Add**.

Step 3 Click **Policy Map**.

Step 4 Enter **Parameter Name**, **Maximum HTTP connections**, **Init-State Timeout(secs)** and choose **webauth** in the **Type** drop-down list.

Step 5 Click **Apply to Device**.

Configuring the Maximum Web Authentication Request Retries

Follow these steps to configure the maximum web authentication request retries:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wireless security web-auth retries** *number* **Example:** `Device(config)# wireless security web-auth retries 2`	*number* is the maximum number of web auth request retries. The valid range is 0 to 20.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring a Local Banner in Web Authentication Page (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

Step 3 In the **General** tab and choose the required Banner Type:

- If you choose **Banner Text**, enter the required banner text to be displayed.
- If you choose **File Name**, specify the path of the file from which the banner text has to be picked up.

Step 4 Click **Update & Apply**.

Configuring a Local Banner in Web Authentication Page (CLI)

Follow the procedure given below to configure a local banner in web authentication pages.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *param-map* **Example:** `Device(config)# parameter-map type webauth param-map`	Configures the web authentication parameters. Enters the parameter map configuration mode.
Step 3	**banner** [*file* | *banner-text* | *title*] **Example:** `Device(config-params-parameter-map)# banner http C My Switch C`	Enables the local banner. Create a custom banner by entering *C banner-text C* (where *C* is a delimiting character), or *file* that indicates a file (for example, a logo or text file) that appears in the banner, or *title* that indicates the title of the banner.
Step 4	**end** **Example:** `Device(config-params-parameter-map)# end`	Returns to privileged EXEC mode.

Configuration Examples for Local Web Authentication

Example: Obtaining Web Authentication Certificate

This example shows how to obtain web authentication certificate.

```
Device# configure terminal
Device(config)# crypto pki import cert pkcs12 tftp://9.1.0.100/ldapserver-cert.p12 cisco
Device(config)# end
Device# 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.
Device# 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.

```
Device# 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: Choosing the Default Web Authentication Login Page

This example shows how to choose a default web authentication login page.

```
Device# configure terminal
Device(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
Device(config)# wlan wlan50
Device(config-wlan)# shutdown
Device(config-wlan)# security web-auth authentication-list test
Device(config-wlan)# security web-auth parameter-map test
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# 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

Device# 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 IPv4 External Web Server

This example shows how to choose a customized web authentication login page from an IPv4 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv4 192.0.2.1.
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login http://9.1.0.100/login.html
Device(config-params-parameter-map)# redirect portal ipv4 9.1.0.100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv4 192.0.2.1.
parameter-map type webauth test
type webauth
redirect for-login http://9.1.0.100/login.html
redirect portal ipv4 9.1.0.100
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: Choosing a Customized Web Authentication Login Page from an IPv6 External Web Server

This example shows how to choose a customized web authentication login page from an IPv6 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv6 2001:DB8::/48
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login http://9:1:1::100/login.html
Device(config-params-parameter-map)# redirect portal ipv6 9:1:1::100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv6 2001:DB8::/48
parameter-map type webauth test
type webauth
redirect for-login http://9:1:1::100/login.html
redirect portal ipv6 9:1:1::100
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.

```
Device# configure terminal
Device(config)# parameter-map type webauth test
Device(config-params-parameter-map)# custom-page login device flash:loginsantosh.html
Device(config-params-parameter-map)# custom-page login expired device flash:loginexpire.html
Device(config-params-parameter-map)# custom-page failure device flash:loginfail.html
Device(config-params-parameter-map)# custom-page success device flash:loginsucess.html
Device(config-params-parameter-map)# end
Device# 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 9.1.0.100
 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 Preauthentication ACL

This example shows how to configure preauthentication ACL.

```
Device# configure terminal
Device(config)# wlan fff
Device(config-wlan)# shutdown
Device(config-wlan)# ip access-group web preauthrule
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# show wlan name fff
```

Example: Configuring Webpassthrough

This example shows how to configure webpassthrough.

```
Device# configure terminal
Device(config)# parameter-map type webauth webparalocal
Device(config-params-parameter-map)# type consent
Device(config-params-parameter-map)# end
Device# 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 9.1.0.100
```

Verifying Web Authentication Type

To verify the web authentication type, run the following command:

```
Device# show parameter-map type webauth all
Type Name
--------------------------------
Global global
Named webauth
Named ext
Named redirect
Named abc
```

```
Named glbal
Named ewa-2
```

```
Device# show parameter-map type webauth global
Parameter Map Name : global
Banner:
Text : CisCo
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
Consent Email : Disabled
Sleeping-Client : Enabled
Sleeping-Client timeout : 60 min
Virtual-ipv4 : 192.0.2.1.
Virtual-ipv4 hostname :
Webauth intercept https : Disabled
Webauth Captive Bypass : Disabled
Webauth bypass intercept ACL :
Trustpoint name :
HTTP Port : 80
Watch-list:
Enabled : no
Webauth login-auth-bypass:
```

```
Device# show parameter-map type webauth name global
Parameter Map Name : global
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
Consent Email : Disabled
Sleeping-Client : Disabled
Webauth login-auth-bypass:
```

External Web Authentication (EWA)

Configuring EWA with Single WebAuth Server Address and Default Ports (80/443) (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**aaa authentication login** **Example:** `Device(config)# aaa authentication login WEBAUTH local`	Defines the authentication method at login.

	Command or Action	Purpose
Step 3	**parameter-map type webauth** *parameter-map-name* **Example:** `Device(config)# parameter-map type webauth ISE-Ext-Webauth_IP`	Creates the parameter map. The *parameter-map-name* must not exceed 99 characters.
Step 4	**type webauth** **Example:** `Device(config-params-parameter-map)# type webauth`	Configures the webauth type parameter.
Step 5	**redirect for-login** *URL-String* **Example:** `Device(config-params-parameter-map)# redirect for-login https://192.168.0.98:8443/portal/PortalSetup.action?portal=[illegible]`	Configures the URL string for redirect during login.
Step 6	**redirect portal ipv4** *ip-address* **Example:** `Device(config-params-parameter-map)# redirect portal ipv4 192.168.0.98`	Configures the external portal IPv4 address.
Step 7	**exit** **Example:** `Device(config-params-parameter-map)# exit`	Returns to global configuration mode.
Step 8	**wlan** *wlan-name wlan-id SSID-name* **Example:** `Device(config)#  wlan EWLC3-GUEST 3 EWLC3-GUEST`	Configures a WLAN.
Step 9	**no security ft adaptive** **Example:** `Device(config-wlan)# no security ft adaptive`	Disables adaptive 11r.
Step 10	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disables WPA security.
Step 11	**no security wpa wpa2** **Example:** `Device(config-wlan)# no security wpa wpa2`	Disables WPA2 security.

	Command or Action	Purpose
Step 12	**no security wpa wpa2 ciphers aes** **Example:** `Device(config-wlan)# no security wpa wpa2 ciphers aes`	Disables WPA2 ciphers for AES.
Step 13	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 14	**security web-auth** **Example:** `Device(config-wlan)# security web-auth`	Enables web authentication for WLAN.
Step 15	**security web-auth authentication-list** *authenticate-list-name* **Example:** `Device(config-wlan)# security web-auth authentication-list WEBAUTH`	Enables authentication list for dot1x security.
Step 16	**security web-auth parameter-map** *parameter-map-name* **Example:** `Device(config-wlan)# security web-auth parameter-map ISE-Ext-Webauth_IP`	Configures the parameter map. **Note** If parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.
Step 17	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.

Configuring EWA with Multiple Web Servers and/or Ports Different than Default (80/443)

Procedure

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

	Command or Action	Purpose
Step 2	**ip access-list extended** *name* **Example:** `Device(config)# ip access-list extended preauth_ISE_Ext_WA`	Defines an extended IPv4 access list using a name, and enters access-list configuration mode.
Step 3	*access-list-number* **permit tcp any host** *external_web_server_ip_address1* **eq** *port-number* **Example:** `Device(config)# 10 permit tcp any host 192.168.0.98 eq 8443`	Permits access from any host to the external web server port number 8443.
Step 4	*access-list-number* **permit tcp any host** *external_web_server_ip_address2* **eq** *port-number* **Example:** `Device(config)# 10 permit tcp any host 192.168.0.99 eq 8443`	Permits access from any host to the external web server port number 8443.
Step 5	*access-list-number* **permit udp any any eq domain** **Example:** `Device(config)# 20 permit udp any any eq domain`	Permits DNS UDP traffic.
Step 6	*access-list-number* **permit udp any any eq bootpc** **Example:** `Device(config)# 30 permit udp any any eq bootpc`	Permits DHCP traffic.
Step 7	*access-list-number* **permit udp any any eq bootps** **Example:** `Device(config)# 40 permit udp any any eq bootps`	Permits DHCP traffic.
Step 8	*access-list-number* **permit tcp host** *external_web_server_ip_address1* **eq** *port_number* **any** **Example:** `Device(config)# 50 permit tcp host 192.168.0.98 eq 8443 any`	Permits the access from the external web server port 8443 to any host.
Step 9	*access-list-number* **permit tcp host** *external_web_server_ip_address2* **eq** *port_number* **any**	Permits the access from the external web server port 8443 to any host.

	Command or Action	Purpose
	Example: `Device(config)# 50 permit tcp host 192.168.0.99 eq 8443 any`	
Step 10	*access-list-number* **permit tcp any any eq domain** **Example:** `Device(config)# 60 permit tcp any any eq domain`	Permits the DNS TCP traffic.
Step 11	*access-list-number* **deny ip any any** **Example:** `Device(config)# 70 deny ip any any`	Denies all the other traffic.
Step 12	**wlan** *wlan-name wlan-id ssid* **Example:** `Device(config)# wlan EWLC3-GUEST 3 EWLC3-GUEST`	Creates the WLAN.
Step 13	**ip access-group web** *name* **Example:** `Device(config-wlan)# ip access-group web preauth_ISE_Ext_WA`	Configures the IPv4 WLAN web ACL. The variable *name* specifies the user-defined IPv4 ACL name.
Step 14	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.

Configuring Wired Guest EWA with Multiple Web Servers and/or Ports Different than Default (80/443)

Before you begin

You cannot assign a manual ACL to a wired guest LAN configuration. The workaround is to use the bypass ACL in the global parameter map.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ip access-list extended** *name* **Example:**	Defines an extended IPv4 access list using a name, and enters access-list configuration mode.

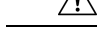
	Command or Action	Purpose
	`Device(config)# ip access-list extended BYPASS_ACL`	
Step 3	*access-list-number* **deny ip any host** *hostname* **Example:** `Device(config)# 10 deny ip any host 192.168.0.45`	Allows the traffic to switch centrally.
Step 4	*access-list-number* **deny ip any host** *hostname* **Example:** `Device(config)# 20 deny ip any host 4.0.0.1`	Allows the traffic to switch centrally.
Step 5	**parameter-map type webauth global** **Example:** `Device(config)# parameter-map type webauth global`	Creates a parameter map and enters parameter-map webauth configuration mode.
Step 6	**webauth-bypass-intercept** *name* **Example:** `Device(config-params-parameter-map)# webauth-bypass-intercept BYPASS_ACL`	Creates a WebAuth bypass intercept using the ACL name. **Note** You cannot apply a manual ACL to the wired guest profile and configure an external web authentication with multiple IP addresses or different ports. The workaround is to use the bypass ACL for wired guest profile.
Step 7	**end** **Example:** `Device(config-params-parameter-map)# end`	Returns to privileged EXEC mode.

Authentication for Sleeping Clients

Information About Authenticating Sleeping Clients

Clients with guest access that have had successful web authentication are allowed to sleep and wake up without having to go through another authentication process through the login page. You can configure the duration for which sleeping clients should be remembered for before reauthentication becomes necessary. The valid range is 10 minutes to 43200 minutes, with the default being 720 minutes. You can also configure this duration on WebAuth parameter map that is mapped to a WLAN. Note that the sleeping client timer comes into effect due to instances such as idle timeout, session timeout, disabling of the WLAN, and the AP being nonoperational.

This feature is supported in the following FlexConnect scenario: local switching and central authentication.



Caution If the MAC address of a client that goes to sleep mode is spoofed, the fake device such as a laptop can be authenticated.

Mobility Scenarios

Following are some guidelines in a mobility scenario:

- L2 roaming in the same subnet is supported.
- Anchor sleeping timer is applicable.
- The sleeping client information is shared between multiple autoanchors when a sleeping client moves from one anchor to another.

A sleeping client does not require reauthentication in the following scenarios:

- Suppose there are two embedded wireless controllers in a mobility group. A client that is associated with one embedded wireless controller goes to sleep and then wakes up and gets associated with the other embedded wireless controller.
- Suppose there are three embedded wireless controllers in a mobility group. A client that is associated with the second embedded wireless controller that is anchored to the first controller goes to sleep, wakes up, and gets associated with the third embedded wireless controller.
- A client sleeps, wakes up and gets associated with the same or different export foreign embedded wireless controller that is anchored to the export anchor.

Restrictions on Authenticating Sleeping Clients

- The sleep client feature works only for WLAN configured with WebAuth security.
- You can configure the sleeping clients only on a per WebAuth parameter-map basis.
- The authentication of sleeping clients feature is supported only on WLANs that have Layer 3 security enabled.
- With Layer 3 security, the Authentication, Passthrough, and On MAC Filter failure web policies are supported. The Conditional Web Redirect and Splash Page Web Redirect web policies are not supported.
- The central web authentication of sleeping clients is not supported.
- The authentication of sleeping clients feature is not supported on guest LANs and remote LANs.
- A guest access sleeping client that has a local user policy is not supported. In this case, the WLAN-specific timer is applied.

Configuring Authentication for Sleeping Clients (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

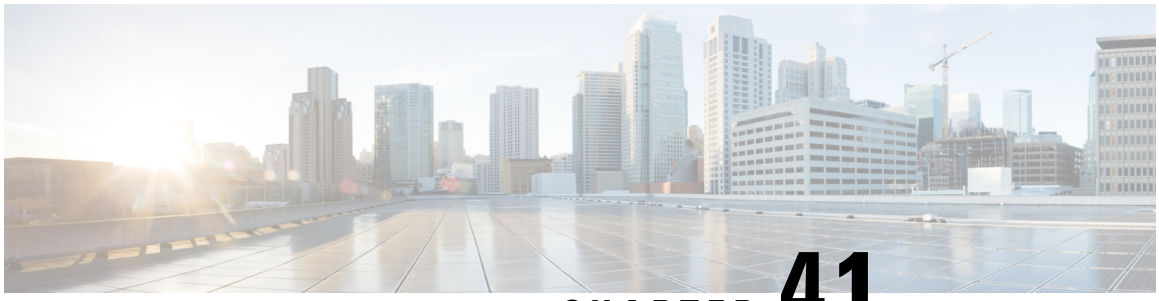
Step 3 Select **Sleeping Client Status** check box.

Step 4 Click **Update & Apply to Device**.

Configuring Authentication for Sleeping Clients (CLI)

Procedure

	Command or Action	Purpose
Step 1	**[no] parameter-map type webauth** {*parameter-map-name* | **global**} **Example:** `Device(config)# parameter-map type webauth global`	Creates a parameter map and enters parameter-map webauth configuration mode.
Step 2	**sleeping-client** [**timeout** *time*] **Example:** `Device(config-params-parameter-map)# sleeping-client timeout 100`	Configures the sleeping client timeout to 100 minutes. Valid range is between 10 minutes and 43200 minutes. **Note** If you do not use the timeout keyword, the sleeping client is configured with the default timeout value of 720 minutes.
Step 3	**end**	Exits parameter-map webauth configuration mode and returns to privileged EXEC mode.
Step 4	(Optional) **show wireless client sleeping-client** **Example:** `Device# show wireless client sleeping-client`	Shows the MAC address of the clients and the time remaining in their respective sessions.
Step 5	(Optional) **clear wireless client sleeping-client** [**mac-address** *mac-addr*] **Example:** `Device# clear wireless client sleeping-client mac-address 00e1.e1e1.0001`	• **clear wireless client sleeping-client**—Deletes all sleeping client entries from the sleeping client cache. • **clear wireless client sleeping-client mac-address** *mac-addr*—Deletes the specific MAC entry from the sleeping client cache.



CHAPTER 41

Central Web Authentication

Information About Central Web Authentication

Central web authentication offers the possibility to have a central device that acts as a web portal (in this example, the ISE). The major difference compared to the usual local web authentication is that it is shifted to Layer 2 along with MAC filtering or dot1x authentication. The concept also differs in that the radius server (ISE in this example) returns special attributes that indicate to the switch that a web redirection must occur. This solution eliminates any delay to start the web authentication.

The following are the different types of web authentication methods:

- Local Web Authentication (LWA): Configured as Layer 3 security on the controller, the web authentication page and the pre-authentication ACL are locally configured on the controller. The controller intercepts htttp(s) traffic and redirects the client to the internal web page for authentication. The credentials entered by the client on the login page is authenticated by the controller locally or through a RADIUS or LDAP server.
- External Web Authentication (EWA): Configured as Layer 3 security on the controller, the controller intercepts htttp(s) traffic and redirects the client to the login page hosted on the external web server. The credentials entered by the client on the login page is authenticated by the controller locally or through a RADIUS or LDAP server. The pre-authentication ACL is configured statically on the controller.
- Central Web Authentication (CWA): Configured mostly as Layer 2 security on the controller, the redirection URL and the pre-authentication ACL reside on ISE and are pushed during layer 2 authentication to the controller. The controller redirects all web traffic from the client to the ISE login page. ISE validates the credentials entered by the client through HTTPS and authenticates the user.

Globally, if the MAC address of the client station is not known by the radius server (but other criteria can also be used), the server returns the redirection attributes, and the embedded wireless controller authorizes the station (using the MAC filtering) but places an access list to redirect the web traffic to the portal.

Once the user logs into the guest portal, it is possible to re-authenticate the client so that a new Layer 2 MAC filtering occurs using the Change of Authorization (CoA). This way, the ISE remembers that it was a webauth

user and pushes the necessary authorization attributes to the embedded wireless controller for accessing the network.

Note In Central Web Authentication (CWA) with dual VLAN posture scenario, Cisco AireOS and IOS-XE controller performs 2 and 3 EAPOL handshakes respectively. If a client is stuck in a quarantine VLAN because of any break in EAPOL handshake due to client or network issue, you need to analyze the client or network issue.

However, you can manually disconnect or reconnect the client to come out of the quarantine loop (or) click the Scan Again on AnyConnect (Or) enable posture lease (Or) use the ISE posture sync feature.

Prerequisites for Central Web Authentication

- Cisco Identity Services Engine (ISE)

How to Configure ISE

To configure ISE, proceed as follows:

1. Create an authorization profile.
2. Create an authentication rule.
3. Create an authorization rule.

Creating an Authorization Profile

Procedure

Step 1	Click **Policy**, and click **Policy Elements**.
Step 2	Click **Results**.
Step 3	Expand **Authorization**, and click **Authorization Profiles**.
Step 4	Click **Add** to create a new authorization profile for central webauth.
Step 5	In the **Name** field, enter a name for the profile. For example, CentralWebauth.
Step 6	Choose **ACCESS_ACCEPT** from the Access Type drop-down list.
Step 7	Check the **Web Redirection (CWA, MDM, NSP, CPP)** check box, and choose **Centralized Web Auth** from the drop-down list.
Step 8	In the **ACL** field, enter the name of the ACL that defines the traffic to be redirected. For example, redirect.
Step 9	In the **Value** field, choose the default or customized values. The Value attribute defines whether the ISE sees the default or a custom web portal that the ISE admin created.
Step 10	Click **Save**.

Creating an Authentication Rule

Follow the procedure given below to use the authentication profile and create the authentication rule:

Procedure

Step 1 In the **Policy** > **Authentication** page, click **Authentication**.

Step 2 Enter a name for your authentication rule. For example, MAB.

Step 3 In the If condition field, select the plus (+) icon.

Step 4 Choose **Compound condition**, and choose **Wireless_MAB**.

Step 5 Click the arrow located next to **and ...** in order to expand the rule further.

Step 6 Click the + icon in the Identity Source field, and choose **Internal endpoints**.

Step 7 Choose **Continue** from the 'If user not found' drop-down list.

This option allows a device to be authenticated even if its MAC address is not known.

Step 8 Click **Save**.

Creating an Authorization Rule

You can configure many rules in the authorization policy. The *MAC not known* rule is configured in this section:

Procedure

Step 1 Click **Policy** > **Authorization**.

Step 2 In the Rule Name field, enter a name. For example: *Mac not known*.

Step 3 In the Conditions field, click the plus (+) icon.

Step 4 Choose **Compound Conditions**, and choose **Wireless_MAB**.

Step 5 From the settings icon, select **Add Attribute/Value** from the options.

Step 6 In the Description field, choose **Network Access** > **AuthenticationStatus** as the attribute from the drop-down list.

Step 7 Choose the **Equals** operator.

Step 8 From the right-hand field, choose **UnknownUser**.

Step 9 In the Permissions field, choose the authorization profile name that you had created earlier.

The ISE continues even though the user (or MAC) is not known.

Unknown users are now presented with the Login page. However, once they enter their credentials, they are presented again with an authentication request on the ISE; therefore, another rule must be configured with a condition that is met if the user is a guest user. For example, if UseridentityGroup Equals Guest is used then it is assumed that all guests belong to this group.

Step 10 In the Conditions field, click the plus (+) icon.

Step 11 Choose **Compound Conditions**, and choose to create a new condition.

The new rule must come before the *MAC not known* rule.

Step 12 From the settings icon, select **Add Attribute/Value** from the options.

Step 13 In the Description field, choose **Network Access** > **UseCase** as the attribute from the drop-down list.

Step 14 Choose the **Equals** operator.

Step 15 From the right-hand field, choose **GuestFlow**.

Step 16 In the Permissions field, click the plus (+) icon to select a result for your rule.

You can choose **Standard** > **PermitAccess** option or create a custom profile to return the attributes that you like.

When the user is authorized on the login page, the ISE triggers a COA that results in the restart of Layer 2 authentication. When the user is identified as a guest user, the user is authorized.

How to Configure Central Web Authentication on the Controller

To configure central web authentication on the controller, proceed as follows:

1. Configure WLAN.
2. Configure policy profile.
3. Configure redirect ACL.
4. Configure AAA for central web authentication.
5. Configure redirect ACL in Flex profile.

Configuring WLAN (GUI)

Before you begin

You need to enable MAC filtering for Layer 2 authentication to download the redirect URL and ACL.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 In the **WLANs** window, click the name of the **WLAN** or click **Add** to create a new one.

Step 3 In the **Add/Edit WLAN** window that is displayed, click the **General** tab to configure the following parameters.

- In the **Profile Name** field, enter or edit the name of the profile.
- In the **SSID** field, enter or edit the SSID name.

  The SSID name can be alphanumeric, and up to 32 characters in length.
- In the **WLAN ID** field, enter or edit the ID number.The valid range is between 1 and 512.
- From the **Radio Policy** drop-down list, choose the **802.11** radio band.

- Using the **Broadcast SSID** toggle button, change the status to either **Enabled** or **Disabled** .
- Using the **Status** toggle button, change the status to either **Enabled** or **Disabled** .

Step 4 Click the **Security** tab, and then **Layer 2** tab to configre the following parameters:

- From the **Layer 2 Security Mode** drop-down list, choose **None**. .This setting disables Layer 2 security.
- Enter the **Reassociation Timeout** value, in seconds. This is the time after which a fast transition reassociation times out.
- Check the **Over the DS** check box to enable Fast Transition over a distributed system.
- Choose OWE, Opportunistic Wireless Encryption (OWE) provides data confidentiality with encryption over the air between an AP radio and a wireless client. OWE Transition Mode is meant to provide a sort of backwards compatibility.
- Choose Fast Transition, 802.11r which is the IEEE standard for fast roaming, introduces a new concept of roaming where the initial handshake with a new AP is done even before the corresponding client roams to the target access point. This concept is called Fast Transition.
- Check the check box to enable MAC filtering in the WLAN.

Step 5 Click **Save & Apply to Device**.

Configuring WLAN (CLI)

Note You need to enable MAC filtering for Layer 2 authentication to download the redirect URL and ACL.

After completing the WLAN configuration, if the changes are not pushed to all the APs, the following syslog message appears:

2021/01/06 16:20:00.597927186 {wncd_x_R0-4}{1}: [wlanmgr-db] [20583]: UUID: 0, ra: 0, TID: 0 (note): Unable to push WLAN config changes to all APs, cleanup required for WlanId: 2, profile: wlan1 state: Delete pending

If the above mentioned syslog message appears for more than six minutes, reload the controller.

If the controller does not reload and still the syslog message appears, then collect the archive logs, wncd core file, and raise a case by clicking the following link: Support Case Manager.

Procedure

	Command or Action	Purpose
Step 1	**wlan `wlan-name wlan-id SSID-name`** **Example:** `Device(config)# wlan wlanProfileName 1 ngwcSSID`	Enters the WLAN configuration sub-mode. **wlan-name** is the name of the configured WLAN. **wlan-id** is the wireless LAN identifier. The range is 1 to 512.

	Command or Action	Purpose
		SSID-name is the SSID name which can contain 32 alphanumeric characters. **Note** If you have already configured this command, enter **wlan wlan-name** command.
Step 2	**mac-filtering** [*name*] **Example:** `Device(config-wlan)# mac-filtering name`	Enables MAC filtering on a WLAN. **Note** While configuring mac-filtering the default authentication list is considered, if the authentication list is not configured earlier.
Step 3	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disable WPA security.
Step 4	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.
Step 5	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.

Example

```
Device# config terminal
Device(config)# wlan wlanProfileName 1 ngwcSSID
Device(config-wlan)# mac-filtering default
Device(config-wlan)# no security wpa
Device(config-wlan)# no shutdown
Device(config-wlan)# end
```

Configuring Policy Profile (CLI)

Note You need a AAA override to apply policies coming from the AAA or ISE servers. When a redirect URL and redirect ACL is received from the ISE server, NAC is used to trigger the Central Web Authentication (CWA).

Both NAC and AAA override must be available in the policy profile to which the client is being associated.

The default policy profile is associated to an AP, if the AP is not associated to any other policy profiles.

Procedure

	Command or Action	Purpose
Step 1	**wireless profile policy default-policy-profile** **Example:** `Device(config)# wireless profile policy default-policy-profile`	Sets the policy profile.
Step 2	**vlan `vlan-id`** **Example:** `Device(config-wireless-policy)# vlan 41`	Maps the VLAN to a policy profile. If vlan-id is not specified, the default native vlan 1 is applied. The valid range for vlan-id is 1 to 4096. Management VLAN is applied if no VLAN is configured on the policy profile.
Step 3	**aaa-override** **Example:** `Device(config-wireless-policy)# aaa-override`	Configures AAA override to apply policies coming from the AAA or ISE servers.
Step 4	**nac** **Example:** `Device(config-wireless-policy)# nac`	Configures Network Access Control in the policy profile. NAC is used to trigger the Central Web Authentication (CWA).
Step 5	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Enables the WLAN.
Step 6	**end** **Example:** `Device(config-wireless-policy)# end`	Returns to privileged EXEC mode.

Example

```
Device# configure terminal
Device(config)# wireless profile policy default-policy-profile
Device(config-wireless-policy)# vlan 41
Device(config-wireless-policy)# aaa-override
Device(config-wireless-policy)# nac
Device(config-wireless-policy)# no shutdown
Device(config-wireless-policy)# end
```

Configuring a Policy Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the **Policy Profile** page, click **Add**.

Step 3 In the **Add Policy Profile** window, in General Tab, enter a name and description for the policy profile.

Step 4 To enable the policy profile, set **Status** as Enabled.

Step 5 Use the slider to enable or disable **Passive Client** and **Encrypted Traffic Analytics**.

Step 6 (Optional) In the **CTS Policy** section, choose the appropriate status for the following:

- Inline Tagging—a transport mechanism using which a embedded wireless controller or access point understands the source SGT.
- SGACL Enforcement

Step 7 Specify a default **SGT**. The valid range is from 2 to 65519.

Step 8 In the WLAN Switching Policy section, choose the following, as required:

- Central Switching
- Central Authentication
- Central DHCP
- Central Association Enable
- Flex NAT/PAT

Step 9 Click **Save & Apply to Device**.

Creating Redirect ACL

The redirect ACL is a punt ACL that needs to be predefined on the controller (or the AP in case of FlexConnect local switching): the AAA server returns the name of the ACL and not its definition. The redirect ACL defines traffic (matching “deny”statements, as it denies redirection for it) that will be allowed through on the data plane and traffic (matching “permit” statements) that will be sent to the control plane towards the CPU for further processing (that is, the web interception and redirection in this case). The ACL has implicit (that is, the invisible) statements allowing DHCP and DNS traffic towards all IPs, just like it is the case with LWA. It also ends with a statement that a security ACL implicit deny.

Procedure

	Command or Action	Purpose
Step 1	**ip access-list extended redirect** **Example:**	The HTTP and HTTPS browsing does not work without authentication (per the other ACL) as

	Command or Action	Purpose
	`Device(config)# ip access-list extended redirect`	ISE is configured to use a redirect ACL (named **`redirect`**).
Step 2	**deny ip any host `ISE-IP-add`** **Example:** `Device(config)# deny ip any host 123.123.134.112`	Allows traffic to ISE and all other traffic is blocked.
Step 3	**deny ip host `ISE-IP-add` any** **Example:** `Device(config)# deny ip host 123.123.134.112 any`	Allows traffic to ISE and all other traffic is blocked. **Note** This ACL is applicable for both local and flex mode.

Step 4	**permit TCP any any eq `web address/port-number`**
	Example:
	In case of HTTP:
	`Device(config)# permit TCP any any eq www`
	`Device(config)# permit TCP any any eq 80`
	Example:
	In case of HTTPS:
	`Device(config)# permit TCP any any eq 443`
		Redirects all HTTP or HTTPS access to the ISE login page. port-number 80 is used for HTTP and port-number 443 is used for HTTPS.
		For the ACE to allow traffic to ISE, ISE should be configured above the HTTP/HTTPS ACE.

Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring AAA for Central Web Authentication

Procedure

	Command or Action	Purpose
Step 1	**aaa server radius dynamic-author** **Example:** `Device(config)# aaa server radius dynamic-author`	Configures the Change of Authorization (CoA) on the embedded wireless controller.
Step 2	**client `ISE-IP-add` server-key `radius-shared-secret`** **Example:**	Specifies a RADIUS client and the RADIUS key to be shared between a device and a RADIUS client.

	Command or Action	Purpose
	`Device(config-locsvr-da-radius)# client 123.123.134.112 server-key 0 SECRET`	**ISE-IP-add** is the IP address of the RADIUS client. **server-key** is the radius client server-key. **radius-shared-secret** covers the following: • **0**—Specifies unencrypted key. • **6**—Specifies encrypted key. • **7**—Specifies HIDDEN key. • **Word**—Unencrypted (cleartext) server key. The RADIUS shared secret should not exceed 240 characters while configuring WSMA data in GUI. **Note** All these steps work only if the AAA configuration is in place. See the *Configuring AAA Authentication* for details.

Example

```
Device# config terminal
Device(config)# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 123.123.134.112 server-key 0 SECRET
Device(config-locsvr-da-radius)# end
```

Configuring Redirect ACL in Flex Profile (GUI)

The redirect ACL definition must be sent to the access point in the FlexConnect profile. For this, the redirect ACL associated with an AP must be configured in the FlexConnect profile where the client is hosted. If an access point is not configured with any of the FlexConnect profiles, the default FlexConnect profile is associated with it.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Flex**.

Step 2 On the **Flex Profile** page, click the name of the FlexConnect profile or click **Add** to create a new FlexConnect profile.

Step 3 In the **Add/Edit Flex Profile** window that is displayed, click the **Policy ACL** tab.

Step 4 Click **Add** to map an ACL to the FlexConnect profile.

Step 5 Choose the ACL name, enable central web authentication, and specify the preauthentication URL filter.

Step 6 Click **Save**.

Step 7 Click **Update & Apply to Device**.

Configuring Redirect ACL in Flex Profile (CLI)

The redirect ACL definition must be sent to the access point in the Flex profile. For this, the redirect ACL associated to an AP must be configured in the Flex profile where the client is being hosted. If an access point is not configured with any of the Flex profiles, the default Flex profile is associated with it.

Procedure

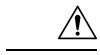
	Command or Action	Purpose
Step 1	**wireless profile flex default-flex-profile** **Example:** `Device(config)# wireless profile flex default-flex-profile`	Creates a new flex policy. The default flex profile name is **`default-flex-profile`**.
Step 2	**acl-policy** *acl policy name* **Example:** `Device(config-wireless-flex-profile)# acl-policy acl1`	Configures ACL policy.
Step 3	**central-webauth** **Example:** `Device(config-wireless-flex-profile-acl)# central-webauth`	Configures central web authentication.
Step 4	**end** **Example:** `Device(config-wireless-flex-profile-acl)# end`	Returns to privileged EXEC mode.

Authentication for Sleeping Clients

Information About Authenticating Sleeping Clients

Clients with guest access that have had successful web authentication are allowed to sleep and wake up without having to go through another authentication process through the login page. You can configure the duration for which sleeping clients should be remembered for before reauthentication becomes necessary. The valid range is 10 minutes to 43200 minutes, with the default being 720 minutes. You can also configure this duration on WebAuth parameter map that is mapped to a WLAN. Note that the sleeping client timer comes into effect due to instances such as idle timeout, session timeout, disabling of the WLAN, and the AP being nonoperational.

This feature is supported in the following FlexConnect scenario: local switching and central authentication.



Caution If the MAC address of a client that goes to sleep mode is spoofed, the fake device such as a laptop can be authenticated.

Mobility Scenarios

Following are some guidelines in a mobility scenario:

- L2 roaming in the same subnet is supported.
- Anchor sleeping timer is applicable.
- The sleeping client information is shared between multiple autoanchors when a sleeping client moves from one anchor to another.

A sleeping client does not require reauthentication in the following scenarios:

- Suppose there are two embedded wireless controllers in a mobility group. A client that is associated with one embedded wireless controller goes to sleep and then wakes up and gets associated with the other embedded wireless controller.
- Suppose there are three embedded wireless controllers in a mobility group. A client that is associated with the second embedded wireless controller that is anchored to the first controller goes to sleep, wakes up, and gets associated with the third embedded wireless controller.
- A client sleeps, wakes up and gets associated with the same or different export foreign embedded wireless controller that is anchored to the export anchor.

Restrictions on Authenticating Sleeping Clients

- The sleep client feature works only for WLAN configured with WebAuth security.
- You can configure the sleeping clients only on a per WebAuth parameter-map basis.
- The authentication of sleeping clients feature is supported only on WLANs that have Layer 3 security enabled.
- With Layer 3 security, the Authentication, Passthrough, and On MAC Filter failure web policies are supported. The Conditional Web Redirect and Splash Page Web Redirect web policies are not supported.
- The central web authentication of sleeping clients is not supported.
- The authentication of sleeping clients feature is not supported on guest LANs and remote LANs.
- A guest access sleeping client that has a local user policy is not supported. In this case, the WLAN-specific timer is applied.

Configuring Authentication for Sleeping Clients (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

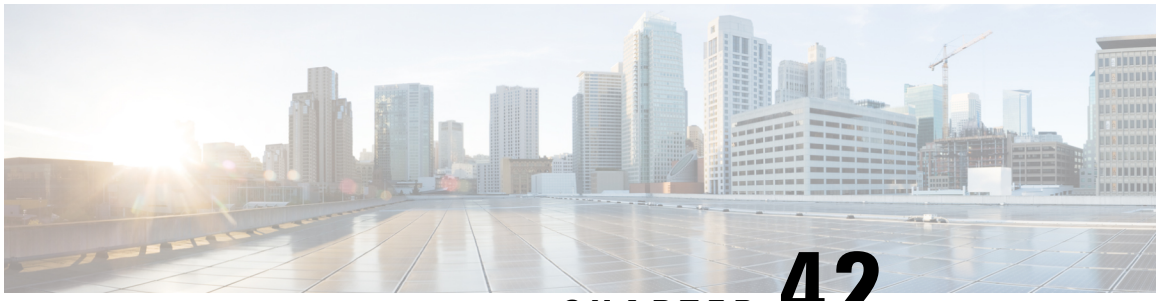
Step 3 Select **Sleeping Client Status** check box.

Step 4 Click **Update & Apply to Device**.

Configuring Authentication for Sleeping Clients (CLI)

Procedure

	Command or Action	Purpose
Step 1	**[no] parameter-map type webauth** {*parameter-map-name* | **global**} **Example:** `Device(config)# parameter-map type webauth global`	Creates a parameter map and enters parameter-map webauth configuration mode.
Step 2	**sleeping-client** [**timeout** *time*] **Example:** `Device(config-params-parameter-map)# sleeping-client timeout 100`	Configures the sleeping client timeout to 100 minutes. Valid range is between 10 minutes and 43200 minutes. **Note** If you do not use the timeout keyword, the sleeping client is configured with the default timeout value of 720 minutes.
Step 3	**end**	Exits parameter-map webauth configuration mode and returns to privileged EXEC mode.
Step 4	(Optional) **show wireless client sleeping-client** **Example:** `Device# show wireless client sleeping-client`	Shows the MAC address of the clients and the time remaining in their respective sessions.
Step 5	(Optional) **clear wireless client sleeping-client** [**mac-address** *mac-addr*] **Example:** `Device# clear wireless client sleeping-client mac-address 00e1.e1e1.0001`	• **clear wireless client sleeping-client**—Deletes all sleeping client entries from the sleeping client cache. • **clear wireless client sleeping-client mac-address** *mac-addr*—Deletes the specific MAC entry from the sleeping client cache.



CHAPTER 42

ISE Simplification and Enhancements

Utilities for Configuring Security

This chapter describes how to configure all the RADIUS server side configuration using the following command:

wireless-default radius server *ip* **key** *secret*

This simplified configuration option provides the following:

- Configures AAA authorization for network services, authentication for web auth and Dot1x.
- Enables local authentication with default authorization.
- Configures the default redirect ACL for CWA.
- Creates global parameter map with virtual IP and enables captive bypass portal.
- Configures all the AAA configuration for a default case while configuring the RADIUS server.
- The method-list configuration is assumed by default on the WLAN.
- Enables the radius accounting by default.
- Disables the radius aggressive failovers by default.
- Sets the radius request timeouts to 5 seconds by default.
- Enables captive bypass portal.

This command configures the following in the background:

```
aaa new-model
aaa authentication webauth default group radius
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa accounting identity default start-stop group radius
!
aaa server radius dynamic-author
```

```
 client <IP> server-key cisco123
!
radius server RAD_SRV_DEF_<IP>
 description Configured by wireless-default
 address ipv4 <IP> auth-port 1812 acct-port 1813
 key <key>
!
aaa local authentication default authorization default
aaa session-id common
!
ip access-list extended CISCO-CWA-URL-REDIRECT-ACL-DEFAULT
remark " CWA ACL to be referenced from ISE "
deny udp any any eq domain
deny tcp any any eq domain
deny udp any eq bootps any
deny udp any any eq bootpc
deny udp any eq bootpc any
deny ip any host <IP>
permit tcp any any eq www
!
parameter-map type webauth global
  captive-bypass-portal
  virtual-ip ipv4 192.0.2.1
  virtual-ip ipv6 1001::1
!
wireless profile policy default-policy-profile
   aaa-override
   local-http-profiling
   local-dhcp-profiling
   accounting
```

Thus, you need not go through the entire Configuration Guide to configure wireless embedded wireless controller for a simple configuration requirement.

Configuring Multiple Radius Servers

Use the following procedure to configure a RADIUS server.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless-default radius server** *ip* **key** *secret* **Example:** `Device(config)# wireless-default radius server 9.2.58.90 key cisco123`	Configures a radius server. **Note** You can configure up to ten RADIUS servers.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying AAA and Radius Server Configurations

To view details of AAA server, use the following command:

```
Device# show run aaa
!
aaa new-model
aaa authentication webauth default group radius
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa accounting Identity default start-stop group radius
!
aaa server radius dynamic-author
 client 9.2.58.90 server-key cisco123
!
radius server RAD_SRV_DEF_9.2.58.90
 description Configured by wireless-default
 address ipv4 9.2.58.90 auth-port 1812 acct-port 1813
 key cisco123
!
aaa local authentication default authorization default
aaa session-id common
!
!
ip access-list extended CISCO-CWA-URL-REDIRECT-ACL-DEFAULT
remark " CWA ACL to be referenced from ISE "
deny udp any any eq domain
deny tcp any any eq domain
deny udp any eq bootps any
deny udp any any eq bootpc
deny udp any eq bootpc any
deny ip any host 9.2.58.90
permit tcp any any eq www
!
parameter-map type webauth global
  captive-bypass-portal
  virtual-ip ipv4 192.0.2.1
  virtual-ip ipv6 1001::1
!
wireless profile policy default-policy-profile
   aaa-override
   local-http-profiling
   local-dhcp-profiling
   accounting
```

Note The **show run aaa** output may change when new commands are added to this utility.

Configuring Captive Portal Bypassing for Local and Central Web Authentication

Information About Captive Bypassing

WISPr is a draft protocol that enables users to roam between different wireless service providers. Some devices (For example, Apple iOS devices) have a mechanism using which they can determine if the device is connected

to Internet, based on an HTTP WISPr request made to a designated URL. This mechanism is used for the device to automatically open a web browser when a direct connection to the internet is not possible. This enables the user to provide his credentials to access the internet. The actual authentication is done in the background every time the device connects to a new SSID.

The client device (Apple iOS device) sends a WISPr request to the embedded wireless controller, which checks for the user agent details and then triggers an HTTP request with a web authentication interception in the embedded wireless controller. After verification of the iOS version and the browser details provided by the user agent, the embedded wireless controller allows the client to bypass the captive portal settings and provides access to the Internet.

This HTTP request triggers a web authentication interception in the embedded wireless controller as any other page requests are performed by a wireless client. This interception leads to a web authentication process, which will be completed normally. If the web authentication is being used with any of the embedded wireless controller splash page features (URL provided by a configured RADIUS server), the splash page may never be displayed because the WISPr requests are made at very short intervals, and as soon as one of the queries is able to reach the designated server, any web redirection or splash page display process that is performed in the background is cancelled, and the device processes the page request, thus breaking the splash page functionality.

For example, Apple introduced an iOS feature to facilitate network access when captive portals are present. This feature detects the presence of a captive portal by sending a web request on connecting to a wireless network. This request is directed to http://www.apple.com/library/test/success.html for Apple iOS version 6 and older, and to several possible target URLs for Apple iOS version 7 and later. If a response is received, then the Internet access is assumed to be available and no further interaction is required. If no response is received, then the Internet access is assumed to be blocked by the captive portal and Apple's Captive Network Assistant (CNA) auto-launches the pseudo-browser to request portal login in a controlled window. The CNA may break when redirecting to an ISE captive portal. The embedded wireless controller prevents this pseudo-browser from popping up.

You can now configure the embedded wireless controller to bypass WISPr detection process, so the web authentication interception is only done when a user requests a web page leading to splash page load in user context, without the WISPr detection being performed in the background.

Configuring Captive Bypassing for WLAN in LWA and CWA (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Web Auth**.

Step 2 In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

Step 3 Select **Captive Bypass Portal** check box.

Step 4 Click **Update & Apply to Device**.

Configuring Captive Bypassing for WLAN in LWA and CWA (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type webauth** *parameter-map-name* **Example:** `Device(config)# parameter-map type webauth WLAN1_MAP`	Creates the parameter map. The *parameter-map-name* must not exceed 99 characters.
Step 3	**captive-bypass-portal** **Example:** `Device(config)# captive-bypass-portal`	Configures captive bypassing.
Step 4	**wlan** *profile-name wlan-id ssid-name* **Example:** `Device(config)# wlan WLAN1_NAME 4 WLAN1_NAME`	Specifies the WLAN name and ID. • *profile-name* is the WLAN name which can contain 32 alphanumeric characters. • *wlan-id* is the wireless LAN identifier. The valid range is from 1 to 512. • *ssid-name* is the SSID which can contain 32 alphanumeric characters.
Step 5	**security web-auth** **Example:** `Device(config-wlan)# security web-auth`	Enables the web authentication for the WLAN.
Step 6	**security web-auth parameter-map** *parameter-map-name* **Example:** `Device(config-wlan)# security web-auth parameter-map WLAN1_MAP`	Maps the parameter map. **Note** If parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.
Step 7	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Sending DHCP Options 55 and 77 to ISE

Information about DHCP Option 55 and 77

The DHCP sensors use the following DHCP options on the ISE for native and remote profiling:

- **Option 12**: Hostname
- **Option 6**: Class Identifier

Along with this, the following options needs to be sent to the ISE for profiling:

- **Option 55**: Parameter Request List
- **Option 77**: User Class

Configuration to Send DHCP Options 55 and 77 to ISE (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the**Policy Profile** page, click **Add** to view the **Add Policy Profile** window.

Step 3 Click **Access Policies** tab, choose the **RADIUS Profiling** and **DHCP TLV Caching** check boxes to configure radius profiling and DHCP TLV Caching on a WLAN.

Step 4 Click **Save & Apply to Device**.

Configuration to Send DHCP Options 55 and 77 to ISE (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-policy* **Example:** `Device(config)# wireless profile policy rr-xyz-policy-1`	Configures WLAN policy profile and enters the wireless policy configuration mode.
Step 3	**dhcp-tlv-caching** **Example:**	Configures DHCP TLV caching on a WLAN.

	Command or Action	Purpose
	`Device(config-wireless-policy)# dhcp-tlv-caching`	
Step 4	**radius-profiling** **Example:** `Device(config-wireless-policy)# radius-profiling`	Configures client radius profiling on a WLAN.
Step 5	**end** **Example:** `Device(config-wireless-policy)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring EAP Request Timeout (GUI)

Follow the steps given below to configure the EAP Request Timeout through the GUI:

Procedure

Step 1 Choose **Configuration** > **Security** > **Advanced EAP**.

Step 2 In the **EAP-Identity-Request Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP identity request to wireless clients using local EAP.

Step 3 In the **EAP-Identity-Request Max Retries** field, specify the maximum number of times that the device attempts to retransmit the EAP identity request to wireless clients using local EAP.

Step 4 Set **EAP Max-Login Ignore Identity Response** to **Enabled** state to limit the number of clients that can be connected to the device with the same username. You can log in up to eight times from different clients (PDA, laptop, IP phone, and so on) on the same device. The default state is **Disabled**.

Step 5 In the **EAP-Request Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP request to wireless clients using local EAP.

Step 6 In the **EAP-Request Max Retries** field, specify the maximum number of times that the device attempts to retransmit the EAP request to wireless clients using local EAP.

Step 7 In the **EAPOL-Key Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP key over the LAN to wireless clients using local EAP.

Step 8 In the **EAPOL-Key Max Retries** field, specify the maximum number of times that the device attempts to send an EAP key over the LAN to wireless clients using local EAP.

Step 9 In the **EAP-Broadcast Key Interval** field, specify the time interval between rotations of the broadcast encryption key used for clients and click **Apply**.

Note After configuring the EAP-Broadcast key interval to a new time period, you must shut down or restart the WLAN for the changes to take effect. Once the WLAN is shut down or restarted, the M5 and M6 packets are exchanged when the configured timer value expires.

Configuring EAP Request Timeout

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps client-exclusion dot1x-timeout** **Example:** `Device(config)# wireless wps client-exclusion dot1x-timeout`	Enables exclusion on timeout and no response. By default, this feature is enabled. To disable, append a **no** at the beginning of the command.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring EAP Request Timeout in Wireless Security (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless security dot1x request** {**retries** *0 - 20* | **timeout** *1 - 120*} **Example:** `Device(config)# wireless security dot1x request timeout 60`	Configures the EAP request retransmission timeout value in seconds.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Captive Portal

Captive Portal Configuration

This feature enables you to configure multiple web authentication URLs (including external captive URLs) for the same SSID based on an AP. The default setting is to use the Global URL for authentication. The override option is available at WLAN and AP level.

The order of precedence is:

- AP
- WLAN
- Global configuration

Restrictions for Captive Portal Configuration

- This configuration is supported in a standalone controller only.
- Export-Anchor configuration is not supported.

Configuring Captive Portal (GUI)

Procedure

Step 1	Choose **Configuration** > **Tags & Profiles** > **WLANs**.
Step 2	Click **Add**.
Step 3	In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.
Step 4	In the **Security** > **Layer2** tab, uncheck the **WPA Policy**, **AES** and **802.1x** check boxes.
Step 5	In the **Security** > **Layer3** tab, choose the parameter map from the **Web Auth Parameter Map** drop-down list and authentication list from the **Authentication List** drop-down list.
Step 6	In the **Security** >**AAA** tab, choose the Authentication list from the **Authentication List** drop-down list.
Step 7	Click **Apply to Device**.
Step 8	Choose **Configuration** > **Security** > **Web Auth**.
Step 9	Choose a **Web Auth Parameter Map**.
Step 10	In the **General** tab, enter the **Maximum HTTP connections**, **Init-State Timeout(secs)** and choose **webauth** from the **Type** drop-down list.
Step 11	In the **Advanced** tab, under the **Redirect to external server** settings, enter the **Redirect for log-in** server.
Step 12	Click **Update & Apply**.

Configuring Captive Portal

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** {*profile-name* | **shutdown**} *network-name* **Example:** `Device(config)# wlan edc6 6 edc`	Configures the WLAN profile. Enables or Disables all WLANs and creates the WLAN identifier. The profile-name and the SSID network name should be up to 32 alphanumeric charcters.
Step 3	**ip** {**access-group** | **verify**} **web** *IPv4-ACL-Name* **Example:** `Device(config-wlan)# ip access-group web CPWebauth`	Configures the WLAN web ACL. **Note** WLAN needs to be disabled before performing this operation.
Step 4	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disables WPA security.
Step 5	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 6	**no security wpa wpa2 ciphers aes** **Example:** `Device(config-wlan)# no security wpa wpa2 ciphers aes`	Disables WPA2 ciphers for AES.

Step 7	**security web-auth** {**authentication-list** *authentication-list-name* | **authorization-list** *authorization-list-name* | **on-macfilter-failure** | **parameter-map** *parameter-map-name*}
	Example:
	`Device(config-wlan)# security web-auth authentication-list cp-webauth`
	`Device(config-wlan)# security web-auth parameter-map parMap6`
		Enables web authentication for WLAN. Here,
		• **authentication-list**
		authentication-list-name: Sets the authentication list for IEEE 802.1x.
		• **authorization-list**
		authorization-list-name: Sets the override-authorization list for IEEE 802.1x.
		• **on-macfilter-failure**: Enables Web authentication on MAC filter failure.

	Command or Action	Purpose
		• **parameter-map** *parameter-map-name*: Configures the parameter map. **Note** When **security web-auth** is enabled, you get to map the default **authentication-list** and global **parameter-map**. This is applicable for authentication-list and parameter-map that are not explicitly mentioned.
Step 8	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.
Step 9	**exit** **Example:** `Device(config-wlan)# exit`	Exits from the WLAN configuration.
Step 10	**parameter-map type webauth** *parameter-map-name* **Example:** `Device(config)# parameter-map type webauth parMap6`	Creates a parameter map and enters parameter-map webauth configuration mode.
Step 11	**parameter-map type webauth** *parameter-map-name* **Example:** `Device(config)# parameter-map type webauth parMap6`	Creates a parameter map and enters parameter-map webauth configuration mode.
Step 12	**type webauth** **Example:** `Device(config-params-parameter-map)# type webauth`	Configures the webauth type parameter.
Step 13	**timeout init-state sec** <*timeout-seconds*> **Example:** `Device(config-params-parameter-map)# timeout inti-state sec 3600`	Configures the WEBAUTH timeout in seconds. Valid range for the time in sec parameter is 60 seconds to 3932100 seconds.
Step 14	**redirect for-login** <*URL-String*> **Example:** `Device(config-params-parameter-map)# redirect for-login https://172.16.100.157/portal/login.html`	Configures the URL string for redirect during login.

	Command or Action	Purpose
Step 15	**exit** **Example:** `Device(config-params-parameter-map)# exit`	Exits the parameters configuration.
Step 16	**wireless tag policy** *policy-tag-name* **Example:** `Device(config)# wireless tag policy policy_tag_edc6`	Configures policy tag and enters policy tag configuration mode.
Step 17	**wlan** *wlan-profile-name* **policy** *policy-profile-name* **Example:** `Device(config-policy-tag)# wlan edc6 policy policy_profile_flex`	Attaches a policy profile to a WLAN profile.
Step 18	**end** **Example:** `Device(config-policy-tag)# end`	Saves the configuration and exits configuration mode and returns to privileged EXEC mode.

Captive Portal Configuration - Example

The following example shows how you can have APs at different locations, broadcasting the same SSID but redirecting clients to different redirect portals:

Configuring multiple parameter maps pointing to different redirect portal:

```
parameter-map type webauth parMap1
type webauth
timeout init-state sec 21600
redirect for-login
https://172.16.12.3:8080/portal/PortalSetup.action?portal=cfdbce00-2ce2-11e8-b83c-005056a06b27
redirect portal ipv4 172.16.12.3
!
!
parameter-map type webauth parMap11
type webauth
timeout init-state sec 21600
redirect for-login
https://172.16.12.4:8443/portal/PortalSetup.action?portal=094e7270-3808-11e8-9797-02421e4cae0c
redirect portal ipv4 172.16.12.4
!
```

Associating these parameter maps to different WLANs:

```
wlan edc1 1 edc
ip access-group web CPWebauth
no security wpa
no security wpa akm dot1x
no security wpa wpa2 ciphers aes
security web-auth
security web-auth authentication-list cp-webauth
security web-auth parameter-map parMap11
```

```
no shutdown
wlan edc2 2 edc
ip access-group web CPWebauth
no security wpa
no security wpa akm dot1x
no security wpa wpa2 ciphers aes
security web-auth
security web-auth authentication-list cp-webauth
security web-auth parameter-map parMap1
no shutdown
```

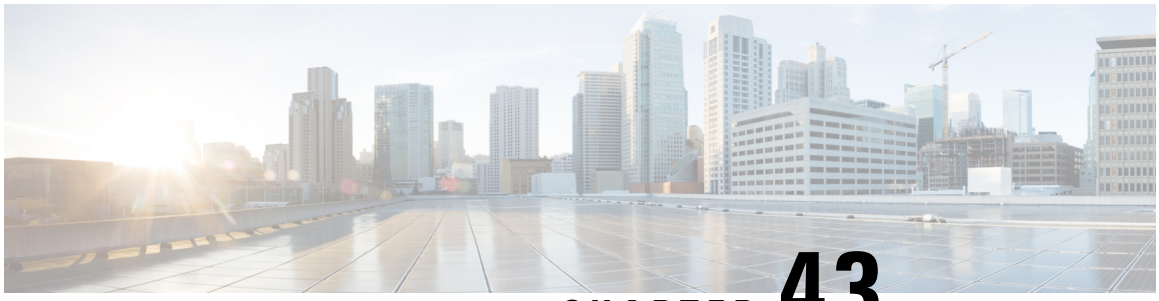
Note All WLANs have identical SSIDs.

Associating WLANs to different policy tags:

```
wireless tag policy policy_tag_edc1
wlan edc1 policy policy_profile_flex
wireless tag policy policy_tag_edc2
wlan edc2 policy policy_profile_flex
```

Assigning these policy tags to the desired APs:

```
ap E4AA.5D13.14DC
policy-tag policy_tag_edc1
site-tag site_tag_flex
ap E4AA.5D2C.3CAC
policy-tag policy_tag_edc2
site-tag site_tag_flex
```



CHAPTER 43

Authentication and Authorization Between Multiple RADIUS Servers

Information About Authentication and Authorization Between Multiple RADIUS Servers

Cisco Embedded Wireless Controller on Catalyst Access Points uses the approach of request and response transaction with a single RADIUS server that combines both authentication and authorization. You can split the authentication and authorization on the controller between multiple RADIUS servers.

A RADIUS sever can assume the role of either an authentication server, authorization server, or both. In cases where there are disparate RADIUS servers for authentication and authorization, the Session Aware Networking (SANet) component on the embedded wireless controller now allows authentication on one server and authorization on another when a client joins the embedded wireless controller.

Authentication can be done using the Cisco ISE, Cisco DNAC, Free RADIUS, or any third-party RADIUS Server. After successful authentication from an authentication server, the embedded wireless controller relays attributes received from the authentication server to another RADIUS sever designated as authorization server.

The authorization server then performs the following:

- Processes received attributes with the other policies or rules defined on the server.
- Derives attributes as part of the authorization response and returns it to the embedded wireless controller.

Note In a split authentication and authorization configuration, both servers must be available and must successfully authenticate and authorize with an ACCESS-ACCEPT for a session to be accepted by the embedded wireless controller.

Note A maximum of 100 entries is supported in the Authentication/Authorization list created through Cisco DNA Center provisioning. The entries beyond 100 do not work even though they can be created.

Configuring 802.1X Security for WLAN with Split Authentication and Authorization Servers

Configuring Explicit Authentication and Authorization Server List (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA**.

Step 2 On the **Authentication Authorization and Accounting** page, click the **Servers/Groups** tab.

Step 3 Click the type of AAA server you want to configure from the following options:

- RADIUS
- TACACS+
- LDAP

In this procedure, the RADIUS server configuration is described.

Step 4 With the **RADIUS** option selected, click **Add**.

Step 5 Enter a name for the RADIUS server and the IPv4 or IPV6 address of the server.

Step 6 Enter the authentication and encryption key to be used between the device and the key string RADIUS daemon running on the RADIUS server. You can choose to use either a PAC key or a non-PAC key.

Step 7 Enter the server timeout value; valid range is 1 to 1000 seconds.

Step 8 Enter a retry count; valid range is 0 to 100.

Step 9 Leave the **Support for CoA** field in **Enabled** state.

Step 10 Click **Save & Apply to Device**.

Step 11 On the **Authentication Authorization and Accounting** page, with **RADIUS** option selected, click the **Server Groups** tab.

Step 12 Click **Add**.

Step 13 In the **Create AAA RADIUS Server Group** window that is displayed, enter a name for the RADIUS server group.

Step 14 From the **MAC-Delimiter** drop-down list, choose the delimiter to be used in the MAC addresses that are sent to the RADIUS servers.

Step 15 From the **MAC Filtering** drop-down list, choose a value based on which to filter MAC addresses.

Step 16 To configure dead time for the server group and direct AAA traffic to alternative groups of servers that have different operational characteristics, in the **Dead-Time** field, enter the amount of time, in minutes, after which a server is assumed to be dead.

Step 17 Choose the servers that you want to include in the server group from the **Available Servers** list and move them to the **Assigned Servers** list.

Step 18 Click **Save & Apply to Device**.

Configuring Explicit Authentication Server List (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA** > **Servers/Groups**.

Step 2 Choose **RADIUS** > **Servers** tab.

Step 3 Click **Add** to add a new server or click an existing server.

Step 4 Enter the **Name**, the **Server Address**, **Key**, **Confirm Key**, **Auth Port** and **Acct Port.** Check the **PAC Key** checkbox and enter the **PAC key** and **Confirm PAC Key**

Step 5 Click **Apply to Device**.

Step 6 Choose **RADIUS** > **Server Groups** and click **Add** to add a new server group or click an existing server group.

Step 7 Enter the **Name** of the server group and choose the servers that you want to include in the server group, from the **Available Servers** list and move them to the **Assigned Servers** list.

Step 8 Click **Apply to Device**.

Configuring Explicit Authentication Server List (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:**	Specifies the RADIUS server name.

	Command or Action	Purpose
	`Device(config)# radius server free-radius-authc-server`	
Step 4	**address ipv4** *address* **auth-port** *auth_port_number* **acct-port** *acct_port_number* **Example:** `Device(config-radius-server)# address ipv4 9.2.62.56 auth-port 1812 acct-port 1813`	Specifies the RADIUS server parameters.
Step 5	[**pac**] **key** *key* **Example:** `Device(config-radius-server)# key cisco`	Specify the authentication and encryption key used between the Device and the key string RADIUS daemon running on the RADIUS server.
Step 6	**exit** **Example:** `Device(config-radius-server)# exit`	Returns to the configuration mode.
Step 7	**aaa group server radius** *server-group* **Example:** `Device(config)# aaa group server radius authc-server-group`	Creates a radius server-group identification. *server-group* refers to the server group name. The valid range is from 1 to 32 alphanumeric characters. If the IP address of the RADIUS server is not added to the routes defined for the controller, the default route is used. We recommend that you define a specific route to source the traffic from the defined SVI in the AAA server group.
Step 8	**server name** *server-name* **Example:** `Device(config)# server name free-radius-authc-server`	Configures the server name.
Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode. For more information, see **Configuring AAA for External Authentication**.

Configuring Explicit Authorization Server List (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **AAA** > **Servers/Groups**.

Step 2 Choose **RADIUS** > **Servers** tab.

Step 3 Click **Add** to add a new server or click an existing server.

Step 4 Enter the **Name**, the **Server Address**, **Key**, **Confirm Key**, **Auth Port** and **Acct Port.** Check the **PAC Key** checkbox and enter the **PAC key** and **Confirm PAC Key**

Step 5 Click **Apply to Device**.

Step 6 Choose **RADIUS** > **Server Groups** and click **Add** to add a new server group or click an existing server group.

Step 7 Enter the **Name** of the server group and choose the servers that you want to include in the server group, from the **Available Servers** list and move them to the **Assigned Servers** list.

Step 8 Click **Apply to Device**.

Configuring Explicit Authorization Server List (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server cisco-dnac-authz-server`	Specifies the RADIUS server name.
Step 4	**address ipv4** *address* **auth-port** *auth_port_number* **acct-port** *acct_port_number* **Example:** `Device(config-radius-server)# address ipv4 9.4.62.32 auth-port 1812 acct-port 1813`	Specifies the RADIUS server parameters.
Step 5	[**pac**] **key** *key* **Example:** `Device(config-radius-server)# pac key cisco`	Specify the authorization and encryption key used between the Device and the key string RADIUS daemon running on the RADIUS server.
Step 6	**exit** **Example:** `Device(config-radius-server)# exit`	Returns to the configuration mode.

	Command or Action	Purpose
Step 7	**aaa group server radius** *server-group* **Example:** `Device(config)# aaa group server radius authz-server-group`	Creates a radius server-group identification.
Step 8	**server name** *server-name* **Example:** `Device(config)# server name cisco-dnac-authz-server`	
Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Authentication and Authorization List for 802.1X Security (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.

Step 4 In the **Security** > **AAA** tab, choose the Authentication list from the **Authentication List** drop-down list.

Step 5 Click **Apply to Device**.

Configuring Authentication and Authorization List for 802.1X Security

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wlan** *wlan-name wlan-id SSID-name* **Example:**	Enters WLAN configuration sub-mode. • *wlan-name*: Is the name of the configured WLAN.

	Command or Action	Purpose
	`Device(config)# wlan wlan-foo 222 foo-ssid`	• *wlan-id*: Is the wireless LAN identifier. Range is from 1 to 512. • *SSID-name*: Is the SSID name which can contain 32 alphanumeric characters. **Note** If you have already configured this command, enter **wlan** *wlan-name* command.
Step 4	**security dot1x authentication-list** *authenticate-list-name* **Example:** `Device(config-wlan)# security dot1x authentication-list authc-server-group`	Enables authentication list for dot1x security.
Step 5	**security dot1x authorization-list** *authorize-list-name* **Example:** `Device(config-wlan)# security dot1x authorization-list authz-server-group`	Specifies authorization list for dot1x security. For more information on the **Cisco Digital Network Architecture Center (DNAC)**, see the **DNAC documentation**.
Step 6	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Web Authentication for WLAN with Split Authentication and Authorization Servers

Configuring Authentication and Authorization List for Web Authentication (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.

Step 3 In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.

Step 4 In the **Security** > **Layer2** tab, uncheck the **WPAPolicy**, **AES** and **802.1x** check boxes.

Step 5 Check the **MAC Filtering** check box to enable the feature. With MAC Filtering enabled, choose the Authorization list from the **Authorization List** drop-down list.

Step 6 In the **Security** > **AAA** tab, choose the Authentication list from the **Authentication List** drop-down list.

Step 7 Click **Apply to Device**.

Configuring Authentication and Authorization List for Web Authentication

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**wlan** *wlan-name wlan-id SSID-name* **Example:** `Device(config)# wlan wlan-bar 1 bar-ssid`	Enters WLAN configuration sub-mode. • *wlan-name*: Is the name of the configured WLAN. • *wlan-id*: Is the wireless LAN identifier. • *SSID-name*: Is the SSID name which can contain 32 alphanumeric characters. **Note** If you have already configured this command, enter **wlan** *wlan-name* command.
Step 4	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disables WPA security.
Step 5	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 6	**no security wpa wpa2** **Example:** `Device(config-wlan)# no security wpa wpa2`	Disables WPA2 security.
Step 7	**security web-auth** {**authentication-list** *authenticate-list-name* | **authorization-list** *authorize-list-name*}	Enables authentication or authorization list for dot1x security.

	Command or Action	Purpose
	Example: `Device(config-wlan)# security web-auth authentication-list authc-server-group`	**Note** You get to view the following error, if you do not disable WPA security, AKM for dot1x, and WPA2 security: *% switch-1:dbm:wireless:web-auth cannot be enabled. Invalid WPA/WPA2 settings.*
Step 8	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Split Authentication and Authorization Configuration

To view the WLAN details, use the following command:

```
Device# show run wlan
wlan wlan-foo 2 foo-ssid
security dot1x authentication-list authc-server-group
security dot1x authorization-list authz-server-group

wlan wlan-bar 3 bar-ssid
security web-auth authentication-list authc-server-group
security web-auth authorization-list authz-server-group
```

To view the AAA authentication and server details, use the following command:

```
Device# show run aaa
!
aaa authentication dot1x default group radius
username cisco privilege 15 password 0 cisco
!
!
radius server free-radius-authc-server
 address ipv4 9.2.62.56 auth-port 1812 acct-port 1813
 key cisco
!
radius server cisco-dnac-authz-server
 address ipv4 9.4.62.32 auth-port 1812 acct-port 1813
 pac key cisco
!
!
aaa new-model
aaa session-id common
!
```

To view the authentication and authorization list for 802.1X security, use the following command:

```
Device# show wlan name wlan-foo | sec 802.1x
802.1x authentication list name               : authc-server-group
802.1x authorization list name                : authz-server-group
            802.1x                            : Enabled
```

To view the authentication and authorization list for web authentication, use the following command:

```
Device# show wlan name wlan-bar | sec Webauth
    Webauth On-mac-filter Failure             : Disabled
    Webauth Authentication List Name          : authc-server-group
    Webauth Authorization List Name           : authz-server-group
    Webauth Parameter Map                     : Disabled
```

Configuration Examples

Configuring Cisco Embedded Wireless Controller on Catalyst Access Points for Authentication with a Third-Party RADIUS Server: Example

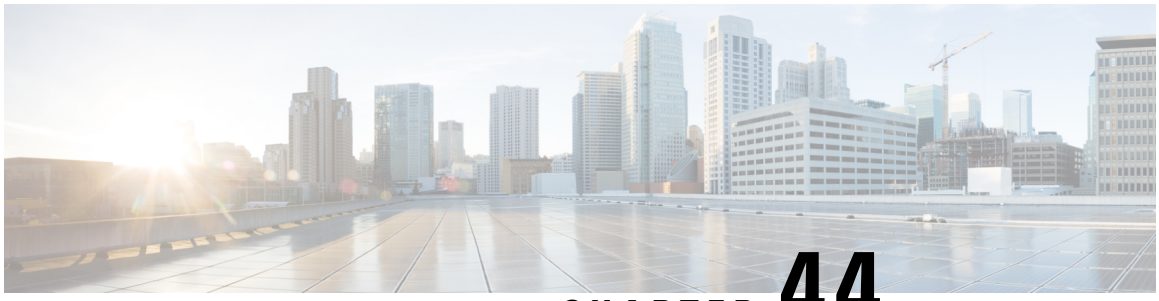
This example shows how to configure Cisco Embedded Wireless Controller on Catalyst Access Points for authentication with a third-party RADIUS server:

```
Device(config)# radius server free-radius-authc-server
Device(config-radius-server)# address ipv4 9.2.62.56 auth-port 1812 acct-port 1813
Device(config-radius-server)# key cisco
Device(config-radius-server)# exit
Device(config)# aaa group server radius authc-server-group
Device(config)# server name free-radius-authc-server
Device(config)# end
```

Configuring Cisco Embedded Wireless Controller on Catalyst Access Points for Authorization with Cisco ISE or DNAC: Example

This example shows how to configure Cisco Embedded Wireless Controller on Catalyst Access Points for authorization with Cisco ISE or DNAC:

```
Device(config)# radius server cisco-dnac-authz-server
Device (config-radius-server)# address ipv4 9.4.62.32 auth-port 1812 acct-port 1813
Device (config-radius-server)# pac key cisco
Device (config-radius-server)# exit
Device(config)# aaa group server radius authz-server-group
Device(config)# server name cisco-dnac-authz-server
Device(config)# end
```



CHAPTER 44

Secure LDAP

Information About SLDAP

Transport Layer Security (TLS)

The Transport Layer Security (TLS) is an application-level protocol that enables secure transactions of data through privacy, authentication, and data integrity. TLS relies upon certificates, public keys, and private keys to prove the identity of clients.

The certificates are issued by the Certificate Authorities (CAs).

Each certificate includes the following:

- The name of the authority that issued it.
- The name of the entity to which the certificate was issued.
- The public key of the entity.
- The timestamps of the entity that indicate the expiration date of the certificate.

You can find the TLS support for LDAP in the RFC2830 which is an extension to the LDAP protocol.

LDAP Operations

Bind

The bind operation is used to authenticate a user to the server. It is used to start a connection with the LDAP server. LDAP is a connection-oriented protocol. The client specifies the protocol version and authentication information.

LDAP supports the following binds:

- Authenticated bind—An authenticated bind is performed when a root Distinguished Name (DN) and password are available.
- Anonymous bind—In the absence of a root DN and password, an anonymous bind is performed.

In LDAP deployments, the search operation is performed first and the bind operation later. This is because, if a password attribute is returned as part of the search operation, the password verification can be done locally on an LDAP client. Thus, there is no need to perform an extra bind operation. If a password attribute is not returned, the bind operation can be performed later. Another advantage of performing a search operation first and a bind operation later is that the DN received in the search result can be used as the user DN instead of forming a DN by prefixing the username (cn attribute) with the base DN. All entries stored in an LDAP server have a unique DN.

The DN consists of two parts:

- Relative Distinguished Name (RDN)
- Location in the LDAP server where the record resides.

Most of the entries that you store in an LDAP server will have a name, and the name is frequently stored in the Common Name (cn) attribute. Because every object has a name, most objects you store in an LDAP will use their cn value as the basis for their RDN.

Search

A search operation is used to search the LDAP server. The client specifies the starting point (base DN) of the search, the search scope (either the object, its children, or the subtree rooted at the object), and a search filter.

For authorization requests, the search operation is directly performed without a bind operation. The LDAP server can be configured with certain privileges for the search operation to succeed. This privilege level is established with the bind operation.

An LDAP search operation can return multiple user entries for a specific user. In such cases, the LDAP client returns an appropriate error code to AAA. To avoid these errors, you must configure appropriate search filters to match a single entry.

Compare

The compare operation is used to replace a bind request with a compare request for an authentication. The compare operation helps to maintain the initial bind parameters for the connection.

LDAP Dynamic Attribute Mapping

The Lightweight Directory Access Protocol (LDAP) is a powerful and flexible protocol for communication with AAA servers. LDAP attribute maps provide a method to cross-reference the attributes retrieved from a server to Cisco attributes supported by the security appliances.

When a user authenticates a security appliance, the security appliance, in turn, authenticates the server and uses the LDAP protocol to retrieve the record for that user. The record consists of LDAP attributes associated with fields displayed on the user interface of the server. Each attribute retrieved includes a value that was entered by the administrator who updates the user records.

Prerequisite for Configuring SLDAP

If you are using a secure Transport Layer Security (TLS) secure connection, you must configure the X.509 certificates.

Restrictions for Configuring SLDAP

- LDAP referrals are not supported.
- Unsolicited messages or notifications from the LDAP server are not handled.
- LDAP authentication is not supported for interactive (terminal) sessions.

Configuring SLDAP

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ldap server** *name* **Example:** `Device(config)# ldap server server1`	Defines a Lightweight Directory Access Protocol (LDAP) server and enters LDAP server configuration mode.
Step 4	**ipv4** *ipv4-address* **Example:** `Device(config-ldap-server)# ipv4 9.4.109.20`	Specifies the LDAP server IP address using IPv4.
Step 5	**timeout retransmit** *seconds* **Example:** `Device(config-ldap-server)# timeout retransmit 20`	Specifies the number of seconds the embedded wireless controller waits for a reply to an LDAP request before retransmitting the request.
Step 6	**bind authenticate root-dn password** [**0** *string* | **7** *string*] *string* **Example:**	Specifies a shared secret text string used between the embedded wireless controller and an LDAP server.

	Command or Action	Purpose
	`Device(config-ldap-server)# bind authenticate root-dn CN=ldapipv6user,CN=Users,DC=ca,DC=ssh2,DC=com password Cisco12345`	Use the **0** line option to configure an unencrypted shared secret. Use the **7** line option to configure an encrypted shared secret.
Step 7	**base-dn** *string* **Example:** `Device(config-ldap-server)# base-dn CN=Users,DC=ca,DC=ssh2,DC=com`	Specifies the base Distinguished Name (DN) of the search.
Step 8	**mode secure** [**no- negotiation**] **Example:** `Device(config-ldap-server)# mode secure no- negotiation`	Configures LDAP to initiate the TLS connection and specifies the secure mode.
Step 9	**end** **Example:** `Device(config-ldap-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring an AAA Server Group (GUI)

Configuring a device to use AAA server groups helps you to group existing server hosts, select a subset of the configured server hosts and use them for a particular service. A server group is used with a global server-host list. The server group lists the IP addresses of the selected server hosts.

You can create the following server groups:

Procedure

Step 1 RADIUS

a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **RADIUS**.
b) Click the **Add** button. The **Create AAA Radius Server Group** dialog box appears.
c) Enter a name for the RADIUS server group in the **Name** field.
d) Choose a desired delimiter from the **MAC-Delimiter** drop-down list. The available options are colon, hyphen, and single-hyphen.
e) Choose a desired filter from the **MAC-Filtering** drop-down list. The available options are mac and Key.
f) Enter a value in the **Dead-Time (mins)** field to make a server non-operational. You must specify a value between 1 and 1440.
g) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
h) Click the **Save & Apply to Device** button.

Step 2 TACACS+

a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **TACACS+**.

b) Click the **Add** button. The **Create AAA Tacacs Server Group** dialog box appears.
c) Enter a name for the TACACS server group in the **Name** field.
d) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
e) Click the **Save & Apply to Device** button.

Step 3 LDAP

a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **LDAP**.
b) Click the **Add** button. The **Create AAA Ldap Server Group** dialog box appears.
c) Enter a name for the LDAP server group in the **Name** field.
d) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
e) Click the **Save & Apply to Device** button.

Configuring a AAA Server Group

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables AAA.
Step 4	**aaa group server ldap** *group-name* **Example:** `Device(config)# aaa group server ldap name1`	Defines the AAA server group with a group name and enters LDAP server group configuration mode. All members of a group must be of the same type, that is, RADIUS, LDAP, or TACACS+.
Step 5	**server** *name* **Example:** `Device(config-ldap-sg)# server server1`	Associates a particular LDAP server with the defined server group. Each security server is identified by its IP address and UDP port number.
Step 6	**exit** **Example:**	Exits LDAP server group configuration mode.

	Command or Action	Purpose
	`Device(config-ldap-sg)# exit`	

Configuring Search and Bind Operations for an Authentication Request

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables AAA.
Step 4	**ldap server** *name* **Example:** `Device(config)# ldap server server1`	Defines a Lightweight Directory Access Protocol (LDAP) server and enters LDAP server configuration mode.
Step 5	**authentication bind-first** **Example:** `Device(config-ldap-server)# authentication bind-first`	Configures the sequence of search and bind operations for an authentication request.
Step 6	**authentication compare** **Example:** `Device(config-ldap-server)# authentication compare`	Replaces the bind request with the compare request for authentication.
Step 7	**exit** **Example:** `Device(config-ldap-server)# exit`	Exits LDAP server group configuration mode.

Configuring a Dynamic Attribute Map on an SLDAP Server

You must create LDAP attribute maps that map your existing user-defined attribute names and values to Cisco attribute names and values that are compatible with the security appliance. You can then bind these attribute maps to LDAP servers or remove them as required.

Note To use the attribute mapping features correctly, you need to understand the Cisco LDAP and user-defined attribute names and values.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ldap attribute-map** *map-name* **Example:** `Device(config)# ldap attribute-map map1`	Configures a dynamic LDAP attribute map and enters attribute-map configuration mode.
Step 4	**map type** *ldap-attr-type aaa-attr-type* **Example:** `Device(config-attr-map)# map type department supplicant-group`	Defines an attribute map.
Step 5	**exit** **Example:** `Device(config-attr-map)# exit`	Exits attribute-map configuration mode.

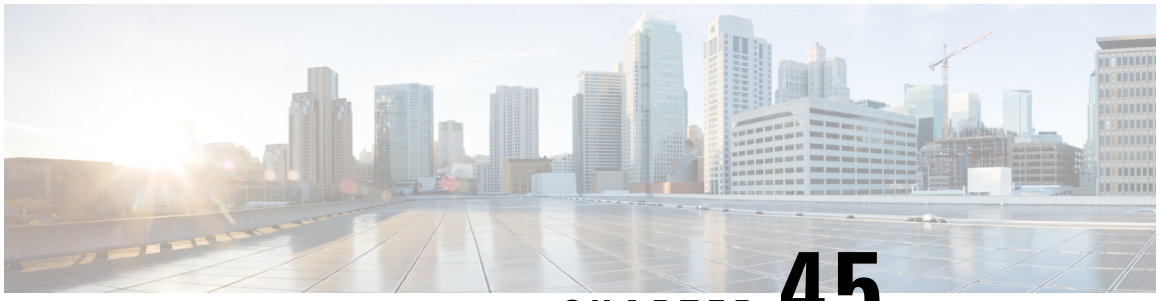
Verifying the SLDAP Configuration

To view details about the default LDAP attribute mapping, use the following command:

```
Device# show ldap attributes
```

To view the LDAP server state information and various other counters for the server, use the following command:

```
Device# show ldap server
```



CHAPTER 45

RADIUS DTLS

Information About RADIUS DTLS

The Remote Authentication Dial-In User Service (RADIUS) is a client or server protocol that provides centralized security for users attempting to gain management access to a network. The RADIUS protocol is a widely deployed authentication and authorization protocol that delivers a complete Authentication, Authorization, and Accounting (AAA) solution.

RADIUS DTLS Port

The RADIUS port (DTLS server) is used for authentication and accounting. The default DTLS server port is 2083.

You can change the RADIUS DTLS port number using **dtls port** *port_number*. For more information, see the Configuring RADIUS DTLS Port Numbersection.

Shared Secret

You can use **radius/dtls** as the shared secret, if you have enabled DTLS for a specific server.

Handling PAC for CTS Communication

You can download PAC from ISE for CTS communication. Once the PAC is downloaded, you need to encrypt all the CTS attributes with the PAC key instead of the shared secret.

The ISE then decrypts these attributes using PAC.

Session Management

The RADIUS client purely depends on the response from the DTLS server. If the session is ideal for ideal timeout, then the session must be closed.

In case of invalid responses, the sessions must be deleted.

If you need to send the radius packets over DTLS, the DTLS session needs to be re-established with the specific server.

Load Balancing

Multiple DTLS servers and load balancing methods are configured.

You need to select the AAA server to which the request needs to be sent. Then use the DTLS context of the specific server to encrypt the RADIUS packet and send it back.

Connection Timeout

After the encrypted RADIUS packet is sent, you need to start the retransmission timer. If you do not get a response before the retransmission timer expires, the packet is re-encrypted and re-transmitted.

You can continue for number of times as per the **dtls retries** configuration or till the default value. Once the number of tries exceeds the limit, the server becomes unavailable and responses are sent back to the AAA clients.

Note The default connection timeout is 5 seconds.

Connection Retries

As the RADIUS DTLS is UDP based, you need to retry the connection after a specific timeout interval for a specific number of retries.

After all retries are exhausted, the DTLS connection performs the following:

- Is marked as unsuccessful.
- Looks up for the next available server for processing the RADIUS requests.

Note The default connection retries is 5.

Idle Timeout

When the idle timer expires and no transactions exists since the last idle timeout, the DTLS session remains closed.

After you establish the DTLS session, you can start the idle timer. If you start the idle timer for 30 seconds and one of the RADIUS DTLS packet is sent, then after 30 seconds, the idle timer expires and checks for number of RADIUS DTLS transactions.

If the idle timer value exceeds zero, the idle timer resets the transaction counter and restarts the timer.

Note The default idle timeout is 60 seconds.

Handling Server and Server Group Failover

You can configure RADIUS servers with and without DTLS. It is recommended to create AAA server groups with DTLS enabled servers and non-DTLS servers. However, you will not find any such restriction while configuring AAA server groups.

Suppose you choose a DTLS server, the DTLS server establishes connection and RADIUS request packet is sent to the DTLS server. If the DTLS server does not respond after all RADIUS retries, it would fall over to the next configured server in the same server group. If the next server is a DTLS server, the processing of the RADIUS request packet continues with the next server. If the next server is a non-DTLS server, the processing of RADIUS request packet does not happen in that server group. Then the server group failover occurs and the same sequence continues with the next server group, if the next server group is available.

Note You need to use either only DTLS or non-DTLS servers in a server group.

Prerequisites

Support for IOS and BINOS AAA

The AAA server runs in IOS and BINOS platforms. Once you complete the RADIUS DTLS support in IOS, the same needs to be ported to BINOS.

Configuring RADIUS DTLS Server

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server R1`	Specifies the RADIUS server name.

	Command or Action	Purpose
Step 4	**dtls** **Example:** `Device(config-radius-server)# dtls`	Configures DTLS parameters.
Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring RADIUS DTLS Connection Timeout

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server R1`	Specifies the RADIUS server name.
Step 4	**dtls connectiontimeout** *timeout* **Example:** `Device(config-radius-server)# dtls connectiontimeout 1`	Configures RADIUS DTLS connection timeout. Here, *timeout* refers to the DTLS connection timeout value. The valid range is from 1 to 65535.
Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring RADIUS DTLS Idle Timeout

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enters privileged EXEC mode.

	Command or Action	Purpose
	`Device# enable`	
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server R1`	Specifies the RADIUS server name.
Step 4	**dtls idletimeout** *idle_timeout* **Example:** `Device(config-radius-server)# dtls idletimeout 2`	Configures RADIUS DTLS idle timeout. Here, *idle_timeout* refers to the DTLS idle timeout value. The valid range is from 1 to 65535.
Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Source Interface for RADIUS DTLS Server

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server R1`	Specifies the RADIUS server name.
Step 4	**dtls ip** {**radius source-interface Ethernet-Internal** *interface_number* **Example:** `Device(config-radius-server)# dtls ip radius source-interface Ethernet-Internal 0`	Configures source interface for RADIUS DTLS server. Here, • *interface_number* refers to the Ethernet-Internal interface number. The default value is 0.

	Command or Action	Purpose
Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring RADIUS DTLS Port Number

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server R1`	Specifies the RADIUS server name.
Step 4	**dtls port** *port_number* **Example:** `Device(config-radius-server)# dtls port 2`	Configures RADIUS DTLS port number. Here, *port_number* refers to the DTLS port number.
Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring RADIUS DTLS Connection Retries

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server R1`	Specifies the RADIUS server name.
Step 4	**dtls retries** *retry_number* **Example:** `Device(config-radius-server)# dtls retries 3`	Configures RADIUS connection retries. Here, *retry_number* refers to the DTLS connection retries. The valid range is from 1 to 65535.
Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring RADIUS DTLS Trustpoint

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**radius server** *server-name* **Example:** `Device(config)# radius server R1`	Specifies the RADIUS server name.

Step 4	**dtls trustpoint** {**client** *LINE* **dtls** | **server** *LINE* **dtls**}
	Example:
	`Device(config-radius-server)# dtls trustpoint client client1 dtls`
	`Device(config-radius-server)# dtls trustpoint server server1 dtls`
		Configures trustpoint for client and server.

Step 5	**end** **Example:** `Device(config-radius-server)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring DTLS Dynamic Author

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa server radius dynamic-author** **Example:** `Device(config)# aaa server radius dynamic-author`	Configures local server profile for RFC 3576 support.
Step 4	**dtls** **Example:** `Device(config-locsvr-da-radius)# dtls`	Configures DTLS source parameters.
Step 5	**end** **Example:** `Device(config-locsvr-da-radius)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Enabling DTLS for Client

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa server radius dynamic-author** **Example:**	Configures local server profile for RFC 3576 support.

	Command or Action	Purpose
	`Device(config)# aaa server radius dynamic-author`	
Step 4	**client** *IP_addr* **dtls** **Example:** `Device(config-locsvr-da-radius)# client 10.104.49.14 dtls`	Enables DTLS for the client.
Step 5	**end** **Example:** `Device(config-locsvr-da-radius)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Client Trustpoint for DTLS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa server radius dynamic-author** **Example:** `Device(config)# aaa server radius dynamic-author`	Configures local server profile for RFC 3576 support.
Step 4	**client** *IP_addr* **dtls** {**client-tp** *client-tp-name* | **server-tp** *server-tp-name*} **Example:** `Device(config-locsvr-da-radius)# client 10.104.49.14 dtls client-tp client_tp_name`	Configures client trustpoint for DTLS.
Step 5	**end** **Example:** `Device(config-locsvr-da-radius)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring DTLS Idle Timeout

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa server radius dynamic-author** **Example:** `Device(config)# aaa server radius dynamic-author`	Configures local server profile for RFC 3576 support.
Step 4	**client** *IP_addr* **dtls idletimeout** *timeout-interval* {**client-tp** *client_tp_name* | **server-tp** *server_tp_name*} **Example:** `Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 62 client-tp dtls_ise`	Configures DTLS idle time. Here, *timeout-interval* refers to the idle timeout interval. The valid range is from 60 to 600.
Step 5	**end** **Example:** `Device(config-locsvr-da-radius)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Server Trustpoint for DTLS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**aaa server radius dynamic-author** **Example:** `Device(config)# aaa server radius dynamic-author`	Configures local server profile for RFC 3576 support.
Step 4	**client** *IP_addr* **dtls server-tp** *server_tp_name* **Example:** `Device(config-locsvr-da-radius)# client 10.104.49.14 dtls server-tp dtls_client`	Configures server trust point.
Step 5	**end** **Example:** `Device(config-locsvr-da-radius)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying the RADIUS DTLS Server Configuration

To view information about the DTLS enabled servers, use the following command:

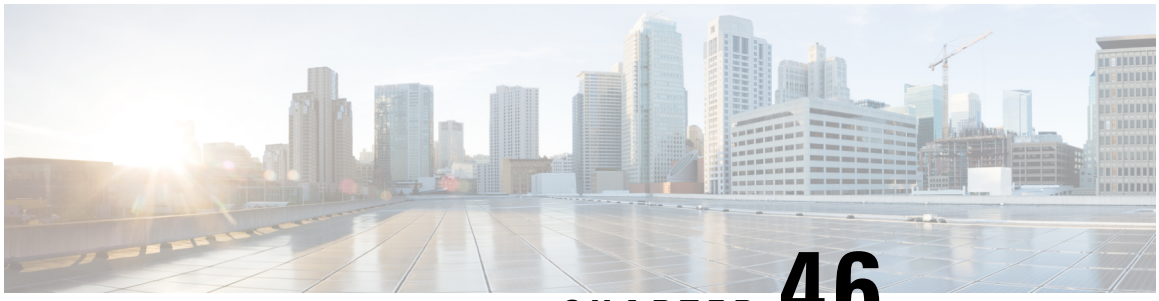
```
Device# show aaa servers
DTLS: Packet count since last idletimeout 1,
Send handshake count 3,
Handshake Success 1,
Total Packets Transmitted 1,
Total Packets Received 1,
Total Connection Resets 2,
Connection Reset due to idle timeout 0,
Connection Reset due to No Response 2,
Connection Reset due to Malformed packet 0,
```

Clearing RADIUS DTLS Specific Statistics

To clear the radius DTLS specific statistics, use the following command:

```
Device# clear aaa counters servers radius {<server-id> | all}
```

Note Here, *server-id* refers to the server ID displayed by **show aaa servers**. The valid range is from 0 to 2147483647.



CHAPTER 46

MAC Filtering

MAC Filtering

You can configure the embedded wireless controller to authorize clients based on the client MAC address by using the MAC filtering feature.

When MAC filtering is enabled, the embedded wireless controller uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. The embedded wireless controller sends the authentication server a RADIUS-access/request frame with a username and password based on the client MAC address as soon as it gets the association request from the client. If authorization succeeds, the embedded wireless controller sends a successful association response to the client. If authorization fails, the embedded wireless controller rejects the client association.

Clients that were authorized with MAC filtering can be re-authenticated through the WLAN session timeout feature.

MAC Filtering Configuration Guidelines

- MAC filtering authentication occurs at the 802.11 association phase and delays the association response until authentication is done. If you use a RADIUS server for MAC filtering, it is advised to keep a low latency between the controller and the RADIUS server. When latency is too high, the client might timeout while waiting for the association response.
- MAC filtering can be combined with other authentication methods such as 802.1X, Pre-Shared Key or it can be used alone.
- MAC addresses can be spoofed and MAC filtering does not consist in a security measure.
- Many clients can use a private MAC address to connect and change it at every session, therefore making it harder to identify devices through their MAC address.

Note If wlan-profile-name is configured for a user, guest user authentication is allowed only from that WLAN.

If wlan-profile-name is not configured for a user, guest user authentication is allowed on any WLAN.

If you want the client to connect to SSID1, but not to SSID2 using mac-filtering, ensure that you configure **aaa-override** in the policy profile.

In the following example, when a client with MAC address 1122.3344.0001 tries to connect to a WLAN, the request is sent to the local RADIUS server, which checks the presence of the client MAC address in its attribute list (FILTER_1 and FILTER_2). If the client MAC address is listed in an attribute list (FILTER_1), the client is allowed to join the WLAN (WLAN_1) that is returned as *ssid attribute* from the RADIUS server. The client is rejected, if the client MAC address is not listed in the attribute list.

Local RADIUS Server Configuration

```
!Configures an attribute list as FILTER_2
aaa attribute list FILTER_2
!Defines an attribute type that is to be added to an attribute list.
attribute type ssid "WLAN_2"

!Username with the MAC address is added to the filter
username 1122.3344.0002 mac aaa attribute list FILTER_2

!
aaa attribute list FILTER_1
attribute type ssid "WLAN_1"
username 1122.3344.0001 mac aaa attribute list FILTER_1
```

Controller Configuration

```
! Sets authorization to the local radius server
aaa authorization network MLIST_MACFILTER local

!A WLAN with the SSID WLAN_2 is created and MAC filtering is set along with security
parameters.
wlan WLAN_2 2 WLAN_2
mac-filtering MLIST_MACFILTER
no security wpa
no security wpa wpa2 ciphers

!WLAN with the SSID WLAN_1 is created and MAC filtering is set along with security parameters.
wlan WLAN_1 1 WLAN_1
mac-filtering MLIST_MACFILTER
no security wpa
no security wpa wpa2 ciphers aes
no security wpa akm dot1x
security web-auth
security web-auth authentication-list WEBAUTH

! Policy profile to be associated with the above WLANs
wireless profile policy MAC_FILTER_POLICY
aaa-override
vlan 504
no shutdown
```

Configuring MAC Filtering for Local Authentication (CLI)

Follow the procedure given below to configure MAB for local authentication.

Before you begin

Configure AAA local authentication.

Configure the username for WLAN configuration (local authentication) using **username** *mac-address* **mac** command.

Note The mac-address must be in the following format: *abcdabcdabcd*

Procedure

	Command or Action	Purpose
Step 1	**wlan** *profile-name wlan-id* **Example:** `wlan CR1_SSID_mab-local-default 1 CR1_SSID_mab-local-default`	Specifies the WLAN name and ID.
Step 2	**mac-filtering default** **Example:** `Device(config-wlan)# mac-filtering default`	Sets MAC filtering support for the WLAN.
Step 3	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disables WPA secuirty.
Step 4	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 5	**no security wpa wpa2** **Example:** `Device(config-wlan)# no security wpa wpa2`	Disables WPA2 security.
Step 6	**no security wpa wpa2 ciphers aes** **Example:** `Device(config-wlan)# no security wpa wpa2 ciphers aes`	Disables WPA2 ciphers for AES.
Step 7	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.

Configuring MAC Filtering (GUI)

Before you begin

Configure AAA external authentication.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **WLANs**.
Step 2 On the **Wireless Networks** page, click the name of the WLAN.
Step 3 In the **Edit WLAN** window, click the **Security** tab.
Step 4 In the **Layer2** tab, check the **MAC Filtering** check box to enable the feature.
Step 5 With MAC Filtering enabled, choose the **Authorization List** from the drop-down list.
Step 6 Save the configuration.

Configuring MAB for External Authentication (CLI)

Follow the procedure given below to configure MAB for external authentication.

Before you begin

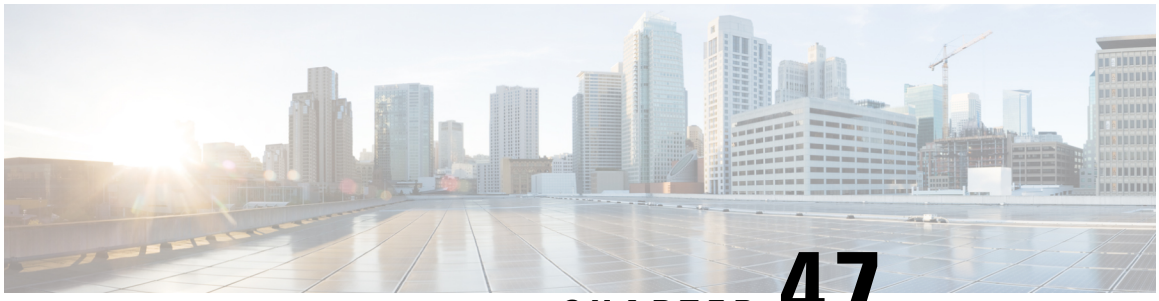
Configure AAA external authentication.

Procedure

	Command or Action	Purpose
Step 1	**wlan** *wlan-name wlan-id ssid-name* **Example:** `wlan CR1_SSID_mab-ext-radius 3 CR1_SSID_mab-ext-radius`	Specifies the WLAN name and ID.
Step 2	**mac-filtering** *list-name* **Example:** `Device(config-wlan)# mac-filtering ewlc-radius`	Sets the MAC filtering parameters. Here, *ewlc-radius* is an example for the *list-name*
Step 3	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disables WPA secuirty.
Step 4	**no security wpa akm dot1x** **Example:**	Disables security AKM for dot1x.

	Command or Action	Purpose
	`Device(config-wlan)# no security wpa akm dot1x`	
Step 5	**no security wpa wpa2** **Example:** `Device(config-wlan)# no security wpa wpa2`	Disables WPA2 security.
Step 6	**mab request format attribute** {**1 groupsize** *size* **separator** *separator* [**lowercase** | **uppercase**] | **2** {**0** | **7** | **LINE**} **LINE** *password* | **32 vlan access-vlan**} **Example:** `Device(config)# mab request format attribute 1 groupsize 4 separator`	Optional. Configures the delimiter while using MAC filtering in a WLAN. Here, **1**- Specifies the username format used for MAB requests. **groupsize** *size*- Specifies the number of hex digits per group. The valid values range from 1 to 12. **separator** *separator*- Specifies how to separate groups. The separators are comma, semicolon, and full stop. **lowercase**- Specifies the username in lowercase format. **uppercase**- Specifies the username in uppercase format. **2**- Specifies the global password used for all the MAB requests. **0**- Specifies the unencrypted password. **7**- Specifies the hidden password. **LINE**- Specifies the encrypted or unencrypted password. *password*- LINE password. **32**- Specifies the NAS-Identifier attribute. **vlan**- Specifies a VLAN. **access-vlan**- Specifies the configured access VLAN.
Step 7	**no security wpa wpa2 ciphers aes** **Example:** `Device(config-wlan)# no security wpa wpa2 ciphers aes`	Disables WPA2 ciphers for AES.
Step 8	**no shutdown** **Example:**	Enables the WLAN.

	Command or Action	Purpose
	`Device(config-wlan)# no shutdown`	



CHAPTER 47

Dynamic Frequency Selection

Information About Dynamic Frequency Selection

Dynamic Frequency Selection (DFS) is the process of detecting radar signals and automatically setting the frequency on a DFS-enabled 5.0-GHz (802.11a/h) radio to avoid interference with the radar signals. Radios configured for use in a regulatory domain must not interfere with radar systems.

In normal DFS, when a radar signal is detected on any of the channels in the 40-MHz or 80-MHz bandwidth, the whole channel is blocked. With Flex DFS, if the radar signals are not detected on the secondary channel, the AP is moved to a secondary channel with a reduction in the bandwidth, usually, by half.

Configuring Dynamic Frequency Selection (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**

Step 2 Choose a profile.

Step 3 In **General** tab, check the **Full sector DFS status** check box.

Step 4 Click **Update & Apply to Device**.

Configuring Dynamic Frequency Selection

Follow the procedure given below to configure DFS:

Before you begin

- The corresponding AP must be on one of the DFS channels.
- Shut down the radio before applying the configuration changes.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**no ap dot11 5ghz dtpc** **Example:** `Device(config)# no ap dot11 5ghz dtpc`	Disables the 802.11a Dynamic Transmit Power Control (DTPC) setting.
Step 3	**ap dot11 5ghz channelswitch mode** *mode-num* **Example:** `Device(config)# ap dot11 5ghz channelswitch mode 1`	Configures the 802.11h channel switch mode.
Step 4	**ap dot11 5ghz power-constraint** *value* **Example:** `Device(config)# ap dot11 5ghz power-constraint 12`	Configures the 802.11h power-constraint value.
Step 5	**ap dot11 5ghz smart-dfs** **Example:** `Device(config)# ap dot11 5ghz smart-dfs`	Configures nonoccupancy time for the radar interference channel.

Verifying DFS

Use the following commands to verify the DFS configuration:

To display the 802.11h configuration, use the following command:

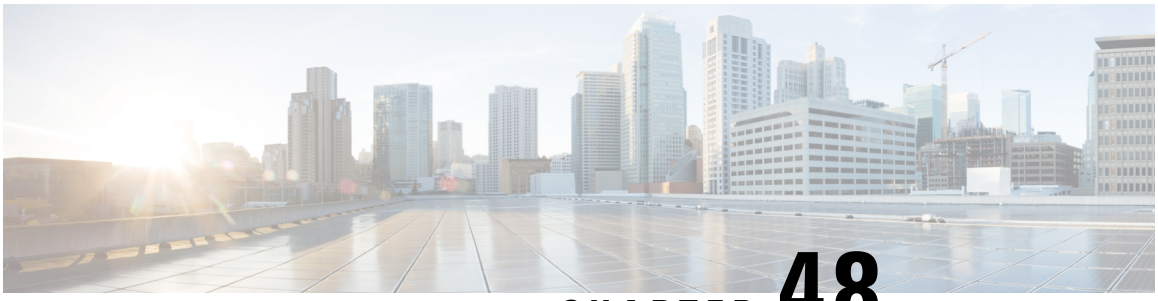
```
Device# show wireless dot11h
```

To display the auto-rF information for 802.11h configuration, use the following command:

```
Device# show ap auto-rf dot11 5ghz
```

To display the auto-rF information for a Cisco AP, use the following command:

```
Device# show ap name ap1 auto-rf dot11 5gh
```



CHAPTER 48

Managing Rogue Devices

Rogue Detection

Rogue Devices

Rogue access points can disrupt wireless LAN operations by hijacking legitimate clients and using plain-text or other denial-of-service or man-in-the-middle attacks. That is, a hacker can use a rogue access point to capture sensitive information, such as usernames and passwords. The hacker can then transmit a series of Clear to Send (CTS) frames. This action mimics an access point, informing a particular client to transmit, and instructing all the other clients to wait, which results in legitimate clients being unable to access network resources. Wireless LAN service providers have a strong interest in banning rogue access points from the air space.

Because rogue access points are inexpensive and readily available, employees sometimes plug unauthorized rogue access points into existing LANs and build ad hoc wireless networks without their IT department's knowledge or consent. These rogue access points can be a serious breach of network security because they can be plugged into a network port behind the corporate firewall. Because employees generally do not enable any security settings on the rogue access point, it is easy for unauthorized users to use the access point to intercept network traffic and hijack client sessions. There is an increased chance of enterprise security breach when wireless users connect to access points in the enterprise network.

The following are some guidelines to manage rogue devices:

- The access points are designed to serve associated clients. These access points spend relatively less time performing off-channel scanning: about 50 milliseconds on each channel. If you want to detect a large number of rogue APs and clients with high sensitivity, a monitor mode access point must be used. Alternatively, you can reduce the scan intervals from 180 seconds to a lesser value, for example, 120 or 60 seconds, ensuring that the radio goes off-channel more frequently, which improves the chances of rogue detection. However, the access point continues to spend about 50 milliseconds on each channel.

- Rogue detection is disabled by default for OfficeExtend access points because these access points, which are deployed in a home environment, are likely to detect many rogue devices.
- Client card implementation might mitigate the effectiveness of containment. This normally happens when a client might quickly reconnect to the network after receiving a "de-association/de-authentication" frame, so it might still be able to pass some traffic. However, the browsing experience of the rogue client would be badly affected when it is contained.
- It is possible to classify and report rogue access points by using rogue states and user-defined classification rules that enable rogues to automatically move between states.
- Each controller limits the number of rogue containments to three and six per radio for access points in the monitor mode.
- When manual containment is performed using configuration, the rogue entry is retained even after the rogue entry expires.
- When a rogue entry expires, the managed access points are instructed to stop any active containment on it.
- To validate a Rogue Client against AAA, add the rogue client MAC to the AAA user-database with relevant delimiter, username, and password being the MAC address with relevant delimiter. The Access-Accept contains the Cisco-AV-pair with one of the following keywords:

  rogue-ap-state=*state*

Note Here, **state** can be of three types, namely: alert, threat, and contain.

For instance, **rogue-ap-state**=*threat*.

If Access-Accept has no AV-Pair rogue-ap-class or an invalid value of rogue-ap-class, such a rogue client state is set to either of the following:

- Contained, if the config is set to autocontain clients or untrusted AP.
- Threat

The Radius Access-Reject for rogue client AAA validation is ignored.

- When Validate Rogue Clients Against AAA is enabled, the controller requests the AAA server for rogue client validation only once. As a result, if rogue client validation fails on the first attempt then the rogue client will not be detected as a threat any more. To avoid this, add the valid client entries in the authentication server before enabling Validate Rogue Clients Against AAA.

Restrictions on Rogue Detection

- Rogue containment is not supported on DFS channels.

A rogue access point is moved to a contained state either automatically or manually. The controller selects the best available access point for containment and pushes the information to the access point. The access point stores the list of containments per radio. For auto containment, you can configure the controller to use only the monitor mode access point. The containment operation occurs in the following two ways:

- The container access point goes through the list of containments periodically and sends unicast containment frames. For rogue access point containment, the frames are sent only if a rogue client is associated.
- Whenever a contained rogue activity is detected, containment frames are transmitted.

Individual rogue containment involves sending a sequence of unicast disassociation and deauthentication frames.

Information About Rogue Containment (Protected Management Frames (PMF) Enabled)

From Cisco IOS XE Amsterdam, 17.3.1 onwards, rogue devices that are enabled with 802.11w Protected Management Frames (PMF) are not contained. Instead, the rogue device is marked as *Contained Pending*, and a WSA alarm is raised to inform about the Contained Pending event. Because the device containment is not performed, access point (AP) resources are not consumed unnecessarily.

Note This feature is supported only on the Wave 2 APs.

Run the **show wireless wps rogue ap detailed** command to verify the device containment, when PMF is enabled on a rogue device.

AP Impersonation Detection

The various methods to detect AP impersonation are:

- AP impersonation can be detected if a managed AP reports itself as Rogue. This method is always enabled and no configuration is required.
- AP impersonation detection is based on MFP.
- AP impersonation detection based on AP authentication.

Infrastructure MFP protects 802.11 session management functions by adding message integrity check (MIC) information elements, to the management frames sent by APs (and not those sent by clients), which are then validated by other APs in the network. If infrastructure MFP is enabled, the managed APs check if the MIC information elements are present and if MIC information elements are as expected. If either of these conditions is not fulfilled, the managed AP sends rogue AP reports with updated AP authentication failure counter.

The AP Authentication functionality allows you to detect AP impersonation. When you enable this functionality, the controller creates an AP domain secret and shares it with other APs in the same network. This allows the APs to authenticate each other.

An AP Authentication information element is attached to beacon and probe response frames. If the AP Authentication information element has an incorrect Signature field, or the timestamp is off, or if the AP Authentication information element is missing, then the AP that has detected such a condition increments the **AP authentication failure count** field. An impersonation alarm is raised after the **AP authentication failure count** field breaches its threshold. The rogue AP is classified as **Malicious** with state **Threat**.

Run the **show wireless wps rogue ap detail** command to see when the impersonation is detected due to authentication errors.

Note Ensure that the **ccx aironet-iesupport** command is run in all the WLAN procedures, else the BSSID will be detected as a rogue.

For AP impersonation detection, Network Time Protocol (NTP) must be enabled instead of CAPWAP based time, under the AP profile.

Configuring Rogue Detection (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 Click the **AP Join Profile Name** to edit the AP join profile properties.

Step 3 In the **Edit AP Join Profile** window, click the **Rogue AP** tab.

Step 4 Check the **Rogue Detection** check box to enable rogue detection.

Step 5 In the **Rogue Detection Minimum RSSI** field, enter the RSSI value.

Step 6 In the **Rogue Detection Transient Interval** field, enter the interval in seconds.

Step 7 In the **Rogue Detection Report Interval** field, enter the report interval value in seconds.

Step 8 In the **Rogue Detection Client Number Threshold** field, enter the threshold for rogue client detection.

Step 9 Check the **Auto Containment on FlexConnect Standalone** check box to enable auto containment.

Step 10 Click **Update & Apply to Device**.

Configuring Rogue Detection (CLI)

Procedure

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

Step 2	**ap profile** *profile-name* **rogue detection min-transient-time** *time in seconds*
	Example:
	`Device(config)# ap profile profile1`
	`Device(config)# rogue detection min-transient-time 120`
		Specify the time interval at which rogues have to be consistently scanned for by APs after the first time the rogues are scanned.
		Valid range for the time in sec parameter is 120 seconds to 1800 seconds, and the default value is 0.

	Command or Action	Purpose
		Note This feature is applicable to all AP modes. Using the transient interval values, you can control the time interval at which APs should scan for rogues. APs can also filter the rogues based on their transient interval values. This feature has thefollowing advantages: • Rogue reports from APs to the controller are shorter • Transient rogue entries are avoided in the controller Unnecessary memory allocation fortransient rogues are avoided

Step 3	**ap profile** *profile-name* **rogue detection containment** {**auto-rate** | **flex-rate**}
	Example:
	`Device(config)# ap profile profile1`
	`Device(config)# rogue detection containment flex-rate`
		Specifies the rogue containment options. The **auto-rate** option enables auto-rate for containment of rogues. The **flex-rate** option enables rogue containment of standalone flexconnect APs.

Step 4	**ap profile** *profile-name* **rogue detection enable** **Example:** `Device(config)# ap profile profile1`	Enables rogue detection on all APs.

Step 5	**ap profile** *profile-name* **rogue detection report-interval** *time in seconds*
	Example:
	`Device(config)# ap profile profile1`
	`Device(config)# rogue detection report-interval 120`
		Configures rogue report interval for monitor mode Cisco APs.
		The valid range for reporting the interval in seconds is 10 seconds to 300 seconds.

Configuring RSSI Deviation Notification Threshold for Rogue APs (CLI)

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**wireless wps rogue ap notify-rssi-deviation** **Example:** `Device(config)# wireless wps rogue ap notify-rssi-deviation`	Configures RSSI deviation notification threshold for Rogue APs.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Management Frame Protection (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies**.

Step 2 In the **Rogue Policy** tab, under the **MFP Configuration** section, check the **Global MFP State** check box and the **AP Impersonation Detection** check box to enable the global MFP state and the AP impersonation detection, respectively.

Step 3 In the **MFP Key Refresh Interval** field, specify the refresh interval in hours.

Step 4 Click **Apply**.

Configuring Management Frame Protection (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps mfp** **Example:** `Device(config)# wireless wps mfp`	Configures a management frame protection.
Step 3	**wireless wps mfp** {**ap-impersonation** | **key-refresh-interval**} **Example:** `Device(config)# wireless wps mfp ap-impersonation`	Configures ap impersonation detection (or) MFP key refresh interval in hours. key-refresh-interval—Refers to the MFP key refresh interval in hours. The valid range is from 1 to 24. Default value is 24.

	Command or Action	Purpose
	`Device(config)# wireless wps mfp key-refresh-interval`	
Step 4	**end** **Example:** `Device(config)# end`	Saves the configuration and exits configuration mode and returns to privileged **EXEC** mode.

Enabling Access Point Authentication

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps ap-authentication** **Example:** `Device(config)# wireless wps ap-authentication`	Configures the wireless WPS AP authentication.
Step 3	**wireless wps ap-authentication threshold** *threshold* **Example:** `Device(config)# wireless wps ap-authentication threshold 100`	Configures AP neighbor authentication and sets the threshold for AP authentication failures.
Step 4	**wlan** *wlan-name wlan-id SSID-name* **Example:** `Device(config)# wlan wlan-demo 1 ssid-demo`	Configures a WLAN.
Step 5	**ccx aironet-iesupport** **Example:** `Device(config-wlan)# ccx aironet-iesupport`	Enables support for Aironet Information Elements on this WLAN.
Step 6	**end** **Example:** `Device# end`	Returns to privileged EXEC mode.

Verifying Management Frame Protection

To verify if the Management Frame Protection (MFP) feature is enabled or not, use the following command:

```
Device# show wireless wps summary
Client Exclusion Policy
  Excessive 802.11-association failures   : unknown
  Excessive 802.11-authentication failures: unknown
  Excessive 802.1x-authentication         : unknown
  IP-theft                                : unknown
  Excessive Web authentication failure    : unknown
  Failed Qos Policy                       : unknown

Management Frame Protection
  Global Infrastructure MFP state : Enabled
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

To view the MFP details, use the following command:

```
Device# show wireless wps mfp summary
Management Frame Protection
  Global Infrastructure MFP state : Enabled
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

Verifying Rogue Detection

This section describes the new command for rogue detection.

The following command can be used to verify rogue detection on the device.

Table 19: Verifying Adhoc Rogues Information

Command	Purpose
show wireless wps rogue adhoc detailed *mac_address*	Displays the detailed information for an Adhoc rogue.
show wireless wps rogue adhoc summary	Displays a list of all Adhoc rogues.

Table 20: Verifying Rogue AP Information

Command	Purpose
show wireless wps rogue ap clients *mac_address*	Displays the list of all rogue clients associated with a rogue.
show wireless wps rogue ap custom summary	Displays the custom rogue AP information.
show wireless wps rogue ap detailed *mac_address*	Displays the detailed information for a rogue AP.
show wireless wps rogue ap friendly summary	Displays the friendly rogue AP information.
show wireless wps rogue ap list *mac_address*	Displays the list of rogue APs detected by a given AP.
show wireless wps rogue ap malicious summary	Displays the malicious rogue AP information.
show wireless wps rogue ap summary	Displays a list of all Rogue APs.
show wireless wps rogue ap unclassified summary	Displays the unclassified rogue AP information.

Table 21: Verifying Rogue Auto-Containment Information

Command	Purpose
show wireless wps rogue auto-contain	Displays the rogue auto-containment information.

Table 22: Verifying Classification Rule Information

Command	Purpose
show wireless wps rogue rule detailed *rule_name*	Displays the detailed information for a classification rule.
show wireless wps rogue rule summary	Displays the list of all rogue rules.

Table 23: Verifying Rogue Statistics

Command	Purpose
show wireless wps rogue stats	Displays the rogue statistics.

Table 24: Verifying Rogue Client Information

Command	Purpose
show wireless wps rogue client detailed *mac_address*	Displays detailed information for a Rogue client.
show wireless wps rogue client summary	Displays a list of all the Rogue clients.

Table 25: Verifying Rogue Ignore List

Command	Purpose
show wireless wps rogue ignore-list	Displays the rogue ignore list.

Examples: Rogue Detection Configuration

This example shows how to configure the minimum RSSI that a detected rogue AP needs to be at, to have an entry created in the device:

```
Device# wireless wps rogue ap notify-min-rssi 100
```

This example shows how to configure the classification interval:

```
Device# configure terminal
Device(config)#
Device(config)#
Device(config)# end
Device# show wireless wps rogue client /show wireless wps rogue ap summary
```

Configuring Rogue Policies (GUI)

Procedure

Step 1 Choose **Configuration > Security > Wireless Protection Policies**.

Step 2 In the **Rogue Policies** tab, use the **Rogue Detection Security Level** drop-down to select the security level.

Step 3 In the **Expiration timeout for Rogue APs (seconds)** field, enter the timeout value.

Step 4 Select the **Validate Rogue Clients against AAA** check box to validate rogue clients against AAA server.

Step 5 Select the **Validate Rogue APs against AAA** check box to validate rogue access points against AAA server.

Step 6 In the **Rogue Polling Interval (seconds)** field, enter the interval to poll the AAA server for rogue information.

Step 7 Select the **Detect and Report Adhoc Networks** check box to enable detection of rogue adhoc networks.

Step 8 In the **Rogue Detection Client Number Threshold** field, enter the threshold to generate SNMP trap.

Step 9 In the **Auto Contain** section, enter the following details.

Step 10 Use the **Auto Containment Level** drop-down to select the level.

Step 11 Select the **Auto Containment only for Monitor Mode APs** check box to limit the auto-containment only to monitor mode APs.

Step 12 Select the **Rogue on Wire** check box to limit the auto-containment only to rogue APs on wire.

Step 13 Select the **Using our SSID** check box to limit the auto-containment only to rogue APs using one of the SSID configured on the controller.

Step 14 Select the **Adhoc Rogue AP** check box to limit the auto-containment only to adhoc rogue APs.

Step 15 Click **Apply**.

Configuring Rogue Policies (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps rogue ap timeout** *number of seconds* **Example:** `Device(config)# wireless wps rogue ap timeout 250`	Configures the expiration time for rogue entries, in seconds. Valid range for the time in seconds 240 seconds to 3600 seconds.
Step 3	**wireless wps rogue client notify-min-rssi** *RSSI threshold* **Example:**	Configures the minimum RSSI notification threshold for rogue clients. Valid range for the RSSI threshold in dB is -128 - dB to -70 dB.

	Command or Action	Purpose
	`Device(config)# wireless wps rogue client notify-min-rssi -128`	
Step 4	**wireless wps rogue client notify-min-deviation** *RSSI threshold* **Example:** `Device(config)# wireless wps rogue client notify-min-deviation 4`	Configures the RSSI deviation notification threshold for rogue clients. Valid range for the RSSI threshold in dB is 0 dB to 10 dB.
Step 5	**wireless wps rogue ap aaa polling-interval** *AP AAA Interval* **Example:** `Device(config)# wireless wps rogue ap aaa polling-interval 120`	Configures rogue AP AAA validation interval. The valid range for the AP AAA interval in seconds is 60 seconds to 86400 seconds.
Step 6	**wireless wps rogue adhoc** **Example:** `Device(config)# wireless wps rogue adhoc`	Enables detecting and reporting adhoc rogue (IBSS).
Step 7	**wireless wps rogue client client-threshold** *threshold* **Example:** `Device(config)# wireless wps rogue client client-threshold 100`	Configures the rogue client per a rogue AP SNMP trap threshold. The valid range for the threshold is 0 to 256.

Rogue Location Discovery Protocol (RLDP)

Rogue Location Discovery Protocol

Rogue Location Discovery Protocol (RLDP) is an active approach, which is used when rogue AP has no authentication (Open Authentication) configured. This mode, which is disabled by default, instructs an active AP to move to the rogue channel and connect to the rogue as a client. During this time, the active AP sends de-authentication messages to all connected clients and then shuts down the radio interface. Then, it associates to the rogue AP as a client. The AP then tries to obtain an IP address from the rogue AP and forwards a User Datagram Protocol (UDP) packet (port 6352) that contains the local AP and rogue connection information to the controller through the rogue AP. If the controller receives this packet, the alarm is set to notify the network administrator that a rogue AP was discovered on the wired network with the RLDP feature. RLDP has 100 % accuracy in rouge AP detection. It detects Open APs and NAT APs.

Following are some guidelines to manage RLDP:

- Rogue Location Discovery Protocol (RLDP) detects rogue access points that are configured for open authentication.
- RLDP detects rogue access points that use a broadcast Basic Service Set Identifier (BSSID), that is, the access point broadcasts its Service Set Identifier in beacons.

- RLDP detects only those rogue access points that are on the same network. If an access list in the network prevents the sending of RLDP traffic from the rogue access point to the embedded wireless controller, RLDP does not work.
- RLDP does not work on 5-GHz Dynamic Frequency Selection (DFS) channels.
- If RLDP is enabled on mesh APs, and the APs perform RLDP tasks, the mesh APs are dissociated from the embedded wireless controller. The workaround is to disable RLDP on mesh APs.
- If RLDP is enabled on non-monitor APs, client connectivity outages occur when RLDP is in process.

The following steps describe the functioning of RLDP:

1. Identify the closest Unified AP to the rogue using signal strength values.
2. The AP then connects to the rogue as a WLAN client, attempting three associations before timing out.
3. If association is successful, the AP then uses DHCP to obtain an IP address.
4. If an IP address was obtained, the AP (acting as a WLAN client) sends a UDP packet to each of the embedded wireless controller's IP addresses.
5. If the embedded wireless controller receives even one of the RLDP packets from the client, that rogue is marked as on-wire.

Note The RLDP packets are unable to reach the embedded wireless controller if filtering rules are placed between the embedded wireless controller's network and the network where the rogue device is located.

The embedded wireless controller continuously monitors all the nearby access points and automatically discovers and collects information on rogue access points and clients. When the embedded wireless controller discovers a rogue access point, it uses the Rogue Location Discovery Protocol (RLDP), if RLDP is enabled, to determine if the rogue is attached to your network.

Embedded Wireless Controller initiates RLDP on rogue devices that have open . If RLDP uses FlexConnect or local mode access points, then clients are disconnected for that moment. After the RLDP cycle, the clients are reconnected to the access points. As and when rogue access points are seen , the RLDP process is initiated.

You can configure the embedded wireless controller to use RLDP on all the access points or only on the access points configured for the monitor (listen-only) mode. The latter option facilitates automated rogue access point detection in a crowded radio frequency (RF) space, allowing monitoring without creating unnecessary interference and without affecting the regular data access point functionality. If you configure the embedded wireless controller to use RLDP on all the access points, the embedded wireless controller always chooses the monitor access point for RLDP operation if a monitor access point and a local (data) access point are both nearby. If RLDP determines that the rogue is on your network, you can choose to contain the detected rogue either manually or automatically.

RLDP detects on wire presence of the rogue access points that are configured with open authentication only once, which is the default retry configuration. Retries can be configured using the .

You can initiate or trigger RLDP from embedded wireless controller in three ways:

1. Enter the RLDP initiation command manually from the embedded wireless controller CLI.
2. Schedule RLDP from the embedded wireless controller CLI.

3. Auto RLDP. You can configure auto RLDP on embedded wireless controller either from embedded wireless controller CLI or GUI but keep in mind the following guidelines:
    - The auto RLDP option can be configured only when the rogue detection security level is set to custom.
    - Either auto RLDP or schedule of RLDP can be enabled at a time.

Restrictions for RLDP

- RLDP only works with open rogue APs broadcasting their SSID with authentication and encryption disabled.
- RLDP requires that the Managed AP acting as a client is able to obtain an IP address via DHCP on the rogue network.
- Manual RLDP can be used to attempt an RLDP trace on a rogue multiple number of times.
- During RLDP process, the AP is unable to serve clients. This negatively impacts performance and connectivity for local mode APs. To avoid this case, RLDP can be selectively enabled for Monitor Mode AP only.
- RLDP does not attempt to connect to a rogue AP operating in a 5GHz DFS channel.
- RLDP is supported only on Cisco IOS APs.

Configuring RLDP for Generating Alarms (GUI)

Procedure

Step 1 Choose **Configuration > Security > Wireless Protection Policies**.

Step 2 In the **RLDP** tab, use the **Rogue Location Discovery Protocol** drop-down to select one of the following options:

a) **Disable**: Disables RLDP on all the access points. **Disable** is the default option.
b) **All APs**: Enables RLDP on all APs.
c) **Monitor Mode APs**: Enables RLDP only on APs in the monitor mode.

Note The **Schedule RLDP** check box is enabled only if the **Disable** option is selected. The Schedule RLDP check box remains disabled when you select the **All APs** option or the **Monitor Mode APs** option.

Step 3 In the **Retry Count** field, specify the number of retries that should be attempted. The range allowed is between 1 and 5.

Step 4 Click **Apply**.

Configuring an RLDP for Generating Alarms (CLI)

Procedure

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

Step 2	**wireless wps rogue ap rldp alarm-only <monitor-ap-only>**
	Example:
	`Device(config)# wireless wps rogue ap rldp alarm-only`
	`Device(config)# wireless wps rogue ap rldp alarm-only monitor-ap-only`
		Enables RLDP to generate alarms. In this method, the RLDP is always enabled.
		The **monitor-ap-only** keyword is optional.
		The command with just the **alarm-only** keyword enables RLDP without any restriction on the AP mode.
		The command with **alarm-only <monitor-ap-only>** keyword enables RLDP in monitor mode access points only.

Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring a Schedule for RLDP (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies**.

Step 2 In the **RLDP** tab, choose the following options from the **Rogue Location Discovery Protocol** drop-down list:

- **Disable (default)**: Disables RLDP on all the access points.

Step 3 In the Retry Count field, specify the number of retries that should be attempted. Provide a valid range between 1 to 5.

Step 4 Check the **Schedule RLDP** check box and then specify the days, start time, and end time for the process to take place.

Step 5 Click **Apply**.

Configuring a Schedule for RLDP (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps rogue ap rldp schedule day** *day* **start** *start-time* **end** *end-time* **Example:** `Device(config)# wireless wps rogue ap rldp schedule day Monday start 10:10:01 end 12:00:00`	Enables RLDP based on a scheduled day, start time, and end time. Here, *day* is the day when the RLDP scheduling can be done. The values are Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday. *start-time* is the start time for scheduling RLDP for the day. You need to enter start time in **HH:MM:SS** format. *end-time* is the end time for scheduling RLDP for the day. You need to enter end time in **HH:MM:SS** format.
Step 3	**wireless wps rogue ap rldp schedule** **Example:** `Device(config)# wireless wps rogue ap rldp schedule`	Enables the schedule.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring an RLDP for Auto-Contain (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies**.

Step 2 In the **Rogue Policies** tab, under the **Auto Contain** section, check the **Rogue on Wire** checkbox.

Step 3 Click **Apply**.

Configuring an RLDP for Auto-Contain (CLI)

Procedure

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

Step 2	**wireless wps rogue ap rldp auto-contain** [**monitor-ap-only**]
	Example:
	`Device(config)# wireless wps rogue ap rldp auto-contain`
	`Device(config)# wireless wps rogue ap rldp auto-contain monitor-ap-only`
		Enables RLDP to perform auto-contain. In this method, the RLDP is always enabled.
		The **monitor-ap-only** keyword is optional.
		The command with just the **auto-contain** keyword enables RLDP without any restriction on the AP mode.
		The command with **auto-contain <monitor-ap-only>** keyword enables RLDP in monitor mode access points only.

Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring RLDP Retry Times on Rogue Access Points (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies**.

Step 2 On the **Wireless Protection Policies** page, click the **RLDP** tab.

Step 3 Enter the RLDP retry attempt value for rogue access points in the **Retry Count** field.

The valid range is between 1 and 5.

Step 4 Save the configuration.

Configuring RLDP Retry Times on Rogue Access Points (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps rogue ap rldp retries** *num-entries* **Example:** `Device(config)# wireless wps rogue ap rldp retries 2`	Enables RLDP retry times on rogue access points. Here, *num-entries* is the number of RLDP retry times for each of the rogue access points. The valid range is 1 to 5.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Rogue AP RLDP

The following commands can be used to verify rogue AP RLDP:

Table 26: Verifying Rogue AP Information

Command	Purpose
show wireless wps rogue ap rldp detailed *mac_address*	Displays the RLDP details for a rogue AP.
show wireless wps rogue ap rldp in progress	Displays the list of in-progress RLDP.
show wireless wps rogue ap rldp summary	Displays the summary of RLDP scheduling information.

Rogue Detection Security Level

The rogue detection security level configuration allows you to set rogue detection parameters.

The available security levels are:

- Critical: Basic rogue detection for highly sensitive deployments.
- High: Basic rogue detection for medium-scale deployments.
- Low: Basic rogue detection for small-scale deployments.
- Custom: Default security-level, where all detection parameters are configurable.

Note When in Critical, High or Low, some rogue parameters are fixed and cannot be configured.

The following table shows parameter details for the three predefined levels:

Table 27: Rogue Detection: Predefined Levels

Parameter	Critical	High	Low
Cleanup Timer	3600	1200	240
AAA Validate Clients	Disabled	Disabled	Disabled
Adhoc Reporting	Enabled	Enabled	Enabled
Monitor-Mode Report Interval	10 seconds	30 seconds	60 seconds
Minimum RSSI	-128 dBm	-80 dBm	-80 dBm
Transient Interval	600 seconds	300 seconds	120 seconds
Auto Contain Works only on Monitor Mode APs.	Disabled	Disabled	Disabled
Auto Contain Level	1	1	1
Auto Contain Same-SSID	Disabled	Disabled	Disabled
Auto Contain Valid Clients on Rogue AP	Disabled	Disabled	Disabled
Auto Contain Adhoc	Disabled	Disabled	Disabled
Containment Auto-Rate	Enabled	Enabled	Enabled
Validate Clients with CMX	Enabled	Enabled	Enabled
Containment FlexConnect	Enabled	Enabled	Enabled
RLDP	Monitor-AP if RLDP scheduling is disabled.	Monitor-AP if RLDP scheduling is disabled	Disabled
Auto Contain RLDP	Disabled	Disabled	Disabled

Setting Rogue Detection Security-level

Follow the procedure given below to set the rogue detection security-level:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 2	**wireless wps rogue security-level custom** **Example:** `Device(config)# wireless wps rogue security-level custom`	Configures rogue detection security level as custom.
Step 3	**wireless wps rogue security-level low** **Example:** `Device(config)# wireless wps rogue security-level low`	Configures rogue detection security level for basic rogue detection setup for small-scale deployments.
Step 4	**wireless wps rogue security-level high** **Example:** `Device(config)# wireless wps rogue security-level high`	Configures rogue detection security level for rogue detection setup for medium-scale deployments.
Step 5	**wireless wps rogue security-level critical** **Example:** `Device(config)# wireless wps rogue security-level critical`	Configures rogue detection security level for rogue detection setup for highly sensitive deployments.

Wireless Service Assurance Rogue Events

Wireless Service Assurance (WSA) rogue events, supported in Release 16.12.x and later releases, consist of telemetry notifications for a subset of SNMP traps. WSA rogue events replicate the same information that is part of the corresponding SNMP trap.

For all the exported events, the following details are provided to the wireless service assurance (WSA) infrastructure:

- MAC address of the rogue AP
- Details of the managed AP and the radio that detected the rogue AP with strongest RSSI
- Event-specific data such as SSID, channel for potential honeypot event, and MAC address of the impersonating AP for impersonation event

The WSA rogue events feature can scale up to four times the maximum number of supported APs and half of the maximum number of supported clients.

The WSA rogue events feature is supported on Cisco DNA Center and other third-party infrastructure.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**network-assurance enable** **Example:** `Device# network-assurance enable`	Enables wireless service assurance.
Step 3	**wireless wps rogue network-assurance enable** **Example:** `Device# wireless wps rogue network-assurance enable`	Enables wireless service assurance for rogue devices. This ensures that the WSA rogue events are sent to the event queue.

Monitoring Wireless Service Assurance Rogue Events

Procedure

- **show wireless wps rogue stats**

  Example:

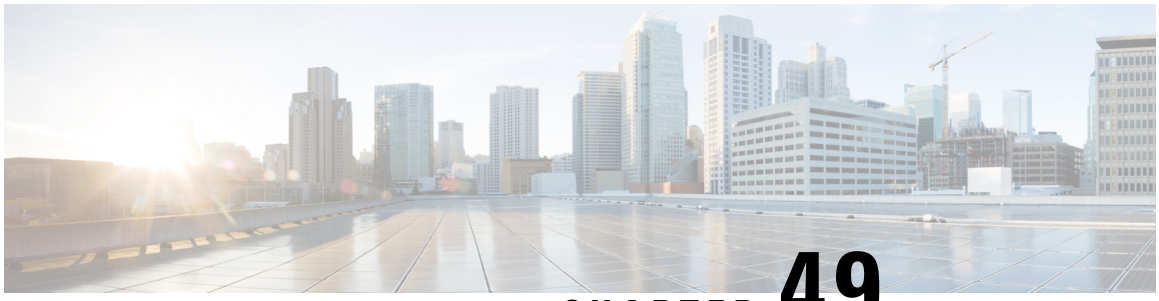
```
Device# show wireless wps rogue stats

WSA Events
  Total WSA Events Triggered            : 9
    ROGUE_POTENTIAL_HONEYPOT_DETECTED   : 2
    ROGUE_POTENTIAL_HONEYPOT_CLEARED    : 3
    ROGUE_AP_IMPERSONATION_DETECTED     : 4
  Total WSA Events Enqueued             : 6
    ROGUE_POTENTIAL_HONEYPOT_DETECTED   : 1
    ROGUE_POTENTIAL_HONEYPOT_CLEARED    : 2
    ROGUE_AP_IMPERSONATION_DETECTED     : 3
```

  In this example, nine events have been triggered, but only six of them have been enqueued. This is because three events were triggered before the WSA rogue feature was enabled.

- **show wireless wps rogue stats internal**

  show wireless wps rogue ap detailed *rogue-ap-mac-addr*

  These commands show information related to WSA events into the event history.



CHAPTER 49

Classifying Rogue Access Points

Information About Classifying Rogue Access Points

The embedded wireless controller software enables you to create rules that can organize and display rogue access points as Friendly, Malicious, or Unclassified.

By default, none of the classification rules are used. You need to enable them. Therefore, all unknown access points are categorized as Unclassified. When you create or change a rule, configure conditions, and enable it, all rogue access points are then reclassified. Whenever you change a rule, it is applied to all the access points (friendly, malicious, and unclassified).

Note

- Rule-based rogue classification does not apply to ad hoc rogues and rogue clients.
- You can configure up to 64 rogue classification rules per embedded wireless controller.

When the embedded wireless controller receives a rogue report from one of its managed access points, it responds as follows:

- If the unknown access point is in the friendly MAC address list, the embedded wireless controller classifies the access point as Friendly.
- If the unknown access point is not in the friendly MAC address list, the embedded wireless controller starts applying the rogue classification rules to the access point.
-
- If the rogue access point matches the configured rules criteria, the embedded wireless controller classifies the rogue based on the classification type configured for that rule.
- If the rogue access point does not match any of the configured rules, the rogue remains unclassified.

  The embedded wireless controller repeats the previous steps for all the rogue access points.

- If the rogue access point is detected on the same wired network, the embedded wireless controller marks the rogue state as Threat and classifies it as Malicious automatically, even if there are no configured rules. You can then manually contain the rogue to change the rogue state to Contained. If the rogue access point is not available on the network, the embedded wireless controller marks the rogue state as Alert. You can then manually contain the rogue.
- If desired, you can manually move the access point to a different classification type and rogue state.

Table 28: Classification Mapping

Rule-Based Classification Type	Rogue State
Friendly	• Internal—If the unknown access point poses no threat to WLAN security, you can manually configure it as Friendly, Internal. An example of this would be the access points in your lab network. • External—If the unknown access point is outside the network and poses no threat to WLAN security, you can manually configure it as Friendly, External. An example of this would be the access point in your neighboring coffee shop. • Alert—
Malicious	• Alert— • Threat—The unknown access point is found to be on the network and poses a threat to WLAN security. • Contained—The unknown access point is contained.
Unclassified	• Alert— • Contained—The unknown access point is contained.

As mentioned earlier, the embedded wireless controller can automatically change the classification type and rogue state of an unknown access point based on user-defined rules. Alternatively, you can manually move the unknown access point to a different classification type and rogue state.

Guidelines and Restrictions for Classifying Rogue Access Points

- Classifying Custom type rogues is tied to rogue rules. Therefore, it is not possible to manually classify a rogue as Custom. Custom class change can occur only when rogue rules are used.
- Some are sent for containment by rule and every 30 minutes for rogue classification change.
- Rogue rules are applied on every incoming new rogue report in the embedded wireless controller in the order of their priority.
- After a rogue satisfies a rule and is classified, it does not move down the priority list for the same report.
- If a rogue AP is classified as friendly

- Until the controller discovers all the APs through neighbor reports from APs, the rogue APs are kept in unconfigured state for three minutes after they are detected. After 3 minutes, the rogue policy is applied on the rogue APs and the APs are moved to unclassified, friendly, malicious, or custom class. Rogue APs kept in unconfigured state means that no rogue policy has yet been applied on them.
- When a rogue BSSID is submitted for a containment on Cisco Catalyst 9800 Series Wireless Controller, if the controller has enough resources, it will contain. The APs that detect the particular contained rogue AP starts broadcasting the DEAUTH packets.

  Wireless client connected to the contained rogue BSSID will disconnect once DEAUTH packets are received. However, when the client assumes being in a connected state, repeatedly tries to reconnect and the wireless client's user browsing experience would be badly affected.

  Also, in a high RF environment like that of a stadium, though DEAUTH packets are broadcasted, client does not receive all of them because of RF disturbance. In this scenario, the client may not be fully disconnected but will be affected badly.
- The rouge AP manual classification limit has been enhanced from 625 to 10,000 configurations at a time. The rouge client manual classification limit has been enhanced from 625 to 10,000 configurations at a time.

How to Classify Rogue Access Points

Classifying Rogue Access Points and Clients Manually (GUI)

Procedure

Step 1 Choose **Monitoring** > **Wireless** > **Rogues**.

Step 2 In the **Unclassified** tab, select an AP to view the detail in the lower pane.

Step 3 Use the **Class Type** drop-down to set the status.

Step 4 Click **Apply**.

Classifying Rogue Access Points and Clients Manually (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps rogue adhoc** {**alert** *mac-addr* | **auto-contain** | **contain** *mac-addr*	Detects and reports the ad hoc rogue.

	Command or Action	Purpose
	containment-level | **internal** *mac-addr* | **external** *mac-addr*} **Example:** `Device(config)# wireless wps rogue adhoc alert 74a0.2f45.c520`	Enter one of these options after you enter the **adhoc** keyword: • **alert**—Sets the ad hoc rogue access point to alert mode. If you choose this option, enter the MAC address for the *mac-addr* parameter. • **auto-contain**—Sets the automatically containing ad hoc rogue to auto-contain mode. • **contain**—Sets the containing ad hoc rogue access point to contain mode. If you choose this option, enter the MAC address for the *mac-addr* parameter and containment level for the *containment-level* parameter. The valid range for *containment-level* is from 1 to 4. • **external**—Sets the ad hoc rogue access point as **external**. If you choose this option, enter the MAC address for the *mac-addr* parameter. • **internal**—Sets the ad hoc rogue access point as **internal**. If you choose this option, enter the MAC address for the *mac-addr* parameter.
Step 3	**wireless wps rogue ap** {**friendly** *mac-addr* **state** [**external** | **internal**] | **malicious** *mac-addr* **state** [**alert** | **contain** *containment-level*]} **Example:** `Device(config)# wireless wps rogue ap malicious 74a0.2f45.c520 state contain 3`	Configures the rogue access points. Enter one of the following options after the **ap** keyword: • **friendly**—Configures the friendly rogue access points. If you choose this option, enter the MAC address for the *mac-addr* parameter. After that enter the **state** keyword followed by either of these options: **internal** or **external**. If you select an **internal** option, it indicates that you trust a foreign access point. If you select an **external** option, it indicates that you acknowledge the presence of a rogue access point. • **malicious**—Configures the malicious rogue access points. If you choose this option, enter the MAC address for the *mac-addr* parameter. After that enter the **state** keyword followed by either of these options: **alert** or **contain**.

	Command or Action	Purpose
		• **alert**—Sets the malicious rogue access point to **alert** mode. • **contain**—Sets the malicious rogue access point to **contain** mode. If you choose this option, enter the containment level for the *containment-level* parameter. The valid range is from 1 to 4.
Step 4	**wireless wps rogue client** {**contain** *mac-addr containment-level*} **Example:** `Device(config)# wireless wps rogue client contain 74a0.2f45.c520 2`	Configures the rogue clients. Enter the following option after you enter the **client** keyword: **contain**—Contains the rogue client. After you choose this option, enter the MAC address for the *mac-addr* parameter and the containment level for *containment-level* parameter. The valid range for *containment-level* is from 1 to 4.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Rogue Classification Rules (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless Protection Policies**.

Step 2 In the **Wireless Protection Policies** page, choose **Rogue AP Rules** tab.

Step 3 On the **Rogue AP Rules** page, click the name of the **Rule** or click **Add** to create a new one.

Step 4 In the **Add/Edit Rogue AP Rule** window that is displayed, enter the name of the rule in the **Rule Name** field.

Step 5 Choose the rule type from the following **Rule Type** drop-down list options:

- Friendly
- Malicious
- Unclassified
- Custom

Configuring Rogue Classification Rules (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless wps rogue rule** *rule-name* **priority** *priority* **Example:** `Device(config)# wireless wps rogue rule rule_3 priority 3`	Creates or enables a rule. While creating a rule, you must enter the priority for the rule. **Note** After creating a rule, you can edit the rule and change the priority only for the rogue rules that are disabled. You cannot change the priority for the rogue rules that are enabled. While editing, changing the priority for a rogue rule is optional.

Step 3	**classify** {**friendly state** {**alert** | **external** | **internal**} | **malicious state** {**alert** | **contained** }}
	Example:
	`Device(config)# wireless wps rogue rule rule_3 priority 3`
	`Device(config-rule)# classify friendly`
		• **friendly**—Configures the friendly rogue access points. After that enter the **state** keyword followed by either of these options: **alert**, **internal**, or **external**. If you select an **internal** option, it indicates that you trust a foreign access point. If you select an **external** option, it indicates that you acknowledge the presence of a rogue access point.
		• **malicious**—Configures the malicious rogue access points. After that enter the state keyword followed by either of these options: alert or contained.
		• **alert**—Sets the malicious rogue access point to alert mode.
		• **contained**—Sets the malicious rogue access point to contained mode.

Step 4	**condition** {**client-count** | **duration** | **encryption** | **infrastructure** | **rssi** | **ssid**}
	Example:
	`Device(config)# wireless wps rogue rule rule_3 priority 3`
	`Device(config-rule)# condition client-count 5`
		Adds the following conditions to a rule, which the rogue access point must meet:
		• **client-count**—Requires that a minimum number of clients be associated to the rogue access point. For example, if the number of clients associated to the rogue access point is greater than or equal to the configured value, the access point could be classified as Malicious. If you choose

	Command or Action	Purpose
		this option, enter the minimum number of clients to be associated to the rogue access point for the parameter. The valid range is from 1 to 10 (inclusive), and the default value is 0. • **duration**—Requires that the rogue access point be detected for a minimum period of time. If you choose this option, enter a value for the minimum detection period for the parameter. The valid range is from 0 to 3600 seconds (inclusive), and the default value is 0 seconds. • **encryption**—Requires that the advertised WLAN does not have encryption enabled. You can choose **any** for any type of encryption, **off** for no encryption, **wpa1** for WPA encryption, **wpa2** for WPA2 encryption, **wpa3-owe** for WPA3 OWE encryption, or **wpa3-sae** for WPA3 SAE encryption. • **infrastructure**—Requires the SSID to be known to the controller. • **rssi**—The valid range is from –95 to –50 dBm (inclusive). • **ssid**—Requires the rogue access point to have a specific SSID. You could specify up to 25 different SSIDs. You should specify an SSID that is not managed by the controller. If you choose this option, enter the SSID for the parameter. • **wildcard-ssid**—Allows you to specify an expression that could match an SSID string. You can specify up to 25 of these SSIDs.

Step 5	**match** {**all** | **any**}
	Example:
	`Device(config)# wireless wps rogue rule rule_3 priority 3`
	`Device(config-rule)# match all`
		Specifies whether a detected rogue access point must meet all or any of the conditions specified by the rule for the rule to be matched and the rogue access point to adopt the classification type of the rule.

Step 6	**default** **Example:** `Device(config)# wireless wps rogue rule rule_3 priority 3`	Sets a command to its default.

	Command or Action	Purpose
	`Device(config-rule)# default`	

Step 7	**exit**
	Example:
	`Device(config)# wireless wps rogue rule rule_3 priority 3`
	`Device(config-rule)# exit`
	`Device(config)#`
		Exits the sub-mode.

Step 8	**shutdown**
	Example:
	`Device(config)# wireless wps rogue rule rule_3 priority 3`
	`Device(config-rule)# shutdown`
		Disables a particular rogue rule. In this example, the rule **rule_3** is disabled.

Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 10	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 11	**wireless wps rogue rule shutdown** **Example:** `Device(config)# wireless wps rogue rule shutdown`	Disables all the rogue rules.
Step 12	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Monitoring Rogue Classification Rules

You can monitor the rogue classification rules using the following commands:

Table 29: Commands for Monitoring Rogue Classification Rules

Command	Purpose
show wireless wps rogue rule detailed	Displays detailed information of a classification rule.
show wireless wps rogue rule summary	Displays a summary of the classification rules.

Examples: Classifying Rogue Access Points

This example shows how to classify a rogue AP with MAC address 00:11:22:33:44:55 as malicious and mark it for being contained by 2 managed APs:

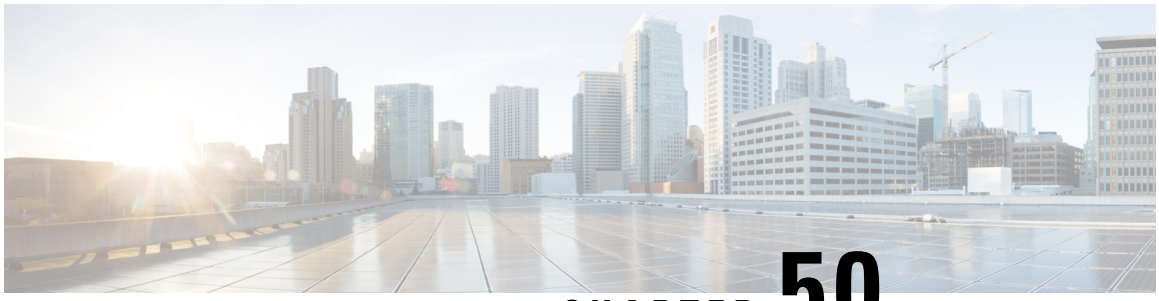
```
Device# configure terminal
Device(config)# wireless wps rogue ap malicious 0011.2233.4455 state contain 2
```

This example shows how to create a rule that can categorize a rogue AP that is using SSID **my-friendly-ssid**, and it is seen for at least for 1000 seconds as friendly internal:

```
Device# configure terminal
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)# condition ssid my-friendly-ssid
Device(config-rule)# condition duration 1000
Device(config-rule)# match all
Device(config-rule)# classify friendly state internal
```

This example shows how to apply a condition that a rogue access point must meet:

```
Device# configure terminal
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)# condition client-count 5
Device(config-rule)# condition duration 1000
Device(config-rule)# end
```



CHAPTER 50

Configuring Secure Shell

Information About Configuring Secure Shell

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 Device Access

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH Servers, Integrated Clients, and Supported Versions

The Secure Shell (SSH) Integrated Client feature is an application that runs over the SSH protocol to provide device authentication and encryption. The SSH client enables a Cisco device to make a secure, encrypted connection to another Cisco device or to any other device running the SSH server. This connection provides functionality similar to that of an outbound Telnet connection except that the connection is encrypted. With authentication and encryption, the SSH client allows for secure communication over an unsecured network.

The SSH server and SSH integrated client are applications that run on the switch. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client works with publicly and commercially available SSH servers. The SSH client supports the ciphers of Data Encryption Standard (DES), 3DES, and password authentication.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

Note The SSH client functionality is available only when the SSH server is enabled.

User authentication is performed like that in the Telnet session to the device. SSH also supports the following user authentication methods:

- TACACS+
- RADIUS
- Local authentication and authorization

SSH Configuration Guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If the SSH server is running on an active switch and the active switch fails, the new active switch uses the RSA key pair generated by the previous active switch.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command.
- When generating the RSA key pair, the message No host name specified might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message No domain specified might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

Secure Copy Protocol Overview

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.

Note When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

Secure Copy Protocol

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying device configurations or switch image files. The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the device can determine whether the user has the correct privilege level. To configure the Secure Copy feature, you should understand the SCP concepts.

SFTP Support

SFTP client support is introduced from Cisco IOS XE Gibraltar 16.10.1 release onwards. SFTP client is enabled by default and no separate configuration required.

The SFTP procedures can be invoked using the **copy** command, which is similar to that of **scp** and **tftp** commands. A typical file download procedure using **sftp** command can be carried out as shown below:

copy sftp://*user* :*password* @*server-ip*/*file-name* **flash0://** *file-name*

For more details on the **copy** command, see the following URL:
https://www.cisco.com/c/m/en_us/techdoc/dc/reference/cli/nxos/commands/fund/copy.html

Prerequisites for Configuring Secure Shell

The following are the prerequisites for configuring the switch for secure shell (SSH):

- For SSH to work, the switch needs an Rivest, Shamir, and Adleman (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.
- The Secure Shell (SSH) server requires an IPsec (Data Encryption Standard [DES] or 3DES) encryption software image; the SSH client requires an IPsec (DES or 3DES) encryption software image.)

- Configure a hostname and host domain for your device by using the **hostname** and **ip domain-name** commands in global configuration mode.

Restrictions for Configuring Secure Shell

The following are restrictions for configuring the device for secure shell.

- From Cisco IOS XE Dublin 17.10.x, Key Exchange and MAC algorithms like diffie-hellman-group14-sha1, hmac-sha1, hmac-sha2-256, and hmac-sha2-512 are not supported by default and it may impact some SSH clients that only support these algorithms. However, you can add them manually if required. For information on manually adding these algorithms, see the **SSH Algorithms for Common Criteria Certification** document available at: https://www.cisco.com/c/en/us/td/docs/routers/ios/config/17-x/sec-vpn/b-security-vpn/m_sec-secure-shell-algorithm-ccc.html
- 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 Data Encryption Standard (DES) (56-bit) and 3DES (168-bit) data encryption software. In DES software images, DES is the only encryption algorithm available. In 3DES software images, both DES and 3DES encryption algorithms are available.
- The device 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.
- When using SCP, you cannot enter the password into the **copy** command. You must enter the password when prompted.
- The login banner is not supported in Secure Shell Version 1. It is supported in Secure Shell Version 2.
- The -l keyword and userid :{number} {ip-address} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for console access.
- To authenticate clients with freeradius over RADSEC, you should generate an RSA key longer than 1024 bit. Use the **crypto key generate rsa general-keys exportable label** *label-name* command to achieve this.

How to Configure SSH

Setting Up the Device to Run SSH

Follow the procedure given below to set up your device to run SSH:

Before you begin

Configure user authentication for local or remote access.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# Device# configure terminal`	Enters global configuration mode.
Step 2	**hostname** *hostname* **Example:** `Device(config)# hostname your_hostname`	Configures a hostname and IP domain name for your device. **Note** Follow this procedure only if you are configuring the device as an SSH server.
Step 3	**ip domain name** *domain_name* **Example:** `Device(config)# ip domain name your_domain`	Configures a host domain for your device.
Step 4	**crypto key generate rsa** **Example:** `Device(config)# crypto key generate rsa`	Enables the SSH server for local and remote authentication on the device and generates an RSA key pair. Generating an RSA key pair for the device automatically enables SSH. We recommend that a minimum modulus size of 1024 bits. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use. **Note** Follow this procedure only if you are configuring the device as an SSH server.
Step 5	**end** **Example:** `Device(config)# end`	Exits configuration mode.

Configuring the SSH Server

Follow the procedure given below to configure the SSH server:

Note This procedure is only required if you are configuring the device as an SSH server.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ip ssh version** [2] **Example:** `Device(config)# ip ssh version 2`	(Optional) Configures the device to run SSH Version 2.
Step 3	**ip ssh** {**timeout** *seconds* | **authentication-retries** *number*} **Example:** `Device(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 device 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:
	`Device(config)# line vty 1 10`
	or
	`Device(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 device prevent non-SSH Telnet connections. This limits the router to only SSH connections.

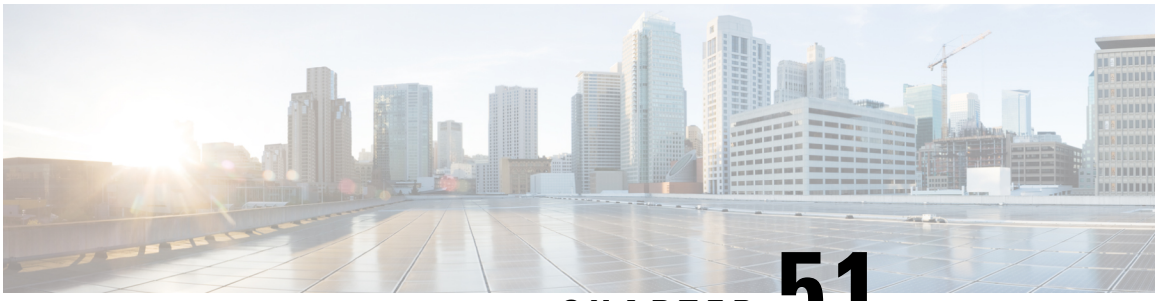
	Command or Action	Purpose
		Note If the Virtual Terminal (VTY) lines are exhausted, Telnet or SSH will fail. You can either disconnect the Telnet or SSH sessions to free up the VTY lines, or follow the recovery steps given below to clear VTY lines and reload Telnet or SSH: `Device# configure terminal` `Device(config)# clear line` *`line number`*
Step 5	**end** **Example:** `Device(config-line)# end`	Returns to privileged EXEC mode.

Monitoring the SSH Configuration and Status

This table displays the SSH server configuration and status.

Table 30: 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.



CHAPTER 51

Private Shared Key

Information About Private Preshared Key

With the advent of Internet of Things (IoT), the number of devices that connect to the internet has increased multifold. Not all of these devices support the 802.1x supplicant and need an alternate mechanism to connect to the internet. One of the security mechanisms, WPA-PSK, could be considered as an alternative. With the current configuration, the PSK is the same for all the clients that connect to the same WLAN. In certain deployments, such as educational institutions, this results in the key being shared to unauthorized users leading to security breach. This necessitates the need to provision unique PSKs for different clients on a large scale.

Identity PSKs are unique PSKs created for individuals or groups of users on the same SSID. No complex configuration is required for the clients. It provides the same simplicity of PSK, making it ideal for IoT, Bring your own device (BYOD), and guest deployments. The default password for PSK SSID is *password*.

Identity PSKs are supported on most devices, in which 802.1X is not, enabling stronger security for IoT. It is possible to easily revoke access, for a single device or individual without affecting everyone else. Thousands of keys can easily be managed and distributed through the AAA server.

Note Special characters, such as '<' and '>' are not supported in SSID Preshared key.

Note PSK supports whitespace in passwords (before or after or in-between) within double quotes only; single quotes for whitespaces are not supported.

IPSK Solution

During client authentication, the AAA server authorizes the client MAC address and sends the passphrase (if configured) as part of the Cisco-AV pair list. The Embedded Wireless Controller receives this as part of the RADIUS response and processes this further for the computation of PSKs.

When a client sends an association request to the SSID broadcast by the corresponding access point, the Embedded Wireless Controller forms the RADIUS request packet with the particular mac address of the client and relays to the RADIUS server.

The RADIUS server performs the authentication and checks whether the client is allowed or not and sends either ACCESS-ACCEPT or ACCESS-REJECT as response to the WLC.

To support Identity PSKs, in addition to sending the authentication response, the authentication server also provides the AV pair passphrase for this specific client. This is used for the computation of the PMK.

The RADIUS server might also provide additional parameters, such as username, VLAN, Quality of Service (QoS), and so on, in the response, that is specific to this client. For multiple devices owned by a single user, the passphrase can remain the same.

Note When the PSK length is less than 15 characters in Federal Information Processing Standard (FIPS), the controller allows the WLAN configuration but displays the following error message on the console:

"AP is allowed to join but corresponding WLAN will not be pushed to the access point"

Configuring a PSK in a WLAN (CLI)

Follow the procedure given below to configure a PSK in a WLAN:

Before you begin

- Security should be configured for a pre-shared key (PSK) in a WLAN.
- If there is no override from the AAA server, the value on the corresponding WLAN is considered for authentication.
- In Federal Information Processing Standard (FIPS) and common criteria mode, ensure that the PSK WLAN has a minimum of 15 ASCII characters, else APs won't join the controller.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name wlan-id ssid* **Example:** `Device(config)# wlan test-profile 4 abc`	Configures the WLAN and SSID.

	Command or Action	Purpose
Step 3	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 4	**security wpa akm psk** **Example:** `Device(config-wlan)# security wpa akm psk`	Configures the security type PSK.
Step 5	**security wpa akm psk set-key** *ascii/hex key* **Example:** `Device(config-wlan)# security wpa akm psk set-key asci 0`	Configures the PSK authenticated key management (AKM) shared key.
Step 6	**security wpa akm psk** **Example:** `Device(config-wlan)# security wpa akm psk`	Configures PSK support.
Step 7	**mac-filtering** *auth-list-name* **Example:** `Device(config-wlan)# mac-filtering test1`	Specifies MAC filtering in a WLAN.

Configuring a PSK in a WLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **Wireless Networks** page, click **Security** tab.

Step 3 In the **Layer 2** window that is displayed, go to the **WPA Parameters** section.

Step 4 From the **Auth Key Mgmt** drop-down, select the PSK format and type.

Step 5 Enter the Pre-Shared Key in hexadecimal characters.

- If you selected the PSK format as HEX, the key length must be exactly 64 characters.
- If you selected the PSK format as ASCII, the key length must be in the range of 8-63 characters.

Note that once you have configured the key, these details are not visible even if you click on the eye icon next to the preshared key box, due to security reasons.

Step 6 Click **Save & Apply to Device**.

Applying a Policy Profile to a WLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 On the**Manage Tags** page, click **Policy** tab.

Step 3 Click **Add** to view the **Add Policy Tag** window.

Step 4 Enter a name and description for the policy tag.

Step 5 Click **Add** to map WLAN and policy.

Step 6 Choose the WLAN profile to map with the appropriate policy profile, and click the tick icon.

Step 7 Click **Save & Apply to Device**.

Applying a Policy Profile to a WLAN (CLI)

Follow the procedure given below to a apply policy profile to a WLAN:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-profile-name* **Example:** `Device(config)# wireless profile policy policy-iot`	Configures the default policy profile.
Step 3	**aaa-override** **Example:** `Device(config-wireless-policy)# aaa-override`	Configures AAA override to apply policies coming from the AAA server or ISE the Cisco Identify Services Engine (ISE) server.

Verifying a Private PSK

Use the following **show** commands to verify the configuration of a WLAN and a client:

```
Device# show wlan id 2

WLAN Profile Name     : test_ppsk
```

```
================================================
Identifier                                      : 2
Network Name (SSID)                             : test_ppsk
Status                                          : Enabled
Broadcast SSID                                  : Enabled
Universal AP Admin                              : Disabled
Max Associated Clients per WLAN                 : 0
Max Associated Clients per AP per WLAN          : 0
Max Associated Clients per AP Radio per WLAN    : 0
Number of Active Clients                        : 0
Exclusionlist Timeout                           : 60
CHD per WLAN                                    : Enabled
Interface                                       : default
Multicast Interface                             : Unconfigured
WMM                                             : Allowed
WifiDirect                                      : Invalid
Channel Scan Defer Priority:
  Priority (default)                            : 4
  Priority (default)                            : 5
  Priority (default)                            : 6
Scan Defer Time (msecs)                         : 100
Media Stream Multicast-direct                   : Disabled
CCX - AironetIe Support                         : Enabled
CCX - Diagnostics Channel Capability            : Disabled
Peer-to-Peer Blocking Action                    : Disabled
Radio Policy                                    : All
DTIM period for 802.11a radio                   : 1
DTIM period for 802.11b radio                   : 1
Local EAP Authentication                        : Disabled
Mac Filter Authorization list name              : test1
Accounting list name                            : Disabled
802.1x authentication list name                 : Disabled
Security
    802.11 Authentication                       : Open System
    Static WEP Keys                             : Disabled
    802.1X                                      : Disabled
    Wi-Fi Protected Access (WPA/WPA2)           : Enabled
        WPA (SSN IE)                            : Disabled
        WPA2 (RSN IE)                           : Enabled
            TKIP Cipher                         : Disabled
            AES Cipher                          : Enabled
        Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Enabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            PMF dot1x                           : Disabled
            PMF PSK                             : Disabled
    CCKM TSF Tolerance                          : 1000
    FT Support                                  : Disabled
        FT Reassociation Timeout                : 20
        FT Over-The-DS mode                     : Enabled
    PMF Support                                 : Disabled
        PMF Association Comeback Timeout        : 1
        PMF SA Query Time                       : 200
    Web Based Authentication                    : Disabled
    Conditional Web Redirect                    : Disabled
    Splash-Page Web Redirect                    : Disabled
    Webauth On-mac-filter Failure               : Disabled
    Webauth Authentication List Name            : Disabled
    Webauth Parameter Map                       : Disabled
    Tkip MIC Countermeasure Hold-down Timer     : 60
Call Snooping                                   : Disabled
```

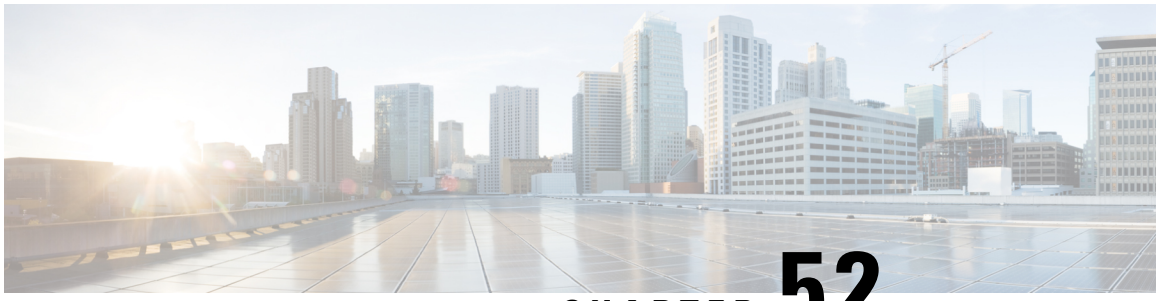
```
Passive Client                                  : Disabled
Non Cisco WGB                                   : Disabled
Band Select                                     : Disabled
Load Balancing                                  : Disabled
Multicast Buffer                                : Disabled
Multicast Buffer Size                           : 0
IP Source Guard                                 : Disabled
Assisted-Roaming
    Neighbor List                               : Disabled
    Prediction List                             : Disabled
    Dual Band Support                           : Disabled
IEEE 802.11v parameters
    Directed Multicast Service                  : Disabled
    BSS Max Idle                                : Disabled
        Protected Mode                          : Disabled
    Traffic Filtering Service                   : Disabled
    BSS Transition                              : Enabled
        Disassociation Imminent                 : Disabled
            Optimised Roaming Timer             : 40
            Timer                               : 200
    WNM Sleep Mode                              : Disabled
802.11ac MU-MIMO                                : Disabled
```

```
Device# show wireless client mac-address a886.adb2.05f9 detail
```

```
Client MAC Address : a886.adb2.05f9
Client IPv4 Address : 9.9.58.246
Client Username : A8-86-AD-B2-05-F9
AP MAC Address : c025.5c55.e400
AP Name: saurabh-3600
AP slot : 1
Client State : Associated
Policy Profile : default-policy-profile
Flex Profile : default-flex-profile
Wireless LAN Id : 6
Wireless LAN Name: SSS_PPSK
BSSID : c025.5c55.e40f
Connected For : 280 seconds
Protocol : 802.11n - 5 GHz
Channel : 60
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Client CCX version : No CCX support
Session Timeout : 320 sec (Remaining time: 40 sec)
Input Policy Name  :
Input Policy State : None
Input Policy Source : None
Output Policy Name  :
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Power Save : OFF
Current Rate : m22
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
```

```
  Mobility Complete Timestamp : 09/27/2017 16:32:25 IST
Policy Manager State: Run
NPU Fast Fast Notified : No
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 280 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : PSK
AAA override passphrase: Yes
Management Frame Protection : No
Protected Management Frame - 802.11w : No
EAP Type : Not Applicable
VLAN : 58
Access VLAN : 58
Anchor VLAN : 0
WFD capable : No
Manged WFD capable : No
Cross Connection capable : No
Support Concurrent Operation : No
Session Manager:
  Interface        : capwap_90000005
  IIF ID           : 0x90000005
  Device Type      : Apple-Device
  Protocol Map     : 0x000001
  Authorized       : TRUE
  Session timeout  : 320
  Common Session ID: 1F3809090000005DC30088EA
  Acct Session ID  : 0x00000000
  Auth Method Status List
        Method : MAB
                SM State        : TERMINATE
                Authen Status   : Success
  Local Policies:
        Service Template : wlan_svc_default-policy-profile (priority 254)
                Absolute-Timer   : 320
                VLAN             : 58
  Server Policies:
  Resultant Policies:
                VLAN             : 58
                Absolute-Timer   : 320
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
FlexConnect Data Switching : Local
FlexConnect Dhcp Status : Local
FlexConnect Authentication : Central
FlexConnect Central Association : No
Client Statistics:
  Number of Bytes Received : 59795
  Number of Bytes Sent : 21404
  Number of Packets Received : 518
  Number of Packets Sent : 274
  Number of EAP Id Request Msg Timeouts :
  Number of EAP Request Msg Timeouts :
  Number of EAP Key Msg Timeouts :
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -32 dBm
```

```
  Signal to Noise Ratio : 58 dB
Fabric status : Disabled
```



CHAPTER 52

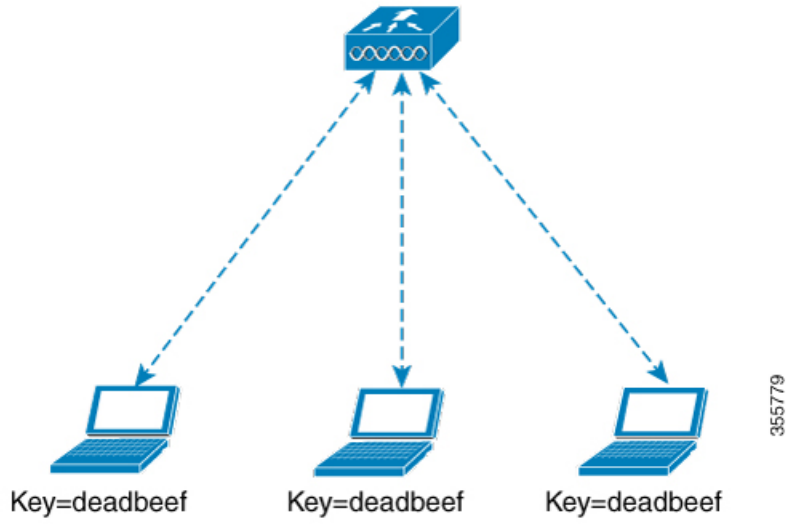
Multi-Preshared Key

Information About Multi-Preshared Key

Multi-PSK feature supports multiple PSKs simultaneously on a single SSID. You can use any of the configured PSKs to join the network. This is different from the Identity PSK (iPSK), wherein unique PSKs are created for individuals or groups of users on the same SSID.

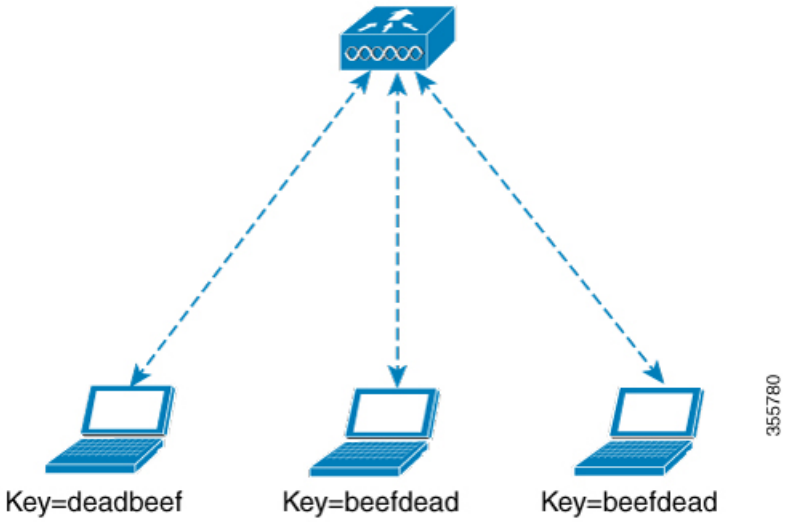
In a traditional PSK, all the clients joining the network use the same password as shown in the below figure.

Figure 19: Traditional PSK



But with multi-PSK, client can use any of the configured pre-shared keys to connect to the network as shown in the below figure.

Figure 20: Multi-PSK



In Multi-PSK, two passwords are configured (deadbeef and beefdead) for the same SSID. In this scenario, clients can connect to the network using either of the passwords.

Restrictions on Multi-PSK

- Central authentication is supported in local, flex, and fabric modes only.
- In central authentication flex mode, the standalone AP allows client join with the highest priority PSK (*priority 0* key). New clients that do not use the highest priority PSK are rejected during the standalone mode.
- Multi-PSK does not support local authentication.

Configuring Multi-Preshared Key (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **Wireless Networks** page, click the name of the WLAN.

Step 3 In the **Edit WLAN** window, click the **Security** tab.

Step 4 In the **Layer2** tab, choose the **Layer2 Security Mode** from the following options:

- None: No Layer 2 security
- 802.1X: WEP 802.1X data encryption type
- WPA + WPA2: Wi-Fi Protected Access
- Static WEP: Static WEP encryption parameters
- Static WEP+802.1X: Both Static WEP and 802.1X parameters

Parameters	Description
802.1X	
WEP Key Size	Choose the key size. The available values are *None*, *40 bits*, and *104 bits*.
WPA + WPA2	
Protected Management Frame	Choose from the following options: • Disabled • Optional • Required
WPA Policy	Check the check box to enable WPA policy.
WPA Encryption	Choose the WPA encryption standard. A WPA encryption standard must be specified if you have enabled WPA policy.
WPA2 Policy	Check the check box to enable WPA2 policy.
WPA2 Encryption	Choose the WPA2 encryption standard. A WPA encryption standard must be specified if you have enabled WPA policy.
Auth Key Mgmt	Choose the rekeying mechanism from the following options: • 802.1X • FT + 802.1X • PSK: You must specify the PSK format and a preshared key • Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value • 802.1X + Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value • FT + 802.1X + Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value
Static WEP	

Parameters	Description
Key Size	Choose the key size from the following options: • 40 bits • 104 bits
Key Index	Choose a key index from 1 to 4. One unique WEP key index can be applied to each WLAN. As there are only four WEP key indexes, only four WLANs can be configured for static WEP Layer2 encryption.
Key Format	Choose the encryption key format as either ASCII or HEX.
Encryption Key	Enter an encryption key that is 13 characters long.
Static WEP + 802.1X	
Key Size	Choose the key size from the following options: • 40 bits • 104 bits
Key Index	Choose a key index from 1 to 4. One unique WEP key index can be applied to each WLAN. As there are only four WEP key indexes, only four WLANs can be configured for static WEP Layer2 encryption.
Key Format	Choose the encryption key format as either ASCII or HEX.
Encryption Key	Enter an encryption key that is 13 characters long.
WEP Key Size	Choose from the following options: • None • 40 bits • 104 bits

Step 5 Click **Save & Apply to Device**.

Configuring Multi-Preshared Key (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-name wlan-id ssid* **Example:** `Device(config)# wlan mywlan 1 SSID_name`	Configures WLAN and SSID.
Step 3	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 4	**security wpa akm psk** **Example:** `Device(config-wlan)# security wpa akm psk`	Configures PSK.
Step 5	**security wpa wpa2 mpsk** **Example:** `Device(config-wlan)# security wpa wpa2 mpsk`	Configures multi-PSK.
Step 6	**priority** *priority_value* **set-key** {**ascii** [**0** | **8**] *pre-shared-key* | **hex** [**0** | **8**] *pre-shared-key*} **Example:** `Device(config-mpsk)# priority 0 set-key ascii 0 deadbeef`	Configures PSK priority and all its related passwords. The *priority_value* ranges from 0 to 4. **Note** You need to configure **priority 0** key for multi-PSK.
Step 7	**no shutdown** **Example:** `Device(config-mpsk)# no shutdown`	Enables WLAN.
Step 8	**exit** **Example:** `Device(config-wlan)# exit`	Exits WLAN configuration mode and returns to configuration mode.

	Command or Action	Purpose
Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Multi-PSK Configurations

To verify the configuration of a WLAN and a client, use the following command:

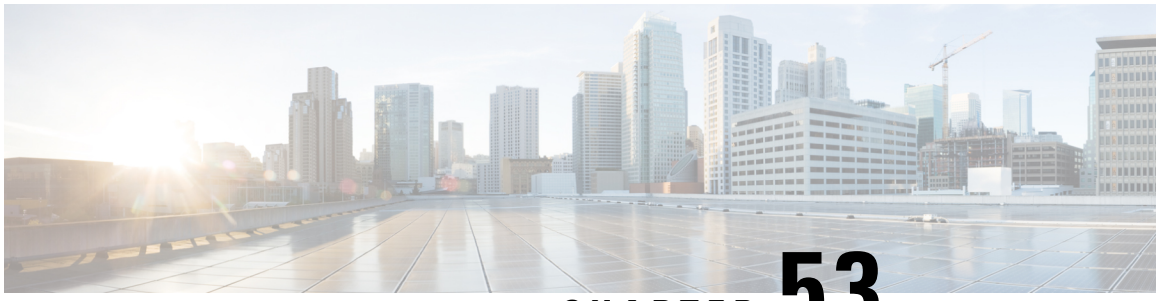
```
Device# show wlan id 8
WLAN Profile Name     : wlan_8
================================================
Identifier                                    : 8
Network Name (SSID)                           : ssid_8
Status                                        : Enabled
Broadcast SSID                                : Enabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
Number of Active Clients                      : 0
CHD per WLAN                                  : Enabled
Multicast Interface                           : Unconfigured
WMM                                           : Allowed
WifiDirect                                    : Invalid
Channel Scan Defer Priority:
  Priority (default)                          : 5
  Priority (default)                          : 6
Scan Defer Time (msecs)                       : 100
Media Stream Multicast-direct                 : Disabled
CCX - AironetIe Support                       : Enabled
CCX - Diagnostics Channel Capability          : Disabled
Peer-to-Peer Blocking Action                  : Disabled
Radio Policy                                  : All
DTIM period for 802.11a radio                 : 1
DTIM period for 802.11b radio                 : 1
Local EAP Authentication                      : Disabled
Mac Filter Authorization list name            : Disabled
Mac Filter Override Authorization list name   : Disabled
Accounting list name                          :
802.1x authentication list name               : Disabled
802.1x authorization list name                : Disabled
Security
    802.11 Authentication                     : Open System
    Static WEP Keys                           : Disabled
    802.1X                                    : Disabled
    Wi-Fi Protected Access (WPA/WPA2/WPA3)    : Enabled
        WPA (SSN IE)                          : Disabled
        WPA2 (RSN IE)                         : Enabled
            MPSK                              : Enabled
            AES Cipher                        : Enabled
            CCMP256 Cipher                    : Disabled
            GCMP128 Cipher                    : Disabled
            GCMP256 Cipher                    : Disabled
        WPA3 (WPA3 IE)                        : Disabled
        Auth Key Management
            802.1x                            : Disabled
            PSK                               : Enabled
```

```
            CCKM                                    : Disabled
            FT dot1x                                : Disabled
            FT PSK                                  : Disabled
            FT SAE                                  : Disabled
            PMF dot1x                               : Disabled
            PMF PSK                                 : Disabled
            SAE                                     : Disabled
            OWE                                     : Disabled
            SUITEB-1X                               : Disabled
            SUITEB192-1X                            : Disabled
    CCKM TSF Tolerance                              : 1000
    FT Support                                      : Adaptive
        FT Reassociation Timeout                    : 20
        FT Over-The-DS mode                         : Enabled
    PMF Support                                     : Disabled
        PMF Association Comeback Timeout            : 1
        PMF SA Query Time                           : 200
    Web Based Authentication                        : Disabled
    Conditional Web Redirect                        : Disabled
    Splash-Page Web Redirect                        : Disabled
    Webauth On-mac-filter Failure                   : Disabled
    Webauth Authentication List Name                : Disabled
    Webauth Authorization List Name                 : Disabled
    Webauth Parameter Map                           : Disabled
    Tkip MIC Countermeasure Hold-down Timer         : 60
Non Cisco WGB                                       : Disabled
Band Select                                         : Enabled
Load Balancing                                      : Disabled
Multicast Buffer                                    : Disabled
Multicast Buffer Size                               : 0
IP Source Guard                                     : Disabled
Assisted-Roaming
    Neighbor List                                   : Disabled
    Prediction List                                 : Disabled
    Dual Band Support                               : Disabled
IEEE 802.11v parameters
    Directed Multicast Service                      : Disabled
    BSS Max Idle                                    : Disabled
        Protected Mode                              : Disabled
    Traffic Filtering Service                       : Disabled
    BSS Transition                                  : Enabled
        Disassociation Imminent                     : Disabled
            Optimised Roaming Timer                 : 40
            Timer                                   : 200
    WNM Sleep Mode                                  : Disabled
802.11ac MU-MIMO                                    : Disabled
802.11ax paramters
    OFDMA Downlink                                  : unknown
    OFDMA Uplink                                    : unknown
    MU-MIMO Downlink                                : unknown
    MU-MIMO Uplink                                  : unknown
    BSS Color                                       : unknown
    Partial BSS Color                               : unknown
    BSS Color Code                                  :
```

To view the WLAN details, use the following command:

```
Device# show run wlan
wlan wlan_8 8 ssid_8
 security wpa psk set-key ascii 0 deadbeef
 no security wpa akm dot1x
 security wpa akm psk
 security wpa wpa2 mpsk
  priority 0 set-key ascii 0 deadbeef
  priority 1 set-key ascii 0 deaddead
```

```
 priority 2 set-key ascii 0 d123d123
 priority 3 set-key hex 0 023456789012345678901234567890123456789012345678901234567890123456789012345678901234

 priority 4 set-key hex 0 123456789012345678901234567890123456789012345678901234567890123456789012345678901234

no shutdown
```



CHAPTER 53

Multiple Authentications for a Client

Information About Multiple Authentications for a Client

Multiple Authentication feature is an extension of Layer 2 and Layer 3 security types supported for client join.

Note You can enable both L2 and L3 authentication for a given SSID.

Note The Multiple Authentication feature is applicable for regular clients only.

Information About Supported Combination of Authentications for a Client

The Multiple Authentications for a Client feature supports multiple combination of authentications for a given client configured in the WLAN profile.

The following table outlines the supported combination of authentications:

Layer 2	Layer 3	Supported
MAB	CWA	Yes
MAB Failure	LWA	Yes
802.1X	CWA	Yes
PSK	CWA	Yes
iPSK + MAB	CWA	Yes
iPSK	LWA	No

MAB Failure + PSK	LWA	No
MAB Failure + PSK	CWA	No

From 16.10.1 onwards, 802.1X configurations on WLAN support web authentication configurations with WPA or WPA2 configuration.

The feature also supports the following AP modes:

- Local
- FlexConnect
- Fabric

Configuring Multiple Authentications for a Client

Configuring WLAN for 802.1X and Local Web Authentication (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.
Step 2 Select the required WLAN from the list of WLANs displayed.
Step 3 Choose **Security** > **Layer2** tab.
Step 4 Select the security method from the **Layer 2 Security Mode** drop-down list.
Step 5 In the **Auth Key Mgmt**, check the **802.1x** check box.
Step 6 Check the **MAC Filtering** check box to enable the feature.
Step 7 After MAC Filtering is enabled, from the **Authorization List** drop-down list, choose an option.
Step 8 Choose **Security** > **Layer3** tab.
Step 9 Check the **Web Policy** check box to enable web authentication policy.
Step 10 From the **Web Auth Parameter Map** and the **Authentication List** drop-down lists, choose an option.
Step 11 Click **Update & Apply to Device**.

Configuring WLAN for 802.1X and Local Web Authentication (CLI)

Procedure

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

	Command or Action	Purpose
Step 2	**wlan** *profile-name wlan-id SSID_Name* **Example:** `Device(config)# wlan wlan-test 3 ssid-test`	Enters WLAN configuration sub-mode. • *profile-name*: Profile name of the configured WLAN. • *wlan-id*: Wireless LAN identifier. Range is from 1 to 512. • *SSID_Name*: SSID that can contain 32 alphanumeric characters. **Note** If you have already configured this command, enter the **wlan** *profile-name* command.
Step 3	**security dot1x authentication-list** *auth-list-name* **Example:** `Device(config-wlan)# security dot1x authentication-list default`	Enables security authentication list for dot1x security. The configuration is similar for all dot1x security WLANs.
Step 4	**security web-auth** **Example:** `Device(config-wlan)# security web-auth`	Enables web authentication.
Step 5	**security web-auth authentication-list** *authenticate-list-name* **Example:** `Device(config-wlan)# security web-auth authentication-list default`	Enables authentication list for dot1x security.
Step 6	**security web-auth parameter-map** *parameter-map-name* **Example:** `Device(config-wlan)# security web-auth parameter-map WLAN1_MAP`	Maps the parameter map. **Note** If a parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.
Step 7	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.

Example

```
wlan wlan-test 3 ssid-test
 security dot1x authentication-list default
 security web-auth
 security web-auth authentication-list default
```

```
security web-auth parameter-map WLAN1_MAP
no shutdown
```

Configuring WLAN for Preshared Key (PSK) and Local Web Authentication (GUI)

Procedure

Step 1	Choose **Configuration** > **Tags & Profiles** > **WLANs**.
Step 2	Select the required WLAN.
Step 3	Choose **Security** > **Layer2** tab.
Step 4	Select the security method from the **Layer 2 Security Mode** drop-down list.
Step 5	In the Auth Key Mgmt, uncheck the **802.1x** check box.
Step 6	Check the **PSK** check box.
Step 7	Enter the **Pre-Shared Key** and choose the PSK Format from the **PSK Format** drop-down list and the PSK Type from the **PSK Type** drop-down list.
Step 8	Choose **Security** > **Layer3** tab.
Step 9	Check the **Web Policy** checkbox to enable web authentication policy.
Step 10	Choose the Web Auth Parameter Map from the **Web Auth Parameter Map** drop-down list and the authentication list from the **Authentication List** drop-down list.
Step 11	Click **Update & Apply to Device**.

Configuring WLAN for Preshared Key (PSK) and Local Web Authentication

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name wlan-id SSID_Name* **Example:** `Device(config)# wlan wlan-test 3 ssid-test`	Enters WLAN configuration sub-mode. • *profile-name*- Is the profile name of the configured WLAN. • *wlan-id* - Is the wireless LAN identifier. Range is from 1 to 512. • *SSID_Name* - Is the SSID which can contain 32 alphanumeric characters.

	Command or Action	Purpose
		Note If you have already configured this command, enter **wlan** *profile-name* command.
Step 3	**security wpa psk set-key** *ascii/hex key password* **Example:** `Device(config-wlan)# security wpa psk set-key ascii 0 PASSWORD`	Configures the PSK shared key.
Step 4	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 5	**security wpa akm psk** **Example:** `Device(config-wlan)# security wpa akm psk`	Configures the PSK support.
Step 6	**security web-auth** **Example:** `Device(config-wlan)# security web-auth`	Enables web authentication for WLAN.
Step 7	**security web-auth authentication-list** *authenticate-list-name* **Example:** `Device(config-wlan)# security web-auth authentication-list webauth`	Enables authentication list for dot1x security.
Step 8	**security web-auth parameter-map** *parameter-map-name* **Example:** `(config-wlan)# security web-auth parameter-map WLAN1_MAP`	Configures the parameter map. **Note** If parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

Example

```
wlan wlan-test 3 ssid-test
 security wpa psk set-key ascii 0 PASSWORD
 no security wpa akm dot1x
 security wpa akm psk
 security web-auth
 security web-auth authentication-list webauth
 security web-auth parameter-map WLAN1_MAP
```

Configuring WLAN for PSK or Identity Preshared Key (iPSK) and Central Web Authentication (GUI)

Procedure

Step 1	Choose **Configuration** > **Tags & Profiles** > **WLANs**.
Step 2	Select the required WLAN.
Step 3	Choose **Security** > **Layer2** tab.
Step 4	Select the security method from the **Layer 2 Security Mode** drop-down list.
Step 5	In the **Auth Key Mgmt**, uncheck the **802.1x** check box.
Step 6	Check the **PSK** check box.
Step 7	Enter the **Pre-Shared Key** and choose the PSK Format from the **PSK Format** drop-down list and the PSK Type from the **PSK Type** drop-down list.
Step 8	Check the **MAC Filtering** check box to enable the feature.
Step 9	With MAC Filtering enabled, choose the Authorization List from the **Authorization List** drop-down list.
Step 10	Choose **Security** > **Layer3** tab.
Step 11	Check the **Web Policy** checkbox to enable web authentication policy.
Step 12	Choose the Web Auth Parameter Map from the **Web Auth Parameter Map** drop-down list and the authentication list from the **Authentication List** drop-down list.
Step 13	Click **Update &Apply to Device**.

Configuring WLAN for PSK or Identity Preshared Key (iPSK) and Central Web Authentication

Configuring WLAN

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name wlan-id SSID_Name* **Example:** `Device(config)# wlan wlan-test 3 ssid-test`	Enters WLAN configuration sub-mode. • *profile-name* - Is the profile name of the configured WLAN. • *wlan-id* - Is the wireless LAN identifier. Range is from 1 to 512.

	Command or Action	Purpose
		• *SSID_Name* - Is the SSID which can contain 32 alphanumeric characters. **Note** If you have already configured this command, enter **wlan** *profile-name* command.
Step 3	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 4	**security wpa psk set-key** *ascii/hex key password* **Example:** `Device(config-wlan)# security wpa psk set-key ascii 0 PASSWORD`	Configures the PSK AKM shared key.
Step 5	**mac-filtering** *auth-list-name* **Example:** `Device(config-wlan)# mac-filtering test-auth-list`	Sets the MAC filtering parameters.

Example

```
wlan wlan-test 3 ssid-test
 no security wpa akm dot1x
 security wpa psk set-key ascii 0 PASSWORD
 mac-filtering test-auth-list
```

Applying Policy Profile to a WLAN

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-profile-name* **Example:** `Device(config)# wireless profile policy policy-iot`	Configures the default policy profile.

	Command or Action	Purpose
Step 3	**aaa-override** **Example:** `Device(config-wireless-policy)# aaa-override`	Configures AAA override to apply policies coming from the AAA or ISE servers.
Step 4	**nac** **Example:** `Device(config-wireless-policy)# nac`	Configures NAC in the policy profile.
Step 5	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Shutdown the WLAN.
Step 6	**end** **Example:** `Device(config-wireless-policy)# end`	Returns to privileged EXEC mode.

Example

```
wireless profile policy policy-iot
 aaa-override
 nac
 no shutdown
```

Verifying Multiple Authentication Configurations

Layer 2 Authentication

After L2 authentication (Dot1x) is complete, the client is moved to *Webauth Pending* state.

To verify the client state after L2 authentication, use the following commands:

```
Device# show wireless client summary
Number of Local Clients: 1
MAC Address  AP Name  WLAN  State  Protocol  Method  Role
-------------------------------------------------------------------------------------------------------------------

58ef.68b6.aa60 ewlc1_ap_1   3  Webauth Pending    11n(5)        Dot1x  Local
Number of Excluded Clients: 0

Device# show wireless client mac-address <mac_address> detail

Auth Method Status List

Method: Dot1x
Webauth State: Init
Webauth Method: Webauth
Local Policies:
Service Template: IP-Adm-V6-Int-ACL-global (priority 100)
```

```
URL Redirect ACL: IP-Adm-V6-Int-ACL-global
Service Template: IP-Adm-V4-Int-ACL-global (priority 100)
URL Redirect ACL: IP-Adm-V4-Int-ACL-global
Service Template: wlan_svc_default-policy-profile_local (priority 254)
Absolute-Timer: 1800
VLAN: 50
```

```
Device# show platform software wireless-client chassis active R0

        ID  MAC Address      WLAN  Client     State
------------------------------------------------------------------------------------------
  0xa0000003      58ef.68b6.aa60    3             L3          Authentication
```

```
Device# show platform software wireless-client chassis active F0

    ID     MAC Address   WLAN  Client     State  AOM ID    Status
-------------------------------------------------------------------------------------------------
0xa0000003    58ef.68b6.aa60    3           L3          Authentication.         730.
Done
```

```
Device# show platform hardware chassis active qfp feature wireless wlclient cpp-client
summary

Client Type Abbreviations:
RG - REGULAR   BLE - BLE
HL - HALO    LI - LWFL INT

Auth State Abbrevations:
UK - UNKNOWN    IP - LEARN    IP IV - INVALID
L3 - L3 AUTH RN - RUN

Mobility State Abbreviations:
UK - UNKNOWN       IN - INIT
LC - LOCAL               AN - ANCHOR
FR - FOREIGN          MT - MTE
IV - INVALID


EoGRE Abbreviations:
N - NON EOGRE Y - EOGRE

CPP IF_H   DP IDX       MAC Address      VLAN   CT   MCVL AS MS E   WLAN      POA
-------------------------------------------------------------------------------------
0X49     0XA0000003    58ef.68b6.aa60     50    RG     0  L3 LC N wlan-test 0x90000003
```

```
Device# show platform hardware chassis active qfp feature wireless wlclient datapath summary
Vlan   DP IDX         MAC Address      VLAN   CT   MCVL AS MS E   WLAN      POA
-------------------------------------------------------------------------------------
0X49   0xa0000003    58ef.68b6.aa60     50    RG     0  L3 LC N wlan-test 0x90000003
```

Layer 3 Authentication

Once L3 authentication is successful, the client is moved to *Run* state.

To verify the client state after L3 authentication, use the following commands:

```
Device# show wireless client summary

Number of Local Clients: 1
MAC Address  AP Name  WLAN  State  Protocol  Method  Role
-----------------------------------------------------------------------------------------------------

58ef.68b6.aa60   ewlc1_ap_1  3     Run    11n(5)   Web Auth      Local
Number of Excluded Clients: 0
```

```
Device# show wireless client mac-address 58ef.68b6.aa60 detail

Auth Method Status List

Method: Web Auth
Webauth State: Authz
Webauth Method: Webauth
Local Policies:
Service Template: wlan_svc_default-policy-profile_local (priority 254)
Absolute-Timer: 1800
VLAN: 50

Server Policies:

Resultant Policies:
VLAN: 50
Absolute-Timer: 1800
```

```
Device# show platform software wireless-client chassis active R0

ID          MAC Address     WLAN    Client State
-------------------------------------------------
0xa0000001 58ef.68b6.aa60    3          Run
```

```
Device# show platform software wireless-client chassis active f0

ID         MAC Address       WLAN    Client State   AOM ID.  Status
-----------------------------------------------------------------------
0xa0000001 58ef.68b6.aa60.   3          Run         11633    Done
```

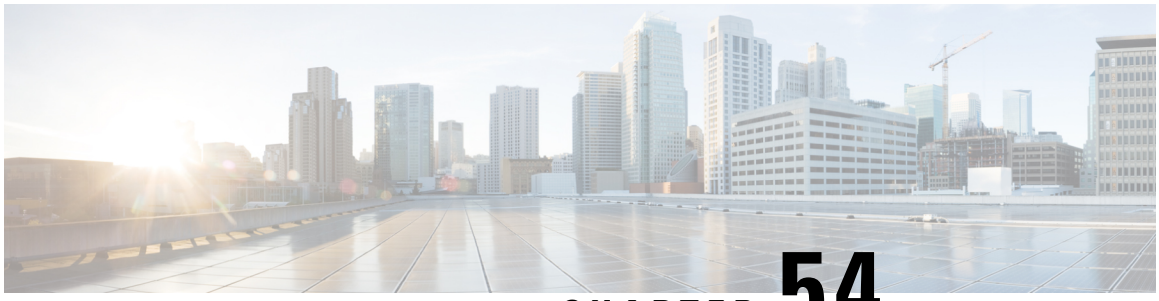
```
Device# show platform hardware chassis active qfp feature wireless wlclient cpp-client
summary

Client Type Abbreviations:
RG - REGULAR   BLE - BLE
HL - HALO      LI - LWFL INT

Auth State Abbrevations:
UK - UNKNOWN    IP - LEARN    IP IV - INVALID
L3 - L3 AUTH RN - RUN
Mobility State Abbreviations:
UK - UNKNOWN       IN - INIT
LC - LOCAL         AN - ANCHOR
FR - FOREIGN       MT - MTE
IV - INVALID
EoGRE Abbreviations:
N - NON EOGRE Y - EOGRE

CPP IF_H   DP IDX       MAC Address   VLAN  CT  MCVL AS MS E   WLAN     POA
-------------------------------------------------------------------------------
0X49     0XA0000003    58ef.68b6.aa60  50   RG   0   RN LC N wlan-test 0x90000003
```

```
Device# show platform hardware chassis active qfp feature wireless wlclient datapath summary

Vlan   pal_if_hd1        mac           Input Uidb     Output Uidb
-----------------------------------------------------------------
50     0xa0000003    58ef.68b6.aa60     95929            95927
```

Verifying PSK+Webauth Configuration

```
Device# show wlan summary

Load for five secs: 0%/0%; one minute: 0%; five minutes: 0%
Time source is NTP, 12:08:32.941 CEST Tue Oct 6 2020
```

```
Number of WLANs: 1

ID Profile Name SSID Status Security
--------------------------------------------------------------------------------------------------------------------------------

23 Gladius1-PSKWEBAUTH Gladius1-PSKWEBAUTH UP [WPA2][PSK][AES],[Web Auth]
```



CHAPTER 54

Cisco Umbrella WLAN

Information About Cisco Umbrella WLAN

The Cisco Umbrella WLAN provides a cloud-delivered network security service at the Domain Name System (DNS) level, with automatic detection of both known and emergent threats.

This feature allows you to block sites that host malware, bot networks, and phishing before they actually become malicious.

Cisco Umbrella WLAN provides the following:

- Policy configuration per user group at a single point.
- Policy configuration per network, group, user, device, or IP address.

  The following is the policy priority order:

  1. Local policy
  2. AP group
  3. WLAN

- Visual security activity dashboard in real time with aggregated reports.
- Schedule and send reports through email.
- Support up to 60 content categories, with a provision to add custom allowed list and blocked list entries.

This feature does not work in the following scenarios:

- If an application or host use an IP address directly, instead of using DNS to query domain names.
- If a client is connected to a web proxy and does not send a DNS query to resolve the server address.

Registering Embedded Wireless Controller to Cisco Umbrella Account

Before you Begin

- You should have an account with Cisco Umbrella.
- You should have an API token from Cisco Umbrella.

The embedded wireless controller is registered to Cisco Umbrella server using the Umbrella parameter map. Each of the Umbrella parameter map must have an API token. The Cisco Umbrella responds with the device ID for the embedded wireless controller. The device ID has a 1:1 mapping with the Umbrella parameter map name.

Fetching API token for Embedded Wireless Controller from Cisco Umbrella Dashboard

From Cisco Umbrella dashboard, verify that your embedded wireless controller shows up under Device Name, along with their identities.

Applying the API Token on Embedded Wireless Controller

Registers the Cisco Umbrella API token on the network.

DNS Query and Response

Once the device is registered and Umbrella parameter map is configured on WLAN, the DNS queries from clients joining the WLAN are redirected to the Umbrella DNS resolver.

Note This is applicable for all domains not configured in the local domain RegEx parameter map.

The queries and responses are encrypted based on the DNScrypt option in the Umbrella parameter map.

For more information on the Cisco Umbrella configurations, see the Integration for ISR 4K and ISR 1100 – Security Configuration Guide.

Limitations and Considerations

The limitations and considerations for this feature are as follows:

- You will be able to apply the wireless Cisco Umbrella profiles to wireless entities, such as, WLAN or AP groups, if the device registration is successful.
- In case of L3 mobility, the Cisco Umbrella must be applied on the anchor embedded wireless controller always.
- When two DNS servers are configured under DHCP, two Cisco Umbrella server IPs are sent to the client from DHCP option 6. If only one DNS server is present under DHCP, only one Cisco Umbrella server IP is sent as part of DHCP option 6.

Configuring Cisco Umbrella WLAN

To configure Cisco Umbrella on the embedded wireless controller, perform the following:

- You must have the API token from the Cisco Umbrella dashboard.
- You must have the root certificate to establish HTTPS connection with the Cisco Umbrella registration server: api.opendns.com. You must import the root certificate from **digicert.com** to the embedded wireless controller using the **crypto pki trustpool import terminal** command.

Importing CA Certificate to the Trust Pool

Before you begin

The following section covers details about how to fetch the root certificate and establish HTTPS connection with the Cisco Umbrella registration server:

Procedure

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

Step 2	Perform either of the following tasks:
	• **crypto pki trustpool import url** *url*
	`Device(config)# crypto pki trustpool import url http://www.cisco.com/security/pki/trs/ios.p7b`
	Imports the root certificate directly from the Cisco website.
	Note The Trustpool bundle contains the root certificate of *digicert.com* together with other CA certificates.
	• **crypto pki trustpool import terminal**
	`Device(config)# crypto pki trustpool import terminal`
	Imports the root certificate by executing the import terminal command.
	• Enter PEM-formatted CA certificate from the following location: See the Related Information section to download the CA certificate.	

	Command or Action	Purpose

	`-----BEGIN CERTIFICATE-----`
	`MIIE6jCCA9KgAwIBAgIQCjUI1VwpKwF9+K1lwA/35DANBgkqhkiG9w0BAQsFADBhMQswCQYDVQQG`
	[illegible]
	`SaZMkE4f97Q=`
	`-----END CERTIFICATE-----`
	
	Imports the root certificate by pasting the CA certificate from the **digicert.com**.	

Step 3	**quit** **Example:** `Device(config)# quit`	Imports the root certificate by entering the **quit** command. **Note** You will receive a message after the certificate has been imported.

Creating a Local Domain RegEx Parameter Map

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type regex** *parameter-map-name* **Example:** `Device(config)# parameter-map type regex dns_wl`	Creates a regex parameter map.
Step 3	**pattern** *regex-pattern*	Configures the regex pattern to match.

	Command or Action	Purpose
	Example: `Device(config-profile)# pattern www.google.com`	**Note** The following patterns are supported: • Begins with `.*`. For example: **`.*facebook.com`** • Begins with `.*` and ends with `*` . For example: **`.*google*`** • Ends with `*`. For example: **`www.facebook*`** • No special character. For example: **`www.facebook.com`**
Step 4	**end** **Example:** `Device(config-profile)# end`	Returns to privileged EXEC mode.

Configuring Parameter Map Name in WLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.
Step 2 Click on the Policy Profile Name. The **Edit Policy Profile** window is displayed.
Step 3 Choose the **Advanced** tab.
Step 4 In the **Umbrella** settings, from the **Umbrella Parameter Map** drop-down list, choose the parameter map.
Step 5 Enable or disable **Flex DHCP Option for DNS** and **DNS Traffic Redirect** toggle buttons.
Step 6 Click **Update & Apply to Device**.

Configuring the Umbrella Parameter Map

Procedure

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

	Command or Action	Purpose
Step 2	**parameter-map type umbrella global** **Example:** `Device(config)# parameter-map type umbrella global`	Creates an umbrella global parameter map.
Step 3	**token** *token-value* **Example:** `Device(config-profile)# token 5XXXXXXXXCXXXXXXXAXXXXXXXFXXXXCXXXXXXXX`	Configures an umbrella token.
Step 4	**local-domain** *regex-parameter-map-name* **Example:** `Device(config-profile)# local-domain dns_wl`	Configures local domain RegEx parameter map.
Step 5	**resolver** {**IPv4** *X.X.X.X* | **IPv6** *X:X:X:X::X*} **Example:** `Device(config-profile)# resolver IPv6 10:1:1:1::10`	Configures the Anycast address. The default address is applied when there is no specific address configured.
Step 6	**end** **Example:** `Device(config-profile)# end`	Returns to privileged EXEC mode.

Enabling or Disabling DNScrypt (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Threat Defence** > **Umbrella**.

Step 2 Enter the **Registration Token** received from Umbrella. Alternatively, you can click on **Click here to get your Token**to get the token from Umbrella.

Step 3 Enter the **Whitelist Domains** that you want to exclude from filtering.

Step 4 Check or uncheck the **Enable DNS Packets Encryption** check box to encrypt or decrypt the DNS packets.

Step 5 Click **Apply**.

Enabling or Disabling DNScrypt

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**parameter-map type umbrella global** **Example:** `Device(config)# parameter-map type umbrella global`	Creates an umbrella global parameter map.
Step 3	[**no**] **dnscrypt** **Example:** `Device(config-profile)# no dnscrypt`	Enables or disables DNScrypt. By default, the DNScrypt option is enabled. **Note** Cisco Umbrella DNScrypt is not supported when DNS-encrypted responses are sent in the data-DTLS encrypted tunnel (either mobility tunnel or AP CAPWAP tunnel).
Step 4	**end** **Example:** `Device(config-profile)# end`	Returns to privileged EXEC mode.

Configuring Timeout for UDP Sessions

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type umbrella global** **Example:** `Device(config)# parameter-map type umbrella global`	Creates an umbrella global parameter map.
Step 3	**udp-timeout** *timeout_value* **Example:** `Device(config-profile)# udp-timeout 2`	Configures timeout value for UDP sessions. The *timeout_value* ranges from 1 to 30 seconds. **Note** The **public-key** and **resolver** parameter-map options are automatically populated with the default values. So, you need not change them.
Step 4	**end** **Example:**	Returns to privileged EXEC mode.

	Command or Action	Purpose
	`Device(config-profile)# end`	

Configuring Parameter Map Name in WLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click on the Policy Profile Name. The **Edit Policy Profile** window is displayed.

Step 3 Choose the **Advanced** tab.

Step 4 In the **Umbrella** settings, from the **Umbrella Parameter Map** drop-down list, choose the parameter map.

Step 5 Enable or disable **Flex DHCP Option for DNS** and **DNS Traffic Redirect** toggle buttons.

Step 6 Click **Update & Apply to Device**.

Configuring Parameter Map Name in WLAN

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-name* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Creates policy profile for the WLAN. The *profile-name* is the profile name of the policy profile.
Step 3	**umbrella-param-map** *umbrella-name* **Example:** `Device(config-wireless-policy)# umbrella-param-map global`	Configures the Umbrella OpenDNS feature for the WLAN.
Step 4	**end** **Example:** `Device(config-wireless-policy)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying the Cisco Umbrella Configuration

To view the Umbrella configuration details, use the following command:

```
Device# show umbrella config
Umbrella Configuration
=======================
Token: 5XXXXXXABXXXXXFXXXXXXXXXXDXXXXXXXXXXXABXX
API-KEY: NONE
OrganizationID: xxxxxxx
Local Domain Regex parameter-map name: dns_bypass
DNSCrypt: Not enabled
Public-key: NONE
UDP Timeout: 5 seconds
Resolver address:
1. 10.1.1.1
2. 5.5.5.5
3. XXXX:120:50::50
4. XXXX:120:30::30
```

To view the Umbrella DNSCrypt details, use the following command:

```
Device# show umbrella dnscrypt
DNSCrypt: Enabled
   Public-key: B111:XXXX:XXXX:XXXX:3E2B:XXXX:XXXX:XXXE:XXX3:3XXX:DXXX:XXXX:BXXX:XXXB:XXXX:FXXX

   Certificate Update Status: In Progress
```

To view the Umbrella global parameter map details, use the following command:

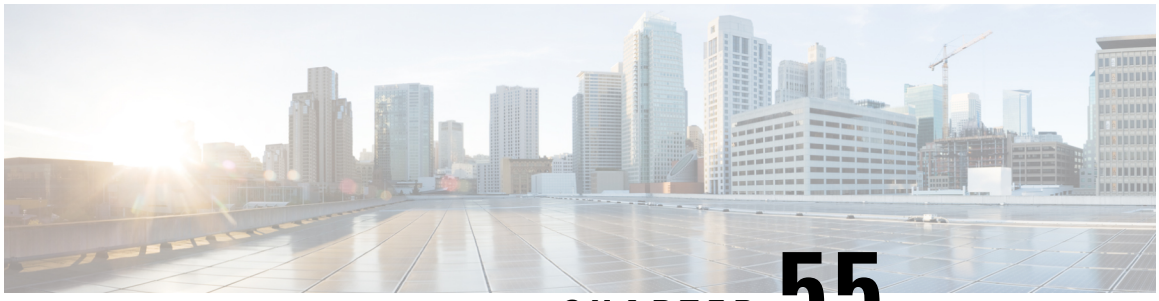
```
Device# show parameter-map type umbrella global
```

To view the regex parameter map details, use the following command:

```
Device# show parameter-map type regex <parameter-map-name>
```

To view the Umbrella details on the AP, use the following command:

```
AP#show client opendns summary
Server-IP role
208.67.220.220 Primary
208.67.222.222 Secondary

Server-IP role
2620:119:53::53 Primary
2620:119:35::35 Secondary

Wlan Id DHCP OpenDNS Override Force Mode
0 true false
1 false false
...

15 false false
Profile-name Profile-id
vj-1 010a29b176b34108
global 010a57bf502c85d4
vj-2 010ae385ce6c1256
AP0010.10A7.1000#

Client to profile command

AP#show client opendns address 50:3e:aa:ce:50:17
Client-mac Profile-name
50:3E:AA:CE:50:17 vj-1
AP0010.10A7.1000#
```



CHAPTER 55

Locally Significant Certificates

Information About Locally Significant Certificates

This module explains how to configure the Cisco Embedded Wireless Controller on Catalyst Access Points and Lightweight Access Points (LAPs) to use the Locally Significant Certificate (LSC). If you choose the Public Key Infrastructure (PKI) with LSC, you can generate the LSC on the APs and embedded wireless controllers. You can then use the certificates to mutually authenticate the embedded wireless controller and the APs.

In Cisco embedded wireless controllers, you can configure the embedded wireless controller to use an LSC. Use an LSC if you want your own PKI to provide better security, have control of your Certificate Authority (CA), and define policies, restrictions, and usages on the generated certificates.

You need to provision the new LSC certificate on the embedded wireless controller and then the Lightweight Access Point (LAP) from the CA Server.

The LAP communicates with the embedded wireless controller using the CAPWAP protocol. Any request to sign the certificate and issue the CA certificates for LAP and embedded wireless controller itself must be initiated from the embedded wireless controller. The LAP does not communicate directly with the CA server. The CA server details must be configured on the embedded wireless controller and must be accessible.

The embedded wireless controller makes use of the Simple Certificate Enrollment Protocol (SCEP) to forward certReqs generated on the devices to the CA and makes use of SCEP again to get the signed certificates from the CA.

The SCEP is a certificate management protocol that the PKI clients and CA servers use to support certificate enrollment and revocation. It is widely used in Cisco and supported by many CA servers. In SCEP, HTTP is used as the transport protocol for the PKI messages. The primary goal of SCEP is the secure issuance of certificates to network devices. SCEP is capable of many operations, but for our release, SCEP is utilized for the following operations:

- CA and Router Advertisement (RA) Public Key Distribution
- Certificate Enrollment

Certificate Provisioning in Controllers

The new LSC certificates, both CA and device certificates, must be installed on the controller.

With the help of SCEP, CA certificates are received from the CA server. During this point, there are no certificates in the controller. After the **get** operation of obtaining the CA certificates, are installed on the controller. The same CA certificates are also pushed to the APs when the APs are provisioned with LSCs.

Note We recommend that you use a new RSA keypair name for the newly configured PKI certificate. If you want to reuse an existing RSA keypair name (that is associated with an old certificate) for a new PKI certificate, do either of the following:

- Do not regenerate a new RSA keypair with an existing RSA keypair name, reuse the existing RSA keypair name. Regenerating a new RSA keypair with an existing RSA keypair name will make all the certificates associated with the existing RSA keypair invalid.
- Manually remove the old PKI certificate configurations first, before reusing the existing RSA keypair name for the new PKI certificate.

Device Certificate Enrollment Operation

For both the LAP and the controller that request a CA-signed certificate, the certRequest is sent as a PKCS#10 message. The certRequest contains the Subject Name, Public Key, and other attributes to be included in the X.509 certificate, and must be digitally signed by the Private Key of the requester. These are then sent to the CA, which transforms the certRequest into an X.509 certificate.

The CA that receives a PKCS#10 certRequest requires additional information to authenticate the requester's identity and verify if the request is unaltered. (Sometimes, PKCS#10 is combined with other approaches, such as PKCS#7 to send and receive the certificate request or response.)

The PKCS#10 is wrapped in a PKCS#7 Signed Data message type. This is supported as part of the SCEP client functionality, while the PKCSReq message is sent to the controller. Upon successful enrollment operation, both the CA and device certificates are available on the controller.

Certificate Provisioning on Lightweight Access Point

In order to provision a new certificate on LAP, while in CAPWAP mode, the LAP must be able to get the new signed X.509 certificate. In order to do this, it sends a certRequest to the controller, which acts as a CA proxy and helps obtain the certRequest signed by the CA for the LAP.

The certReq and the certResponses are sent to the LAP with the LWAPP payloads.

Both the LSC CA and the LAP device certificates are installed in the LAP, and the system reboots automatically. The next time when the system comes up, because it is configured to use LSCs, the AP sends the LSC device certificate to the controller as part of the JOIN Request. As part of the JOIN Response, the controller sends the new device certificate and also validates the inbound LAP certificate with the new CA root certificate.

What to Do Next

To configure, authorize, and manage certificate enrollment with the existing PKI infrastructure for controller and AP, you need to use the LSC provisioning functionality.

Restrictions for Locally Significant Certificates

- LSC workflow is different in FIPS+WLANCC mode. CA server must support Enrollment over Secure Transport (EST) protocol and should be capable of issuing EC certificates in FIPS+WLANCC mode.
- Elliptic Curve Digital Signature Algorithm (ECDSA) cipher works only if both AP and controller are having EC certificates, provisioned with LSC.
- EC certificates (LSC-EC) can be provisioned only if CA server supports EST (and not SCEP).
- FIPS + CC security modes is required to be configured in order to provision EC certificate.

Provisioning Locally Significant Certificates

Configuring RSA Key for PKI Trustpoint

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**crypto key generate rsa** [**exportable**] **general-keys modulus** *key_size* **label** *RSA_key* **Example:** `Device(config)# crypto key generate rsa exportable general-keys modulus 2048 label lsc-tp`	Configures RSA key for PKI trustpoint. **exportable** is an optional keyword. You may or may not want to configure an exportable-key. If selected, you can export the key out of the box, if required • *key_size*: Size of the key modulus. The valid range is from 2048 to 4096. • *RSA_key*: RSA key pair label.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring PKI Trustpoint Parameters

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**crypto pki trustpoint** *trustpoint_name* **Example:** `Device(config)# crypto pki trustpoint microsoft-ca`	Creates a new trustpoint for an external CA server. Here, *trustpoint_name* refers to the trustpoint name.
Step 3	**enrollment url** *HTTP_URL* **Example:** `Device(ca-trustpoint)# enrollment url http://CA_server/certsrv/mscep/mscep.dll`	Specifies the URL of the CA on which your router should send certificate requests. **url** *url*: URL of the file system where your router should send certificate requests. An IPv6 address can be added in the URL enclosed in brackets. For example: http://[2001:DB8:1:1::1]:80. For more enrollment method options, see the enrollment url (ca-trustpoint) command page.
Step 4	**subject-name** *subject_name* **Example:** `Device(ca-trustpoint)# subject-name C=IN, ST=KA, L=Bengaluru, O=Cisco, CN=eagle-eye/emailAddress=support@abc.com`	Creates subject name parameters for the trustpoint.
Step 5	**rsakeypair** *RSA_key key_size* **Example:** `Device(ca-trustpoint)# rsakeypair ewlc-tp1`	Maps RSA key with that of the trustpoint. • *RSA_key*: RSA key pair label. • *key_size*: Signature key length. Range is from 360 to 4096.
Step 6	**revocation** {**crl** | **none** | **ocsp**} **Example:** `Device(ca-trustpoint)# revocation none`	Checks revocation.
Step 7	**end** **Example:** `Device(ca-trustpoint)# end`	Returns to privileged EXEC mode.

Authenticating and Enrolling a PKI Trustpoint (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management**.

Step 2 In the **PKI Management** window, click the **Trustpoints** tab.

Step 3 In the **Add Trustpoint** dialog box, provide the following information:

a) In the **Label** field, enter the RSA key label.
b) In the **Enrollment URL** field, enter the enrollment URL.
c) Check the **Authenticate** check box to authenticate the Public Certificate from the enrollment URL.
d) In the **Subject Name** section, enter the **Country Code**, **State**, **Location**, **Organisation**, **Domain Name**, and **Email Address**.
e) Check the **Key Generated** check box to view the available RSA keypairs. Choose an option from the **Available RSA Keypairs** drop-down list.
f) Check the **Enroll Trustpoint** check box.
g) In the **Password** field, enter the password.
h) In the **Re-Enter Password** field, confirm the password.
i) Click **Apply to Device**.

The new trustpoint is added to the trustpoint name list.

Authenticating and Enrolling the PKI Trustpoint with CA Server (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**crypto pki authenticate** *trustpoint_name* **Example:** `Device(config)# crypto pki authenticate microsoft-ca`	Fetches the CA certificate.

Step 3	**yes**
	Example:
	`Device(config)# % Do you accept this certificate? [yes/no]:`
	`yes Trustpoint CA certificate accepted.`	

Step 4	**crypto pki enroll** *trustpoint_name* **Example:**	Enrolls the client certificate.

	Command or Action	Purpose

	`Device(config)# crypto pki enroll microsoft-ca`
	`%`
	`% Start certificate enrollment ..`
	`% Create a challenge password. You will need to verbally`
	`provide this password to the CA Administrator in order to`
	`revoke your certificate. For security reasons your password`
	`will not be saved in the configuration.`
	`Please make a note of it.`	

Step 5	**password** **Example:** `Device(config)# abcd123`	Enters a challenge password to the CA server.
Step 6	**password** **Example:** `Device(config)# abcd123`	Re-enters a challenge password to the CA server.

Step 7	**yes**
	Example:
	`Device(config)# % Include the router serial number`
	`in the subject name? [yes/no]: yes`	

Step 8	**no**
	Example:
	`Device(config)# % Include an IP address`
	`in the subject name? [no]: no`	

Step 9	**yes**
	Example:
	`Device(config)#`
	`Request certificate from CA? [yes/no]: yes`
	`% Certificate request sent to Certificate Authority`
	`% The 'show crypto pki certificate verbose`
	`client' command will show the fingerprint.`	

Step 10	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring AP Join Attempts with LSC Certificate (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **All Access Points** window, click the LSC Provision name.

Step 3 From the **Status** drop-down list, choose a status to enable LSC.

Step 4 From the **Trustpoint Name** drop-down list, choose the trustpoint.

Step 5 In the **Number of Join Attempts** field, enter the number of retry attempts that will be permitted.

Step 6 Click **Apply**.

Configuring AP Join Attempts with LSC Certificate (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap lsc-provision join-attempt** *number_of_attempts* **Example:** `Device(config)# ap lsc-provision join-attempt 10`	Specifies the maximum number of AP join failure attempts with the newly provisioned LSC certificate. When the number of AP joins exceed the specified limit, AP joins back with the Manufacturer Installed Certificate (MIC).
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Subject-Name Parameters in LSC Certificate

Procedure

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

	Command or Action	Purpose
Step 2	**ap lsc-provision subject-name-parameter country** *country-str* **state** *state-str* **city** *city-str* **domain** *domain-str* **org** *org-str* **email-address** *email-addr-str* **Example:** `Device(config)# ap lsc-provision subject-name-parameter country India state Karnataka city Bangalore domain domain1 org Right email-address adc@gfe.com`	Specifies the attributes to be included in the subject-name parameter of the certificate request generated by an AP.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring Key Size for LSC Certificate

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap lsc-provision key-size** { **2048** | **3072** | **4096}**} **Example:** `Device(config)# ap lsc-provision key-size 2048`	Specifies the size of keys to be generated for the LSC on AP.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Trustpoint for LSC Provisioning on an Access Point

Procedure

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

	Command or Action	Purpose
Step 2	**ap lsc-provision trustpoint** *tp-name* **Example:** `Device(config)# ap lsc-provision trustpoint microsoft-ca`	Specifies the trustpoint with which the LCS is provisioned to an AP. *tp-name*: The trustpoint name.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring an AP LSC Provision List (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **All Access Points** window, click the corresponding LSC Provision name.

Step 3 From the **Status** drop-down list, choose a status to enable LSC.

Step 4 From the **Trustpoint Name** drop-down list, choose a trustpoint.

Step 5 In the **Number of Join Attempts** field, enter the number of retry attempts that are allowed.

Step 6 From the **Key Size** drop-down list, choose a key.

Step 7 In the **Edit AP Join Profile** window, click the **CAPWAP** tab.

Step 8 In the **Add APs to LSC Provision List** section, click **Select File** to upload the CSV file that contains AP details.

Step 9 Click **Upload File**.

Step 10 In the **AP MAC Address** field, enter the AP MAC address. and add them. (The APs added to the provision list are displayed in the **APs in provision List** .)

Step 11 In the **Subject Name Parameters** section, enter the following details:

- **Country**
- **State**
- **City**
- **Organisation**
- **Department**
- **Email Address**

Step 12 Click **Apply**.

Configuring an AP LSC Provision List (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap lsc-provision mac-address** *mac-addr* **Example:** `Device(config)# ap lsc-provision mac-address 001b.3400.02f0`	Adds the AP to the LSC provision list. **Note** You can provision a list of APs using the **ap lsc-provision provision-list** command. (Or) You can provision all the APs using the **ap lsc-provision** command.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring LSC Provisioning for all the APs (GUI)

Procedure

Step 1 Choose **Configuration** > **Wireless** > **Access Points**.

Step 2 In the **Access Points** window, expand the **LSC Provision** section.

Step 3 Set **Status** to **Enabled** state.

Note If you set **Status** to **Provision List**, LSC provisioning will be configured only for APs that are a part of the provision list.

Step 4 From the **Trustpoint Name** drop-down list, choose the appropriate trustpoint for all APs.

Step 5 In the **Number of Join Attempts** field, enter the number of retry attempts that the APs can make to join the embedded wireless controller.

Step 6 From the **Key Size** drop-down list, choose the appropriate key size of the certificate:

- **2048**
- **3072**
- **4096**

Step 7 In the **Add APs to LSC Provision List** section, click **Select File** to upload the CSV file that contains the AP details.

Step 8 Click **Upload File**.

Step 9 In the **AP MAC Address** field, enter the AP MAC address. (The APs that are added to the provision list are displayed in the **APs in Provision List** section.)

Step 10 In the **Subject Name Parameters** section, enter the following details:

a. **Country**

b. **State**

c. **City**

d. **Organization**

e. **Department**

f. **Email Address**

Step 11 Click **Apply**.

Configuring LSC Provisioning for All APs (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap lsc-provision** **Example:** `Device(config)# ap lsc-provision`	Enables LSC provisioning for all APs. By default, LSC provisioning is disabled for all APs.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Configuring LSC Provisioning for the APs in the Provision List

Procedure

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

	Command or Action	Purpose
Step 2	**ap lsc-provision provision-list** **Example:** `Device(config)# ap lsc-provision provision-list`	Enables LSC provisioning for a set of APs configured in the provision list.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Unprovisioning Local Significant Certificates

To unprovision the Local Significant Certificates (LSC), complete the following steps:

1. Move the chassis to WLAN Common Criteria (WLANCC) mode.
2. Reload the APs by provisioning LSC and the wireless management trustpoint. For more information, refer to Configuring LSC Provisioning and Management Trustpoint, on page 602.
3. Remove Federal Information Processing Standard (FIPS) and WLANCC. For more information, refer to Removing FIPS and WLAN Common Criteria, on page 603.
4. Remove LSC provisioning. For more information, refer to Removal of LSC Provisioning, on page 604.

Configuring LSC Provisioning and Management Trustpoint

Before you begin

When EWC HA pair is used note the name of the Standby Access Point. Use the **show chassis** command.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap lsc-provision** **Example:** `Device(config)# ap lsc-provision`	Configures the AP LSC Provisioning parameters.
Step 3	**wireless management trustpoint** *trustpoint_name* **Example:** `Device(config)# wireless management trustpoint trustpoint-name`	Configures the management trustpoint to LSC.

	Command or Action	Purpose
Step 4	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves the configuration. Wait for the standby AP to join the controller. The HA pair will not be formed at this point.
Step 5	**wireless ewc-ap ap reload** **Example:** `Device# wireless ewc-ap ap reload`	Reloads the internal AP. This will also reload the controller on the AP. Standby AP starts the controller and becomes new Active for HA pair.

Removing FIPS and WLAN Common Criteria

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dtls-version dtls_1_2** **Example:** `Device(config)# ap dtls-version dtls_1_2`	Configures the AP DTLS version.
Step 3	**ap dtls-cipher** *ECDHE-ECDSA-AES256-GCM-SHA384* **Example:** `Device(config)# ap dtls-cipher ECDHE-ECDSA-AES256-GCM-SHA384`	Configures the AP DTLS ciphersuite.
Step 4	**no wireless wlancc** **Example:** `Device(config)# no wireless wlancc`	Disables WLAN CC on the controller.
Step 5	**no fips authorization-key** **Example:** `Device(config)# no fips authorization-key`	Disables the authorization key for FIPS.
Step 6	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 7	**write memory** **Example:**	Saves the configuration.

	Command or Action	Purpose
	`Device# write memory`	
Step 8	**reload** **Example:** `Device# reload`	Reloads the internal AP to move on to non-FIPS and non-CC mode.

Removal of LSC Provisioning

Before you begin

Wait for the standby AP to come up.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**no ap lsc-provision** **Example:** `Device(config)# no ap lsc-provision`	Disables AP LSC provisioning parameters.
Step 3	**no ap dtls-cipher ECDHE-ECDSA-AES256-GCM-SHA384** **Example:** `Device(config)# no ap dtls-cipher ECDHE-ECDSA-AES256-GCM-SHA384`	Disables AP DTLS cipher suite.
Step 4	**no ap dtls-version dtls_1_2** **Example:** `Device(config)# no ap dtls-version dtls_1_2`	Disables the DTLS version.
Step 5	**no wireless management trustpoint** **Example:** `Device(config)# no wireless management trustpoint`	Disables the wireless management trustpoint.
Step 6	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves the configuration changes.

	Command or Action	Purpose
Step 7	**wireless ewc-ap ap reload** **Example:** `Device# wireless ewc-ap ap reload`	Reloads the internal AP.

Importing a CA Certificate to the Trustpool (GUI)

PKI Trustpool Management is used to store a list of trusted certificates (either downloaded or built in) used by the different services on the controller. This is also used to authenticate a multilevel CA certificate. The built in CA certificate bundle in the PKI trustpool receives automatic updates from Cisco if they are not current, are corrupt, or if certain certificates need to be updated.

Perform this task to manually update the CA certificates in the PKI trustpool.

Note If your LSC has been issued by an intermediate CA, you must import the complete chain of CA certificates into the trustpool. Otherwise, you will not be able to provision the APs without the complete chain being present on the controller. The import step is not required if the certificate has been issued by a root CA.

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management**.

Step 2 In the **PKI Management** window, click the **Trustpool** tab.

Step 3 Click **Import**.

Step 4 In the **CA Certificate** field, copy and paste the CA certificate. Link together the multiple CA certificates in **.pem** format.

Step 5 Click **Apply to Device**.

Importing a CA Certificate to the Trustpool (CLI)

Procedure

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

Step 2	**crypto pki trust pool import terminal**
	Example:
	`Device(config)# crypto pki trust pool import terminal`
	`% Enter PEM-formatted CA certificate.`
		Imports the root certificate. For this, you need to paste the CA certificate from the **digicert.com**.

	Command or Action	Purpose

	`% End with a blank line or "quit" on a line by itself.`
	`-----BEGIN CERTIFICATE-----`
	`-----END CERTIFICATE-----`
	`-----BEGIN CERTIFICATE-----`
	`-----END CERTIFICATE-----`
	`-----BEGIN CERTIFICATE-----`
	`-----END CERTIFICATE-----`
	`Aug 23 02:47:33.450:`
	`%PKI-6-TRUSTPOOL_DOWNLOAD_SUCCESS:`
	`Trustpool Download is successful`	

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

Cleaning the CA Certificates Imported in Trustpool (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management**.

Step 2 In the **PKI Management** window, click the **Trustpool** tab.

Step 3 Click **Clean**.

Note This erases the downloaded CA certificate bundles. However, it does not erase the built-in CA certificate bundles.

Step 4 Click **Yes**.

Cleaning CA Certificates Imported in Trustpool (CLI)

You cannot delete a specific CA certificate from the trustpool. However, you can clear all the CA certificates that are imported to the Trustpool.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**crypto pki trustpool clean** **Example:** `Device(config)# crypto pki trustpool clean`	Erases the downloaded CA certificate bundles. However, it does not erase the built-in CA certificate bundles.

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

Creating a New Trustpoint Dedicated to a Single CA Certificate

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**crypto pki trustpoint** *tp-name* **Example:** `Device(config)# crypto pki trustpoint tp_name`	Creates a trustpoint.
Step 3	**enrollment terminal** **Example:** `Device(ca-trustpoint)# enrollment terminal`	Creates an enrollment terminal for the trustpoint.
Step 4	**exit** **Example:** `Device(ca-trustpoint)# exit`	Exits from the truspoint configuration.

Step 5	**crypto pki authenticate** *tp-name*
	Example:
	`Device(config)# crypto pki authenticate tp_name`
	`<<< PASTE CA-CERT in PEM format followed by quit >>>`
		Authenticates the trustpoint.

Verifying LSC Configuration

To view the details of the wireless management trustpoint, use the following command:

```
Device# show wireless management trustpoint

Trustpoint  Name : microsoft-ca
Certificate Info : Available
Certificate Type : LSC
Certificate Hash : 9e5623adba5307facf778e6ea2f5082877ea4beb
 Private key Info : Available
```

To view the LSC provision-related configuration details for an AP, use the following command:

```
Device# show ap lsc-provision summary

AP LSC-provisioning : Disabled
Trustpoint used for LSC-provisioning : microsoft-ca
LSC Revert Count in AP reboots : 10

AP LSC Parameters :
Country : IN
State : KA
City : BLR
Orgn : ABC
Dept : ABC
Email : support@abc.com
Key Size : 2048

AP LSC-provision List : Enabled
Total number of APs in provision list: 3

Mac Address
-----------
0038.df24.5fd0
2c5a.0f22.d4ca
e4c7.22cd.b74f
```

```
Device# show ap lsc-provision summary

AP LSC-provisioning : Disabled
Trustpoint used for LSC-provisioning : lsc-root-tp
Certificate chain status : Available
Number of certs on chain : 2
Certificate hash : 7f9d05183deecac4e5a79db65d538245685e8e30
LSC Revert Count in AP reboots : 1

AP LSC Parameters :
Country : IN
State : KA
City : BLR
Orgn : ABC
Dept : ABC
Email : support@abc.com
Key Size : 2048
EC Key Size : 384 bit

AP LSC-provision List :

Total number of APs in provision list: 2

Mac Addresses :
--------------
1880.90f5.1540
2c5a.0f70.84dc
```

Configuring Management Trustpoint to LSC (GUI)

Procedure

Step 1 Choose **Administration** > **Management** > **HTTP/HTTPS**.

Step 2 In the **HTTP Trust Point Configuration** section, set **Enable Trust Point** to the **Enabled** state.

Step 3 From the **Trust Points** drop-down list, choose the appropriate trustpoint.

Step 4 Save the configuration.

Configuring Management Trustpoint to LSC (CLI)

After LSC provisioning, the APs will automatically reboot and join at the LSC mode after bootup. Similarly, if you remove the AP LSC provisioning, the APs reboot and join at non-LSC mode.

In EWC, the internal APs will not automatically reboot. You should manually reboot the internal AP to make it work in LSC and non-LSC mode.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless management trustpoint** *trustpoint_name* **Example:** `Device(config)# wireless management trustpoint microsoft-ca`	Configures the management trustpoint to LSC. The internal AP will not able to join before a reload, so follow the steps given below to reload the internal AP.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 4	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Saves the configuration.
Step 5	**wireless ewc-ap ap reload** **Example:** `Device# wireless ewc-ap ap reload`	Reloads the internal AP. This will also reload the controller on the AP.

Information About MIC and LSC Access Points Joining the Controller

Overview of Support for MIC and LSC Access Points Joining the Controller

In Cisco IOS XE Bengaluru 17.4.1 and earlier releases, APs with a default certificate (Manufacturing Installed Certificates [MIC]) or Secure Unique Device Identifier [SUDI]) fail to join a Locally Significant Certificate-deployed (LSC-deployed) controller, where the management certificate of the controller is an LSC. To resolve this issue, you must provision LSC on these APs using the provisioning controller before moving them to the LSC-deployed controller.

From Cisco IOS XE Bengaluru 17.5.1 onwards, the new authorization policy configuration allows MIC APs to join the LSC-deployed controller, so that the LSC and MIC APs can coexist in the controller at the same time.

Recommendations and Limitations

- When the CA server is configured with manual enrollment (manual intervention) to accept Certificate Signing Request (CSR), the controller waits for the CA server to send the pending response. If there is no response from the CA server for 10 minutes, the fallback mode comes into effect.
  - Cisco Wave 2 APs regenerate CSR, and a fresh CSR is sent to the CA server.
  - Cisco IOS APs restart, and then Cisco IOS APs send a fresh CSR, which is in turn sent to the CA server.
- Locally significant certificate (LSC) on the controller does not work on the password challenge. Therefore, for LSC to work, you must disable password challenge on the CA server.
- If you are using Microsoft CA, we recommend that you use Windows Server 2012 or later as the CA server.

Configuration Workflow

1. Configuring LSC on the Controller (CLI), on page 610
2. Enabling the AP Certificate Policy on the APs (CLI), on page 611
3. Configuring the AP Policy Certificate (GUI), on page 612
4. Configuring the Allowed List of APs to Join the Controller (CLI), on page 613

Configuring LSC on the Controller (CLI)

The server certificate used by the controller for CAPWAP-DTLS is based on the following configuration.

Before you begin

- Ensure that you enable LSC by setting the appropriate trustpoints for the following wireless management services:
  - AP join process: CAPWAP DTLS server certificate
  - Mobility connections: Mobility DTLS certificate
  - NMSP and CMX connections: NMSP TLS certificate

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**[no] wireless management trustpoint** *trustpoint-name* **Example:** `Device(config)# wireless management trustpoint trustpoint-name`	Configures the LSC trustpoint in the LSC-deployed controller.

Enabling the AP Certificate Policy on the APs (CLI)

- If the management trustpoint is an LSC, by default, MIC APs fail to join the controller. This configuration acts as an enable or disable configuration knob that allows MIC APs to join the controller.
- This configuration is a controller authorization to allow APs to join MIC at the time of DTLS handshake.

To prevent manufacturing installed certificate (MIC) expiry failures, ensure that you configure a policy, as shown here:

- Create a certificate map and add the rules:

```
configure terminal
crypto pki certificate map map1 1
issuer-name co Cisco Manufacturing CA
```

Note You can add multiple rules and filters under the same map. The rule mentioned in the example above specifies that any certificate whose issuer-name contains *Cisco Manufacturing CA* (case insensitive) is selected under this map.

- Use the certificate map under the trustpool policy:

```
configure terminal
crypto pki trustpool policy
match certificate map1 allow expired-certificate
```

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap auth-list ap-cert-policy allow-mic-ap trustpoint** *trustpoint-name* **Example:** `Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint trustpoint-name`	Configures the trustpoint name for the controller certificate chain. **Note** The **allow-mic-ap trustpoint** command is required only for the virtual controller (Cisco Catalyst 9800-CL Wireless Controller for Cloud). In all the other appliance controller platforms, the default certificate is selected. This default certificate is manufacturer-installed SUDI.
Step 3	**ap auth-list ap-cert-policy allow-mic-ap** **Example:** `Device(config)# ap auth-list ap-cert-policy allow-mic-ap`	Enables the AP certificate policy during CAPWAP-DTLS handshake.
Step 4	**ap auth-list ap-cert-policy** {**mac-address** *H.H.H* | **serial-number** *serial-number-ap*} **policy-type mic** **Example:** `Device(config)# ap auth-list ap-cert-policy mac-address 1111.1111.1111 policy-type mic`	Enables the AP certificate policy as MIC.

Configuring the AP Policy Certificate (GUI)

Procedure

Step 1 Choose **Configuration > Wireless > Access Points**

Step 2 In the **All Access Points** window, click **AP Certificate Policy** .

Step 3 In the **AP Policy Certificate** window, complete the following actions:

a) Click the **Authorize APs joining with MIC** toggle button to enable AP authorization.

b) From the **Trustpoint Name** drop-down list, choose the required trustpoint.

c) Click **Add MAC or Serial Number** to add a MAC address or a serial number manually or through a .csv file.
The **Add MAC or Serial Number** window is displayed.

d) Click the **AP Authlist Type** and enter the MAC address or the serial number. Upload the .csv file or enter the MAC address in the list box.
The newly added MAC address and serial numbers are displayed under **List of MAC Address and Serial Numbers**.

e) Click **Apply**.

The AP certificate policy is added to the **AP Inventory** window.

Note To add a new AP with MIC, perform Step 1 to Step 3 described in Configuring the AP Policy Certificate (GUI) section. To add a new AP with LSC, perform the procedure described in the Configuring AP LSC Provision List (GUI) and Step 1 to Step 3 in the Configuring the AP Policy Certificate (GUI) section.

Configuring the Allowed List of APs to Join the Controller (CLI)

The allowed list of APs can either be populated based on the Ethernet MAC address or based on the serial number of the APs.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap auth-list ap-cert-policy** {**mac-address** *AP-Ethernet-MAC-address* | **serial-number** *AP-serial-number*} **policy-type mic** **Example:** `Device# ap auth-list ap-cert-policy mac-address 00b0.e192.0d98 policy-type mic`	Configures the AP certificate policy based on the Ethernet MAC address or based on the assembly serial number of the AP.

Verifying the Configuration Status

To verify if the APs have been authorized by the AP certificate policy, use the following command:

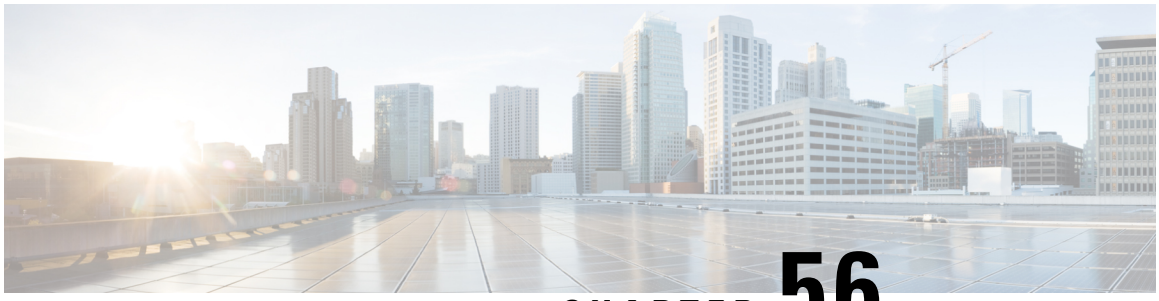
```
Device# show ap auth-list ap-cert-policy
Authorize APs joining with MIC : ENABLED
MIC AP policy trustpoint
Name : CISCO_IDEVID_SUDI
Certificate status : Available
Certificate Type : MIC
Certificate Hash : xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

To verify the AP certificate policy on the MAC address and the serial number of the AP, use the following commands:

```
Device# show ap auth-list ap-cert-policy mac-address
MAC address      AP cert policy
---------------------------------
1111.2222.3333   MIC

Device# show ap auth-list ap-cert-policy serial-number
Serial number    AP cert policy
---------------------------------
F1234567890      MIC
```

Note If you set an invalid trustpoint (not SSC), the **allow-mic-ap policy** is not enabled. If you set an invalid trustpoint, the following error is displayed on the console:

```
Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint lsc-root-tp
Dec 18 07:38:29.944: %CERT_MGR_ERRMSG-3-CERT_MGR_GENERAL_ERR: Chassis 1 R0/0: wncd: General
 error: MIC AP Policy trustpoint: 'lsc-root-tp' cert-chain type is LSC, It must be either
MIC or vWLC-SSC
```



CHAPTER 56

Certificate Management

About Public Key Infrastructure Management (GUI)

The Public Key Infrastructure (PKI) Management page displays the following tabs:

Trustpoints tab: Used to add, create or enroll a new trustpoint. This page also displays the current trustpoints configured on the controller and other details of the trustpoint. You can also view if the trustpoint is in use for any of the features. For example, Webadmin or AP join (Wireless Management Interface), and others.

CA Server tab: Used to enable or disable the Certificate Authority (CA) server functionality on the controller. The CA server functionality should be enabled for the controller to generate a Self Signed Certificate (SSC).

Key Pair Generation tab: Used to generate key pairs.

Certificate Management tab: Used to generate and manage certificates, and perform all certificate related operations, on the controller.

Authenticating and Enrolling a PKI Trustpoint (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management**.

Step 2 In the **PKI Management** window, click the **Trustpoints** tab.

Step 3 In the **Add Trustpoint** dialog box, provide the following information:

a) In the **Label** field, enter the RSA key label.

b) In the **Enrollment URL** field, enter the enrollment URL.

c) Check the **Authenticate** check box to authenticate the Public Certificate from the enrollment URL.

d) In the **Subject Name** section, enter the **Country Code**, **State**, **Location**, **Organisation**, **Domain Name**, and **Email Address**.
e) Check the **Key Generated** check box to view the available RSA keypairs. Choose an option from the **Available RSA Keypairs** drop-down list.
f) Check the **Enroll Trustpoint** check box.
g) In the **Password** field, enter the password.
h) In the **Re-Enter Password** field, confirm the password.
i) Click **Apply to Device**.

The new trustpoint is added to the trustpoint name list.

Generating an AP Self-Signed Certificate (GUI)

Note This section is valid only for virtual controllers (Cisco Catalyst 9800-CL Wireless Controller for Cloud) and not applicable for appliance based controllers (Cisco Catalyst 9800-40 Wireless Controller, Cisco Catalyst 9800-80 Wireless Controller, Cisco Catalyst 9800-L Wireless Controller (Copper Uplink), and Cisco Catalyst 9800-L Wireless Controller (Fiber Uplink)).

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management**.
Step 2 In the **AP SSC Trustpoint** area, click **Generate** to generate an AP SSC trustpoint.
Step 3 From the **RSA Key-Size** drop-down list, choose a key size.
Step 4 From the **Signature Algorithm** drop-down list, choose an option.
Step 5 From the **Password Type** drop-down list, choose a password type.
Step 6 In the **Password** field, enter a password. The valid range is between 8 and 32 characters.
Step 7 Click **Apply to Device**.

Adding the Certificate Authority Server (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management**.
Step 2 In the **PKI Management** window, click the **CA Server** tab.
Step 3 In the **CA Server** section, click the **Shutdown Status** toggle button, to enable the status. If you choose the shutdown status as **Enabled**, you must enter the password and confirm the same.
Step 4 If you choose the shutdown status as **Disabled**, you must enter the **Country Code**, **State**, **Location**, **Organisation**, **Domain Name**, and **Email Address**.

Step 5 Click **Apply** to add the CA server.

Step 6 Click **Remove CA Server** to delete the CA server.

Adding an RSA or EC Key for PKI Trustpoint (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management**.

Step 2 In the **PKI Management** window, click the **Key Pair Generation** tab.

Step 3 In the **Key Pair Generation** section, click **Add**.

Step 4 In the dialog box that is displayed, provide the following information:

a) In the **Key Name** field, enter the key name.

b) In the **Key Type** options, select either **RSA Key** or **EC Key**.

c) In the **Modulus Size** field, enter the modulus value for the RSA key or the EC key. The default modulus size for the RSA key is 4096 and the default value for the EC key is 521.

d) Check the **Key Exportable** check box to export the key. By default, this is checked.

e) Click **Generate**.

Adding and Managing Certificates

To add and manage certificates, use one of the following methods:

Note While configuring a password for the .pfx file, do not use the following ASCII characters: "*, ^, (), [], \, ", and +"

Using these ASCII characters results in error with bad configuration and does not import the certificate to the controller.

Method 1

Procedure

Step 1 Choose **Configuration** > **Security** > **PKI Management** > **Add Certificate**.

Step 2 Click **Generate Certificate Signing Request**.

a) In the **Certificate Name** field, enter the certificate name.

b) From the **Key Name** drop-down list, choose an RSA key pair. (Click the plus (+) icon under the **Key Pair Generation** tab to create new RSA key pairs.).

c) Enter values the **Country Code**, **Location**, **Organisation**, **State**, **Organizational Unit**, and the **Domain Name** fields.

d) Click **Generate**.
The generated Certificate Signing Request (CSR) is displayed on the right. Click **Copy** to copy and save a local copy. Click **Save to Device** to save the generated CSR to the /bootflash/csr directory.

Step 3 Click **Authenticate Root CA** .

a) From the **Trustpoint** drop-down list, choose the trustpoint label generated in Step 2, or any other trustpoint label that you want to authenticate.

b) In the **Root CA Certificate (.pem)** field, copy and paste the certificate that you have received from the CA.

Note Ensure that you copy and paste the PEM Base64 certificate of the issuing CA of the device certificate.

c) Click **Authenticate**.

Step 4 Click **Import Device Certificate** .

a) From the **Trustpoint** drop-down list, choose the trustpoint label that was generated in Step 2, or any other trustpoint label that you want to authenticate.

b) In the **Signed Certificate (.pem)** field, copy and paste the signed certificate that you received, from your CA.

c) Click **Import**.

This completes the device certificate import process and the certificate can now be assigned to features.

Method 2

Procedure

Click **Import PKCS12 Certificate** .

Note You can import an entire certificate chain in the PKCS12 format using different transport types.

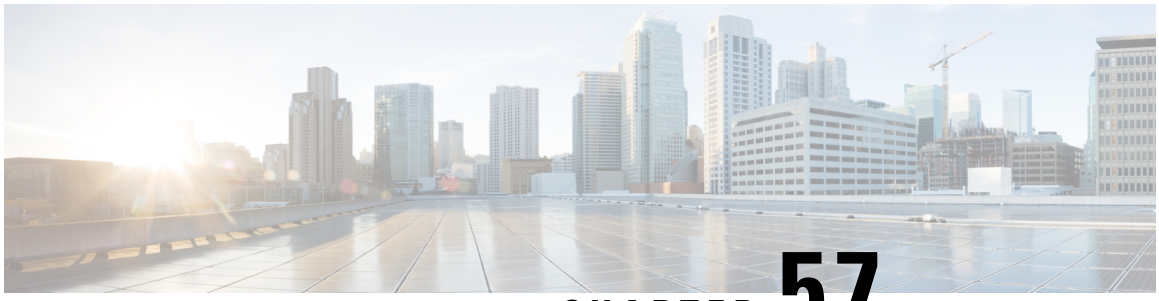
a) From the **Transport Type** drop-down list, choose either **FTP**, **SFTP**, **TFTP**, **SCP**, or **Desktop (HTTPS)**.

For **FTP**, **SFTP**, and **SCP**, enter values in the **Server IP Address (IPv4/IPv6)**, **Username**, **Password**, **Certificate File Path**, **Certificate Destination File Name**, and **Certificate Password** fields.

For **TFTP**, enter values in the **Server IP Address (IPv4/IPv6)**, **Certificate File Path**, **Certificate Destination File Name**, and **Certificate Password** fields.

For **Desktop (HTTPS)**, enter values in the **Source File Path** and **Certificate Password** fields.

b) Click **Import**.



CHAPTER 57

User and Entity Behavior Analysis

Information About User and Entity Behavior Analysis

User and Entity Behavior Analysis (UEBA) is a solution that has a number of security techniques, which allow you to profile and track the behavior of users and devices, in order to identify potential inside threats and targeted attacks in networks, when anomalies occur.

For instance, employees of an enterprise may unintentionally download a malicious piece of software that might include some backdoor or leakage in company secrets. This is detected by the change in the pattern of communication from one or more devices or users in the network, compared to an established baseline.

User and Entity Behavior Analysis can be deployed using two methods:

- User Datagram Protocol (UDP) collector (Cisco Digital Network Architecture (DNA) Center is a UDP collector)
- Stealthwatch Cloud (SwC) - The Embedded Wireless Controller (EWC) directly uploads data to SwC.

Configuring User and Entity Behavior Analysis (Using UDP Collector)

In a Cisco DNA Center-based deployment, the controller acts as the collector of NetFlow information that is sent to Cisco DNA Center. In turn, Cisco DNA Center compresses the information for SwC. The controller enables Application Visibility and Control (AVC) on the access points (APs) and maintains the communication channel with Cisco DNA Center.

In EWC, you can also send FnFv9 data through the UDP to a UDP collector.

In the Non-Cisco DNA-C based deployment, the FnF flow records are directly sent to SwC from the controller.

Configuring User and Entity Behavior Analysis (Using Stealthwatch Cloud)

The following sections provide information about configuring the User and Entity Behavior Analysis solution using Stealthwatch Cloud (GUI and CLI).

Configuring User and Entity Behavior Analysis Using Stealthwatch Cloud (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Threat Defense**.

Step 2 Click **Cisco StealthWatch Integration**.

Step 3 On the Stealthwatch page, in the **Service Key** field, enter the Stealthwatch cloud service key.

Step 4 Click the cloud icon to view the detailed statistics of Stealthwatch.

Step 5 In the **Sensor Name** field, enter a sensor name for Stealthwatch Cloud registration.

Step 6 In the **URL** field, enter the Stealthwatch Cloud server URL.

Step 7 Click **Apply**.

Step 8 (Optional) Click **Unconfigure StealthWatch**, to unconfigure Stealthwatch Cloud.

What to do next

You can view and verify the Stealthwatch Cloud's health status in the **Stealthwatch Health Status**

Configuring Stealthwatch Cloud (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**stealthwatch-cloud-monitor** **Example:** `Device(config)# stealthwatch-cloud-monitor`	Configures the Stealthwatch Cloud monitor. Enters the Stealthwatch Cloud Monitor configuration mode.

	Command or Action	Purpose
Step 3	**service-key** *swc-service-key* **Example:** `Device(config-stealthwatch-cloud-monitor)# service-key xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx`	(Optional) Sets the Stealthwatch Cloud service key. Service key is provided by the SwC portal. The alternative to service key is the authentication through the IP address allowed list. For more information about service key and allowed lists, see the appropriate SwC guide.
Step 4	**sensor-name** *swc-sensor-name* **Example:** `Device(config-stealthwatch-cloud-monitor)# sensor-name swc-sensor-name`	(Optional) Provides a sensor name for the Stealthwatch Cloud registration. The device serial number is the default value.
Step 5	**url** *SwC-server-url* **Example:** `Device(config-stealthwatch-cloud-monitor)# url https://sensors.eu-2.obsrvbl.com`	Sets the Stealthwatch Cloud server URL.

Mapping Stealthwatch Cloud to Flow Measurements

There are two options to map Stealthwatch Cloud to flow measurements, namely the flow-exporter configuration and the flow-monitor configuration.

Note At any given period, there can be only one internal and one external active flow exporter. An active flow exporter is an exporter that is bound to the flow monitor that is bound to a wireless profile.

Configuring Flow Exporter for Stealthwatch Cloud

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow exporter** *flow-exporter-name* **Example:**	Defines the flow exporter.

	Command or Action	Purpose
	`Device(config)# flow exporter flow-exporter-name`	**Note** At a given moment, there can be only one internal and one external active flow exporter. An active flow exporter is an exporter that is bound to the flow monitor, which is bound to a wireless profile.
Step 3	**destination stealthwatch-cloud** **Example:** `Device(config-flow-exporter)# destination stealthwatch-cloud`	Exports the flow information to Stealthwatch Cloud.

Configuring Flow Monitor for Stealthwatch Cloud

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**flow monitor** *flow-monitor-name* **Example:** `Device(config)# flow monitor flow-monitor-name`	Defines the flow monitor.
Step 3	**exporter** *flow-exporter-name* **Example:** `Device(config-flow-monitor)# exporter flow-exporter-name`	Exports the flow information to the exporter.
Step 4	**record wireless avc basic** **Example:** `Device(config-flow-monitor)# record wireless avc basic`	Specifies the flow record with basic IPv4 wireless AVC template.
Step 5	**end** **Example:** `Device(config-flow-monitor)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Example: Stealthwatch Cloud Configuration

The following example shows a complete CLI configuration of Stealthwatch Cloud:

```
stealthwatch-cloud-monitor
  service-key XXXXXXXXXXXXXXXXXXXXXXXXXXXX
  sensor-name ewc-sensor
  url https://sensors.eu-2.obsrvbl.com

flow exporter fexp-swc
 destination stealthwatch-cloud

flow monitor fm-avc-swc
 exporter fexp-swc
 record wireless avc basic

wireless profile policy swc-policy-profile
 ipv4 flow monitor fm-avc-swc input
 ipv4 flow monitor fm-avc-swc output
 ipv6 flow monitor fm-avc-swc input
 ipv6 flow monitor fm-avc-swc output

wlan my-wlan 1 my-wlan

wireless tag policy swc-policy-tag
 wlan my-wlan policy swc-policy-profile

ap 0000.0000.0001
 policy-tag swc-policy-tag
```

Verifying Stealthwatch Cloud Details

To verify the state and statistics of Stealthwatch Cloud, use the **show stealthwatch-cloud wireless-shim** command:

```
Device# show stealthwatch-cloud wireless-shim
Stealthwatch-Cloud wireless shim

Total
 RX records        : 15
 RX bytes          : 2345
 TX records        : 10
 TX bytes          : 1234
 TX batches        : 1
 Failed batches    : 0
 Non-SWC records   : 5

Buffers
 Status        : TX
 Size          : 1272000
 Compressed    : 8
 Uncompressed  : 0
 Records       : 8

 Status        : Filling
 Size          : 1272000
 Compressed    : 2
 Uncompressed  : 0
 Records       : 2
```

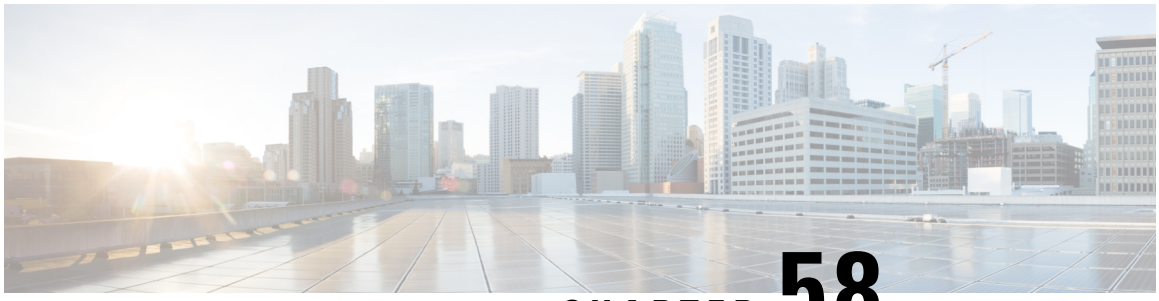
To verify the Stealthwatch Cloud connection details, use the **show stealthwatch-cloud connection** command.

```
Device# show stealthwatch-cloud connection
Stealthwatch-Cloud details
  Registration
    #ID         : 0xe6000001
```

```
    URL         : https://sensors.eu-2.obsrvbl.com
    Service Key : XXXXXXXXXXXXXXXXXXXXXXXXXXXX
    Sensor Name : ewc-sensor
    Registered  : Yes
  Connection
    Status               : UP
    Last status update   : 03/17/2020 21:44:55
    # Flaps              : 0
    # Heartbeats         : 9
    # Lost heartbeats    : 1
    Total RX bytes       : 4567
    Total TX bytes       : 1234
    Upload Speed (B/s)   : 247
    Download Speed (B/s) : 269
    # Open sessions      : 0
    # Redirections       : 0
    # Timeouts           : 0

  HTTP Events
    GET response                  : 1
    GET request                   : 1
    GET Status Code 2XX           : 1
    PUT response                  : 1
    PUT request                   : 1
    PUT Status Code 2XX           : 1
    POST response                 : 12
    POST request                  : 12
    POST Status Code 2XX          : 11
    POST Status Code 4XX          : 1
API Events
    Abort                         : 1

  Event History
Timestamp               #Times   Event               RC Context
----------------------- -------- ------------------- --
----------------------------------------
03/21/2020 10:42:06.161 9        HEARTBEAT_OK         0

03/20/2020 06:49:05.717 1        HEARTBEAT_FAIL       0 HTTPCON_EV_TIMEOUT (6)

03/20/2020 06:47:05.717 1        SEND_START           0 ID:0001
03/20/2020 06:49:05.717 3        SIGNAL_DATA_FAIL     0 ID:0001, attempt : 3
03/18/2020 09:23:39.375 1        REGISTER_OK          0

03/18/2020 09:23:13.276 1        REGISTER_SEND        0

03/18/2020 09:23:12.154 1        SEND_ABORT_ALL       0 config change

03/18/2020 09:23:12.154 1       OPTIONS_CONFIG      0 URL https://sensor.staging.obsrvbl.com

03/18/2020 09:23:12.154 1       OPTIONS_CONFIG      0 Service-key XXXXXXXXXXXXXXXXXXXXXXXX
03/18/2020 09:23:12.154 1        OPTIONS_CONFIG       0 Host ewc-sensor => reset

03/18/2020 09:23:12.154 1        OPTIONS_CONFIG       0 cfg-mode manual => reset
```

PART VII

High Availability



CHAPTER 58

High Availability

High Availability Active and Standby

The Cisco Embedded Wireless Controller on Catalyst Access Points (EWC), is supported on the Cisco Catalyst 9100 series APs. The active AP election process determines which of the Cisco Catalyst 9100 series APs is elected to run the EWC controller function. Once the active AP is elected and other subordinate EWC-capable Cisco Catalyst 9100 series APs join the active AP, it selects a standby AP and redundancy is formed.

This High Availability (HA) architecture is based on the Cisco Catalyst 9800 HA architecture, with a few additions:

HA pairing is different in EWC. For the initial bring-up, the EWC active AP waits until all the APs join the controller. The active AP then selects the designated standby AP (either by auto-selection or configuration), and communicates the role and the HA parameters (local/peer IP, keepalive interval, priority) to the selected AP, through a CAPWAP control message.

Note After a power outage, the standby AP does not come up in the EWC HA pair. The standby AP tries to come up but fails. Then another EWC capable AP is selected as standby, which fails to come up. To avoid this situation, ensure that the APs have the same IP version to be elected as a HA pair.

The selected standby AP starts and dynamically configures the HA parameters without manual intervention.

Monitoring Redundancy between Active and Standby Access Points

To view the redundancy between active AP and standby APs, follow the steps given below:

Procedure

Step 1 Open the Cisco Embedded Wireless Controller for Catalyst Access Points GUI.

Step 2 Choose **Monitoring** > **General** > **System**.

Step 3 Click the **Redundancy** tab.

In the **General** tab, you can view the current state, peer state, redundancy modes, and the chassis details of the active and standyby APs.

Active Access Point election Process

The EWC election process is used to choose the AP on which the controller is started. Virtual Router Redundancy Protocol (VRRP) is used to elect the active AP. The logic used to elect the EWC active AP and standby AP is described in the following sections.

Selecting the Active EWC Access Point

The following points are used to compare and select an Active EWC AP:

- If you have configured an AP to be a preferred controller, it takes the highest precedence.
- The AP type is compared next. The APs with higher model numbers have higher values. The AP having the highest value becomes the active AP.
- If the APs have the same AP type, the client load (number of associated clients) is compared, and the AP with the smallestclient load is selected.
- If all the methods mentioned above fail (all are equal among the APs), then the AP with the lowest MAC address becomes the active AP.

Selecting the Standby EWC Access Points

The standby EWC AP is not selected using VRRP. The following is the selection process for the standby EWC AP, on day-1:

- After the active EWC AP is selected, the active AP waits for the external APs to join, to begin the standby AP selection.
- Once the external APs join, the active AP assigns a priority to all the joined APs. The AP with the highest priority is selected as the standby AP. If multiple APs match the same highest priority, the AP with the lowest MAC address gets selected. Only EWC-capable APs with an EWC image installed are considered for the selection process.
- Priority is calculated based on the following parameters:
  - Explicit user configuration to choose a particular AP as the next preferred controller (highest priority)
  - AP type
  - AP join time

Note There is no concept of standby on day 0. On day 0, you there is only one active EWC AP. If the active EWC AP goes down for some reason, the VRRP election takes place again, to elect a new active EWC AP.

Note If a controller is running on an AP, this AP will have a higher priority compared to the other APs not running as the controller. For example, if you bring-up a Cisco Catalyst 9115AX Series AP, since there are no other APs to choose from, this AP become the active AP and starts the controller. Later, if you bring-up a Cisco Catalyst 9117AX Series AP on this network, although the Cisco Catalyst 9117AX Series AP has a higher model number, it does not become the controller, since you already have a controller running in the network. Election will take place only if you bring-up two APs at the same time.

Selecting the Preferred Controller

To select the preferred controller and to make it the controller, follow the steps given below:

Before you begin

The active EWC AP and standby EWC APs are selected by the process described in the earlier topics. For some reason, if you want to select another AP as the standby, you can select any EWC-capable AP as a preferred controller, from the GUI.

Note When you select another AP that is not the current standby AP to be the preferred controller, the current standby AP goes down and the new EWC AP you have selected becomes the standby EWC AP.

Procedure

Step 1 Open the Cisco Embedded Wireless Controller for Catalyst Access Points GUI.

Step 2 Choose **Configuration** > **Wireless** > **Access Points**.

Step 3 Click the AP that you want to make as the preferred controller.
The **Edit AP** window is displayed.

Step 4 Click the **Advanced** tab.

Step 5 In the **Embedded Wireless Controller** section, check the **Preferred Controller** check box.

Step 6 Click **Update & Apply to Device**.

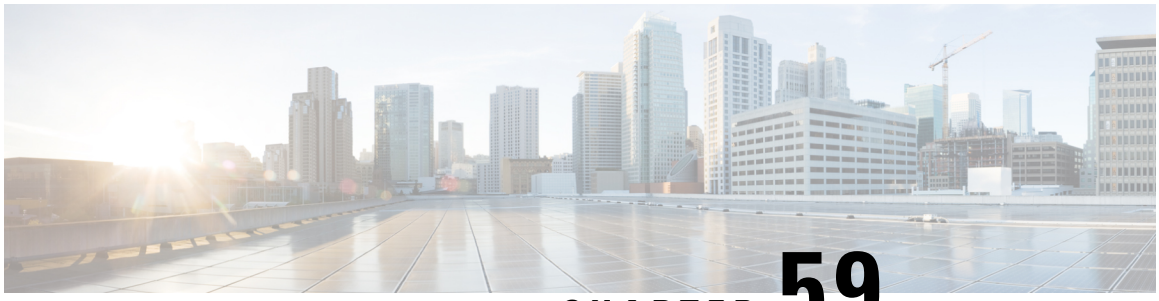
What to do next

Return to the **Advanced** tab, and click **Make Controller**. Then click **Update & Apply to Device**.

Note A warning message is displayed mentioning that this operation will disrupt the network, as the controller will reset.

PART VIII

Quality of Service



CHAPTER 59

Quality of Service

Wireless QoS Overview

Quality of Service (QoS), provides the ability to prioritize the traffic by giving preferential treatment to specific traffic over the other traffic types. Without QoS, the device offers best-effort service for each packet, regardless of the packet contents or size. The device sends the packets without any assurance of reliability, delay bounds, or throughput.

A target is the entity where the policy is applied. Wireless QoS policies for SSID and client are applied in the upstream and (or) downstream direction. The flow of traffic from a wired source to a wireless target is known as downstream traffic. The flow of traffic from a wireless source to a wired target is known as upstream traffic.

The following are some of the specific features provided by wireless QoS:

- SSID and client policies on wireless QoS targets
- Marking and Policing (also known as Rate Limiting) of wireless traffic

Wireless QoS Targets

This section describes the various wireless QoS targets available on a device.

SSID Policies

You can create QoS policies on SSID in both the ingress and egress directions. If not configured, there is no SSID policy applied.

The policy is applicable per AP per SSID.

You can configure policing and marking policies on SSID.

Client Policies

Client policies are applicable in the ingress and egress direction. You can configure policing and marking policies on clients. AAA override is also supported.

Supported QoS Features on Wireless Targets

This table describes the various features available on wireless targets.

Table 31: QoS Features Available on Wireless Targets

Target	Features	Direction Where Policies Are Applicable
SSID	• Set • Police • Drop	Upstream and downstream
Client	• Set • Police • Drop	Upstream and downstream

Note For Drop support, the Drop action is achieved by the following configuration:

```
police <rate>
   conform-action drop
   exceed-action drop
```

Direct **action drop** is not supported.

Precious Metal Policies for Wireless QoS

The precious metal policies are system-defined policies that are available on the embedded wireless controller. They cannot be removed or changed.

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 are pre-configured. They cannot be modified.

For client metal policies, they can be pushed using AAA.

Based on the policies applied, the 802.11e (WMM), and DSCP fields in the packets are affected.

For more information about metal policies format see the Metal Policy Format, on page 636 section.

For more information about DSCP to UP mapping, see the Architecture for Voice, Video and Integrated Data (AVVID), on page 642 table.

Prerequisites for Wireless QoS

Before configuring wireless QoS, you must have a thorough understanding of these items:

- Wireless concepts and network topologies.
- Understanding of QoS implementation.
- Modular QoS CLI (MQC). For more information on Modular QoS, see the MQC guide
- The types of applications used and the traffic patterns on your network.
- Bandwidth requirements and speed of the network.

Restrictions for QoS on Wireless Targets

General Restrictions

A target is an entity where a policy is applied. A policy can be applied to a wireless target, which can be an SSID or client target, in the downstream and/or upstream direction. Downstream indicates that traffic is flowing from the controller to the wireless client. Upstream indicates that traffic is flowing from wireless client to the controller.

- Hierarchical (Parent policy and child policy) QoS is not supported.
- One policy per target per direction is supported.
- Only BSSID and client targets are supported, on both directions.
- The following policy formats are supported:
  - QoS Policy Action
    - Police:

      ```
      police [cir | rate] bps [conform-action action] [exceed-action action]
      ```

      Policer action types are **transmit** or **drop**.

    - Set:

      ```
      set dscp
       set wlan user-priority
      ```

Note **set wlan user-priority** (downstream only; BSSID only)

- QoS Policy Classification

```
match [not] access-group
 match [not] dscp
 match [not] protocol
```

AP Side Restrictions

- In Cisco Embedded Wireless Controller, FlexConnect local switching, and SDA deployments, the QoS policies are enforced on the AP. Due to this AP-side restriction, police actions (e.g., rate limiting) are only enforced at a per flow (5-tuple) level and not per client.

Control Plane Rate Limiting and Policing

You need not explicitly configure control plane rate limiting or policing on the controller. The controller has embedded mechanisms (like policers) to protect the CPU by policing control plane traffic directed towards it. If you're migrating from AireOS to IOS-XE, this change is taken care of at the code level.

Metal Policy Format

Metal Policy Format

Metal Policies are system defined, and you cannot change it or delete it. There are four levels of metal policy - Platinum, Gold, Silver, and Bronze.

Note Each metal policy defines a DSCP ceiling so that the DSCP or the UP marking does not exceed a certain value.

For Platinum the value is 46, Gold is AF41, Silver is 22, and Bronze is CS1.

Policy Name	Policy-map Format	Class-map Format

platinum	policy-map platinum
	 class cm-dscp-34
	  set dscp af41
	 class cm-dscp-45
	  set dscp 45
	 class cm-dscp-46
	  set dscp ef
	 class cm-dscp-47
	  set dscp 47
		class-map match-any cm-dscp-34
		 match dscp af41
		class-map match-any cm-dscp-45
		 match dscp 45
		class-map match-any cm-dscp-46
		 match dscp ef
		class-map match-any cm-dscp-47
		 match dscp 47
		class-map match-any cm-dscp-0
		 match dscp default

gold	policy-map gold
	 class cm-dscp-45
	  set dscp af41
	 class cm-dscp-46
	  set dscp af41
	 class cm-dscp-47
	  set dscp af41	

silver	policy-map silver
	 class cm-dscp-34
	  set dscp default
	 class cm-dscp-45
	  set dscp default
	 class cm-dscp-46
	  set dscp default
	 class cm-dscp-47
	  set dscp default	

bronze	policy-map bronze
	 class cm-dscp-0
	  set dscp cs1
	 class cm-dscp-34
	  set dscp cs1
	 class cm-dscp-45
	  set dscp cs1
	 class cm-dscp-46
	  set dscp cs1
	 class cm-dscp-47
	  set dscp cs1	

Policy Name	Policy-map Format	Class-map Format

platinum-up	policy-map platinum-up
	 class cm-dscp-set1-for-up-4
	  set dscp af41
	 class cm-dscp-set2-for-up-4
	  set dscp af41
	 class cm-dscp-for-up-5
	  set dscp af41
	 class cm-dscp-for-up-6
	  set dscp ef
	 class cm-dscp-for-up-7
	  set dscp ef
		class-map match-any cm-dscp-for-up-0
		  match dscp default
		  match dscp cs2
		class-map match-any cm-dscp-for-up-1
		  match dscp cs1
		class-map match-any cm-dscp-set1-for-up-4
		  match dscp cs3
		  match dscp af31
		  match dscp af32
		  match dscp af33
		class-map match-any cm-dscp-set2-for-up-4
		  match dscp af41
		  match dscp af42
		  match dscp af43
		class-map match-any cm-dscp-for-up-5
		  match dscp cs4
		  match dscp cs5
		class-map match-any cm-dscp-for-up-6
		  match dscp 44
		  match dscp ef
		class-map match-any cm-dscp-for-up-7
		  match dscp cs6
		  match dscp cs7

gold-up	policy-map gold-up
	 class cm-dscp-for-up-6
	  set dscp af41
	 class cm-dscp-for-up-7
	  set dscp af41	

silver-up	policy-map silver-up
	 class cm-dscp-set1-for-up-4
	  set dscp default
	 class cm-dscp-set2-for-up-4
	  set dscp default
	 class cm-dscp-for-up-5
	  set dscp default
	 class cm-dscp-for-up-6
	  set dscp default
	 class cm-dscp-for-up-7
	  set dscp default	

bronze-up	policy-map bronze-up
	 class cm-dscp-for-up-0
	  set dscp cs1
	 class cm-dscp-for-up-1
	  set dscp cs1
	 class cm-dscp-set1-for-up-4
	  set dscp cs1
	 class cm-dscp-set2-for-up-4
	  set dscp cs1
	 class cm-dscp-for-up-5
	  set dscp cs1
	 class cm-dscp-for-up-6
	  set dscp cs1
	 class cm-dscp-for-up-7
	  set dscp cs1	

Policy Name	Policy-map Format	Class-map Format
clwmm-platinum	policy-map clwmm-platinum class voice-plat set dscp ef class video-plat set dscp af41 class class-default set dscp default	class-map match-any voice-plat match dscp ef class-map match-any video-plat match dscp af41 class-map match-any voice-gold match dscp ef class-map match-any video-gold match dscp af41
clwmm-gold	policy-map clwmm-gold class voice-gold set dscp af41 class video-gold set dscp af41 class class-default set dscp default	
clnon-wmm-platinum	policy-map clnon-wmm-platinum class class-default set dscp ef	
clnon-wmm-gold	policy-map clnon-wmm-gold class class-default set dscp af41	
clsilver	policy-map clsilver class class-default set dscp default	
clbronze	policy-map clbronze class class-default set dscp cs1	

Auto QoS Policy Format

Policy Name	Policy-map Format	Class-map Format

enterprise-avc	policy-map AutoQos-4.0-wlan-ET-SSID-Input-AVC-Policy
	 class AutoQos-4.0-wlan-Voip-Data-Class
	  set dscp ef
	 class AutoQos-4.0-wlan-Voip-Signal-Class
	  set dscp cs3
	 class AutoQos-4.0-wlan-Multimedia-Conf-Class
	  set dscp af41
	 class AutoQos-4.0-wlan-Transaction-Class
	  set dscp af21
	 class AutoQos-4.0-wlan-Bulk-Data-Class
	  set dscp af11
	 class AutoQos-4.0-wlan-Scavanger-Class
	  set dscp cs1
	 class class-default
	  set dscp default
	policy-map AutoQos-4.0-wlan-ET-SSID-Output-Policy
	class AutoQos-4.0-RT1-Class
	set dscp ef
	class AutoQos-4.0-RT2-Class
	set dscp af31
	class class-default	

Policy Name	Policy-map Format	Class-map Format
		class-map match-any AutoQos-4.0-wlan-Voip-Data-Class match dscp ef class-map match-any AutoQos-4.0-wlan-Voip-Signal-Class match protocol skinny match protocol cisco-jabber-control match protocol sip match protocol sip-tls class-map match-any AutoQos-4.0-wlan-Multimedia-Conf-Class match protocol cisco-phone-video match protocol cisco-jabber-video match protocol ms-lync-video match protocol webex-media class-map match-any AutoQos-4.0-wlan-Transaction-Class match protocol cisco-jabber-im match protocol ms-office-web-apps match protocol salesforce match protocol sap class-map match-any AutoQos-4.0-wlan-Bulk-Data-Class match protocol ftp match protocol ftp-data match protocol ftps-data match protocol cifs class-map match-any AutoQos-4.0-wlan-Scavanger-Class match protocol netflix match protocol youtube match protocol skype match protocol bittorrent class-map match-any AutoQos-4.0-RT1-Class match dscp ef

Policy Name	Policy-map Format	Class-map Format
		match dscp cs6 class-map match-any AutoQos-4.0-RT2-Class match dscp cs4 match dscp cs3 match dscp af41
voice	policy-map platinum-up class dscp-for-up-4 set dscp 34 class dscp-for-up-5 set dscp 34 class dscp-for-up-6 set dscp 46 class dscp-for-up-7 set dscp 46 policy-map platinum class cm-dscp-34 set dscp 34 class cm-dscp-46 set dscp 46	
guest	Policy Map AutoQos-4.0-wlan-GT-SSID-Output-Policy Class class-default set dscp default Policy Map AutoQos-4.0-wlan-GT-SSID-Input-Policy Class class-default set dscp default	
port (only applies to Local Mode)	policy-map AutoQos-4.0-wlan-Port-Output-Policy class AutoQos-4.0-Output-CAPWAP-C-Class priority level 1 class AutoQos-4.0-Output-Voice-Class priority level 2 class class-default ip access-list extended AutoQos-4.0-Output-Acl-CAPWAP-C permit udp any eq 5246 16666 any	class-map match-any AutoQos-4.0-Output-CAPWAP-C-Class match access-group name AutoQos-4.0-Output-Acl-CAPWAP-C class-map match-any AutoQos-4.0-Output-Voice-Class match dscp ef

Architecture for Voice, Video and Integrated Data (AVVID)

IETF DiffServ Service Class	DSCP	IEEE 802.11e	
		User Priority	Access Category
Network Control	(CS7) CS6	0	AC_BE
Telephony	EF	6	AC_VO
VOICE-ADMIT	44	6	AC_VO
Signaling	CS5	5	AC_VI

IETF DiffServ Service Class	DSCP	IEEE 802.11e	
		User Priority	Access Category
Multimedia Conferencing	AF41 AF42 AF43	4	AC_VI
Real-Time Interactive	CS4	5	AC_VI
Multimedia Streaming	AF31 AF32 AF33	4	AC_VI
Broadcast Video	CS3	4	AC_VI
Low-Latency Data	AF21 AF22 AF23	3	AC_BE
OAM	CS2	0	AC_BE
High-Throughput Data	AF11 AF12 AF13	2	AC_BK
Standard	DF	0	AC_BE
Low-Priority Data	CS1	1	AC_BK
Remaining	Remaining	0	

How to apply Bi-Directional Rate Limiting

Information about Bi-Directional Rate Limiting

Bi-Directional Rate Limiting (BDRL) feature defines rate limits on both upstream and downstream traffic. These rate limits are individually configured. The rate limits can be configured on WLAN directly instead of QoS profiles, which will override QoS profile values. The WLAN rate limiting will always supersede Global QoS setting for controller and clients.

BDRL feature defines throughput limits for clients on their wireless networks and allows setting a priority service to a particular set of clients.

The following four QoS profiles are available to configure the rate limits:

- Gold

- Platinum
- Silver
- Bronze

The QoS profile is applied to all clients on the associated SSID. Therefore all clients connected to the same SSID will have the same rate limits.

To configure BDRL, select the QoS profile and configure the various rate limiting parameters. When rate limiting parameters are set to *0*, the rate limiting feature is not functional. Each WLAN has a QoS profile associated with it in addition to the configuration in the QoS profile.

Note BDRL in a mobility Anchor-Foreign setup must be configured both on Anchor and Foreign controller. As a best practice, it is recommended to perform identical configuration on both the controllers to avoid breakage of any feature.

BDRL is is supported on Guest anchor scenarios. The feature is supported on IRCM guest scenarios with AireOS as Guest anchor or Guest Foreign. Cisco Catalyst 9800 Series Wireless Controller uses **Policing** option to rate limit the traffic.

To apply metal policy with BDRL, perform the following tasks:

- Configure Metal Policy on SSID
- Configure Metal Policy on Client
- Configure Bi-Directional Rate Limiting for All Traffic, on page 646
- Configure Bi-Directional Rate Limiting Based on Traffic Classification, on page 646
- Apply Bi-Directional Rate Limiting Policy Map to Policy Profile, on page 648
- Apply Metal Policy with Bi-Directional Rate Limiting, on page 649

Prerequisites for Bi-Directional Rate Limiting

- Client metal policy is applied through AAA-override.
- You must specify the metal policy on ISE server.
- AAA-override must be enabled on policy profile.

Configure Metal Policy on SSID

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**wireless profile policy** *policy-profile-name* **Example:** `Device(config)# wireless profile policy policy-profile1`	Configures WLAN policy profile and enters wireless policy configuration mode.
Step 3	**description** *description* **Example:** `Device(config-wireless-policy)# description policy-profile1`	Adds a user defined description to the new wireless policy.
Step 4	**service-policy input** *input-policy* **Example:** `Device(config-wireless-policy)# service-policy input platinum-up`	Sets platinum policy for input.
Step 5	**service-policy output** *output-policy* **Example:** `Device(config-wireless-policy)# service-policy output platinum`	Sets platinum policy for output.

Configure Metal Policy on Client

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-profile-name* **Example:** `Device(config)# wireless profile policy policy-profile1`	Configures WLAN policy profile and enters wireless policy configuration mode.
Step 3	**description** *description* **Example:** `Device(config-wireless-policy)# description profile with aaa override`	Adds a user defined description to the new wireless policy.
Step 4	**aaa-override** **Example:**	Enables AAA override on the WLAN.

	Command or Action	Purpose
	`Device(config-wireless-policy)# aaa-override`	**Note** After AAA-override is enabled and ISE server starts sending policy, client policy defined in service-policy client will not take effect.

Configure Bi-Directional Rate Limiting for All Traffic

Use the police action in the policy-map to configure BDRL.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**policy-map** *policy-map* **Example:** `Device(config)# policy-map policy-sample 1`	Creates a named object representing a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class** *class-map-name* **Example:** `Device(config-pmap)# class class-default`	Associates a class map with the policy map, and enters policy-map class configuration mode.
Step 4	**police** *rate* **Example:** `Device(config-pmap-c)# police 500000`	Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.

Configure Bi-Directional Rate Limiting Based on Traffic Classification

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**policy-map** *policy-map* **Example:**	Creates a named object representing a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain

	Command or Action	Purpose
	`Device(config)# policy-map policy-sample2`	alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	**class** *class-map-name* **Example:** `Device(config-pmap)# class class-sample-youtube`	Associates a class map with the policy map, and enters policy-map class configuration mode.
Step 4	**police** *rate* **Example:** `Device(config-pmap-c)# police 1000000`	Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.
Step 5	**conform-action drop** **Example:** `Device(config-pmap-c-police)# conform-action drop`	Specifies the drop action to take on packets that conform to the rate limit.
Step 6	**exceed-action drop** **Example:** `Device(config-pmap-c-police)# exceed-action drop`	Specifies the drop action to take on packets that exceeds the rate limit.
Step 7	**exit** **Example:** `Device(config-pmap-c-police)# exit`	Exits the policy-map class configuration mode.
Step 8	**set dscp default** **Example:** `Device(config-pmap-c)# set dscp default`	Sets the DSCP value to default.
Step 9	**police** *rate* **Example:** `Device(config-pmap-c)# police 500000`	Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.
Step 10	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits the policy-map class configuration mode.
Step 11	**exit** **Example:** `Device(config-pmap)# exit`	Exits the policy-map configuration mode.
Step 12	**class-map match-any** *class-map-name* **Example:**	Selects a class map.

	Command or Action	Purpose
	`Device(config)# class-map match-any class-sample-youtube`	
Step 13	**match protocol** *protocol* **Example:** `Device(config-cmap)# match protocol youtube`	Configures the match criteria for a class map on the basis of the specified protocol.

Apply Bi-Directional Rate Limiting Policy Map to Policy Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-profile-name* **Example:** `Device(config)# wireless profile policy policy-profile3`	Configures WLAN policy profile and enters wireless policy configuration mode.
Step 3	**description** *description* **Example:** `Device(config-wireless-policy)# description policy-profile3`	Adds a user defined description to the new wireless policy.
Step 4	**service-policy client input** *input-policy* **Example:** `Device(config-wireless-policy)# service-policy client input platinum-up`	Sets the input client service policy as platinum.
Step 5	**service-policy client output** *output-policy* **Example:** `Device(config-wireless-policy)# service-policy client output platinum`	Sets the output client service policy as platinum.
Step 6	**service-policy input** *input-policy* **Example:** `Device(config-wireless-policy)# service-policy input platinum-up`	Sets the input service policy as platinum.
Step 7	**service-policy output** *output-policy* **Example:** `Device(config-wireless-policy)# service-policy output platinum`	Sets the output service policy as platinum.

Apply Metal Policy with Bi-Directional Rate Limiting

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-profile-name* **Example:** `Device(config)# wireless profile policy policy-profile3`	Configures WLAN policy profile and enters wireless policy configuration mode.
Step 3	**description** *description* **Example:** `Device(config-wireless-policy)# description policy-profile3`	Adds a user defined description to the new wireless policy.
Step 4	**service-policy client input** *input-policy* **Example:** `Device(config-wireless-policy)# service-policy client input platinum-up`	Sets the input client service policy as platinum.
Step 5	**service-policy client output** *output-policy* **Example:** `Device(config-wireless-policy)# service-policy client output platinum`	Sets the output client service policy as platinum.
Step 6	**service-policy input** *input-policy* **Example:** `Device(config-wireless-policy)# service-policy input platinum-up`	Sets the input service policy as platinum.
Step 7	**service-policy output** *output-policy* **Example:** `Device(config-wireless-policy)# service-policy output platinum`	Sets the output service policy as platinum.
Step 8	**exit** **Example:** `Device(config-wireless-policy)# exit`	Exits the policy configuration mode.
Step 9	**policy-map** *policy-map* **Example:** `Device(config)# policy-map policy-sample 1`	Creates a named object representing a set of policies that are to be applied to a set of traffic classes. Policy map names can contain alphabetic, hyphen, or underscore characters,

	Command or Action	Purpose
		are case sensitive, and can be up to 40 characters.
Step 10	**class** *class-map-name* **Example:** `Device(config-pmap)# class class-default`	Associates a class map with the policy map, and enters configuration mode for the specified system class.
Step 11	**police** *rate* **Example:** `Device(config-pmap-c)# police 500000`	Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.

How to apply Per Client Bi-Directional Rate Limiting

Information About Per Client Bi-Directional Rate Limiting

The Per Client Bi-Directional Rate Limiting feature adds bi-directional rate limiting for each wireless clients on 802.11ac Wave 2 APs in a Flex local switching configuration. Earlier, the Wave 2 APs supported only per-flow rate limiting for a wireless client. When wireless client starts multiple streams of traffic, the client-based rate limiting does not work as expected. This limitation is addressed by this feature.

For instance, if the controller is configured with QoS policy and you expect each client to have a rate limiting cap of 1000 kbps. Due to per-flow rate limiting on the AP, if the wireless client starts a Youtube stream and FTP stream, each of them will be rate limited at 1000 Kbps, therefore the client will be 2000 Kbps rates. This is not desirable.

Use Cases

The following are the use cases supported by the Per Client Bi-Directional Rate Limiting feature:

Use Case -1

Configuring only default class map

If policy map is configured only with default class map and mapped only to QoS client policy, AP does a per client rate limit to the client connected to AP.

Use Case-2

Changing from per client rate limit to per flow rate limit

If policy map is configured with another different class map along with a default class map and mapped to QoS client policy, AP performs per flow rate limit to client. As policy map has different class map along with the default class map. The per client rate limit values are cleared, if the AP has previously configured per client rate limit.

If the policy map has more than one class map, then additional class map is configured along with the default class map. So, the rate limit is applied from per client to per flow. The per client rate limit value is deleted from the rate info token bucket.

Use Case-3

Changing from per flow rate limit to per client limit

If different class map is removed from policy map and policy map has only one default class map, AP performs a per client rate limit to client.

The following covers the high-level steps for Per Client Bi-Directional Rate Limiting feature:

1. Configure a policy map to WLAN through policy profile.
2. Map the QoS related policy map to WLAN.
3. Configure policy map with the default class map.
4. Configure different police rate value for class Default map.

Note If policy map has class Default with valid police rate value, AP applies that rate limit to the overall client data traffic flow.

5. Apply the policy map with class Default to QoS client policy in WLAN policy profile.

Prerequisites for Per Client Bi-Directional Rate Limiting

- This feature is exclusive to QoS client policy, that is, the policy profile must have only QoS Policy or policy target as client.
- If policy map has class default with valid police rate value, AP applies that rate limit value to the overall client data traffic flow.

Restrictions on Per Client Bi-Directional Rate Limiting

- If policy map has class map other than the class Default map, the per client rate limit does not work in AP.

Configuring Per Client Bi-Directional Rate Limiting (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click the Policy Profile Name.

The **Edit Policy Profile** window is displayed.

Note The **Edit Policy Profile** window is displayed and configured in default class map only.

Step 3 Choose the **QOS And AVC** tab.

Step 4 In the **QoS Client Policy** settings, choose the policies from the **Egress** and **Ingress** drop-down lists.

Note You need to apply the default policy map to the QoS Client Policy.

Step 5 Click **Update & Apply to Device**.

Verifying Per Client Bi-Directional Rate Limiting

To verify whether per client is applied in AP, use the following command:

```
Device# show rate-limit client
Config:
            mac vap rt_rate_out rt_rate_in rt_burst_out rt_burst_in nrt_rate_out nrt_rate_in
 nrt_burst_out nrt_burst_in
A0:D3:7A:12:6C:5E   0           0          0             0            0             0
   0             0            0
Statistics:
            name     up down
        Unshaped      0    0
  Client RT pass 697610 8200
 Client NRT pass      0    0
 Client RT drops      0    0
Client NRT drops      0   16
               9    180    0
Per client rate limit:
             mac vap rate_out rate_in            policy
A0:D3:7A:12:6C:5E   0       88      23 per_client_rate_2
```

Configuring BDRL Using AAA Override

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-name* **Example:** `Device (config)# wireless profile policy default-policy-profile`	Configures the WLAN policy profile and enters wireless policy configuration mode.
Step 3	**aaa-override** **Example:** `Device(config-wireless-policy)# aaa`	Configures AAA override to apply policies coming from the AAA server or ISE the Cisco Identify Services Engine (ISE) server. The following attributes are available in the RADIUS server: • Airespace-Data-Bandwidth-Average-Contract: 8001 • Airespace-Real-Time-Bandwidth-Average-Contract: 8002

	Command or Action	Purpose
		• Airespace-Data-Bandwidth-Burst-Contract: 8003 • Airespace-Real-Time-Bandwidth-Burst-Contract: 8004 • Airespace-Data-Bandwidth-Average-Contract-Upstream: 8005 • Airespace-Real-Time-Bandwidth-Average-Contract-Upstream: 8006 • Airespace-Data-Bandwidth-Burst-Contract-Upstream: 8007 • Airespace-Real-Time-Bandwidth-Burst-Contract-Upstream: 8008 **Note** 8001, 8002, 8003, 8004, 8005, 8006, 8007, and 8008 are the desired rate-limit values configured as an example.

Verifying Bi-Directional Rate-Limit

To verify the bi-directional rate limit, use the following command:

```
Device# show wireless client mac-address E8-8E-00-00-00-71 detailClient MAC Address :
e88e.0000.0071
Client MAC Type     : Universally Administered Address
Client IPv4 Address : 100.0.7.94
Client Username     : e88e00000071
AP MAC Address      : 0a0b.0c00.0200
AP Name             : AP6B8B4567-0002
AP slot             : 0
Client State        : Associated
Policy Profile      : dnas_qos_profile_policy
Flex Profile        : N/A
Wireless LAN Id     : 10
WLAN Profile Name   : QoS_wlan
Wireless LAN Network Name (SSID): QoS_wlan
BSSID : 0a0b.0c00.0200
Connected For       : 28 seconds
Protocol            : 802.11n - 2.4 GHz
Channel             : 1
Client IIF-ID       : 0xa0000034
Association Id      : 10
Authentication Algorithm : Open System
Idle state timeout  : N/A
Session Timeout     : 1800 sec (Remaining time: 1777 sec)
Session Warning Time : Timer not running
Input Policy Name    : None
Input Policy State   : None
Input Policy Source  : None
Output Policy Name   : None
Output Policy State  : None
```

```
Output Policy Source : None
WMM Support          : Enabled
U-APSD Support       : Disabled
Fastlane Support     : Disabled
Client Active State  : In-Active
Power Save           : OFF
Supported Rates : 1.0,2.0,5.5,6.0,9.0,11.0,12.0,18.0,24.0,36.0,48.0,54.0
AAA QoS Rate Limit Parameters:
  QoS Average Data Rate Upstream             : 8005 (kbps)
  QoS Realtime Average Data Rate Upstream    : 8006 (kbps)
  QoS Burst Data Rate Upstream               : 8007 (kbps)
  QoS Realtime Burst Data Rate Upstream      : 8008 (kbps)
  QoS Average Data Rate Downstream           : 8001 (kbps)
  QoS Realtime Average Data Rate Downstream  : 8002 (kbps)
  QoS Burst Data Rate Downstream             : 80300 (kbps)
  QoS Realtime Burst Data Rate Downstream    : 8004 (kbps)
```

To verify the rate-limit details from the AP terminal, use the following command

```
Device# show rate-limit client
Config:
mac vap rt_rate_out rt_rate_in rt_burst_out rt_burst_in nrt_rate_out nrt_rate_in nrt_burst_out
 nrt_burst_in
00:1C:F1:09:85:E7 0 8001 8002 8003 8004 8005 8006 8007 8008
Statistics:
name up down
Unshaped 0 0
Client RT pass 0 0
Client NRT pass 0 0
Client RT drops 0 0
Client NRT drops 0 0
Per client rate limit:
mac vap rate_out rate_in policy
```

How to Configure Wireless QoS

Configuring a Policy Map with Class Map (GUI)

Procedure

Step 1 Choose **Configuration** > **Services** > **QoS**.

Step 2 Click **Add** to view the **Add QoS** window.

Step 3 In the text box next to the **Policy Name**, enter the name of the new policy map that is being added.

Step 4 Click **Add Class-Maps**.

Step 5 Configure **AVC** based policies or **User Defined** policies. To enable **AVC** based policies, and configure the following:

a) Choose either **Match Any** or **Match All**.

b) Choose the required **Mark Type**. If you choose **DSCP** or **User Priority**, you must specify the appropriate **Mark Value**.

c) Check the **Drop** check box to drop traffic from specific sources.

Note When **Drop** is enabled, the **Mark Type** and **Police(kbps)** options are disabled.

d) Based on the chosen **Match Type**, select the required protocols from the **Available Protocol(s)** list and move them to the **Selected Protocol(s)** list. These selected protocols are the ones from which traffic is dropped.
e) Click **Save**.

Note To add more Class Maps, repeat steps 4 and 5.

Step 6 To enable **User-Defined** QoS policy, and the configure the following:

a) Choose either **Match Any** or **Match All**.
b) Choose either **ACL** or **DSCP** as the **Match Type** from the drop-down list, and then specify the appropriate **Match Value**.
c) Choose the required **Mark Type** to associate with the mark label. If you choose *DSCP*, you must specify an appropriate **Mark Value**.
d) Check the **Drop** check box to drop traffic from specific sources.

Note When **Drop** is enabled, the **Mark Type** and **Police(kbps)** options are disabled.

e) Click **Save**.

Note To define actions for all the remaining traffic, in the Class Default, choose **Mark** and/or **Police(kbps)** accordingly.

Step 7 Click **Save & Apply to Device**.

Configuring a Class Map (CLI)

Follow the procedure given below to configure class maps for voice and video traffic:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**class-map** *class-map-name* **Example:** `Device(config)# class-map test`	Creates a class map.
Step 3	**match dscp** *dscp-value* **Example:** `Device(config-cmap)# match dscp 46`	Matches the DSCP value in the IPv4 and IPv6 packets. **Note** By default for the class map the value is match-all.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config-cmap)# end`	Exits the class map configuration and returns to the privileged EXEC mode.
Step 5	**show class-map** *class-map-name* **Example:** `Device# show class-map class_map_name`	Verifies the class map details.

Configuring Policy Profile to Apply QoS Policy (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 On the **Policy Profile** page, click the name of the policy profile.

Step 3 In the **Edit Policy Profile** window, click the **QoS and AVC** tab.

Step 4 Under **QoS SSID Policy**, choose the appropriate **Ingress** and **Egress** policies for WLANs.

Note The ingress policies can be differentiated from the egress policies by the suffix *-up*. For example, the Platinum ingress policy is named *platinum-up*.

Step 5 Under **QoS Client Policy**, choose the appropriate **Ingress** and **Egress** policies for clients.

Step 6 Click **Update & Apply to Device**.

Note Only custom policies are displayed under **QoS Client Policy**. AutoQoS policies are auto generated and not displayed for user selection.

Configuring Policy Profile to Apply QoS Policy (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-policy* **Example:** `Device(config)# wireless profile policy qostest`	Configures WLAN policy profile and enters the wireless policy configuration mode.

	Command or Action	Purpose
Step 3	**service-policy client** {**input** | **output**} *policy-name* **Example:** `Device(config-wireless-policy)# service-policy client input policy-map-client`	Applies the policy. The following options are available. • **input**—Assigns the client policy for ingress direction on the policy profile. • **output**—Assigns the client policy for egress direction on the policy profile.
Step 4	**service-policy** {**input** | **output**} *policy-name* **Example:** `Device(config-wireless-policy)# service-policy input policy-map-ssid`	Applies the policy to the BSSID. The following options are available. • **input**—Assigns the policy-map to all clients in WLAN. • **output**—Assigns the policy-map to all clients in WLAN.
Step 5	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Enables the wireless policy profile.

Applying Policy Profile to Policy Tag (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Tags**.

Step 2 On the **Manage Tags** page in the **Policy** tab, click **Add**.

Step 3 In the **Add Policy Tag** window that is displayed, enter a name and description for the policy tag.

Step 4 Map the required WLAN IDs and WLAN profiles with appropriate policy profiles.

Step 5 Click **Update & Apply to Device**.

Applying Policy Profile to Policy Tag (CLI)

Procedure

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

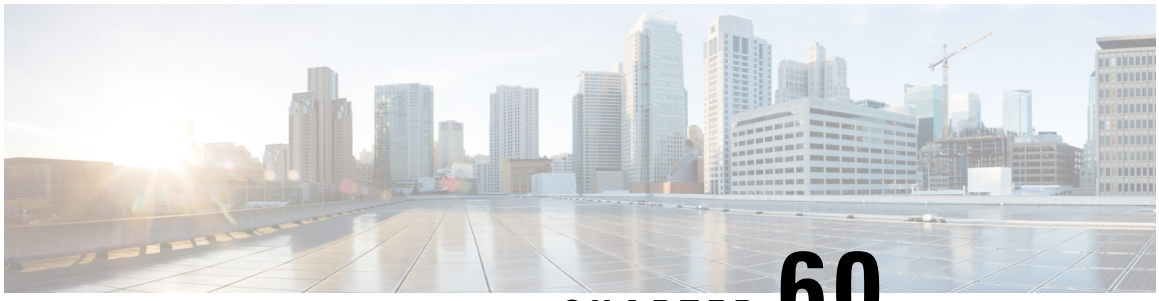
	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**wireless tag policy** *policy-tag-name* **Example:** `Device(config-policy-tag)# wireless tag policy qostag`	Configures policy tag and enters the policy tag configuration mode.
Step 3	**wlan** *wlan-name* **policy** *profile-policy-name* **Example:** `Device(config-policy-tag)# wlan test policy qostest`	Maps a policy profile to a WLAN profile.
Step 4	**end** **Example:** `Device(config-policy-tag)# end`	Saves the configuration and exits the configuration mode and returns to privileged EXEC mode.
Step 5	**show wireless tag policy summary** **Example:** `Device# show wireless tag policy summary`	Displays the configured policy tags. **Note** To view the detailed information of a policy tag, use the **show wireless tag policy detailed** *policy-tag-name* command.

Attaching Policy Tag to an AP

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap** *mac-address* **Example:** `Device(config)# ap F866.F267.7DFB`	Configures Cisco APs and enters the ap profile configuration mode.
Step 3	**policy-tag** *policy-tag-name* **Example:** `Device(config-ap-tag)# policy-tag qostag`	Maps a Policy tag to the AP.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config-ap-tag)# end`	Saves the configuration and exits the configuration mode and returns to privileged EXEC mode.
Step 5	**show ap tag summary** **Example:** `Device# show ap tag summary`	Displays the ap details and tags associated to it.



CHAPTER 60

Wireless Auto-QoS

Information About Auto QoS

Wireless Auto QoS automates deployment of wireless QoS features. It has a set of predefined profiles which can be further modified by the customer to prioritize different traffic flows. Auto-QoS matches traffic and assigns each matched packet to qos-groups. This allows the output policy map to put specific qos-groups into specific queues, including into the priority queue.

AutoQoS Policy Configuration

Table 32: AutoQoS Policy Configuration

Mode	Client Ingress	Client Egress	BSSID Ingress	BSSID Egress	Port Ingress	Port Egress	Radio
Voice	N/A	N/A	P3	P4	N/A	P7	ACM on
Guest	N/A	N/A	P5	P6	N/A	P7	
Fastlane	N/A	N/A	N/A	N/A	N/A	P7	edca-parameters fastlane
Enterprise-avc	N/A	N/A	P1	P2	N/A	P7	

P1	AutoQos-4.0-wlan-ET-SSID-Input-AVC-Policy
P2	AutoQos-4.0-wlan-ET-SSID-Output-Policy
P3	platinum-up
P4	platinum
P5	AutoQos-4.0-wlan-GT-SSID-Input-Policy

P6	AutoQos-4.0-wlan-GT-SSID-Output-Policy
P7	AutoQos-4.0-wlan-Port-Output-Policy

How to Configure Wireless AutoQoS

Configuring Wireless AutoQoS on Profile Policy

You can enable AutoQoS on a profile policy.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**wireless autoqos policy-profile** *policy-name* **mode { enterprise-avc | fastlane | guest | voice}** **Example:** `Device# wireless autoqos policy-profile test-profile mode voice`	Configures AutoQoS wireless policy. • **enterprise-avc**—Enables AutoQos Wireless Enterprise AVC Policy. • **fastlane**—Enable AutoQos Wireless Fastlane Policy. • **guest**—Enable AutoQos Wireless Guest Policy. • **voice**—Enable AutoQos Wireless Voice Policy. **Note** AutoQoS MIB attribute does not support full functionality with service policy. Service policy must be configured manually. Currently, there is only support for AutoQoS mode.

What to do next

Note After enabling AutoQoS, we recommend that you wait for a few seconds for the policy to install and then try and modify the AutoQoS policy maps if required; or retry if the modification is rejected.

Disabling Wireless AutoQoS

To globally disable Wireless AutoQoS:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**shutdown** **Example:** `Device# shutdown`	Shuts down the policy profile.
Step 3	**wireless autoqos disable** **Example:** `Device# wireless autoqos disable`	Globally disables wireless AutoQoS.
Step 4	**[no] shutdown** **Example:** `Device# no shutdown`	Enables the wireless policy profile. **Note** Disabling Auto QoS does not reset global radio configurations like CAC and EDCA parameters.

Rollback AutoQoS Configuration (GUI)

Procedure

Step 1 Choose **Configuration** > **Services** > **QoS**.
Step 2 Click **Disable AutoQoS**.
Step 3 Click **Yes** to confirm.

Rollback AutoQoS Configuration

Before you begin

Note AutoQoS MIB attribute does not support the full functionality with service policy. Currently, there is only support for AutoQoS mode. Service policy must be configured manually.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device`**`enable`**	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**clear platform software autoqos config template { enterprise_avc | guest}** **Example:** `Device# clear platform software autoqos config template guest`	Resets AutoQoS configuration. • enterprise-avc—Resets AutoQoS Enterprise AVC Policy Template. • guest—Resets AutoQoS Guest Policy Template.

Clearing Wireless AutoQoS Policy Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.
Step 2 Click on the **Policy Profile Name**.
Step 3 Go to **QOS and AVC** tab.
Step 4 From the **Auto Qos** drop-down list, choose **None**.
Step 5 Click **Update & Apply to Device**.

Clearing Wireless AutoQoS Policy Profile

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**shutdown** **Example:** `Device# shutdown`	Shuts down the policy profile.
Step 3	**wireless autoqos policy-profile** *policy-name* **mode clear** **Example:**	Clears the configured AutoQoS wireless policy.

	Command or Action	Purpose
	`Device# wireless autoqos policy-profile test-profile mode clear`	
Step 4	**[no] shutdown** **Example:** `no shutdown`	Enables the wireless policy profile.

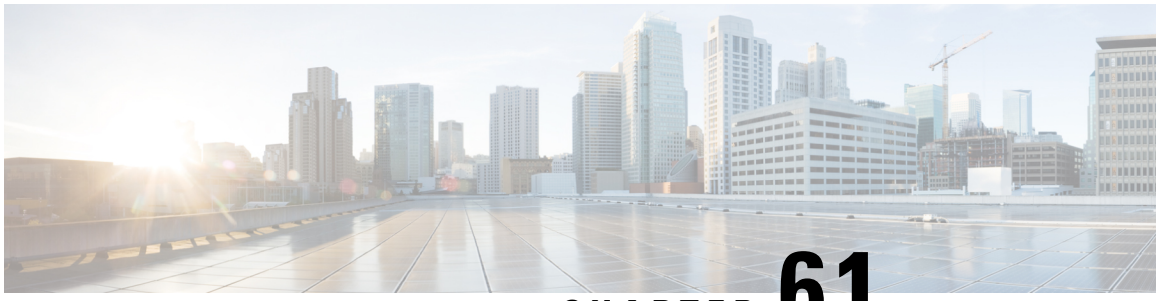
Viewing AutoQoS on policy profile

Before you begin

Autoqos is supported on the local mode and flex mode. Autoqos configures a set of policies and radio configurations depending on the template. It is possible to override the service-policy that is configured by autoqos. The latest configuration takes effect, with AAA override policy being of highest priority.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Deviceenable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**show wireless profile policy detailed** *policy-profile-name* **Example:** `Device# show wireless profile policy detailed testqos`	Shows policy-profile detailed parameters.



CHAPTER 61

Native Profiling

Information About Native Profiling

You can profile devices based on HTTP and DHCP to identify the end devices on the network. You can configure device-based policies and enforce these policies per user or per device policy on the network.

Policies allow profiling of mobile devices and basic onboarding of the profiled devices to a specific VLAN. They also assign ACL and QoS or configure session timeouts.

You can configure policies as two separate components:

- Defining policy attributes as service templates that are specific to clients joining the network and applying policy match criteria
- Applying match criteria to the policy.

Note Before proceeding with the native profile configuration, ensure that HTTP Profiling and DHCP Profiling are enabled.

To configure Native Profiling, use one of the following procedures:

- Create a service template
- Create a class map

Note You can apply a service template using either a class map or parameter map.

- Create a parameter-map and associate the service template to parameter-map
  - Create a policy map
    1. If class-map has to be used: Associate the class-map to the policy-map and associate the service-template to the class-map.
    2. If parameter-map has to be used: Associate the parameter-map to the policy-map
  - Associate the policy-map to the policy profile.

Creating a Class Map (GUI)

Procedure

Step 1 Click **Configuration** > **Services** > **QoS**.

Step 2 In the **Qos – Policy** area, click **Add** to create a new QoS Policy or click the one you want to edit.

Step 3 Add **Add Class Map** and enter the details.

Step 4 Click **Save**.

Step 5 Click **Update and Apply to Device**.

Creating a Class Map (CLI)

Note Configuration of class maps via CLI offer more options and can be more granular than GUI.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**class-map type control subscriber match-any** *class-map-name* **Example:**	Specifies the class map type and name.

	Command or Action	Purpose
	`Device(config)# class-map type control subscriber match-any cls_user`	
Step 3	**match username** *username* **Example:** `Device(config-filter-control-classmap)# match username ciscoise`	Specifies the class map attribute filter criteria.
Step 4	**class-map type control subscriber match-any** *class-map-name* **Example:** `Device(config)# class-map type control subscriber match-any cls_userrole`	Specifies the class map type and name.
Step 5	**match user-role** *user-role* **Example:** `Device(config-filter-control-classmap)# match user-role engineer`	Specifies the class map attribute filter criteria.
Step 6	**class-map type control subscriber match-any** *class-map-name* **Example:** `Device(config)# class-map type control subscriber match-any cls_oui`	Specifies the class map type and name.
Step 7	**match oui** *oui-address* **Example:** `Device(config-filter-control-classmap)# match oui 48.f8.b3`	Specifies the class map attribute filter criteria.
Step 8	**class-map type control subscriber match-any** *class-map-name* **Example:** `Device(config)# class-map type control subscriber match-any cls_mac`	Specifies the class map type and name.
Step 9	**match mac-address** *mac-address* **Example:** `Device(config-filter-control-classmap)# match mac-address 0040.96b9.4a0d`	Specifies the class map attribute filter criteria.
Step 10	**class-map type control subscriber match-any** *class-map-name* **Example:** `Device(config)# class-map type control subscriber match-any cls_devtype`	Specifies the class map type and name.

	Command or Action	Purpose
Step 11	**match device-type** *device-type* **Example:** `Device(config-filter-control-classmap)# match device-type windows`	Specifies the class map attribute filter criteria.
Step 12	**match join-time-of-day** *start-time end-time* **Example:** `Device(config-filter-control-classmap)# match join-time-of-day 10:30 12:30`	Specifies a match to the time of day. Here, join time is considered for matching. For example, if the match filter is set from 11:00 am to 2:00 pm, a device joining at 10:59 am is not considered, even if it acquires credentials after 11:00 am. Here, *start-time* and *end-time* specifies the 24-hour format. Use the **show class-map type control subscriber name** *name* command to verify the configuration. **Note** You should also disable AAA override for this command to work.

Creating a Service Template (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Local Policy**.

Step 2 On the **Local Policy** page, **Service Template** tab, click **ADD**.

Step 3 In the **Create Service Template** window, enter the following parameters:

- **Service Template Name:** Enter a name for the template.
- **VLAN ID:** Enter the VLAN ID for the template. Valid range is between 1 and 4094.
- **Session Timeout (secs):** Sets the timeout duration for the template. Valid range is between 1 and 65535.
- **Access Control List:** Choose the Access Control List from the drop-down list.
- **Ingress QOS:** Choose the input QoS policy for the client from the drop-down list
- **Egress QOS:** Choose the output QoS policy for the client from the drop-down list.

Step 4 Click **Save & Apply to Device**.

Creating a Service Template (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**service-template** *service-template-name* **Example:** `Device(config)# service-template svc1`	Enters service template configuration mode.
Step 3	**access-group** *access-list-name* **Example:** `Device(config-service-template)# access-group acl-auto`	Specifies the access list to be applied.
Step 4	**vlan** *vlan-id* **Example:** `Device(config-service-template)# vlan 10`	Specifies VLAN ID. Valid range is from 1-4094.
Step 5	**absolute-timer** *timer* **Example:** `Device(config-service-template)# absolute-timer 1000`	Specifies session timeout value for a service template. Valid range is from 1-65535.
Step 6	**service-policy qos input** *qos-policy* **Example:** `Device(config-service-template)# service-policy qos input in_qos`	Configures an input QoS policy for the client.
Step 7	**service-policy qos output** *qos-policy* **Example:** `Device(config-service-template)# service-policy qos output out_qos`	Configures an output QoS policy for the client.

Creating a Parameter Map

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**parameter-map type subscriber attribute-to-service** *parameter-map-name* **Example:** `Device(config)# parameter-map type subscriber attribute-to-service param`	Specifies the parameter map type and name.
Step 3	*map-index***map device-type eq***filter-name* **Example:** `Device(config-parameter-map-filter)# 1 map device-type eq "windows" mac-address eq 3c77.e602.2f91 username eq "cisco"`	Specifies the parameter map attribute filter criteria. Multiple filters are used in the example provided here.
Step 4	*map-index***service-template***service-template-name* **precedence** *precedence-num* **Example:** `Device(config-parameter-map-filter-submode)# 1 service-template svc1 precedence 150`	Specifies the service template and its precedence.

Creating a Policy Map (GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Local Policy** > **Policy Map** tab..

Step 2 Enter a name for the Policy Map in the **Policy Map Name** text field.

Step 3 Click **Add**

Step 4 Choose the service template from the **Service Template** drop-down list.

Step 5 For the following parameters select the type of filter from the drop-down list and enter the required match criteria

- Device Type
- User Role
- User Name

- OUI
- MAC Address

Step 6 Click **Add Criteria**

Step 7 Click **Update & Apply to Device**.

Creating a Policy Map (CLI)

Before you begin

Before removing a policy map or parameter map, you should remove it from the target or shut down the WLAN profile or delete the session.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**policy-map type control subscriber** *policy-map-name* **Example:** `Device(config)# policy-map type control subscriber polmap5`	Specifies the policy map type.
Step 3	**event identity-update match-all** **Example:** `Device(config-event-control-policymap)# event identity-update match-all`	Specifies the match criteria to the policy map.
Step 4	You can apply a service template using either a class map or a parameter map, as shown here. • *class-num* **class** *class-map-name* **do-until-failure** • *action-index* **activate service-template** *service-template-name* • *action-index* **map attribute-to-service table** *parameter-map-name* **Example:** The following example shows how a class-map with a service-template has to be applied: `Device(config-class-control-policymap)# 10 class cls_mac do-until-failure`	Configures the local profiling policy class map number and specifies how to perform the action or activates the service template or maps an identity-update attribute to an auto-configured template.

	Command or Action	Purpose

	`Device(config-action-control-policymap)# 10 activate service-template svc1`
	Example:
	The following example shows how a parameter map has to be applied (service template is already associated with the parameter map 'param' while creating it):
	`Device(config-action-control-policymap)#1 map attribute-to-service table param`	

Step 5	**end** **Example:** `Device(config-action-control-policymap)# end`	Exits configuration mode.
Step 6	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 7	**wireless profile policy** *wlan-policy-profile-name* **Example:** `Device(config)# wireless profile policy wlan-policy-profilename`	Configures a wireless policy profile. **Caution** Do not configure aaa-override for native profiling under a named wireless profile policy. Native profiling is applied at a lower priority than AAA policy. If aaa-override is enabled, the AAA policies will override native profile policy.
Step 8	**description** *profile-policy-description* **Example:** `Device(config-wireless-policy)# description "default policy profile"`	Adds a description for the policy profile.
Step 9	**dhcp-tlv-caching** **Example:** `Device(config-wireless-policy)# dhcp-tlv-caching`	Configures DHCP TLV caching on a WLAN.
Step 10	**http-tlv-caching** **Example:** `Device(config-wireless-policy)# http-tlv-caching`	Configures client HTTP TLV caching on a WLAN.
Step 11	**subscriber-policy-name** *policy-name* **Example:**	Configures the subscriber policy name.

	Command or Action	Purpose
	`Device(config-wireless-policy)# subscriber-policy-name polmap5`	
Step 12	**vlan** *vlan-id* **Example:** `Device(config-wireless-policy)# vlan 1`	Configures a VLAN name or VLAN ID.
Step 13	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Saves the configuration.

Configuring Native Profiling in Local Mode

To configure native profiling in the local mode, you must follow the steps described in Creating a Policy Map (CLI), on page 673. In the policy profile, you must enable central switching as described in the step given below in order to configure native profiling.

Procedure

	Command or Action	Purpose
Step 1	**central switching** **Example:** `Device(config-wireless-policy)# central switching`	Enables central switching.

Verifying Native Profile Configuration

Use the following **show** commands to verify the native profile configuartion:

```
Device# show wireless client device summary

Active classified device summary
MAC Address       Device-type                     User-role
Protocol-map
-----------------------------------------------------------------------------------------------
1491.82b8.f94b    Microsoft-Workstation           sales
          9
1491.82bc.2fd5    Windows7-Workstation            sales
          41


Device# show wireless client device cache

Cached classified device info

MAC Address       Device-type                     User-role
Protocol-map
-----------------------------------------------------------------------------------------------
```

```
2477.031b.aa18    Microsoft-Workstation
           9
30a8.db3b.a753    Un-Classified Device
           9
4400.1011.e8b5    Un-Classified Device
           9
980c.a569.7dd0    Un-Classified Device
```

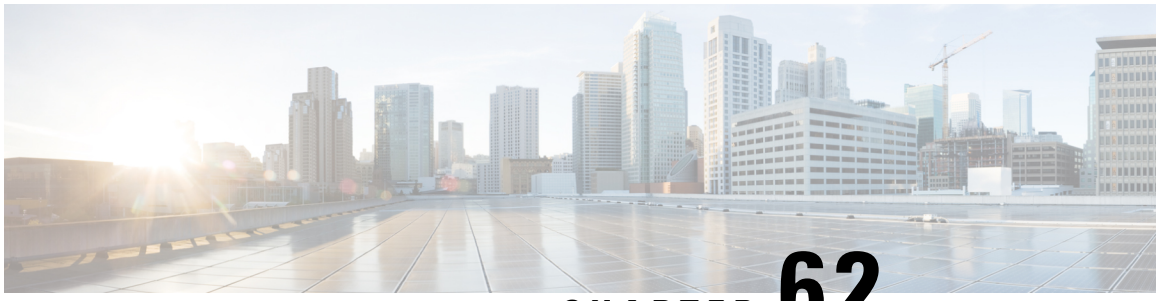
```
Device# show wireless client mac-address 4c34.8845.e32c detail | s
Session Manager:
  Interface :
  IIF ID            : 0x90000002
  Device Type       : Microsoft-Workstation
  Protocol Map      : 0x000009
  Authorized        : TRUE
  Session timeout   : 1800
  Common Session ID: 78380209000000174BF2B5B9
  Acct Session ID   : 0
  Auth Method Status List
   Method : MAB
    SM State         : TERMINATE
    Authen Status    : Success
  Local Polices:
   Service Template : wlan_svc_C414.3CCA.0A51 (priority 254)
    Absolute-Timer   : 1800
  Server Polices:
  Resultant Policies:
  Filter-ID         : acl-auto
  Input QOS         : in_qos
  Output QOS        : out_qos
  Idle timeout      : 60 sec
  VLAN               : 10
  Absolute-Timer    : 1000
```

Use the following **show** command to verify the class map details for a class map name:

```
Device# show class-map type control subscriber name test
Class-map               Action                        Exec  Hit  Miss  Comp
---------                ------                       ----  ---  ----   ---
match-any test     match day Monday                     0     0    0     0
match-any test     match join-time-of-day 8:00 18:00    0     0    0     0
Key:
  "Exec" - The number of times this line was executed
  "Hit"  - The number of times this line evaluated to TRUE
  "Miss" - The number of times this line evaluated to FALSE
  "Comp" - The number of times this line completed the execution of its
       condition without a need to continue on to the end
```

PART IX

IPv6



CHAPTER 62

IPv6 Client Address Learning

Information About IPv6 Client Address Learning

Client Address Learning is configured on embedded wireless controller to learn the IPv4 and IPv6 address of wireless client, and the client's transition state maintained by the embedded wireless controller on association and timeout.

There are three ways for an IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLAAC)
- Stateful DHCPv6
- Static Configuration

In all of these methods, the IPv6 client always sends a neighbor solicitation Duplicate Address Detection (DAD) request to ensure that there is no duplicate IP address on the network. The embedded wireless controller snoops on the Neighbor Discovery Protocol (NDP) and DHCPv6 packets of the client to learn about its client IP addresses.

Address Assignment Using SLAAC

The most common method for IPv6 client address assignment is SLAAC, which provides simple plug-and-play connectivity, where clients self-assign an address based on the IPv6 prefix.

SLAAC is configured as follows:

- A host sends a Router Solicitation message.
- The host waits for a Router Advertisement message.
- The host take the first 64 bits of the IPv6 prefix from the Router Advertisement message and combines it with the 64 bit EUI-64 address (in the case of Ethernet, this is created from the MAC address) to create a global unicast message. The host also uses the source IP address, in the IP header, of the Router Advertisement message, as its default gateway.

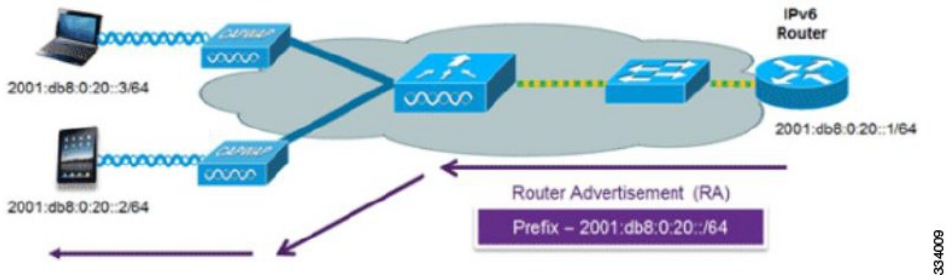
- Duplicate Address Detection is performed by the IPv6 clients to ensure that random addresses that are picked do not collide with other clients.

Note The choice of algorithm is up to the client and is often configurable.

The last 64 bits of the IPv6 address can be learned by using one of the following algorithms:

- EUI-64, which is based on the MAC address of the interface
- Private addresses that are randomly generated

Figure 21: Address Assignment Using SLAAC



The following Cisco IOS configuration commands from a Cisco-capable IPv6 router are used to enable SLAAC addressing and router advertisements:

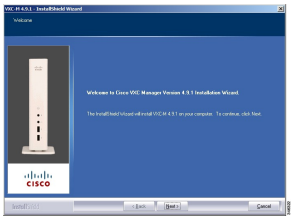
```
ipv6 unicast-routing
interface Vlan20
description IPv6-SLAAC
ip address 192.168.20.1 255.255.255.0
ipv6 address FE80:DB8:0:20::1 linklocal
ipv6 address 2001:DB8:0:20::1/64
ipv6 enable
end
```

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, that is, 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, becuase this is already provided by SLAAC. This information includes the DNS domain name, DNS servers, and other DHCP vendor-specific options.

Figure 22: Stateful DHCPv6 Address Assignment



The following 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
```

Static IP Address Assignment

Statically configured address on a client.

Router Solicitation

A Router Solicitation message is issued by a host controller to facilitate local routers to transmit a Router Advertisement from which the controller can obtain information about local routing, or perform stateless auto configuration. Router Advertisements are transmitted periodically and the host prompts with an immediate Router Advertisement using a Router Solicitation such as - when it boots or following a restart operation.

Router Advertisement

A Router Advertisement message is issued periodically by a router or in response to a Router Solicitation message from a host. The information contained in these messages is used by a host to perform stateless auto configuration and to modify its routing table.

Neighbor Discovery

IPv6 Neighbor Discovery is a set of messages and processes that determine relationships between neighboring nodes. Neighbor Discovery replaces the Address Resolution Protocol (ARP), Internet Control Message Protocol (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 tracks each IPv6 address and its associated MAC address. Clients are removed from the table according to neighbor-binding timers.

Neighbor Discovery Suppression

The IPv6 addresses of wireless clients are cached by a device. When the device receives an NS multicast looking for an IPv6 address, and if the target address is known to the device and belongs to one of its clients, the device will reply with an NA message on behalf of the client. At the end of this process, the equivalent of the ARP table of IPv4 is generated, but is more efficient because it uses fewer messages.

Note The device acts as a proxy and responds with NA, only when the **ipv6 nd suppress** command is configured.

If the device does not have the IPv6 address of a wireless client, the device does not respond with NA; instead, it forwards the NS packet to the wireless side. To resolve this, an NS Multicast Forwarding knob is provided. If this knob is enabled, the device 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.

Note that this cache miss scenario occurs rarely, and only very few clients who do not implement complete IPv6 stack may not advertise their IPv6 address during NDP.

Router Advertisement Guard

- Port on which the frame is received
- IPv6 source address
- Prefix list
- Trusted or Untrusted ports for receiving the router advertisement guard messages
- Trusted/Untrusted IPv6 source addresses of the router advertisement sender
- Trusted/Untrusted Prefix list and Prefix ranges
- Router preference

Router Advertisement Throttling

RA throttling allows the controller to enforce limits to the RA packets headed toward the wireless network. By enabling RA throttling, routers that send multiple 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 unicast to the client. This process ensures that the new clients or roaming clients are not affected by the RA throttling.

Prerequisites for IPv6 Client Address Learning

Before configuring IPv6 client address learning, configure the embedded wireless controller clients to support IPv6.

Configuring IPv6 on Embedded Wireless Controller Interface

Follow the procedure given below to configure IPv6 on an interface:

Before you begin

Enable IPv6 on the client and IPv6 support on the wired infrastructure.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface GigabitEthernet0** **Example:** `Device(config)# interface GigabitEthernet0`	Creates the GigabitEthernet interface and enters interface configuration mode.

Step 4	**ip address fe80::1 link-local**
	Example:
	`Device(config-if)# ip address 198.51.100.1 255.255.255.0`
	`Device(config-if)# ipv6 address fe80::1 link-local`
	`Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64`
	`Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64`
		Configures IPv6 address on the GigabitEthernet interface using the link-local option.

Step 5	**ipv6 enable** **Example:** `Device(config)# ipv6 enable`	(Optional) Enables IPv6 on the GigabitEthernet interface.
Step 6	**end** **Example:** `Device(config)# end`	Exits interface mode.

Native IPv6

Information About IPv6

IPv6 is a packet-based protocol used to exchange data, voice, and video traffic over digital networks. IPv6 is based on IP, but with a much larger address space, and improvements such as a simplified main header and extension headers. The architecture of IPv6 has been designed to allow existing IPv4 users to transition easily to IPv6 while continuing to use services such as end-to-end security, quality of service (QoS), and globally unique addresses. The larger IPv6 address space allows networks to scale and provide global reachability.

Note The features and functions that work on IPv4 networks with IPv4 addresses also work on IPv6 networks with IPv6 addresses.

General Guidelines

- You must configure the **ipv6 unicast-routing** command on the embedded wireless controller for the IPv6 feature to work.
- The Wireless Management interface should have only one static IPv6 address.
- Router advertisement should be suppressed on the wireless management interface and client VLANs (if IPv6 is configured on the client VLAN).
- Preferred mode is part of an AP join profile. When you configure the preferred mode as IPv6, an AP attempts to join over IPv6 first. If it fails, the AP falls back to IPv4.
- You should use MAC addresses for RA tracing of APs and clients.

Unsupported Features

- UDP Lite is not supported.
- AP sniffer over IPv6 is not supported.
- IPv6 is not supported for the HA port interface.
- Auto RF grouping over IPv6 is not supported. Only static RF grouping is supported.

Configuring IPv6 Addressing

Follow the procedure given below to configure IPv6 addressing:

Note All the features and functions that work on IPv4 networks with IPv4 addresses will work on IPv6 networks with IPv6 addresses too.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ipv6 unicast-routing** **Example:** `Device(config)# ipv6 unicast-routing`	Configures IPv6 for unicasting.

	Command or Action	Purpose
Step 3	**interface GigabitEthernet0** **Example:** `Device(config)# interface GigabitEthernet0`	Creates the GigabitEthernet interface and enters interface configuration mode.
Step 4	**ipv6 address** *ipv6-address* **Example:** `Device(config-if)# ipv6 address FD09:9:2:49::53/64`	Specifies a global IPv6 address.
Step 5	**ipv6 enable** **Example:** `Device(config-if)# ipv6 enable`	Enables IPv6 on the interface.
Step 6	**ipv6 nd ra suppress all** **Example:** `Device(config-if)# ipv6 nd ra suppress all`	Suppresses IPv6 router advertisement transmissions on the interface.
Step 7	**exit** **Example:** `Device(config-if)# exit`	Returns to global configuration mode.
Step 8	**wireless management interface gigabitEthernet** *gigabitEthernet-interface-vlan 64* **Example:** `Device(config)# wireless management interface gigabitEthernet vlan 64`	Configures the ports that are connected to the supported APs with the wireless management interface.
Step 9	**ipv6 route** *ipv6-address* **Example:** `Device(config)# ipv6 route ::/0 FD09:9:2:49::1`	Specifies IPv6 static routes.

Creating an AP Join Profile (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** window, click the **General** tab and click **Add**.

Step 3 In the **Name** field enter, a name for the AP join profile.

Step 4 (Optional) Enter a description for the AP join profile.

Step 5 Choose **CAPWAP** > **Advanced**.

Step 6 Under the **Advanced** tab, from the **Preferred Mode** drop-down list, choose **IPv6**. This sets the preferred mode of APs as IPv6.

Step 7 Click **Save & Apply to Device**.

Creating an AP Join Profile (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *ap-profile* **Example:** `Device(config)# ap profile xyz-ap-profile`	Configures an AP profile and enters AP profile configuration mode.
Step 3	**description** *ap-profile-name* **Example:** `Device(config-ap-profile)# description "xyz ap profile"`	Adds a description for the AP profile.
Step 4	**preferred-mode ipv6** **Example:** `Device(config-ap-profile)# preferred-mode ipv6`	Sets the preferred mode of APs as IPv6.

Configuring the Primary and Backup Embedded Wireless Controller (GUI)

Before you begin

Ensure that you have configured an AP join profile prior to configuring the primary and backup embedded wireless controllers.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** window, click the AP join profile name.

Step 3 In the **Edit AP Join Profile** window, click the **CAPWAP** tab.

Step 4 In the **High Availability** tab, under **Backup Controller Configuration**, check the **Enable Fallback** check box.

Step 5 Enter the primary and secondary controller names and IP addresses.

Step 6 Click **Update & Apply to Device**.

Configuring Primary and Backup Controller (CLI)

Follow the procedure given below to configure the primary and secondary controllers for a selected AP:

Procedure

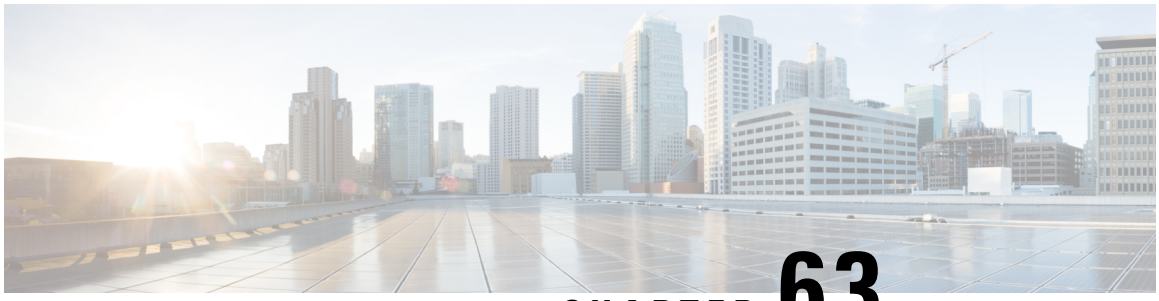
	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap profile** *profile-name* **Example:** `Device(config)# ap profile yy-ap-profile`	Configures an AP profile and enters AP profile configuration mode.
Step 3	**capwap backup primary** *primary-controller-name primary-controller-ip* **Example:** `Device(config)# capwap backup primary WLAN-Controller-A 2001:DB8:1::1`	Configures AP CAPWAP parameters with the primary backup controller's name. **Note** You need to enable fast heartbeat for **capwap backup primary** and **capwap backup secondary** to work. AP disconnection may occur if the link between the controller and AP is not reliable and fast heartbeat is enabled.
Step 4	**ap capwap backup secondary** *secondary-controller-name secondary-controller-ip* **Example:** `Device(config)# capwap backup secondary WLAN-Controller-B 2001:DB8:1::1`	Configures AP CAPWAP parameters with the secondary backup controller's name.
Step 5	**syslog host** *ipaddress* **Example:** `Device(config)# syslog host 2001:DB8:1::1`	Configures the system logging settings for the APs.
Step 6	**tftp-downgrade** *tftp-server-ip imagename* **Example:**	Initiates AP image downgrade from a TFTP server for all the APs.

	Command or Action	Purpose
	`Device(config)# tftp-downgrade 2001:DB8:1::1 testimage`	

Verifying IPv6 Configuration

Use the following **show** command to verify the IPv6 configuration:

```
Device# show wireless interface summary

Interface Name   Interface Type VLAN ID IP Address   IP Netmask     NAT-IP Address   MAC
Address
-----------------------------------------------------------------------------------------------
GigabitEthernet0 Management     0       0.0.0.0     255.255.255.0  0.0.0.0
d4c9.3ce6.b854
                                      fd09:9:2:49::54/64
```



CHAPTER 63

IPv6 ACL

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 embedded wireless controller). ACLs are configured on the devicend 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 embedded wireless 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

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 Cisco Secure Access Control Server (Cisco Secure ACS).

The ACE is not configured on the Controller Embedded Wireless Controller. The ACE is sent to the device in the `ACCESS-Accept` attribute and applies it directly for the client. When a wireless client roams into an foreign device, the ACEs are sent to the foreign device as an AAA attribute in the mobility Handoff message. Output direction, using per-user ACL is not supported.

Filter ID IPv6 ACL

For the filter-Id ACL, the full ACEs and the `acl name(filter-id)` is configured on the device and only the `filter-id` is configured on the Cisco Secure ACS.

The `filter-id` is sent to the device in the ACCESS-Accept attribute, and the device looks up the filter-id for the ACEs, and then applies the ACEs to the client. When the client L2 roams to the foreign device, only the filter-id is sent to the foreign device in the mobility Handoff message. Output filtered ACL, using per-user ACL is not supported. The foreign device has to configure the filter-id and ACEs beforehand.

Downloadable IPv6 ACL

For the downloadable ACL (dACL), all the full ACEs and the `dacl` name are configured only on the Cisco Secure ACS.

The Cisco Secure ACS sends the `dacl` name to the device in its `ACCESS-Accept` attribute, which takes the `dacl` name and sends the `dACL` name back to the Cisco Secure ACS for the ACEs, using the `ACCESS-request` attribute.

Prerequisites for Configuring 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 Network Essentials license.

Restrictions for Configuring 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 device supports most of the Cisco IOS-supported IPv6 ACLs with some exceptions:

- The device does not support matching on these keywords: **flowlabel**, **routing header**, and **undetermined-transport**.
- The device does not support reflexive ACLs (the **reflect** keyword).
- The device 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 device 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 device does not allow the ACE to be added to the ACL that is currently attached to the interface

Configuring IPv6 ACLs

Follow the procedure given below to filter IPv6 traffic:

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.
4. Apply the IPv6 ACL to an interface. For router ACLs, you must also configure an IPv6 address on the Layer 3 interface to which the ACL is applied.

Default IPv6 ACL Configuration

There are no IPv6 ACLs configured or applied.

Interaction with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

  You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface. If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.

- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, for any additional configured ACLs, packets are dropped to the CPU, and the ACLs are applied in software. When the hardware is full a message is printed to the console indicating the ACL has been unloaded and the packets will be dropped on the interface.

How To Configure an IPv6 ACL

Creating an IPv6 ACL

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 access-list** *acl_name* **Example:** `Device# ipv6 access-list access-list-name`	Use a name to define an IPv6 access list and enter IPv6 access-list configuration mode.
Step 4	**{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

	Command or Action	Purpose
		in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range. If the operator follows the source-ipv6-prefix/prefix-length argument, it must match the source port. If the operator follows the destination-ipv6-prefix/prefix-length argument, it must match the destination port. • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6. • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295 • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.

	Command or Action	Purpose
Step 5	**{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 6	**{deny|permit} udp** **Example:** `{deny | permit} udp {source-ipv6-prefix/prefix-length | any | hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length | any | hostdestination-ipv6-address} [operator [port-number]][dscp value] [log][log-input] [neq {port |protocol}] [range {port |protocol}] [routing][sequence value][time-range name]`	(Optional) Define a UDP access list and the access conditions. Enter udp for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the operator [port]] port number or name must be a UDP port number or name, and the established parameter is not valid for UDP.
Step 7	**{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`	(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:

	Command or Action	Purpose
	`[icmp-code] |icmp-message] [dscpvalue] [log] [log-input] [routing] [sequence value][time-range name]`	• 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 8	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 9	**show ipv6 access-list** **Example:** `show ipv6 access-list`	Verify the access list configuration.
Step 10	**copy running-config startup-config** **Example:** `copy running-config startup-config`	(Optional) Save your entries in the configuration file.

Creating WLAN IPv6 ACL

Verifying IPv6 ACL

Displaying IPv6 ACLs

To display IPv6 ACLs, perform this procedure:

Procedure

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

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**show access-list** **Example:** `Device# show access-lists`	Displays all access lists configured on the device
Step 4	**show ipv6 access-list** *acl_name* **Example:** `Device# 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 an IPv6 ACL

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

Note Logging is supported only on Layer 3 interfaces.

```
Device(config)# ipv6 access-list CISCO
Device(config-ipv6-acl)# deny tcp any any gt 5000
Device (config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Device(config-ipv6-acl)# permit icmp any any
Device(config-ipv6-acl)# permit any any
```

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.

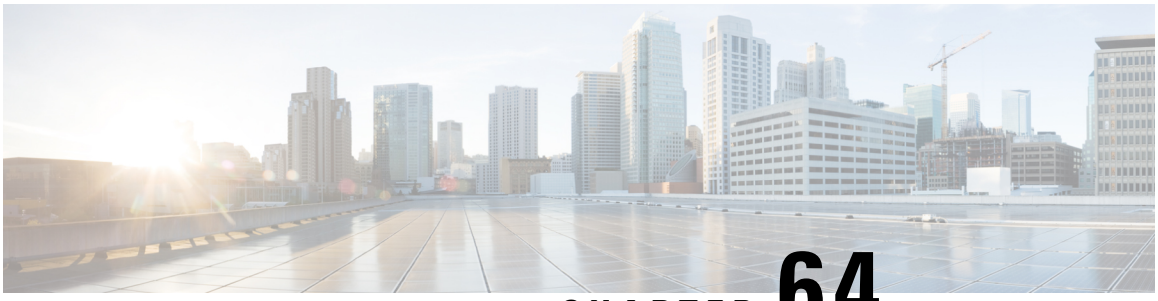
```
Device #show access-lists
Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

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.

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

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

PART X

CleanAir



CHAPTER 64

Cisco CleanAir

Information About Cisco CleanAir

Cisco CleanAir is a solution designed to proactively manage the challenges of a shared wireless spectrum. It allows you to see all the users of a shared spectrum (both native devices and foreign interferers). It also enables the network to act upon this information. For example, you can manually remove the interfering device, or the system can automatically change the channel away from the interference. CleanAir provides spectrum management and Radio Frequency (RF) visibility.

A Cisco CleanAir system consists of CleanAir-enabled access points. These access points collect information about all the 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 embedded wireless controller. The controller embedded wireless controller controls the access points.

For every device operating in the unlicensed band, Cisco CleanAir provides information about what it is, how it is impacting your wireless network, and what actions you or your network should take. It simplifies RF.

Wireless LAN systems operate in unlicensed 2.4-GHz and 5-GHz ISM bands. Many devices, such as 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 802.11 radio communications, might 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 RF interference.

Cisco CleanAir-Related Terms

Table 33: CleanAir-Related Terms

Term	Decription
AQI	Air Quality Index. The AQI is an indicator of air quality, based on the air pollutants. An AQI of 0 is bad and an AQI > 85 is good.
AQR	Air Quality Report. AQRs contain information about total interference from all the identified sources represented by AQI and the summary of the most severe interference categories. AQRs are sent every 15 minutes to the Mobility Controller and every 30 seconds in the Rapid mode.
DC	Duty Cycle. Percentage of time that the channel is utilized by a device.
EDRRM	Event-Driven RRM. EDRRM allows an access point in distress to bypass normal RRM intervals and immediately change channels.
IDR	Interference Device Reports that an access point sends to the embedded wireless controller.
ISI	Interference Severity Index. The ISI is an indicator of the severity of the interference.
RSSI	Received Signal Strength Indicator. RSSI is a measurement of the power present in a received radio signal. It is the power at which an access point sees the interferer device.

Cisco CleanAir Components

The basic Cisco CleanAir architecture consists of Cisco CleanAir-enabled APs and device.

An access point equipped with Cisco CleanAir technology collects information about Wi-Fi interference sources processes it. The access point sends the Air Quality Report (AQR) and Interference Device Report (IDR) to the embedded wireless controller.

The controller controls and configures CleanAir-capable access points, and collects and processes spectrum data. The provides local user interfaces (GUI and CLI) to configure basic CleanAir features and services and display current spectrum information. The also detects, merges, and mitigates interference devices using RRM TPC and DCA For details, see Interference Device Merging.

The device performs the following tasks in a Cisco CleanAir system:

- Configures Cisco CleanAir capabilities on the access point.
- Provides interfaces (CLI) for configuring Cisco CleanAir features and retrieving data.
- Displays spectrum data.
- Collects and processes AQRs from the access point and stores them in the air quality database. AQRs contain information about the total interference from all the identified sources represented by the Air Quality Index (AQI) and the summary for the most severe interference categories. The CleanAir system can also include unclassified interference information under per-interference type reports that enable you to take action in scenarios where interference because of unclassified interfering devices is more.
- Collects and processes IDRs from the access point and stores them in the interference device database.

Interference Types that Cisco CleanAir can Detect

Cisco CleanAir .

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 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.

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) which, 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.

Microwave Ovens, Outdoor Ethernet bridges are two classes of devices that qualify as persistent, since once detected, it is likely that these devices will continue to be a random problem and are not likely to move. For these types of devices we can tell RRM of the detection and Bias the affected channel so that RRM "remembers" that there is a high potential for client impacting interference for the Detecting AP on the detected channel. For more information, see https://www.cisco.com/c/en/us/td/docs/wireless/controller/technotes/8-3/b_RRM_White_Paper/b_RRM_White_Paper_chapter_0100.html?bookSearch=true#id_15217.

CleanAir PDA devices include:

- Microwave Oven
- WiMax Fixed
- WiMax Mobile
- Motorola Canopy

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.

EDRRM and AQR Update Mode

EDRRM is a feature that allows an access point that is in distress to bypass normal RRM intervals and immediately change channels. A CleanAir access point always monitors AQ and reports the AQ every 15 minutes. AQ only reports classified interference devices. The key benefit of EDRRM is fast action time. If an interfering device is operating on an active channel and causes enough AQ degradation to trigger an EDRRM, then no clients will be able to use that channel or the access point. You must remove the access point from the channel. EDRRM is not enabled by default, you must first enable CleanAir and then enable EDRRM.

Prerequisites for CleanAir

You can configure Cisco CleanAir only on CleanAir-enabled access points.

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. An AP can only measure air quality and interference when the AP is not busy transmitting Wi-Fi frames. This implies that CleanAir detections will be drastically lower if the AP is having a high channel utilization.
- 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 channels are legal within a regulatory domain

Restrictions for CleanAir

- Access points in monitor mode do not transmit Wi-Fi traffic or 802.11 packets. They are excluded from radio resource management (RRM) planning and are not included in the neighbor access point list. IDR clustering depends on the device's ability to detect neighboring in-network access points. Correlating interference device detections from multiple access points is limited between monitor-mode access points.
- For 4800 AP slot 1 5 GHz is dedicated and cannot be individually moved to monitor mode. However, slot 0 is XOR and can be moved to monitor as well as 2.4/5 GHz. Slot 2 is dedicated monitor and will operate in 5GHz and in AP monitor mode, slot 2 will be disabled because a monitor radio is already available in both 2.4/5GHz. 3700 AP has dedicated 2.4GHz (slot0) and 5GHz (slot1).
- Do not connect access points in SE connect mode directly to any physical port on the controller.
- CleanAir is not supported wherein the channel width is 160 MHz.

How to Configure CleanAir

Enabling CleanAir for the 2.4-GHz Band (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**

Step 2 On the **CleanAir** page, click the me**2.4 GHz Band** > **General** tab.

Step 3 Check the **Enable CleanAir** checkbox.

Step 4 Click **Apply**.

Enabling CleanAir for the 2.4-GHz Band (CLI)

Procedure

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

Step 2	**ap dot11 24ghz cleanair**
	Example:
	`Device(config)#ap dot11 24ghz cleanair`
	`Device(config)#no ap dot11 24ghz cleanair`
		Enables the CleanAir feature on the 802.11b network. Run the **no** form of this command to disable CleanAir on the 802.11b network.

Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Interference Reporting for a 2.4-GHz Device (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**.

Step 2 Click the **2.4 GHz Band** tab.

Step 3 Choose the interference types and add them to the **Interference Types to detect** section.

The following interference types are available:

- BLE Beacon—Bluetooth low energy beacon
- Bluetooth Discovery
- Bluetooth Link
- Canopy
- Continuous Transmitter
- DECT-like Phone—Digital Enhanced Cordless Technology phone
- 802.11 FH—802.11 frequency hopping device
- WiFi Inverted—Device using spectrally inverted Wi-Fi signals
- Jammer
- Microwave Oven
- WiFi Invalid Channel—Device using nonstandard Wi-Fi channels
- TDD Transmitter
- Video Camera
- SuperAG—802.11 SuperAG device
- WiMax Mobile
- WiMax Fixed
- 802.15.4
- Microsoft Device
- SI_FHSS

Step 4 Click **Apply**.

Configuring Interference Reporting for a 2.4-GHz Device (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# 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** |	Configures the 2.4-GHz interference devices to report to the device. Run the **no** form of this command to disable the configuration.

	Command or Action	Purpose

	superag | tdd-tx | video | wimax-fixed | wimax-mobile | xbox | zigbee }
	Example:
	`Device(config)# ap dot11 24ghz cleanair device bt-discovery`
	`Device(config)# ap dot11 24ghz cleanair device bt-link`
	`Device(config)# ap dot11 24ghz cleanair device canopy`
	`Device(config)# ap dot11 24ghz cleanair device cont-tx`
	`Device(config)# ap dot11 24ghz cleanair device dect-like`
	`Device(config)# ap dot11 24ghz cleanair device fh`
	`Device(config)# ap dot11 24ghz cleanair device inv`
	`Device(config)# ap dot11 24ghz cleanair device jammer`
	`Device(config)# ap dot11 24ghz cleanair device mw-oven`
	`Device(config)# ap dot11 24ghz cleanair device nonstd`
	`Device(config)# ap dot11 24ghz cleanair device report`
	`Device(config)# ap dot11 24ghz cleanair device superag`
	`Device(config)# ap dot11 24ghz cleanair device tdd-tx`
	`Device(config)# ap dot11 24ghz cleanair device video`
	`Device(config)# ap dot11 24ghz cleanair device wimax-fixed`
	`Device(config)# ap dot11 24ghz cleanair device wimax-mobile`
	`Device(config)# ap dot11 24ghz cleanair device xbox`
	`Device(config)# ap dot11 24ghz cleanair device zigbee`
		The following is a list of the keyword descriptions:
		• **bt-discovery**—Bluetooth discovery
		• **bt-link**—Bluetooth link
		• **canopy**—Canopy device
		• **cont-tx**—Continuous transmitter
		• **dect-like**—Digital Enhanced Cordless Communication-like phone
		• **fh**—802.11-frequency hopping device
		• **inv**—Device using spectrally inverted Wi-Fi signals
		• **jammer**—Jammer
		• **mw-oven**—Microwave oven
		• **nonstd**—Device using nonstandard Wi-Fi channels
		• **report**—Interference device reporting
		• **superag**—802.11 SuperAG device
		• **tdd-tx**—TDD transmitter
		• **video**—Video camera
		• **wimax-fixed**—WiMax Fixed
		• **wimax-mobile**—WiMax Mobile
		• **microsoft xbox**—Microsoft Xbox device
		• **zigbee**—802.15.4 device

	Command or Action	Purpose
	`Device(config)# ap dot11 24ghz cleanair device alarm`	
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Enabling CleanAir for the 5-GHz Band (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**

Step 2 On the **CleanAir** page, click the me**5 GHz Band** > **General** tab.

Step 3 Check the **Enable CleanAir** checkbox.

Step 4 Click **Apply**.

Enabling CleanAir for the 5-GHz Band (CLI)

Procedure

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

Step 2	**ap dot11 5ghz cleanair**
	Example:
	`Device(config)#ap dot11 5ghz cleanair`
	`Device(config)#no ap dot11 5ghz cleanair`
		Enables the CleanAir feature on a 802.11a network. Run the **no** form of this command to disable CleanAir on the 802.11a network.

Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Interference Reporting for a 5-GHz Device (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **CleanAir**.

Step 2 Click the **5 GHz Band** tab.

Step 3 Choose the interference types and add them to the **Interference Types to detect** section.

The following interference types are available:

- Canopy
- Continuous Transmitter
- DECT-like Phone—Digital Enhanced Cordless Technology phone
- 802.11 FH—802.11 frequency hopping device
- WiFi Inverted—Device using spectrally inverted Wi-Fi signals
- Jammer
- WiFi Invalid Channel—Device using nonstandard Wi-Fi channels
- SuperAG—802.11 SuperAG device
- TDD Transmitter
- WiMax Mobile
- WiMax Fixed
- Video Camera

Step 4 Click **Apply**.

Configuring Interference Reporting for a 5-GHz Device (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11 5ghz cleanair device** {**canopy** | **cont-tx** | **dect-like** | **inv** | **jammer** | **nonstd** | **report** | **superag** | **tdd-tx** | **video** | **wimax-fixed** | **wimax-mobile**}	Configures a 5-GHz interference device to report to the device. Run the **no** form of this command to disable interference device reporting.

	Command or Action	Purpose

	Example:
	
	`Device(config)#ap dot11 5ghz cleanair device canopy`
	
	`Device(config)#ap dot11 5ghz cleanair device cont-tx`
	
	`Device(config)#ap dot11 5ghz cleanair device dect-like`
	
	`Device(config)#ap dot11 5ghz cleanair device inv`
	
	`Device(config)#ap dot11 5ghz cleanair device jammer`
	
	`Device(config)#ap dot11 5ghz cleanair device nonstd`
	
	`Device(config)#ap dot11 5ghz cleanair device report`
	
	`Device(config)#ap dot11 5ghz cleanair device superag`
	
	`Device(config)#ap dot11 5ghz cleanair device tdd-tx`
	
	`Device(config)#ap dot11 5ghz cleanair device video`
	
	`Device(config)#ap dot11 5ghz cleanair device wimax-fixed`
	
	`Device(config)#ap dot11 5ghz cleanair device wimax-mobile`
	
	`Device(config)#ap dot11 5ghz cleanair device si_fhss`
	
	`Device(config)#ap dot11 5ghz cleanair device alarm`
		The following is a list of the keyword descriptions:
		• **canopy**—Canopy device
		• **cont-tx**—Continuous transmitter
		• **dect-like**—Digital Enhanced Cordless Communication-like phone
		• **fh**—802.11-frequency hopping device
		• **inv**—Device using spectrally-inverted Wi-Fi signals
		• **jammer**—Jammer
		• **nonstd**—Device using nonstandard Wi-Fi channels
		• **superag**—802.11 SuperAG device
		• **tdd-tx**—TDD transmitter
		• **video**—Video camera
		• **wimax-fixed**—WiMax fixed
		• **wimax-mobile**—WiMax mobile

Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Event Driven RRM for a CleanAir Event (GUI)

Procedure

Step 1 Choose **Configuration** > **Radio Configurations** > **RRM**.
The **Radio Resource Management** page is displayed.

Step 2 Click the **DCA** tab.

Step 3 In the **Event Driven RRM** section, check the **EDRRM** check box to run RRM when CleanAir-enabled AP detects a significant level of interference.

Step 4 Configure the **Sensitivity Threshold** level at which RRM has to be invoked from the following options:

- **Low**: Represents a decreased sensitivity to changes in the environment and its value is set at 35.
- **Medium**: Represents medium sensitivity to changes in the environment at its value is set at 50.
- **High**: Represents increased sensitivity to changes in the environment at its value is set at 60.
- **Custom**: If you choose this option, you must specify a custom value in the **Custom Threshold** box.

Step 5 To configure rogue duty cycle, check the **Rogue Contribution** check box and then specify the **Rogue Duty-Cycle** in terms of percentage. The default value of rogue duty cycle is 80 percent.

Note Rogue Contribution is a new component included in ED-RRM functionality. Rogue Contribution allows ED-RRM to trigger based on identified Rogue Channel Utilization, which is completely separate from CleanAir metrics. Rogue Duty Cycle comes from normal off channel RRM metrics, and invokes a channel change based on neighboring rogue interference. Because this comes from RRM metrics and not CleanAir, the timing - assuming normal 180 second off channel intervals - would be within 3 minutes or 180 seconds worst case. It is configured separately from CleanAir ED-RRM and is disabled by default. This allows the AP to become reactive to Wi-Fi interference that is not coming from own network and is measured at each individual AP.

Step 6 Save the configuration.

Configuring EDRRM for a CleanAir Event (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11** {**24ghz** | **5ghz**} **rrm channel cleanair-event** **Example:** `Device(config)#ap dot11 24ghz rrm channel cleanair-event`	Enables EDRRM CleanAir event. Run the **no** form of this command to disable EDRRM.

	Command or Action	Purpose
	`Device(config)#no ap dot11 24ghz rrm channel cleanair-event`	
Step 3	**ap dot11** {**24ghz** | **5ghz**} **rrm channel cleanair-event** [**sensitivity** {**high** | **low** | **medium**}] **Example:** `Device(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high`	Configures the EDRRM sensitivity of the CleanAir event. The following is a list of the keyword descriptions: • **High**—Specifies the most sensitivity to non-Wi–Fi interference as indicated by the AQ value. • **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:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying CleanAir Parameters

You can verify CleanAir parameters using the following commands:

Table 34: Commands for verifying CleanAir

Command Name	Description
show ap dot11 24ghz cleanair device type all	Displays all the CleanAir interferers for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type bt-discovery	Displays CleanAir interferers of type BT Discovery for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type bt-link	Displays CleanAir interferers of type BT Link for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type canopy	Displays CleanAir interferers of type Canopy for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type cont-tx	Displays CleanAir interferers of type Continuous transmitter for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type dect-like	Displays CleanAir interferers of type DECT Like for the 2.4-GHz band.

Command Name	Description
show ap dot11 24ghz cleanair device type fh	Displays CleanAir interferers of type 802.11FH for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type inv	Displays CleanAir interferers of type Wi-Fi Inverted for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type jammer	Displays CleanAir interferers of type Jammer for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type mw-oven	Displays CleanAir interferers of type MW Oven for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type nonstd	Displays CleanAir interferers of type Wi-Fi inverted channel for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type superag	Displays CleanAir interferers of type SuperAG for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type tdd-tx	Displays CleanAir interferers of type TDD Transmit for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type video	Displays CleanAir interferers of type Video Camera for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type wimax-fixed	Displays CleanAir interferers of type WiMax Fixed for the 2.4-GHz band.

Monitoring Interference Devices

When a CleanAir-enabled access point detects interference devices, detections of the same device from multiple sensors are merged together to create clusters. Each cluster is given a unique ID. Some devices conserve power by limiting the transmit time until actually needed, which results in the spectrum sensor to stop detecting the device temporarily. This device is then correctly marked as down. Such a device is correctly removed from the spectrum database. In cases when all the interferer detections for a specific device are reported, the cluster ID is kept alive for an extended period of time to prevent possible device-detection bouncing. If the same device is detected again, it is merged with the original cluster ID and the device-detection history is preserved.

For example, some bluetooth headsets operate on battery power. These devices employ methods to reduce power consumption, such as turning off the transmitter when not actually needed. Such devices can appear to come and go from the classification. To manage these devices, CleanAir keeps the cluster IDs for longer and they are remerged into a single record upon detection. This process smoothens the user records and accurately represents the device history.

Configuration Examples for CleanAir

This example shows how to enable CleanAir on the 2.4-GHz band and an access point operating in the channel:

```
Device#configure terminal
Device(config)#ap dot11 24ghz cleanair
```

```
Device(config)#exit
Device#ap name TAP1 dot11 24ghz cleanair
Device#end
```

This example shows how to enable an EDRRM CleanAir event in the 2.4-GHz band and configure high sensitivity to non-Wi–Fi interference:

```
Device#configure terminal
Device(config)#ap dot11 24ghz rrm channel cleanair-event
Device(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high
Device(config)#end
```

CleanAir FAQs

Q. Multiple access points detect the same interference device. However, the device shows them as separate clusters or different suspected devices clustered together. Why does this happen?

A. Access points must be RF neighbors for the device to consider merging the devices that are detected by these access points. An access point takes time to establish neighbor relationships. A few minutes after the device reboots or after there is a change in the RF group, and similar events, clustering will not be very accurate.

Q. How do I view neighbor access points?

A. To view neighbor access points, use the **show ap** *ap_name* **auto-rf dot11** { **24ghz** | **5ghz** } command.

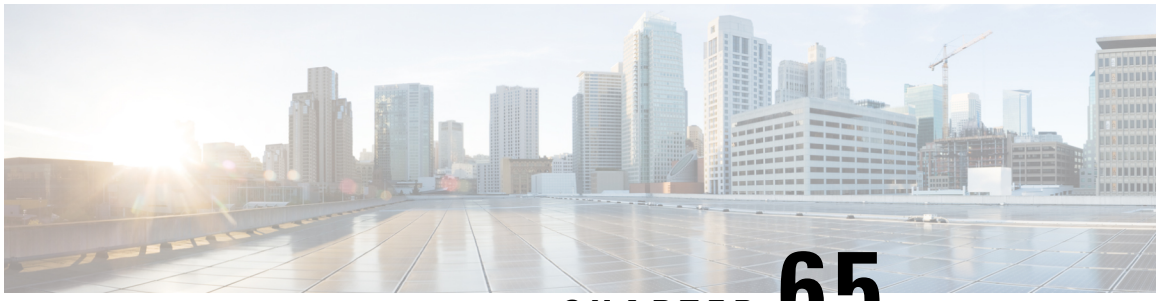
This example shows how to display the neighbor access points:

```
Device#show ap name AS-5508-5-AP3 auto-rf dot11 24ghz

<snippet>
Nearby APs
  AP 0C85.259E.C350 slot 0                          :  -12 dBm on   1 (10.10.0.5)
  AP 0C85.25AB.CCA0 slot 0                          :  -24 dBm on   6 (10.10.0.5)
  AP 0C85.25C7.B7A0 slot 0                          :  -26 dBm on  11 (10.10.0.5)
  AP 0C85.25DE.2C10 slot 0                          :  -24 dBm on   6 (10.10.0.5)
  AP 0C85.25DE.C8E0 slot 0                          :  -14 dBm on  11 (10.10.0.5)
  AP 0C85.25DF.3280 slot 0                          :  -31 dBm on   6 (10.10.0.5)
  AP 0CD9.96BA.5600 slot 0                          :  -44 dBm on   6 (10.0.0.2)
  AP 24B6.5734.C570 slot 0                          :  -48 dBm on  11 (10.0.0.2)
<snippet>
```

Q. What are the AP debug commands available for CleanAir?

A. The AP debug commands for CleanAir are:

-
-



CHAPTER 65

Spectrum Intelligence

Spectrum Intelligence

The Spectrum Intelligence feature scans for non-Wi-Fi radio interference on 2.4-GHz and 5-GHz bands. Spectrum intelligence provides basic functions to detect interferences of three types, namely microwave, continuous wave (like video bridge and baby monitor), wi-fi and frequency hopping (bluetooth and frequency-hopping spread spectrum (FHSS) cordless phone).

The following Cisco access points (APs) support Spectrum Intelligence feature:

- Cisco Catalyst 9115 Series Wi-Fi 6 APs
- Cisco Aironet 1852E/I APs
- Cisco Aironet 1832I APs
- Cisco Aironet 1815W/T/I/M APs
- Cisco Aironet 1810W/T APs
- Cisco Aironet 1800I/S APs
- Cisco Aironet 1542D/I APs

Note You must enable Spectrum Intelligence feature on the Cisco Aironet 1832 and 1852 series APs to get radio details, such as noise, air-quality, interference, and radio utilization on the Cisco DNA Center Assurance AP health.

Restrictions

- SI APs only report a single interference type in Local mode.

- SI does not support high availability for air quality or interference reports. High Availability is not supported because interference report/device reported will not be copied to standby after switchover. We expect AP to send it again, if at all interferer is still there.
- Spectrum Intelligence detects only three types of devices:
  - Microwave
  - Continuous wave—(video recorder, baby monitor)
  - SI-FHSS—(Bluetooth, Frequency hopping Digital European Cordless Telecommunications (DECT) phones)

Configuring Spectrum Intelligence

Follow the procedure given below to configure spectrum intelligence:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ap dot11** {**24ghz** | **5ghz**} **SI** **Example:** `Device(config)# ap dot11 24ghz SI`	Configures the 2.4-GHz or 5-GHz Spectrum Intelligence feature on the 802.11a or 802.11b network. Add **no** form of the command to disable SI on the 802.11a or 802.11b network.

Verifying Spectrum Intelligence Information

Use the following commands to verify spectrum intelligence information:

To display the SI information for a 2.4-GHz or 5-GHz band, use the following command:

```
Device# show ap dot11 24ghz SI config

SI Solution..................................... : Enabled
Interference Device Settings:
        SI_FHSS............................... : Enabled
    Interference Device Types Triggering Alarms:
        SI_FHSS............................... : Disabled
```

To display SI interferers of type Continuous transmitter for a 2.4-GHz band, use the following command:

```
Device# show ap dot11 24ghz SI device type cont_tx

DC    = Duty Cycle (%)
ISI = Interference Severity Index (1-Low Interference, 100-High Interference)
RSSI  = Received Signal Strength Index (dBm)
```

```
DevID = Device ID
AP type = CA, clean air, SI spectrum intelligence

No  ClusterID       DevID  Type     AP Type AP Name                 ISI  RSSI  DC   Channel
--- --------------- ------ -------- ------- ------------------ ---- ----- ----  ---  ------
   xx:xx:xx:xx  0014  BT     CA    myAP1              --   -69 00  133
   xx:xx:xx:xx  0014  BT     SI    myAP1              --   -69 00  133
```

To display 802.11a interference devices information for the given AP for 5-GHz, use the following command:

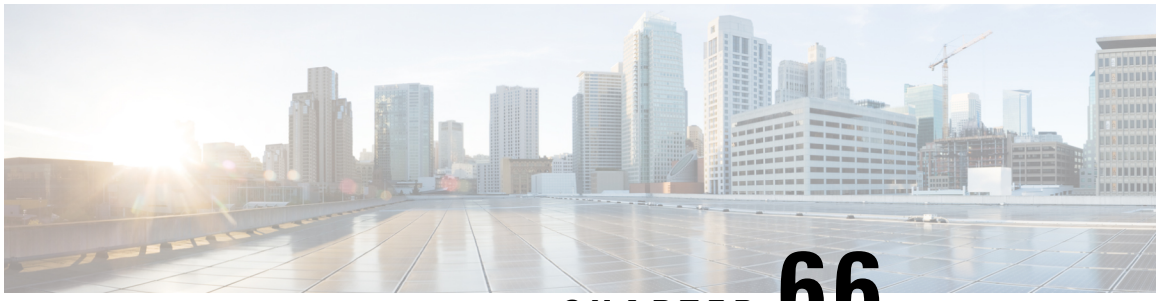
```
Device# show ap dot11 5ghz SI device type ap

DC    = Duty Cycle (%)
ISI   = Interference Severity Index (1-Low Interference, 100-High Interference)
RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
AP type = CA, clean air, SI spectrum intelligence

No  ClusterID/BSSID    DevID  Type    AP Type AP Name                 ISI  RSSI  DC   Channel
--- ------------------ ------ ------- ------  ------------------------ ---- ----- ----
----------
```

To display all Cisco CleanAir interferers for a 2.4-GHz band, use the following command:

```
Device# show ap dot11 24ghz cleanair device type all
```

PART XI

WLAN



CHAPTER 66

WLANs

Information About WLANs

This feature enables you to control WLANs for lightweight access points. Each WLAN has a separate WLAN ID, a separate profile name, and a WLAN SSID. All access points can advertise up to 16 WLANs. However, you can create up to 4096 WLANs and then selectively advertise these WLANs (using profiles and tags) to different APs for better manageability.

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 select enables client radios that are capable of dual-band (2.4 and 5-GHz) operations 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 3 nonoverlapping channels. To prevent these sources of interference and improve overall network performance, configure band selection on the device.

Off-Channel Scanning Deferral

A lightweight access point, in normal operational conditions, periodically goes off-channel and scans another channel. This is in order to perform RRM operations such as the following:

- Transmitting and receiving Neighbor Discovery Protocol (NDP) packets with other APs.
- Detecting rogue APs and clients.
- Measuring noise and interference.

During the off-channel period, which normally is about 70 milliseconds, the AP is unable to transmit or receive data on its serving channel. Therefore, there is a slight impact on its performance and some client transmissions might be dropped.

While the AP is sending and receiving important data, it is possible to configure off-channel scanning deferral so that the AP does not go off-channel and its normal operation is not impacted. You can configure off-channel scanning deferral on a per-WLAN basis, per WMM UP class basis, with a specified time threshold in milliseconds. If the AP sends or receives, on a particular WLAN, a data frame marked with the given UP class within the specified threshold, the AP defers its next RRM off-channel scan. For example, by default, off-channel scanning deferral is enabled for UP classes 4, 5, and 6, with a time threshold of 100 milliseconds. Therefore, when RRM is about to perform an off-channel scan, a data frame marked with UP 4, 5, or 6 is received within the last 100 milliseconds, RRM defers going off-channel. The AP radio does not go off-channel when a voice call sending and receiving audio samples are marked as UP class 6 for every active 20 milliseconds.

Off-channel scanning deferral does come with a tradeoff. Off-channel scanning can impact throughput by 2 percent or more, depending on the configuration, traffic patterns, and so on. Throughput can be slightly improved if you enable off-channel scanning deferral for all traffic classes and increase the time threshold. However, by not going off-channel, RRM can fail to identify AP neighbors and rogues, resulting in negative impact to security, DCA, TPC, and 802.11k messages.

DTIM Period

In the 802.11 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 broadcast and multicast frames after every other beacon). For instance, if the beacon period of the 802.11 network is 100 ms and the DTIM value is set to 1, the access point transmits buffered broadcast and multicast frames for 10 times every 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 for 5 times every 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). The only recommended DTIM values are 1 and 2; higher DTIM values will likely cause communications problems.

Note A beacon period, which is specified in milliseconds on the device, is converted internally by the software to 802.11 Time Units (TUs), where 1 TU = 1.024 milliseconds. Depending on the AP model, the actual beacon period may vary slightly; for example, a beacon period of 100 ms may in practice equate to 104.448 ms.

Session Timeouts

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.

If a WLAN is configured with Layer 2 security, for example WPA2-PSK, and a Layer 3 authentication is also configured, the WLAN session timeout value is overridden with the 802.1X reauthentication timeout value. If APF reauthentication timeout value is greater than 65535, the WLAN session timeout is by default set to 65535; else, the configured 802.1X reauthentication timeout value is applied as the WLAN session timeout.

This section contains the following subsections:

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 devices 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 device 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 device sends Aironet IEs 0x85 and 0x95 (which contains the management IP address of the device and the IP address of the access point) in the reassociation response if it receives Aironet IE 0x85 in the reassociation request.

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 device, dropped by the device, or forwarded to the upstream VLAN.

Peer-to-peer blocking is supported for clients that are associated with local and central switching WLANs.

Note Peer-to-peer blocking feature is VLAN-based. WLANs using the same VLAN has an impact, if Peer-to-peer blocking feature is enabled.

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 device GUI or CLI to enable the diagnostic channel, and you can use the device **diag-channel** 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. CCX Diagnostic feature has been tested only with clients having Cisco ADU card

Prerequisites for WLANs

- You can associate up to 16 WLANs with each policy tag.
- We recommend that you assign one set of VLANs for WLANs and a different set of VLANs for management interfaces to ensure that devices properly route VLAN traffic.

Restrictions for WLANs

- Do not configure PSK and CCKM in a WLAN, as this configuration is not supported and impacts client join flow.
- Ensure that TKIP or AES ciphers are enabled with WPA1 configuration, else ISSU may break during upgrade process.
- When you change the WLAN profile name, then FlexConnect APs (using AP-specific VLAN mapping) will become WLAN-specific. If FlexConnect Groups are configured, the VLAN mapping will become Group-specific.
- Do not enable IEEE 802.1X Fast Transition on Flex Local Authentication enabled WLAN, as client association is not supported with Fast Transition 802.1X key management.
- Peer-to-peer blocking does not apply to multicast traffic.
- In FlexConnect, peer-to-peer blocking configuration cannot be applied only to a particular FlexConnect AP or a subset of APs. It is applied to all the FlexConnect APs that broadcast the SSID.
- The WLAN name and SSID can have up to 32 characters.
- WLAN and SSID names support only the following ASCII characters:
  - Numerals: 48 through 57 hex (0 to 9)
  - Alphabets (uppercase): 65 through 90 hex (A to Z)
  - Alphabets (lowercase): 97 through 122 hex (a to z)
  - ASCII space: 20 hex
  - Printable special characters: 21 through 2F, 3A through 40, and 5B through 60 hex, that is: ! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~
- WLAN name cannot be a keyword; for example, if you try to create a WLAN with the name as 's' by entering the **wlan s** command, it results in shutting down all WLANs because 's' is used as a keyword for shutdown.
- You cannot map a WLAN to VLAN 0. Similarly, you cannot map a WLAN to VLANs 1002 to 1006.
- Dual stack clients with a static-IPv4 address is not supported.
- In a dual-stack with IPv4 and IPv6 configured in the Cisco 9800 controller, if an AP tries to join controller with IPv6 tunnel before its IPv4 tunnel gets cleaned, you would see a traceback and AP join will fail.
- When creating a WLAN with the same SSID, you must create a unique profile name for each WLAN.

- When multiple WLANs with the same SSID get assigned to the same AP radio, you must have a unique Layer 2 security policy so that clients can safely select between them.
- If the newly configured SSID is on a 5-GHz DFS channel, beaconing does not start immediately.
- RADIUS server overwrite is not configured on a per WLAN basis, but rather on a per AAA server group basis.
- Downloadable ACL (DACL) is not supported in the FlexConnect mode or the local mode.

Caution Some clients might not be able to connect to WLANs properly if they detect the same SSID with multiple security policies. Use this WLAN feature with care.

How to Configure WLANs

Creating WLANs (GUI)

Procedure

Step 1 In the **Configuration** > **Tags & Profiles** > **WLANs** page, click **Add**.

The **Add WLAN** window is displayed.

Step 2 Under the **General** tab and **Profile Name** field, enter the name of the WLAN. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

Step 3 Click **Save & Apply to Device**.

Creating WLANs (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name wlan-id* [*ssid*] **Example:** `Device(config)# wlan mywlan 34 mywlan-ssid`	Specifies the WLAN name and ID: • For the *profile-name*, enter the profile name. The range is from 1 to 32 alphanumeric characters.

	Command or Action	Purpose
		• 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** • You can create SSID using GUI or CLI. However, we recommend that you use CLI to create SSID. • By default, the WLAN is disabled.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Deleting WLANs (GUI)

Procedure

Step 1 In the **Configuration** > **Tags & Profiles** > **WLANs** page, check the checkbox adjacent to the WLAN you want to delete.

To delete multiple WLANs, select multiple WLANs checkboxes.

Step 2 Click **Delete**.

Step 3 Click **Yes** on the confirmation window to delete the WLAN.

Deleting WLANs

Procedure

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

	Command or Action	Purpose
Step 2	**no wlan** *wlan-name wlan-id ssid* **Example:** `Device(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:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Searching WLANs (CLI)

To verify the list of all WLANs configured on the controller, use the following show command:

```
Device# 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
```

To use wild cards and search for WLANs, use the following `show` command:

```
Device# show wlan summary | include test-wlan-ssid
1    test-wlan                       test-wlan-ssid                   137   UP
```

Enabling WLANs (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **WLANs** page, click the WLAN name.

Step 3 In the **Edit WLAN** window, toggle the **Status** button to **ENABLED**.

Step 4 Click **Update & Apply to Device**.

Enabling WLANs (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.
Step 4	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.

Disabling WLANs (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 In the **WLANs** window, click the WLAN name.

Step 3 In the **Edit WLAN** window, set the **Status** toggle button as **DISABLED**.

Step 4 Click **Update & Apply to Device**.

Disabling WLANs (CLI)

Procedure

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

	Command or Action	Purpose
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**shutdown** **Example:** `Device(config-wlan)# shutdown`	Disables the WLAN.
Step 4	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.
Step 5	**show wlan summary** **Example:** `Device# show wlan summary`	Displays the list of all WLANs configured on the device. You can search for the WLAN in the output.

Configuring General WLAN Properties (CLI)

You can configure the following properties:

- Media stream
- Broadcast SSID
- Radio

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**shutdown** **Example:** `Device(config-wlan)# shutdown`	Disables the WLAN.
Step 4	**broadcast-ssid** **Example:** `Device(config-wlan)# broadcast-ssid`	Broadcasts the SSID for this WLAN.

	Command or Action	Purpose
Step 5	**dot11bg 11g** **Example:** `Device(config-wlan)# dot11bg 11g`	Configures the WLAN radio policy for dot11 radios.
Step 6	**media-stream multicast-direct** **Example:** `Device(config-wlan)# media-stream multicast-direct`	Enables multicast VLANs on this WLAN.
Step 7	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.
Step 8	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.

Configuring Advanced WLAN Properties (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**chd** **Example:** `Device(config-wlan)# chd`	Enables coverage hole detection for this WLAN.
Step 4	**ccx aironet-iesupport** **Example:** `Device(config-wlan)# ccx aironet-iesupport`	Enables support for Aironet IEs for this WLAN.
Step 5	**client association limit** {*clients-per-wlan* | **ap** *clients-per-ap-per-wlan* | **radio***clients-per-ap-radio--per-wlan*} **Example:**	Sets the maximum number of clients, clients per AP, or clients per AP radio that can be configured on a WLAN.

	Command or Action	Purpose
	`Device(config-wlan)# client association limit ap 400`	
Step 6	**ip access-group web** *acl-name* **Example:** `Device(config-wlan)# ip access-group web test-acl-name`	Configures the IPv4 WLAN web ACL. The variable *acl-name* specifies the user-defined IPv4 ACL name.
Step 7	**peer-blocking** [**drop** | **forward-upstream**] **Example:** `Device(config-wlan)# 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**—No action is taken and forwards packets to the upstream.
Step 8	**channel-scan** {**defer-priority** {**0-7**} | **defer-time** {**0 - 6000**}} **Example:** `Device(config-wlan)# 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 6000. The default is 100.
Step 9	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode.

Configuring Advanced WLAN Properties (GUI)

Before you begin

Ensure that you have configured an AP Join Profile prior to configuring the primary and backup controllers.

Procedure

Step 1 Choose **Configuration** > **Wireless** > **WLANs** > **Wireless Networks**.

Step 2 In the **Wireless Networks** window, click **Add**.

Step 3 Under the **Advanced** tab, check the **Coverage Hole Detection** check box.

Step 4 Check the **Aironet IE** check box to enable Aironet IE on the WLAN.

Step 5 Check the **Diagnostic Channel** check box to enable diagnostic channel on the WLAN.

Step 6 From the **P2P Blocking Action** drop-down list, choose the required value.

Step 7 Set the **Multicast Buffer** toggle button as enabled or diabled.

Step 8 Check the **Media Stream Multicast-Direct** check box to enable the feature.

Step 9 In the **Max Client Connections** section, specify the maximumui number of client connections for the following:

- In the **Per WLAN** field, enter a value. The valid range is between 0 and 10000.
- In the **Per AP Per WLAN** field, enter a value. The valid range is between 0 and 400.
- In the **Per AP Radio Per WLAN** field, enter a value. The valid range is between 0 and 200.

Step 10 In the **11v BSS Transition Support** section, perform the following configuration tasks:

a) Check the BSS Transition check box to enable 802.11v BSS Transition support.
b) In the **Disassociation Imminent** field, enter a value. The valid range is between 0 and 3000.
c) In the **Optimized Roaming Disassociation Timer** field, enter a value. The valid range is between 0 and 40.
d) Select the check box to enable the following:

- BSS Max Idle Service
- BSS Max Idle Protected
- Disassociation Imminent Service
- Directed Multicast Service
- Universal Admin
- Load Balance
- Band Select
- IP Source Guard

Step 11 From the **WMM Policy** drop-down list, choose the policy as Allowed, Disabled, or Required. By default, the WMM policy is Allowed.

Step 12 In the **Off Channel Scanning Defer** section, choose the appropriate **Defer Priority** values and then specify the required Scan Defer Time value in milliseconds.

Step 13 In the **Assisted Roaming (11k)** section, choose the appropriate status for the following:

- Prediction Optimization
- Neighbor List
- Dual-Band Neighbor List

Step 14 In the **DTIM Period (in beacon intervals)** section, specify a value for 802.11a/n and 802.11b/g/n radios. The valid range is from 1 to 255.

Step 15 Click **Save & Apply to Device**.

Verifying WLAN Properties (CLI)

To verify the WLAN properties based on the WLAN ID, use the following `show` command:

```
Device# show wlan id wlan-id
```

To verify the WLAN properties based on the WLAN name, use the following `show` command:

```
Device# show wlan name wlan-name
```

To verify the WLAN properties of all the configured WLANs, use the following `show` command:

```
Device# show wlan all
```

To verify the summary of all WLANs, use the following `show` command:

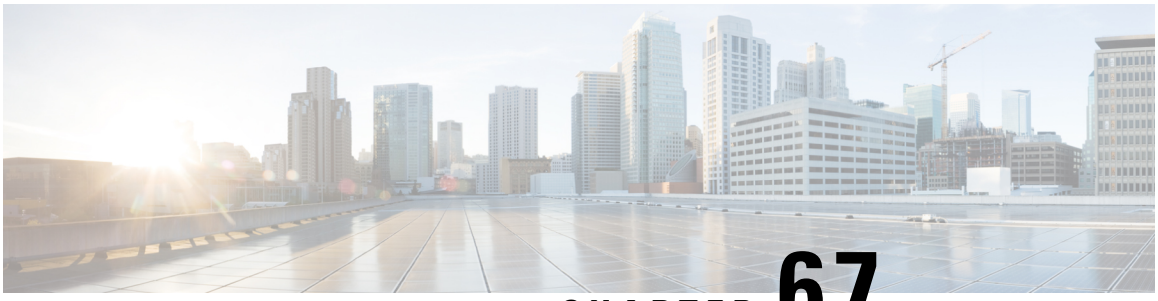
```
Device# show wlan summary
```

To verify the running configuration of a WLAN based on the WLAN name, use the following `show` command:

```
Device# show running-config wlan wlan-name
```

To verify the running configuration of all WLANs, use the following `show` command:

```
Device# show running-config wlan
```



CHAPTER 67

Network Access Server Identifier

Information About Network Access Server Identifier

Network access server identifier (NAS-ID) is used to notify the source of a RADIUS access request, which enables the RADIUS server to choose a policy for that request. You can configure one on each WLAN profile, or VLAN interface. The NAS-ID is sent to the RADIUS server by the embedded wireless controller through an authentication request to classify users to different groups. This enables the RADIUS server to send a customized authentication response.

Note The acct-session-id is sent with the RADIUS access request only when accounting is enabled on the policy profile.

If you configure a NAS-ID for a WLAN profile, it overrides the NAS-ID that is configured for the VLAN interface.

The following options can be configured for a NAS ID:

- sys-name (System Name)
- sys-ip (System IP Address)
- sys-mac (System MAC Address)
- ap-ip (AP's IP address)
- ap-name (AP's Name)
- ap-mac (AP's MAC Address)
- ap-eth-mac (AP's Ethernet MAC Address)

- ap-policy-tag (AP's policy tag name)
- ap-site-tag (AP's site tag name)
- ssid (SSID Name)
- ap-location (AP's Location)

Creating a NAS ID Policy(GUI)

Procedure

Step 1 Choose **Configuration** > **Security** > **Wireless AAA Policy**.

Step 2 On the **Wireless AAA Policy** page, click the name of the **Policy** or click **Add** to create a new one.

Step 3 In the **Add/Edit Wireless AAA Policy** window that is displayed, enter the name of the policy in the **Policy Name** field.

Step 4 Choose from one of the NAS ID options from the **Option 1** drop-down list.

Step 5 Choose from one of the NAS ID options from the **Option 2** drop-down list.

Step 6 Choose from one of the NAS ID options from the **Option 3** drop-down list.

Step 7 Save the configuration.

Creating a NAS ID Policy

Follow the procedure given below to create NAS ID policy:

Before you begin

- NAS ID can be a combination of multiple NAS ID options; the maximum options are limited to 3.
- The maximum length of the NAS ID attribute is 253. Before adding a new attribute, the attribute buffer is checked, and if there is no sufficient space, the new attribute is ignored.
- By default, a wirleess aaa policy (default-aaa-policy) is created with the default configuration (sys-name). You can update this policy with various NAS ID options. However, the default-aaa-policy cannot be deleted.
- If a NAS ID is not configured, the default sys-name is considered as the NAS ID for all wireless-specific RADIUS packets (authentication and accounting) from the embedded wireless controller.

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**wireless aaa policy** *policy-name* **Example:** `Device(config)# wireless aaa policy test`	Configures a new AAA policy.
Step 3	**nas-id option1 sys-name** **Example:** `Device(config-aaa-policy)# nas-id option1 sys-name`	Configures NAS ID for option1.
Step 4	**nas-id option2 sys-ip** **Example:** `Device(config-aaa-policy)# nas-id option2 sys-ip`	Configures NAS ID for option2.
Step 5	**nas-id option3 sys-mac** **Example:** `Device(config-aaa-policy)# nas-id option3 sys-mac`	Configures NAS ID for option3.

Attaching a Policy to a Tag (GUI)

Procedure

Step 1 Choose **Configuration > Tags & Profiles > Tags** page, click **Policy** tab.

Step 2 Click **Add** to view the **Add Policy Tag** window.

Step 3 Enter a name and description for the policy tag.

Step 4 Click **Add** to map WLAN profile and Policy profile.

Step 5 Choose the **WLAN Profile** to map with the appropriate **Policy Profile**, and click the tick icon.

Step 6 Click **Save & Apply to Device**.

Attaching a Policy to a Tag (CLI)

Follow the procedure given below to attach a NAS ID policy to a tag:

Procedure

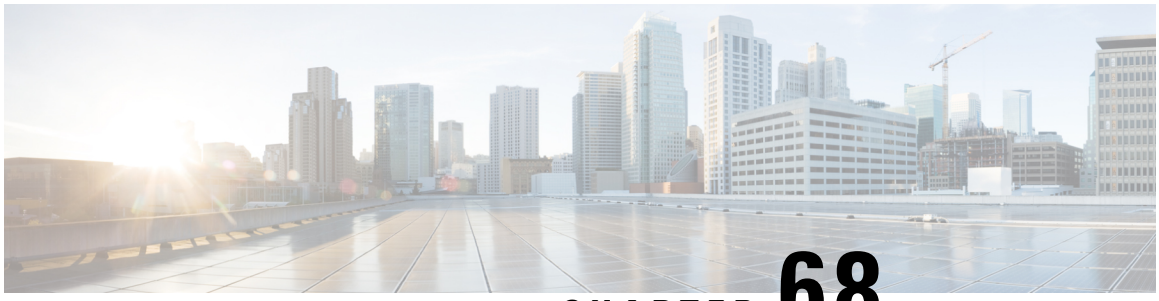
	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *policy-name* **Example:** `Device(config)# wireless profile policy test1`	Configures a WLAN policy profile.
Step 3	**aaa-policy** *aaa-policy-name* **Example:** `Device(config-wireless-policy)# aaa-policy policy-aaa`	Configures a AAA policy profile.
Step 4	**exit** **Example:** `Device(config-wireless-policy)# exit`	Returns to global configuration mode.
Step 5	**wireless tag policy** *policy-tag* **Example:** `Device(config)# wireless tag policy policy-tag1`	Configures a wireless policy tag.
Step 6	**wlan wlan1 policy** *policy-name* **Example:** `Device(config)# wlan wlan1 policy test1`	Maps a WLAN profile to a policy profile. **Note** You can also use the **ap-tag** option to configure a NAS ID for an AP group, which will override the NAS ID that is configured for a WLAN profile or the VLAN interface.

Verifying the NAS ID Configuration

Use the following **show** command to verify the NAS ID configuration:

```
Device# show wireless profile policy detailed test1

Policy Profile Name           : test1
Description                   :
Status                        : ENABLED
VLAN                          : 1
Client count                  : 0

:
:
AAA Policy Params
```

```
AAA Override                  : DISABLED
NAC                           : DISABLED
AAA Policy name               : test
```



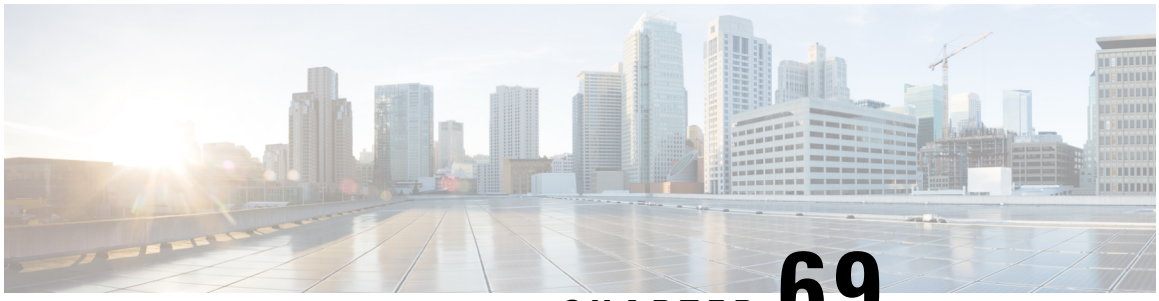
CHAPTER 68

DHCP for WLANs

DHCP for WLANs

DHCP packets sent by the wireless clients are released in their respective VLANs as broadcast by the AP and relies on the fact that the network gateway of that VLAN forwards the requests to the DHCP server.

Note Internal DHCP server is not supported in EWC.



CHAPTER 69

WLAN Security

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.

Prerequisites for Layer 2 Security

WLANs with the same SSID must have unique Layer 2 security policies so that clients can make a WLAN selection based on the information advertised in beacon and probe responses. The available Layer 2 security policies are as follows:

- None (open WLAN)
- WPA+WPA2

Note

- Although WPA and WPA2 cannot be used by multiple WLANs with the same SSID, you can configure two WLANs with the same SSID with WPA/TKIP with PSK and Wi-Fi Protected Access (WPA)/Temporal Key Integrity Protocol (TKIP) with 802.1X, or with WPA/TKIP with 802.1X or WPA/AES with 802.1X.
- A WLAN configured with TKIP support will not be enabled on an RM3000AC module.

- Static WEP (not supported on Wave 2 APs)

How to Configure WLAN Security

Configuring Static WEP Layer 2 Security Parameters (CLI)

Before you begin

You must have administrator privileges.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**end** **Example:** `Device(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)

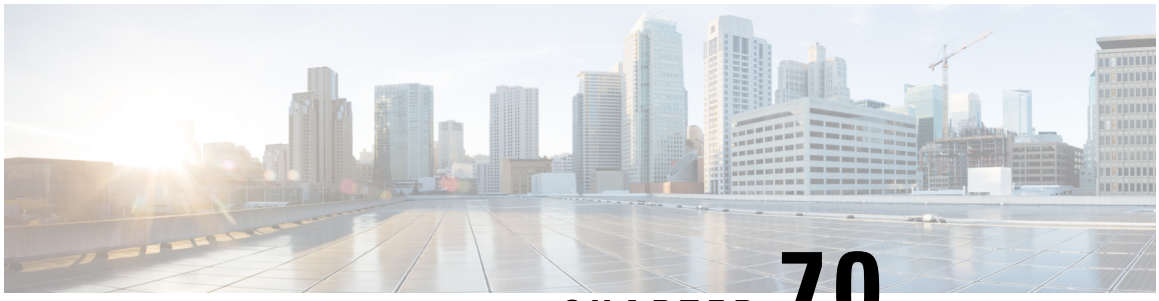
Before you begin

You must have administrator privileges.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**security wpa** **Example:** `Device(config-wlan)# security wpa`	
Step 3	**security wpa wpa1** **Example:** `Device(config-wlan)# security wpa wpa1`	Enables .
Step 4	**security wpa wpa1 ciphers** [**aes** | **tkip**] **Example:**	Specifies the WPA1 cipher. Choose one of the following encryption types: • **aes**—Specifies WPA/AES support.

	Command or Action	Purpose
	`Device(config-wlan)# security wpa wpa1 ciphers aes`	• **tkip**—Specifies WPA/TKIP support.
Step 5	**security wpa wpa2** **Example:** `Device(config-wlan)# security wpa wpa2`	Enables WPA2.
Step 6	**security wpa wpa2 ciphers aes** **Example:** `Device(config-wlan)# security wpa wpa2` **Example:**	Configure WPA2 cipher.



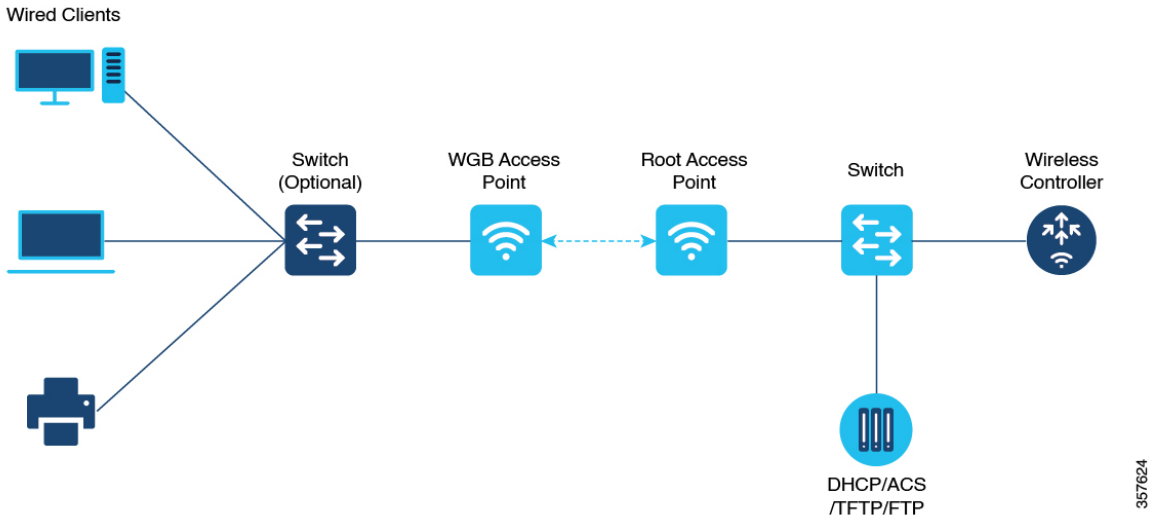
CHAPTER 70

Workgroup Bridges

Cisco Workgroup Bridges

A workgroup bridge (WGB) is an Access Point (AP) mode to provide wireless connectivity to wired clients that are connected to the Ethernet port of the WGB AP. 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 WLC through infrastructure AP using Internet Access Point Protocol (IAPP) messaging. The WGB establishes a single wireless connection to the root AP, which in turn, treats the WGB as a wireless client.

Figure 23: Example of a WGB



The mode supported in WGB for Embedded Wireless Controller is:

- Flex Mode: Central authentication and local switching.

Note Cenral authentication is supported on Wave 1 and Wave 2 APs, whereas local switching is supported only on Wave 2 APs.

The following features are supported for use with a WGB:

Table 35: WGB Feature Matrix

Feature	Cisco Wave 1 APs	Cisco Wave 2
802.11r	Supported	Supported
QOS	Supported	Supported
UWGB mode	Supported	Supported on Wave 2 APs
IGMP Snooping or Multicast	Supported	Supported
802.11w	Supported	Supported
PI support (without SNMP)	Supported	Not supported
IPv6	Supported	Supported
VLAN	Supported	Supported
802.11i (WPAv2)	Supported	Supported
Broadcast tagging/replicate	Supported	Supported
Unified VLAN client	Implicitly supported (No CLI required)	Supported
WGB client	Supported	Supported
802.1x – PEAP, EAP-FAST, EAP-TLS	Supported	Supported
NTP	Supported	Supported
Wired client support on all LAN ports	Supported in Wired-0 and Wired-1 interfaces	Supported in all Wired-0, 1 and LAN ports 1, 2, and 3

Table 36: Supported Access Points and Requirements

Access Points	Requirements
Cisco Aironet 2700, 3700, and 1572 Series	Requires autonomous image.
Cisco Aironet 2800, 3800, 4800, 1562, and Cisco Catalyst 9105, 9115, IW6300 and ESW6300 Series	CAPWAP image starting from Cisco AireOS 8.8 release.

- MAC filtering is not supported for wired clients.

- Idle timeout is not supported for both WGB and wired clients.
- Session timeout is not applicable for wired clients.
- Web authentication is not supported.
- WGB supports only up to 20 clients.
- If you want to use a chain of certificates, copy all the CA certificates to a file and install it under a trust point on the WGB, else server certificate validation may fail.
- Wired clients connected to the WGB are not authenticated for security. Instead, the WGB is authenticated against the access point to which it associates. Therefore, we recommend that you physically secure the wired side of the WGB.
- Wired clients connected to a WGB inherit the WGB's QoS and AAA override attributes.
- To enable the WGB to communicate with the root AP, create a WLAN and make sure that Aironet IE is enabled under the Advanced settings.

Configuring Workgroup Bridge on a WLAN

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan wlan-profile`	Enters WLAN configuration submode. The *wlan-profile* is the profile name of the configured WLAN.
Step 3	**ccx aironet-iesupport** **Example:** `Device(config-wlan)# ccx aironet-iesupport`	Enables support for Aironet IEs for this WLAN.
Step 4	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Restarts the WLAN.

Verifying the Status of Workgroup Bridges

- To verify the number of WGBs, use the following command:

  show wireless wgb summary

The following is a sample output:

```
Device#show wireless wgb summary
Number of WGBs: 1
MAC Address    AP Name                              WLAN State              Clients
-------------------------------------------------------------------------------------
7070.8b7a.7030 Ed2-JFW-AP1                          1    Run                1
```

- To verify WGB details, use the following command:

  show wireless wgb mac-address *MAC-address* **detail**

  The following is a sample output:

```
Device#show wireless wgb mac-address 7XXX.8XXa.7XXX detail

Work Group Bridge

MAC Address        : 7XXX.8XXa.7XXX
AP Name            : Ed2-JFW-AP1
WLAN ID            : 1
State              : Run

Number of Clients: 1

MAC Address
------------
d8XX.97XX.bXXX
```

- To view the client details on the controller, use the following command:

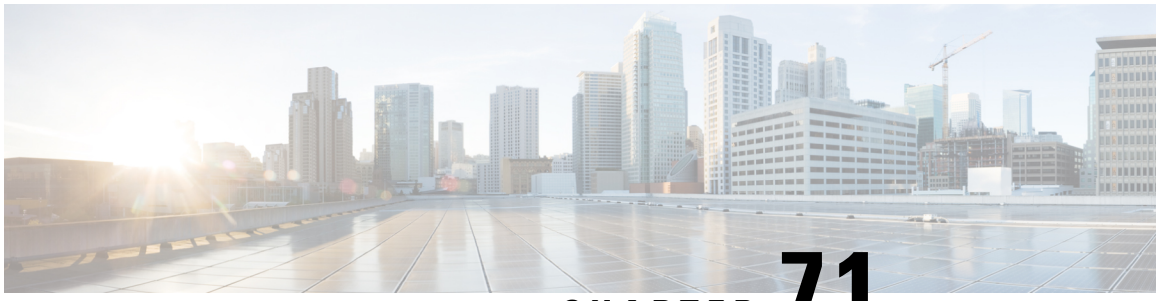
  show wireless client mac-address *MAC-address* **detail**

  The following is a sample output:

```
Device#show wireless client mac-address 7XXX.8bXX.70XX detail

Workgroup Bridge
Wired Client count : 1
```

- The following is a sample output:

```
Device#show wireless client mac-address d8XX.97XX.b0XX detail
Workgroup Bridge Client
WGB MAC Address : 7XXX.8bXX.70XX
```



CHAPTER 71

Peer-to-Peer Client Support

Information About Peer-to-Peer Client Support

Peer-to-peer client support can be applied to individual WLANs, with each client inheriting the peer-to-peer blocking setting of the WLAN to which it is associated. The peer-to-Peer Client Support feature provides a granular control over how traffic is directed. For example, you can choose to have traffic bridged locally within a device, dropped by a device, or forwarded to the upstream VLAN.

Peer-to-peer blocking is supported for clients that are associated with local and central switching WLANs.

Restrictions

- Peer-to-peer blocking does not apply to multicast traffic.
- Peer-to-peer blocking is not enabled by default.
- In FlexConnect, peer-to-peer blocking configuration cannot be applied only to a particular FlexConnect AP or a subset of APs. It is applied to all the FlexConnect APs that broadcast the SSID.
- FlexConnect central switching clients supports peer-to-peer upstream-forward. However, this is not supported in the FlexConnect local switching. This is treated as peer-to-peer drop and client packets are dropped.

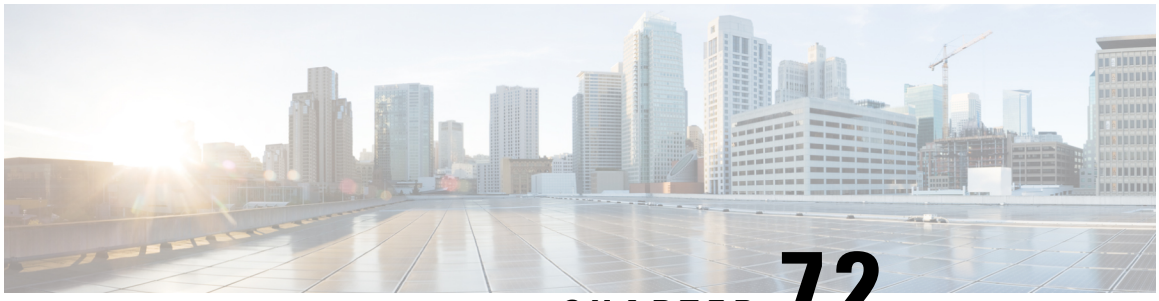
  FlexConnect central switching clients supports peer-to-peer blocking for clients associated with different APs. However, for FlexConnect local switching, this solution targets only clients connected to the same AP. FlexConnect ACLs can be used as a workaround for this limitation.

Configure Peer-to-Peer Client Support

Follow the procedure given below to configure Peer-to-Peer Client Support:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan wlan1`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**peer-blocking** [**drop** | **forward-upstream**] **Example:** `Device(config-wlan)# 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**—No action is taken and forwards packets to the upstream.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 5	**show wlan id** *wlan-id* **Example:** `Device# show wlan id 12`	Displays the details of the selected WLAN.



CHAPTER 72

802.11r BSS Fast Transition

Information About 802.11r Fast Transition

802.11r, which is the IEEE standard for fast roaming, introduces a new concept of roaming where the initial handshake with a new AP is done even before the corresponding client roams to the target access point. This concept is called Fast Transition. The initial handshake allows a client and the access points to do the Pairwise Transient Key (PTK) calculation in advance. These PTK keys are applied to the client and the access points after the client responds to the reassociation request or responds to the exchange with new target AP.

The FT key hierarchy is designed to allow clients to make fast BSS transitions between APs without requiring reauthentication at every AP. WLAN configuration contains a new Authenticated Key Management (AKM) type called FT (Fast Transition).

Client Roaming

For a client to move from its current AP to a target AP using the FT protocols, message exchanges are performed using one of the following methods:

- Over-the-Air—The client communicates directly with the target AP using IEEE 802.11 authentication with the FT authentication algorithm.
- Over-the-Distribution System (DS)—The client communicates with the target AP through the current AP. The communication between the client and the target AP is carried in FT action frames between the client and the current AP and is then sent through the device.

Figure 24: Message Exchanges when Over–the–Air Client Roaming is Configured

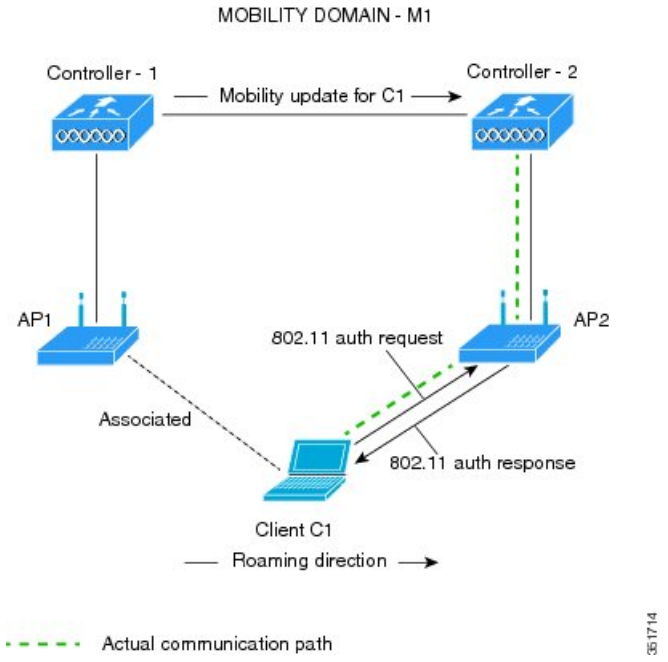
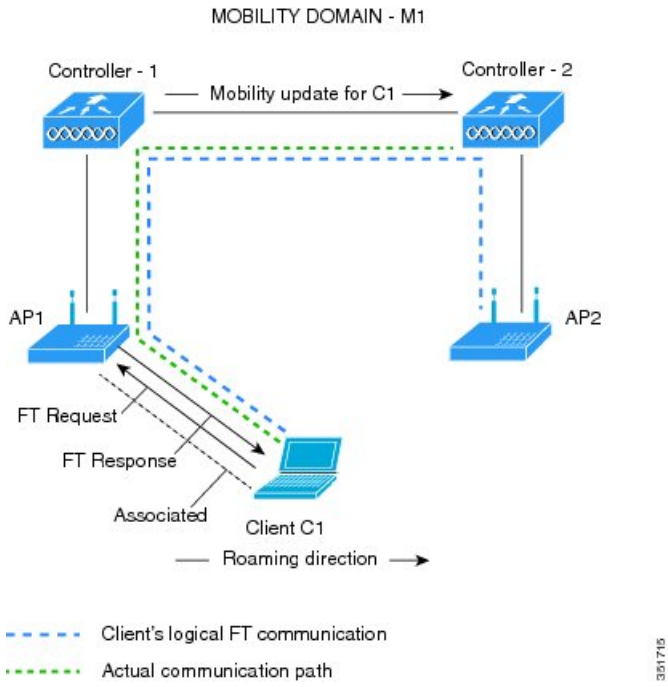


Figure 25: Message Exchanges when Over–the–DS Client Roaming is Configured



Restrictions for 802.11r Fast Transition

- EAP LEAP method is not supported.

- Traffic Specification (TSPEC) is not supported for 802.11r fast roaming. Therefore, RIC IE handling is not supported.
- If WAN link latency exists, fast roaming is also delayed. Voice or data maximum latency should be verified. The Cisco WLC handles 802.11r Fast Transition authentication requests during roaming for both Over-the-Air and Over-the-DS methods.
- Legacy clients cannot associate with a WLAN that has 802.11r enabled if the driver of the supplicant that is responsible for parsing the Robust Security Network Information Exchange (RSN IE) is old and not aware of the additional AKM suites in the IE. Due to this limitation, clients cannot send association requests to WLANs. These clients, however, can still associate with non-802.11r WLANs. Clients that are 802.11r-capable can associate as 802.11i clients on WLANs that have both 802.11i and 802.11r Authentication Key Management Suites enabled.

  The workaround is to enable or upgrade the driver of the legacy clients to work with the new 802.11r AKMs, after which the legacy clients can successfully associate with 802.11r-enabled WLANs.

  Another workaround is to have two SSIDs with the same name, but with different security settings (FT and non-FT).
- Fast Transition resource–request protocol is not supported because clients do not support this protocol. Also, the resource–request protocol is an optional protocol.
- To avoid any Denial of Service (DoS) attack, each Cisco WLC allows a maximum of three Fast Transition handshakes with different APs.
- Non-802.11r–capable devices will not be able to associate with FT-enabled WLAN.
- We do not recommend 802.11r FT + PMF.
- We recommend 802.11r FT Over-the-Air roaming for FlexConnect deployments.

Monitoring 802.11r Fast Transition (CLI)

The following command can be used to monitor 802.11r Fast Transition:

Command	Description
show wlan name *wlan-name*	Displays a summary of the configured parameters on the WLAN.

Command	Description
show wireless client mac-address *mac-address*	Displays the summary of the 802.11r authentication key management configuration on a client. . . . . . . Client Capabilities CF Pollable : Not implemented CF Poll Request : Not implemented Short Preamble : Not implemented PBCC : Not implemented Channel Agility : Not implemented Listen Interval : 15 Fast BSS Transition : Implemented Fast BSS Transition Details : Client Statistics: Number of Bytes Received : 9019 Number of Bytes Sent : 3765 Number of Packets Received : 130 Number of Packets Sent : 36 Number of EAP Id Request Msg Timeouts : 0 Number of EAP Request Msg Timeouts : 0 Number of EAP Key Msg Timeouts : 0 Number of Data Retries : 1 Number of RTS Retries : 0 Number of Duplicate Received Packets : 1 Number of Decrypt Failed Packets : 0 Number of Mic Failured Packets : 0 Number of Mic Missing Packets : 0 Number of Policy Errors : 0 Radio Signal Strength Indicator : -48 dBm Signal to Noise Ratio : 40 dB . . . . . .

Configuring 802.11r BSS Fast Transition on a Dot1x Security Enabled WLAN (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.

	Command or Action	Purpose
Step 3	**client vlan** *vlan-name* **Example:** `Device(config-wlan)# client vlan 0120`	Associates the client VLAN to this WLAN.
Step 4	**security dot1x authentication-list default** **Example:** `Device(config-wlan)# security dot1x authentication-list default`	Enables security authentication list for dot1x security. The configuration is similar for all dot1x security WLANs.
Step 5	**security ft** **Example:** `Device(config-wlan)# security ft`	Enables 802.11r Fast Transition on the WLAN.
Step 6	**security wpa akm ft dot1x** **Example:** `Device(config-wlan)# security wpa akm ft dot1x`	Enables 802.1x security on the WLAN.
Step 7	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.
Step 8	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode

Configuring 802.11r Fast Transition in an Open WLAN (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**client vlan** *vlan-id* **Example:** `Device(config-wlan)# client vlan 0120`	Associates the client VLAN to the WLAN.

	Command or Action	Purpose
Step 4	**no security wpa** **Example:** `Device(config-wlan)# no security wpa`	Disables WPA secuirty.
Step 5	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 6	**no security wpa wpa2** **Example:** `Device(config-wlan)# no security wpa wpa2`	Disables WPA2 security.
Step 7	**no wpa wpa2 ciphers aes** **Example:** `Device(config-wlan)# no security wpa wpa2 ciphers aes`	Disables WPA2 ciphers for AES.
Step 8	**security ft** **Example:** `Device(config-wlan)# security ft`	Specifies the 802.11r Fast Transition parameters.
Step 9	**no shutdown** **Example:** `Device(config-wlan)# shutdown`	Shuts down the WLAN.
Step 10	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode

Configuring 802.11r Fast Transition on a PSK Security–Enabled WLAN (CLI)

Procedure

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

	Command or Action	Purpose
Step 2	**wlan** *profile-name* **Example:** `Device# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**client vlan** *vlan-name* **Example:** `Device(config-wlan)# client vlan 0120`	Associates the client VLAN to this WLAN.
Step 4	**no security wpa akm dot1x** **Example:** `Device(config-wlan)# no security wpa akm dot1x`	Disables security AKM for dot1x.
Step 5	**security wpa akm ft psk** **Example:** `Device(config-wlan)# security wpa akm ft psk`	Configures Fast Transition PSK support.
Step 6	**security wpa akm psk set-key** {**ascii** {**0** | **8**} | **hex** {**0** | **8**}} **Example:** `Device(config-wlan)# security wpa akm psk set-key ascii 0 test`	Configures PSK AKM shared key.
Step 7	**security ft** **Example:** `Device(config-wlan)# security ft`	Configures 802.11r Fast Transition.
Step 8	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN.
Step 9	**end** **Example:** `Device(config-wlan)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-z** to exit global configuration mode

Disabling 802.11r Fast Transition (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 On the **WLANs** page, click the WLAN name.

Step 3 In the **Edit WLAN** window, click the **Security** > **Layer2** tab.

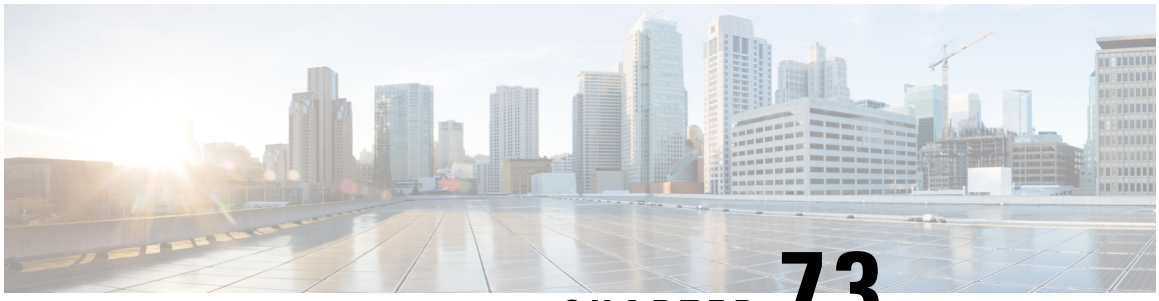
Step 4 From the **Fast Transition** drop-down list, choose **Disabled**. Note that you cannot enable or disable Fast Transition, if you have configured an SSID with Open Authentication.

Step 5 Click **Update & Apply to Device**.

Disabling 802.11r Fast Transition (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device# wlan test4`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**no security ft** [**over-the-ds** | **reassociation-timeout** *timeout-in-seconds*] **Example:** `Device(config-wlan)# no security ft over-the-ds`	Disables 802.11r Fast Transition on the WLAN.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.



CHAPTER 73

Assisted Roaming

802.11k Neighbor List and Assisted Roaming

The 802.11k standard allows an AP to inform 802.11k-capable clients of neighboring BSSIDs (APs in the same SSID). This can help the client to optimize its scanning and roaming behavior. Additionally, the Assisted Roaming Prediction Optimization feature can be used with non-802.11k clients, to discourage them from roaming to suboptimal APs.

Note We recommend not configuring two SSIDs with the same name in the controller, which may cause roaming issues.

Prediction Based Roaming - Assisted Roaming for Non-802.11k Clients

You can optimize roaming for non-802.11k clients by generating a prediction neighbor list for each client without sending an 802.11k neighbor list request. When prediction based roaming enables a WLAN, after each successful client association/re-association, the same neighbor list optimization applies on the non-802.11k client to generate and store the neighbor list in the mobile station software data structure. Clients at different locations have different lists because the client probes are seen with different RSSI values by the different neighbors as the clients usually probe before any association or re-association. This list is created with the most updated probe data and predicts the next AP that the client is likely to roam to.

The wireless infrastructure discourages clients from roaming to those less desirable neighbors by denying association if the association request to an AP does not match the entries on the stored prediction neighbor list.

- Denial count: Maximum number of times a client is refused association.
- Prediction threshold: Minimum number of entries required in the prediction list for the assisted roaming feature to activate.

For more information, see https://www.cisco.com/c/en/us/td/docs/wireless/controller/8-5/Enterprise-Mobility-8-5-Design-Guide/Enterprise_Mobility_8-5_Deployment_Guide/Chapter-11.html#pgfId-1140097.

Restrictions for Assisted Roaming

- This feature is supported only on 802.11n capable indoor access points. For a single band configuration, a maximum of 6 neighbors are visible in a neighbor list. For dual band configuration, a maximum of 12 neighbors are visible.
- You can configure assisted roaming only using the device CLI.

How to Configure Assisted Roaming

Configuring Assisted Roaming (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless assisted-roaming floor-bias** *dBm* **Example:** `Device(config)# wireless assisted-roaming floor-bias 20`	Configures neighbor floor label bias. The valid range is from 5 to 25 dBm, and the default value is 15 dBm.
Step 3	**wlan** *wlan-id* **Example:** `Device(config)# wlan wlan1`	Enters the WLAN configuration submode. The *wlan-name* is the profile name of the configured WLAN.
Step 4	**assisted-roaming neighbor-list** **Example:** `Device(wlan)# assisted-roaming neighbor-list`	Configures an 802.11k neighbor list for a WLAN. By default, assisted roaming is enabled on the neighbor list when you create a WLAN. The **no** form of the command disables assisted roaming neighbor list.
Step 5	**assisted-roaming dual-list** **Example:** `Device(wlan)# assisted-roaming dual-list`	Configures a dual-band 802.11k dual list for a WLAN. By default, assisted roaming is enabled on the dual list when you create a WLAN. The **no** form of the command disables assisted roaming dual list.

	Command or Action	Purpose
Step 6	**assisted-roaming prediction** **Example:** `Device(wlan)# assisted-roaming prediction`	Configures assisted roaming prediction list feature for a WLAN. By default, the assisted roaming prediction list is disabled. **Note** A warning message is displayed and load balancing is disabled for the WLAN if load balancing is already enabled for the WLAN.
Step 7	**wireless assisted-roaming prediction-minimum** *count* **Example:** `Device# wireless assisted-roaming prediction-minimum`	Configures the minimum number of predicted APs required for the prediction list feature to be activated. The default value is 3. **Note** If the number of the AP in the prediction assigned to the client is less than the number that you specify, the assisted roaming feature will not apply on this roam.
Step 8	**wireless assisted-roaming denial-maximum** *count* **Example:** `Device# wireless assisted-roaming denial-maximum 8`	Configures the maximum number of times a client can be denied association if the association request is sent to an AP does not match any AP on the prediction. The valid range is from 1 to 10, and the default value is 5.
Step 9	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Assisted Roaming

The following command can be used to verify assisted roaming configured on a WLAN:

Command	Description
show wlan id *wlan-id*	Displays the WLAN parameters on the WLAN.

Configuration Examples for Assisted Roaming

This example shows how to configure the neighbor floor label bias:

```
Device# configure terminal
Device(config)# wireless assisted-roaming floor-bias 10
```

```
Device(config)# end
Device# show wlan id 23
```

This example shows how to disable neighbor list on a specific WLAN:

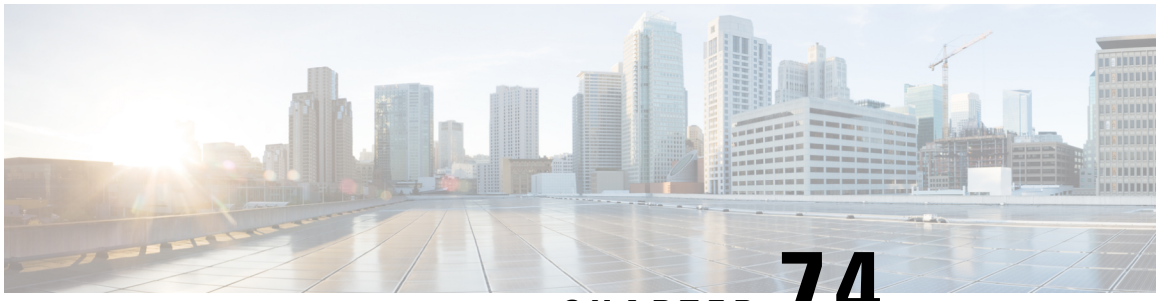
```
Device# configure terminal
Device(config)# wlan test1
Device(config (wlan)# no assisted-roaming neighbor-list
Device(config)(wlan)# end
Device# show wlan id 23
```

This example shows how to configure the prediction list on a specific WLAN:

```
Device# configure terminal
Device(config)# wlan test1
Device(config)(wlan)# assisted-roaming prediction
Device(config)(wlan)# end
Device# show wlan id 23
```

This example shows how to configure the prediction list based on assisted roaming prediction threshold and maximum denial count on a specific WLAN:

```
Device# configure terminal
Device(config)# wireless assisted-roaming prediction-minimum 4
Device(config)# wireless assisted-roaming denial-maximum 4
Device(config)(wlan)# end
Device# show wlan id 23
```



CHAPTER 74

802.11v

Information About 802.11v

The embedded wireless controller supports 802.11v amendment for wireless networks, which describes numerous enhancements to wireless network management.

One such enhancement is Network assisted Power Savings which helps clients to improve the battery life by enabling them to sleep longer. As an example, mobile devices typically use a certain amount of idle period to ensure that they remain connected to access points and therefore consume more power when performing the following tasks while in a wireless network.

Another enhancement is Network assisted Roaming which enables the WLAN to send requests to associated clients, advising the clients as to better APs to associate to. This is useful for both load balancing and in directing poorly connected clients.

Enabling 802.11v Network Assisted Power Savings

Wireless devices consume battery to maintain their connection to the clients, in several ways:

- By waking up at regular intervals to listen to the access point beacons containing a DTIM, which indicates buffered broadcast or multicast traffic that the access point delivers to the clients.
- By sending null frames to the access points, in the form of keepalive messages– to maintain connection with access points.
- Devices also periodically listen to beacons (even in the absence of DTIM fields) to synchronize their clock to that of the corresponding access point.

All these processes consume battery and this consumption particularly impacts devices (such as Apple), because these devices use a conservative session timeout estimation, and therefore, wake up often to send keepalive messages. The 802.11 standard, without 802.11v, does not include any mechanism for the controller or the access points to communicate to wireless clients about the session timeout for the local client.

To save the power of clients due to the mentioned tasks in wireless network, the following features in the 802.11v standard are used:

- Directed Multicast Service
- Base Station Subsystem (BSS) Max Idle Period

Directed Multicast Service

Using Directed Multicast Service (DMS), the client requests the access point to transmit the required multicast packet as unicast frames. This allows the client to receive the multicast packets it has ignored while in sleep mode and also ensures Layer 2 reliability. Furthermore, the unicast frame is transmitted to the client at a potentially higher wireless link rate which enables the client to receive the packet quickly by enabling the radio for a shorter duration, thus also saving battery power. Since the wireless client also does not have to wake up at each DTIM interval in order to receive multicast traffic, longer sleeping intervals are allowed.

BSS Max Idle Period

The BSS Max Idle period is the timeframe during which an access point (AP) does not disassociate a client due to nonreceipt of frames from the connected client. This helps ensure that the client device does not send keepalive messages frequently. The idle period timer value is transmitted using the association and reassociation response frame from the access point to the client. The idle time value indicates the maximum time that a client can remain idle without transmitting any frame to an access point. As a result, the clients remain in sleep mode for a longer duration without transmitting the keepalive messages often. This in turn contributes to saving battery power.

Prerequisites for Configuring 802.11v

- Applies for Apple clients like Apple iPad, iPhone, and so on, that run on Apple iOS version 7 or later.
- Supports local mode; also supports FlexConnect access points in central authentication modes only.

Restrictions for 802.11v

Client needs to support 802.11v BSS Transition.

Enabling 802.11v BSS Transition Management

802.11v BSS Transtion is applied in the following three scenarios:

- Solicited request—Client can send an 802.11v Basic Service Set (BSS) Transition Management Query before roaming for a better option of AP to reassociate with.
- Unsolicited Load Balancing request—If an AP is heavily loaded, it sends out an 802.11v BSS Transition Management Request to an associated client.
- Unsolicited Optimized Roaming request—If a client's RSSI and rate do not meet the requirements, the corresponding AP sends out an 802.11v BSS Transition Management Request to this client.

Note 802.11v BSS Transition Management Request is a suggestion (or advice) given to a client, which the client can choose to follow or ignore. To force the task of disassociating a client, turn on the disassociation-imminent function. This disassociates the client after a period if the client is not reassociated to another AP.

Configuring 802.11v BSS Transition Management (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add** to create WLANs.

The **Add WLAN** page is displayed.

Step 3 In the **Advanced** tab and **11v BSS Transition Support** section, select the **BSS Transition** check box to enable BSS transition per WLAN.

Step 4 Enable the **Dual Neighbor List** check box to include the neighbours of other radio slots of the same AP in the BSS transition response.

Note This is applicable only for 2.4 GHz and 5 GHz radio slots.

Step 5 Enable the **BSS Max Idle Service** check box to help clients and APs efficiently decide how long to remain associated when no traffic is being transmitted. The device uses this information to preserve device battery life.

Step 6 Enable the **BSS Max Idle Protected** check box to enable the AP to accept only authenticated frames (encrypted with Robust Security Network (RSN) information) from the client to reset the BSS Max Idle period counter. Without protected mode, any data or management frame (encrypted or unencrypted) sent by the client will reset the idle timer for the client.

Step 7 Enable the **Directed Multicast Service** check box to request the AP to send a multicast stream as unicast, to any DMS capable client on this WLAN.

Step 8 Click **Save & Apply to Device**.

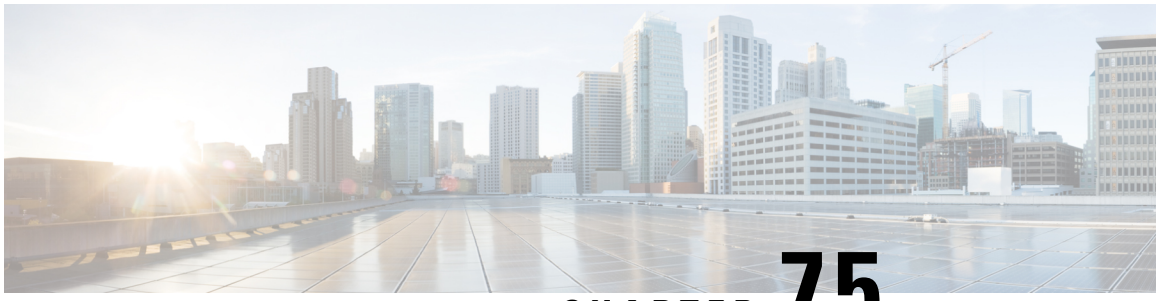
Configuring 802.11v BSS Transition Management (CLI)

802.11v BSS Transtion is applied in the following three scenarios:

Procedure

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

	Command or Action	Purpose
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan test-wlan`	Configures WLAN profile and enters the WLAN profile configuration mode.
Step 3	**shut** **Example:** `Device(config-wlan)# shut`	Shutdown the WLAN profile.
Step 4	**bss-transition** **Example:** `Device(config-wlan)# bss-transition`	Configure BSS transition per WLAN.
Step 5	**bss-transition disassociation-imminent** **Example:** `Device(config-wlan)# bss-transition disassociation-imminent`	Configure BSS transition disassociation Imminent per WLAN.
Step 6	**no shutdown** **Example:** `Device(config-wlan)# no shutdown`	Enables the WLAN profile.
Step 7	**end** **Example:** `Device(config-wlan)# end`	Return to privilege EXEC mode. Alternatively, you can press **CTRL + Z** to exit global configuration mode.



CHAPTER 75

802.11w

Information About 802.11w

Wi-Fi is a broadcast medium that enables any device to eavesdrop and participate either as a legitimate or rogue device. Management frames such as authentication, de-authentication, association, dissociation, beacons, and probes are used by wireless clients to initiate and tear down sessions for network services. Unlike data traffic, which can be encrypted to provide a level of confidentiality, these frames must be heard and understood by all clients and therefore must be transmitted as open or unencrypted. While these frames cannot be encrypted, they must be protected from forgery to protect the wireless medium from attacks. For example, an attacker could spoof management frames from an AP to attack a client associated with the AP.

The 802.11w protocol applies only to a set of robust management frames that are protected by the Protected Management Frames (PMF) service. These include Disassociation, De-authentication, and Robust Action frames.

Management frames that are considered as robust action and therefore protected are the following:

- Spectrum Management
- QoS
- DLS
- Block Ack
- Radio Measurement
- Fast BSS Transition
- SA Query
- Protected Dual of Public Action
- Vendor-specific Protected

When 802.11w is implemented in the wireless medium, the following occur:

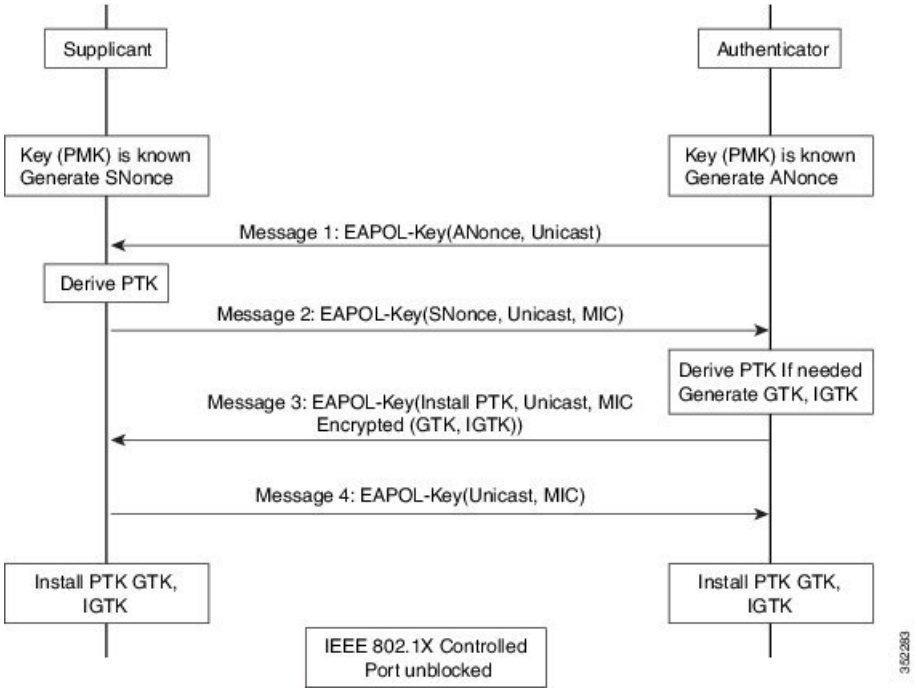
- Client protection is added by the AP adding cryptographic protection to de-authentication and dissociation frames preventing them from being spoofed in a DOS attack.
- Infrastructure protection is added by adding a Security Association (SA) tear down protection mechanism consisting of an Association Comeback Time and an SA-Query procedure preventing spoofed association request from disconnecting an already connected client.

802.11w has introduced a new IGTK Key, which is used to protect broadcast/multicast robust management frames:

- IGTK is a random value assigned by the authenticator STA (WLC) and used to protect MAC management protocol data units (MMPDUs) from that source STA.

When Management Frame Protection is negotiated, the AP encrypts the GTK and IGTK values in the EAPOL-Key frame, which is delivered in Message 3 of 4-way handshake.

Figure 26: IGTK Exchange in 4-way Handshake

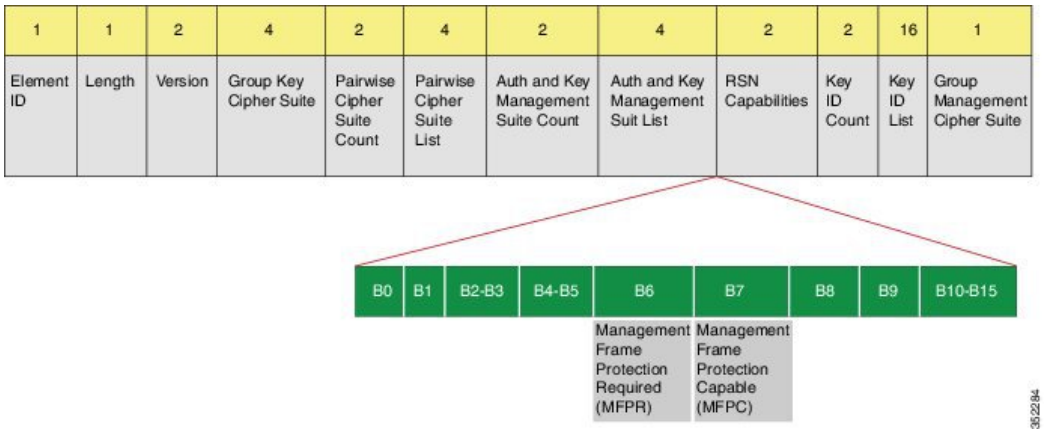


- If the AP later changes the GTK, it sends the new GTK and IGTK to the client using the Group Key Handshake .

802.11w defines a new Broadcast/Multicast Integrity Protocol (BIP) that provides data integrity and replay protection for broadcast/multicast robust management frames after successful establishment of an IGTKSA - It adds a MIC that is calculated using the shared IGTK key.

802.11w Information Elements (IEs)

Figure 27: 802.11w Information Elements

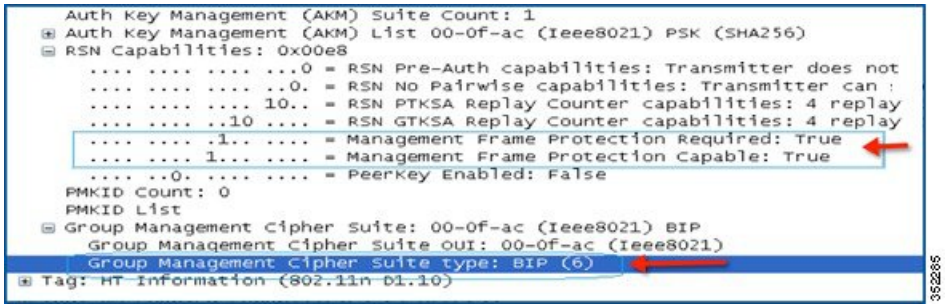


1. Modifications made in the RSN capabilities field of RSNIE.
   a. Bit 6: Management Frame Protection Required (MFPR)
   b. Bit 7: Management Frame Protection Capable (MFPC)
2. Two new AKM Suites, 5 and 6 are added for AKM Suite Selectors.
3. New Cipher Suite with type 6 is added to accommodate BIP.

The WLC adds this modified RSNIE in association and re-association responses and the APs add this modified RSNIE in beacons and probe responses.

The following Wireshark captures shows the RSNIE capabilities and the Group Management Cipher Suite elements.

Figure 28: 802.11w Information Elements



Security Association (SA) Teardown Protection

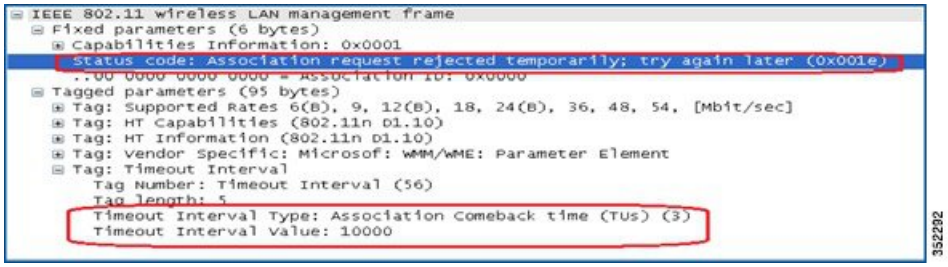
SA teardown protection is a mechanism to prevent replay attacks from tearing down the session of an existing client. It consists of an Association Comeback Time and an SA-Query procedure preventing spoofed association requests from disconnecting an already connected client.

If a client has a valid security association, and has negotiated 802.11w, the AP shall reject another Association Request with status code 30. This status code stands for "Association request rejected temporarily; Try again later". The AP should not tear down or otherwise modify the state of the existing association until the SA-Query

procedure determines that the original SA is invalid and shall include in the Association Response an Association Comeback Time information element, specifying a comeback time when the AP would be ready to accept an association with this client.

The following capture shows the Association Reject message with status code 0x1e (30) and the Association comeback time set to 10 seconds.

Figure 29: Association Reject with Comeback Time



Following this, if the AP is not already engaged in an SA Query with the client, the AP shall issue an SA Query until a matching SA Query response is received or the Association Comeback time expires. An AP may interpret reception of a valid protected frame as an indication of a successfully completed SA Query.

If a SA QUERY response with a matching transaction identifier within the time period, the AP shall allow the association process to be started without starting additional SA Query procedures.

Prerequisites for 802.11w

- To configure 802.11w feature for optional and mandatory, you must have WPA and AKM configured.

Note The RNS (Robust Secure Network) IE must be enabled with an AES Cipher.

Restrictions for 802.11w

- 802.11w cannot be applied on an open WLAN, WEP-encrypted WLAN, or a TKIP-encrypted WLAN.
- Cisco Catalyst 9800 Series Wireless Controller supports 802.11w + PMF combination for non-Apple clients. But Apple iOS version 11 and earlier require fix from the Apple iOS side to resolve the association issues.
- The controller will ignore disassociation or deauthentication frames sent by the clients if they are not using 802.11w PMF. The client entry will only get deleted immediately upon reception of such a frame if the client uses PMF. This is to avoid denial of service by malicious device since there is no security on those frames without PMF.

How to Configure 802.11w

Configuring 802.11w (GUI)

Before you begin

WPA and AKM must be configured.

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add** to create WLANs.

The **Add WLAN** page is displayed.

Step 3 In the **Security** > **Layer2** tab, navigate to the **Protected Management Frame** section.

Step 4 Choose **PMF** as *Disabled*, *Optional*, or *Required*. By default, the PMF is *disabled*.

If you choose **PMF** as *Optional* or *Required*, you get to view the following fields:

- **Association Comeback Timer**—Enter a value between 1 and 10 seconds to configure 802.11w association comeback time.
- **SA Query Time**—Enter a value between 100 to 500 (milliseconds). This is required for clients to negotiate 802.11w PMF protection on a WLAN.

Step 5 Click **Save & Apply to Device**.

Configuring 802.11w (CLI)

Before you begin

WPA and AKM must be configured.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name wlan-id ssid* **Example:** `Device(config)# wlan wlan-test 12 alpha`	Configures a WLAN and enters configuration mode.

	Command or Action	Purpose
Step 3	**security wpa akm pmf dot1x** **Example:** `Device(config-wlan)#security wpa akm pmf dot1x`	Configures 802.1x support.
Step 4	**security pmf association-comeback** *comeback-interval* **Example:** `Device(config-wlan)# security pmf association-comeback 10`	Configures the 802.11w association comeback time.
Step 5	**security pmf mandatory** **Example:** `Device(config-wlan)# security pmf mandatory`	Requires clients to negotiate 802.11w PMF protection on a WLAN.
Step 6	**security pmf saquery-retry-time** *timeout* **Example:** `Device(config-wlan)# security pmf saquery-retry-time 100`	Time interval identified in milliseconds before which the SA query response is expected. If the device does not get a response, another SQ query is tried.

Disabling 802.11w

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name wlan-id ssid* **Example:** `Device(config)# wlan wlan-test 12 alpha`	Configures a WLAN and enters configuration mode.
Step 3	**no security wpa akm pmf dot1x** **Example:** `Device(config-wlan)# no security wpa akm pmf dot1x`	Disables 802.1x support.
Step 4	**no security pmf association-comeback** *comeback-interval* **Example:** `Device(config-wlan)# no security pmf association-comeback 10`	Disables the 802.11w association comeback time.

	Command or Action	Purpose
Step 5	**no security pmf mandatory** **Example:** `Device(config-wlan)# no security pmf mandatory`	Disables client negotiation of 802.11w PMF protection on a WLAN.
Step 6	**no security pmf saquery-retry-time** *timeout* **Example:** `Device(config-wlan)# no security pmf saquery-retry-time 100`	Disables SQ query retry.

Monitoring 802.11w

Use the following commands to monitor 802.11w.

Procedure

Step 1 **show wlan name** *wlan-name*

Displays the WLAN parameters on the WLAN. The PMF parameters are displayed.

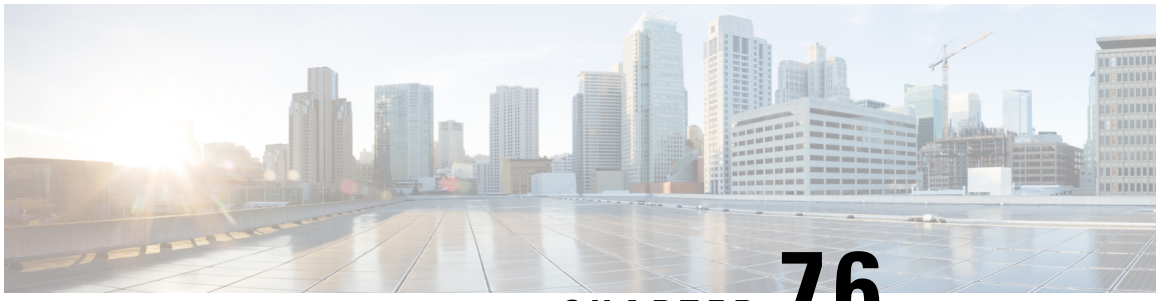
```
. . . .
. . . .
Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Disabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            FT SAE                              : Disabled
            Dot1x-SHA256                        : Enabled
            PSK-SHA256                          : Disabled
            SAE                                 : Disabled
            OWE                                 : Disabled
            SUITEB-1X                           : Disabled
            SUITEB192-1X                        : Disabled
    CCKM TSF Tolerance                          : 1000
    FT Support                                  : Adaptive
        FT Reassociation Timeout                : 20
        FT Over-The-DS mode                     : Enabled
    PMF Support                                 : Required
        PMF Association Comeback Timeout        : 1
        PMF SA Query Time                       : 500
. . . .
. . . .
```

Step 2 **show wireless client mac-address** *mac-address* **detail**

Displays the summary of the 802.11w authentication key management configuration on a client.

```
. . . .
. . . .
Policy Manager State: Run
```

```
NPU Fast Fast Notified : No
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 497 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : 802.1x-SHA256
Encrypted Traffic Analytics : No
Management Frame Protection : No
Protected Management Frame - 802.11w : Yes
EAP Type : LEAP
VLAN : 39
Multicast VLAN : 0
Access VLAN : 39
Anchor VLAN : 0
WFD capable : No
Manged WFD capable : No
. . . .
. . . .
```



CHAPTER 76

802.11ax Per WLAN

Information About 802.11ax Mode Per WLAN

Prior to Cisco IOS XE Bengaluru Release 17.4.1, the 802.11ax mode was configured per radio band. In this configuration, the 11ax mode was either enabled or disabled for all WLANs (AP) that were configured per radio, all at once. When 11ax was enabled per radio, the 11ac clients were not able to scan or connect to the SSID if the beacon had 11ax information elements. Client could not probe an access point (AP), if the beacon has 11ax IE.

Therefore, a 11ax configuration knob per AP is introduced, from Cisco IOS XE Bengaluru Release 17.5.1. This knob is introduced under the WLAN profile. By default, the 11ax knob per WLAN is now enabled on the controller.

Configuring 802.11ax Mode Per WLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.

Step 2 Click **Add**.
The **Add WLAN** window is displayed.

Step 3 Click the **Advanced** tab.

Step 4 In the **11ax** section, check the **Enable 11ax** check box to enable 802.11ax operation status on the WLAN.

Note When 11ax is disabled, beacons will not display 11ax IE, and all the 11ax features will be operationally disabled on the WLAN.

Step 5 Click **Apply to Device**.

Configuring 802.11ax Mode Per WLAN (CLI)

Procedure

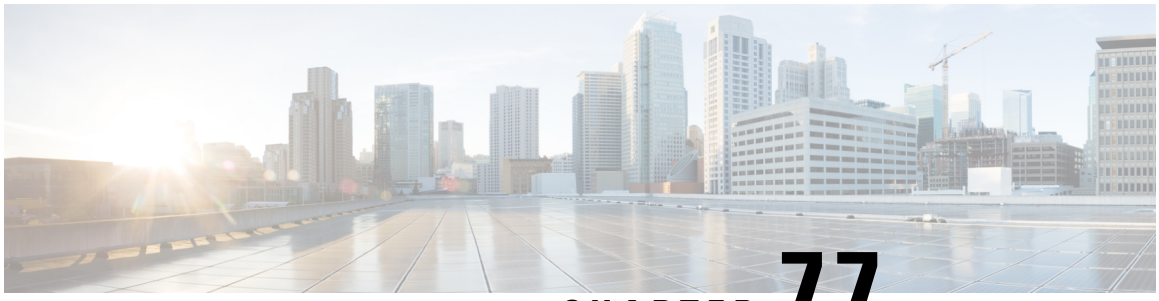
	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *wlan-profile-name* **Example:** `Device(config)# wlan wlan-profile`	Specifies the WLAN name and enters the WLAN configuration mode.
Step 3	**dot11ax** **Example:** `Device(config-wlan)# dot11ax`	Configures 802.11ax on a WLAN.
Step 4	**no dot11ax** **Example:** `Device(config-wlan)# no dot11ax`	Disables 802.11ax on the WLAN profile.

Verifying 802.11ax Mode Per WLAN

To display the status of the 11ax parameter, run the following command:

```
Device# show wlan id 6
WLAN Profile Name     : power
================================================
Identifier                                    : 6
Description                                   :
Network Name (SSID)                           : power
Status                                        : Enabled
Broadcast SSID                                : Enabled
Advertise-Apname                              : Disabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
.
.
.
802.11ac MU-MIMO                              : Enabled
802.11ax parameters
    802.11ax Operation Status                 : Enabled
    OFDMA Downlink                            : Enabled
    OFDMA Uplink                              : Enabled
    MU-MIMO Downlink                          : Enabled
    MU-MIMO Uplink                            : Enabled
    BSS Target Wake Up Time                   : Enabled
    BSS Target Wake Up Time Broadcast Support  : Enabled
```

.
.
.



CHAPTER 77

Deny Wireless Client Session Establishment Using Calendar Profiles

Information About Denial of Wireless Client Session Establishment

Denial of client session establishment feature allows the controller to stop client session establishment based on a particular time. This helps control the network in efficient and controlled manner without any manual intervention.

In Embedded Wireless Controller, you can deny the wireless client session based on the following recurrences:

- Daily
- Weekly
- Monthly

The Calendar Profiles created are then mapped to the policy profile. By attaching the calendar profile to a policy profile, you will be able to create different recurrences for the policy profile using different policy tag.

Note You need to create separate Calendar Profile for Daily, Weekly, and Monthly sub-categories.

The following is the workflow for denial of wireless client session establishment feature:

- Create a calendar profile.
- Apply the calendar profile to a policy profile.

Note A maximum of 100 calendar profile configuration and 5 calendar profile association to policy profile is supported.

Points to Remember

If you boot up your controller, the denial of client session establishment feature kicks in after a minute from the system boot up.

If you change the system time after the calendar profile is associated to a policy profile, you can expect a maximum of 30 second delay to adapt to the new clock timings.

Note You cannot use the **no action deny-client** command to disable action while associating the calendar profile to a policy profile.

If you want to disable the action command, you need to disassociate the calendar profile from the policy profile, and re-configure again.

Configuring Daily Calendar Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile calendar-profile name** *name* **Example:** `Device(config)# wireless profile calendar-profile name daily_calendar_profile`	Configures a calendar profile. Here, *name* refers to the name of the calendar profile.
Step 3	**start** *start_time* **end** *end_time* **Example:**	Configures start and end time for the calendar profile.

	Command or Action	Purpose
	`Device(config-calendar-profile)# start 09:00:00 end 17:00:00`	Here, *start_time* is the start time for the calendar profile. You need to enter start time in **HH:MM:SS** format. *end_time* is the end time for the calendar profile. You need to enter end time in **HH:MM:SS** format.
Step 4	**recurrence daily** **Example:** `Device(config-calendar-profile)# recurrence daily`	Configures daily recurrences for a calendar profile.
Step 5	**end** **Example:** `Device(config-calendar-profile)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode. **Note** When the calendar profile kicks in, the AP power profile rules (for example, radio state and USB device state) that are defined for the Ethernet speed are not applied and continue to be as per the fixed power profile.

Configuring Weekly Calendar Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile calendar-profile name** *name* **Example:** `Device(config)# wireless profile calendar-profile name weekly_calendar_profile`	Configures a calendar profile. Here, *name* refers to the name of the calendar profile.
Step 3	**start** *start_time* **end** *end_time* **Example:** `Device(config-calendar-profile)# start 18:00:00 end 19:00:00`	Configures start and end time for the calendar profile. Here,

	Command or Action	Purpose
		start_time is the start time for the calendar profile. You need to enter start time in **HH:MM:SS** format. *end_time* is the end time for the calendar profile. You need to enter end time in **HH:MM:SS** format.
Step 4	**recurrence weekly** **Example:** `Device(config-calendar-profile)# recurrence weekly`	Configures weekly recurrences for the calendar profile.

Step 5	**day** {**friday** | **monday** | **saturday** | **sunday** | **thursday** | **tuesday** | **wednesday**}
	Example:
	`Device(config-calendar-profile)# day friday`
	`Device(config-calendar-profile)# day monday`
		Configure days when the weekly calendar needs to be active.
		Note You can configure multiple days using this command.

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

Configuring Monthly Calendar Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile calendar-profile name** *name* **Example:** `Device(config)# wireless profile calendar-profile name monthly_calendar_profile`	Configures a calendar profile. Here, *name* refers to the name of the calendar profile.
Step 3	**start** *start_time* **end** *end_time* **Example:** `Device(config-calendar-profile)# start 18:00:00 end 19:00:00`	Configures start and end time for the calendar profile. Here,

	Command or Action	Purpose
		start_time is the start time for the calendar profile. You need to enter start time in **HH:MM:SS** format. *end_time* is the end time for the calendar profile. You need to enter end time in **HH:MM:SS** format.
Step 4	**recurrence monthly** **Example:** `Device(config-calendar-profile)# recurrence monthly`	Configures monthly recurrences for the calendar profile.
Step 5	**date** *value* **Example:** `Device(config-calendar-profile)# date 25`	Configures a date for the calendar profile. **Note** If the requirement is to perform denial of service in certain timing, such as, 2,10, and 25 of every month, all three days need to be configured using the **date** command. There is no range for date. You need to configure the dates as per your requirement.
Step 6	**end** **Example:** `Device(config-calendar-profile)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Mapping a Daily Calendar Profile to a Policy Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-name* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Creates policy profile for the WLAN. The *profile-name* is the profile name of the policy profile.
Step 3	**calender-profile name** *calendar-profile-name* **Example:**	Maps a calender profile to a policy profile.

	Command or Action	Purpose
	`Device(config-wireless-policy)# calender-profile name daily_calendar_profile`	The *calendar-profile-name* is the name of the calendar profile name created in Configuring Daily Calendar Profile, on page 782. **Note** You need to disable Policy Profile before associating a calendar profile to a policy profile. The following needs to be done: `Device(config-wireless-policy)# shutdown`
Step 4	**action deny-client** **Example:** `Device(config-policy-profile-calender)# action deny-client`	Configures deny client session establishment during calendar profile interval. **Note** Client associations are denied daily between timeslot 9:00:00 to 17:00:00. For start and end time details, see Configuring Daily Calendar Profile, on page 782.
Step 5	**end** **Example:** `Device(config-policy-profile-calender)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Mapping a Weekly Calendar Profile to a Policy Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-name* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Creates policy profile for the WLAN. The *profile-name* is the profile name of the policy profile.
Step 3	**calender-profile name** *calendar-profile-name* **Example:** `Device(config-wireless-policy)# calender-profile name weekly_calendar_profile`	Maps a calender profile to a policy profile. The *calendar-profile-name* is the name of the calendar profile name created in Configuring Weekly Calendar Profile, on page 783.

	Command or Action	Purpose
		Note You need to disable Policy Profile before associating a calendar profile to a policy profile. The following needs to be done: `Device(config-wireless-policy)# shutdown`
Step 4	**action deny-client** **Example:** `Device(config-policy-profile-calender)# action deny-client`	Configures deny client session establishment during calendar profile interval. **Note** Client associations are denied daily between timeslot 9:00:00 to 17:00:00. For start and end time details, see Configuring Weekly Calendar Profile, on page 783. On Monday and Tuesday, clients are denied between 17:30:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00. On 25th of every month, clients are denied between 18:00:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00.
Step 5	**end** **Example:** `Device(config-policy-profile-calender)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Mapping a Monthly Calendar Profile to a Policy Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-name* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Creates policy profile for the WLAN. The *profile-name* is the profile name of the policy profile.
Step 3	**calender-profile name** *calendar-profile-name*	Maps a calender profile to a policy profile.

	Command or Action	Purpose
	Example: `Device(config-wireless-policy)# calender-profile name monthly_calendar_profile`	The *calendar-profile-name* is the name of the calendar profile name created in Configuring Monthly Calendar Profile, on page 784.
Step 4	**action deny-client** **Example:** `Device(config-policy-profile-calender)# action deny-client`	Configures deny client session establishment for the defined calendar profile interval. **Note** Every day client associations are denied between timeslot 9:00:00 to 17:00:00. For start and end time details, see Configuring Monthly Calendar Profile, on page 784. On Monday and Tuesday, clients are denied between 17:30:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00. On 25th of every month, clients are denied between 18:00:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00.
Step 5	**end** **Example:** `Device(config-policy-profile-calender)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Calendar Profile Configuration

To view the summary of calendar profiles, use the following command:

```
Device# show wireless profile calendar-profile summary
Number of Calendar Profiles: 3

Profile-Name
---------------------------------
monthly_25_profile
weekly_mon_profile
daily_calendar_profile
```

To view the calendar profile details for a specific profile name, use the following command:

```
Device# show wireless profile calendar-profile detailed daily_calendar_profile
Calendar profiles                          : daily_calendar_profile
-------------------------------------------------------------------
Recurrence                                 : DAILY
Start Time                                 : 09:00:00
End Time                                   : 17:00:00
```

Verifying Policy Profile Configuration

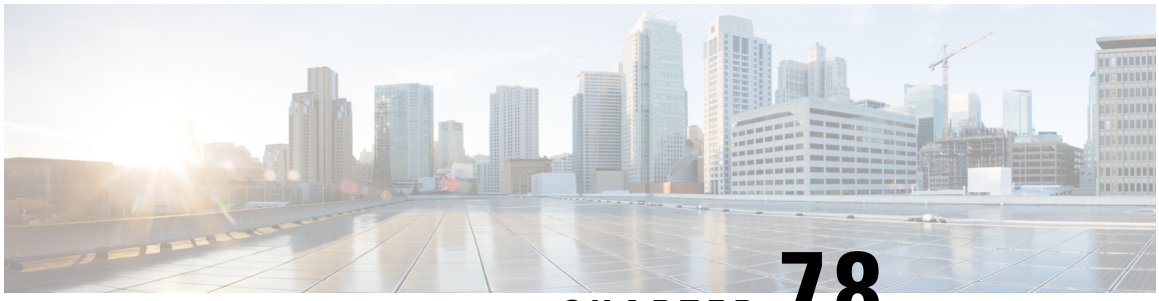
To view the detailed parameters for a specific policy profile, use the following command:

```
Device# show wireless profile policy detailed default-policy-profile
Tunnel Profile
  Profile Name                      : Not Configured
Calendar Profile
  Profile Name                      : monthly_25_profile
  Wlan Enable                       : Not Configured
  Client Block                      : Client Block Configured
  ---------------------------------------------------
  Profile Name                      : weekly_mon_profile
  Wlan Enable                       : Not Configured
  Client Block                      : Client Block Configured
  ---------------------------------------------------
  Profile Name                      : daily_calendar_profile
  Wlan Enable                       : Not Configured
  Client Block                      : Client Block Configured
  ---------------------------------------------------
Fabric Profile
  Profile Name                      : Not Configured
```

To view the configured calendar profile information under policy profile, use the following command:

```
Device# show wireless profile policy all
Tunnel Profile
Profile Name : Not Configured
Calendar Profile
Profile Name : daily_calendar_profile
Wlan Enable : Not Configured
Client Block : Client Block Configured
---------------------------------------------------
Profile Name : weekly_calendar_profile
Wlan Enable : Not Configured
Client Block : Client Block Configured
---------------------------------------------------
Fabric Profile
Profile Name : Not Configured
```

Note The anchor priority is always displayed as local. Priorities can be assigned on the foreign controller.



CHAPTER 78

Ethernet over GRE Tunnels

Introduction to EoGRE

Ethernet over GRE (EoGRE) is an aggregation solution for grouping Wi-Fi traffic from hotspots. This solution enables customer premises equipment (CPE) devices to bridge the Ethernet traffic coming from an end-host, and encapsulate the traffic in Ethernet packets over an IP Generic Routing Encapsulation (GRE) tunnel. When the IP GRE tunnels are terminated on a service provider's broadband network gateway, the end-host traffic is forwarded and subscriber sessions are initiated.

Client IPv6

EoGRE for WLAN

To enable EoGRE for a WLAN, the wireless policy profile should be mapped to a tunnel profile, which may contain the following:

- AAA override: Allows you to bypass rule filtering for a client.
- Gateway RADIUS proxy: Allows forwarding of AAA requests to tunnel gateways.
- Tunnel rules: Defines the domain to use for each realm. They also define VLAN tagging for the client traffic towards tunnel gateways.
- DHCP option 82: Provides a set of predefined fields.

EoGRE Deployment with Multiple Tunnel Gateways

The embedded wireless controller sends keepalive pings to the primary and secondary tunnel gateways and keeps track of the missed pings. When a certain threshold level is reached for the missed pings, switchover

is performed and the secondary tunnel is marked as active. This switchover deauthenticates all the clients to enable them to rejoin the access points (APs). When the primary tunnel come back online, all the client traffic are reverted to the primary tunnel. However, this behavior depends on the type of redundancy.

Load Balancing in EtherChannels

Load balancing of tunneled traffic over Etherchannels works by hashing the source or destination IP addresses or mac addresses of the tunnel endpoint pair. Because the number of tunnels is very limited when compared to clients (each tunnel carries traffic for many clients), the spreading effect of hashing is highly reduced and optimal utilization of Etherchannel links can be hard to achieve.

Using the EoGRE configuration model, you can use the *tunnel source* option of each tunnel interface to adjust the load-balancing parameters and spread tunnels across multiple links.

You can use different source interfaces on each tunnel for load balancing based on the source or destination IP address. For that choose the source interface IP address in such a way that traffic flows take different links for each src-dest IP pair. The following is an example with four ports:

```
Client traffic on Tunnel1 – Src IP: 40.143.0.72  Dest IP: 40.253.0.2
Client traffic on Tunnel2 – Src IP: 40.146.0.94  Dest IP: 40.253.0.6
Client traffic on Tunnel3 – Src IP: 40.147.0.74  Dest IP: 40.253.0.10
```

Use the **show platform software port-channel link-select interface port-channel 4 ipv4** *src_ip dest_ip* command to determine the link that a particular flow will take.

EoGRE Configuration Overview

The EoGRE solution can be deployed in two different ways:

- Central-Switching: EoGRE tunnels connect the embedded wireless controller to the tunnel gateways.
- Flex or Local-Switching: EoGRE tunnels are initiated on the APs and terminated on the tunnel gateways.

To configure EoGRE, perform the following tasks:

1. Create a set of tunnel gateways.
2. Create a set of tunnel domains.
3. Create a tunnel profile with rules that define how to match clients to domains.
4. Create a policy profile and attach the tunnel profile to it.
5. Map the policy profile to WLANs using policy tags.

Note The EoGRE tunnel fallback to the secondary tunnel is triggered after the *max-skip-count* ping fails in the last measurement window. Based on the starting and ending instance of the measurement window, the fall-back may take more time than the duration that is configured.

Table 37: EoGRE Authentication Methods

Method Name	First Supported Release	Mode
PSK	17.2.1	Local/Flex (central authentication)
Open	16.12.1	Local/Flex (central authentication)
LWA	16.12.1	Local/Flex (central authentication)
Dot1x	16.12.1	Local/Flex (central authentication)
CWA	16.12.1	Local/Flex (central authentication)

Create a Tunnel Gateway

Note In the Cisco Embedded Wireless Controller on Catalyst Access Points, a tunnel gateway is modeled as a tunnel interface.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**interface tunnel** *tunnel_number* **Example:** `Device(config)# interface tunnel 21`	Configures a tunnel interface and enters interface configuration mode.
Step 3	**tunnel source** *source_intf* **Example:** `Device(config-if)# tunnel source 22`	Sets the source address of the tunnel interface. The source interface can be VLAN, Gigabit Ethernet or loopback.
Step 4	**tunnel destination** *tunnel-address* **Example:** `Device(config-if)# tunnel destination 10.11.12.13`	Sets the destination address of the tunnel.
Step 5	**tunnel mode ethernet gre** {**ipv4** | **ipv6**} **p2p** **Example:** `Device(config-if)# tunnel mode ethernet gre ipv4 p2p`	Sets the encapsulation mode of the tunnel to Ethernet over GRE IPv4 or Ethernet over GRE IPv6.

Configuring a Tunnel Domain

Note Tunnel domains are a redundancy grouping of tunnels. The following configuration procedure specifies a primary and a secondary tunnel, along with a redundancy model.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**tunnel eogre domain** *domain* **Example:** `Device(config)# tunnel eogre domain dom1`	Configures EoGRE redundancy domain.
Step 3	**primary tunnel** *primary-tunnel_intf* **Example:** `Device(config-eogre-domain)# primary tunnel 21`	Configures the primary tunnel.
Step 4	**secondary tunnel** *secondary-tunnel_intf* **Example:** `Device(config-eogre-domain)# secondary tunnel 22`	Configures the secondary tunnel.
Step 5	**redundancy revertive** **Example:** `Device(config-eogre-domain)# redundancy revertive`	Sets the redundancy model as revertive. When redundancy is set to revertive and the primary tunnel goes down, a switchover to secondary tunnel is performed. When the primary tunnel comes back up, a switchover to the primary tunnel is performed, because the primary tunnel has priority over the secondary tunnel. When redundancy is not set to revertive, tunnels will have the same priority, and a switchover to the primary tunnel is not performed if the active tunnel is the secondary tunnel and the primary tunnel comes back up.

Configuring EoGRE Global Parameters

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**tunnel eogre heartbeat interval** *interval-value* **Example:** `Device(config)# tunnel eogre heartbeat interval 600`	Sets EoGRE tunnel heartbeat periodic interval.
Step 3	**tunnel eogre heartbeat max-skip-count** *skip-count* **Example:** `Device(config)# tunnel eogre heartbeat max-skip-count 7`	Sets the maximum number of tolerable dropped heartbeats. After reaching the maximum number of heartbeats that can be dropped, the tunnel is declared as down and a switchover is performed.
Step 4	**tunnel eogre source loopback** *tunnel_source* **Example:** `Device(config)# tunnel eogre source loopback 12`	Sets the tunnel EoGRE source interface.
Step 5	**tunnel eogre interface tunnel** *tunnel-intf* **aaa proxy key** *key key-name* **Example:** `Device(config)# tunnel eogre interface tunnel 21 aaa proxy key 0 mykey`	(Optional) Configures AAA proxy RADIUS key for the AAA proxy setup. **Note** When the tunnel gateway is behaving as the AAA proxy server, only this step is required for the configuration.

Configuring a Tunnel Profile

Before you begin

Ensure that you define the destination VLAN on the controller. If you do not define the VLAN, clients will not be able to connect.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-policy-name* **Example:** `Device(config)# wireless profile policy eogre_policy`	Configures a WLAN policy profile.
Step 3	**tunnel-profile** *tunnel-profile-name* **Example:** `Device(config-wireless-policy)# tunnel-profile tunnel1`	Creates a tunnel profile.
Step 4	**exit** **Example:** `Device(config-wireless-policy)# exit`	Returns to global configuration mode.
Step 5	**wireless profile tunnel** *tunnel-profile-name* **Example:** `Device(config)# wireless profile tunnel wl-tunnel-1`	Configures a wireless tunnel profile.
Step 6	**dhcp-opt82 enable** **Example:** `Device(config-tunnel-profile)# dhcp-opt82 enable`	Activates DHCP Option 82 for the tunneled clients.
Step 7	**dhcp-opt82 remote-id** *remote-id* **Example:** `Device(config-tunnel-profile)# dhcp-opt82 remote-id vlan`	Configures Remote ID options. Choose from the comma-separated list of options such as **ap-mac**, **ap-ethmac**, **ap-name**, **ap-group-name**, **flex-group-name**, **ap-location**, **vlan**, **ssid-name**, **ssid-type**, and **client-mac**.
Step 8	**aaa-override** **Example:** `Device(config-tunnel-profile)# aaa-override`	Enables AAA policy override.
Step 9	**gateway-radius-proxy** **Example:** `Device(config-tunnel-profile)# gateway-radius-proxy`	Enables the gateway RADIUS proxy.

	Command or Action	Purpose
Step 10	**gateway-accounting-radius-proxy** **Example:** `Device(config-tunnel-profile)# gateway-accounting-radius-proxy`	Enables the gateway accounting RADIUS proxy.
Step 11	**rule** *priority* **realm-filter** *realm* **domain** *domain-name* **vlan** *vlan-id* **Example:** `Device(config-tunnel-profile)# rule 12 realm-filter realm domain dom1 vlan 5`	Creates a rule to choose a domain, using the realm filter, for client Network Access Identifier (NAI), tunneling domain name, and destination VLAN.

Associating WLAN to a Wireless Policy Profile

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless tag policy** *policy-tag-name* **Example:** `Device(config)# wireless tag policy eogre_tag`	Configures a policy tag and enters policy tag configuration mode.
Step 3	**wlan** *wlan-name* **policy** *profile-policy-name* **Example:** `Device(config-policy-tag)# wlan eogre_open_eogre policy eogre_policy`	Maps an EoGRE policy profile to a WLAN profile.
Step 4	**end** **Example:** `Device(config-policy-tag)# end`	Saves the configuration, exits configuration mode, and returns to privileged EXEC mode.

Attaching a Policy Tag and a Site Tag to an AP

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**ap** *mac-address* **Example:** `Device(config)# ap 80E8.6FD4.0BB0`	Configures an AP and enters AP profile configuration mode.
Step 3	**policy-tag** *policy-tag-name* **Example:** `Device(config-ap-tag)# policy-tag eogre_tag`	Maps the EoGRE policy tag to the AP.
Step 4	**site-tag** *site-tag-name* **Example:** `Device(config-ap-tag)# site-tag sp-flex-site`	Maps a site tag to the AP.
Step 5	**end** **Example:** `Device(config-ap-tag)# end`	Saves the configuration, exits configuration mode, and returns to privileged EXEC mode.

Verifying the EoGRE Tunnel Configuration

The `show tunnel eogre` command displays the EoGRE clients, domains, gateways, global-configuration, and manager information in the local mode.

To display the EoGRE domain summary in the local mode, use the following command:

```
Device# show tunnel eogre domain summary

Domain Name          Primary GW   Secondary GW      Active GW      Redundancy
------------------------------------------------------------------------------
domain1              Tunnel1      Tunnel2           Tunnel1        Non-Revertive
eogre_domain         Tunnel1      Tunnel2           Tunnel1        Non-Revertive
```

To display the details of an EoGRE domain in the local mode, use the following command:

```
Device# show tunnel eogre domain detailed domain-name

Domain Name    : eogre_domain
 Primary GW    : Tunnel1
 Secondary GW  : Tunnel2
 Active GW     : Tunnel1
 Redundancy    : Non-Revertive
```

To view the EoGRE tunnel gateway summary and statistics in the local mode, use the following command:

```
Device# show tunnel eogre gateway summary

Name              Type  Address                                 AdminState  State  Clients
-------------------------------------------------------------------------------------------
Tunnel1            IPv4  9.51.1.11                               Up          Up     0
```

```
Tunnel2            IPv4  9.51.1.12                          Up          Down   0

Tunnel10           IPv6  fd09:9:8:21::90                    Down        Down   0

Tunnel11           IPv4  9.51.1.11                          Up          Up     0

Tunnel12           IPv6  fd09:9:8:21::90                    Up          Down   0

Tunnel100          IPv4  9.51.1.100                         Up          Down   0
```

To view the details of an EoGRE tunnel gateway in the local mode, use the following command:

```
Device# show tunnel eogre gateway detailed gateway-name

Gateway : Tunnel1
 Mode   : IPv4
 IP     : 9.51.1.11
 Source : Vlan51 / 9.51.1.1
 State  : Up
 SLA ID : 56
 MTU    : 1480
 Up Time: 4 minutes 45 seconds

 Clients
  Total Number of Wireless Clients      : 0
 Traffic
  Total Number of Received Packets      : 0
  Total Number of Received Bytes        : 0
  Total Number of Transmitted Packets   : 0
  Total Number of Transmitted Bytes     : 0
 Keepalives
  Total Number of Lost Keepalives       : 0
  Total Number of Received Keepalives   : 5
  Total Number of Transmitted Keepalives: 5
  Windows                               : 1
  Transmitted Keepalives in last window : 2
  Received Keepalives in last window    : 2
```

To view the client summary of EoGRE in the local mode, use the following command:

```
Device# show tunnel eogre client summary

Client MAC      AP MAC          Domain                              Tunnel        VLAN  Local
-------------------------------------------------------------------------------------------
74da.3828.88b0  80e8.6fd4.9520  eogre_domain                         N/A           2121   No
```

To view the details of an EoGRE global configuration in the local mode, use the following command:

```
Device# show tunnel eogre global-configuration

Heartbeat interval       : 60
Max Heartbeat skip count : 3
Source Interface         : (none)
```

To view the details of the global tunnel manager statistics in the local mode, use the following command:

```
Device# show tunnel eogre manager stats global

Tunnel Global Statistics
Last Updated                    : 02/18/2019 23:50:35
EoGRE Objects
```

```
  Gateways                       : 6
  Domains                        : 2

EoGRE Flex Objects
  AP Gateways                    : 2
  AP Domains                     : 1
  AP Gateways HA inconsistencies : 0
  AP Domains HA inconsistencies  : 0

Config events
  IOS Tunnel updates             : 806
  IOS Domain updates             : 88
  Global updates                 : 48
  Tunnel Profile updates         : 120
  Tunnel Rule updates            : 16
  AAA proxy key updates          : 0

AP events
  Flex AP Join                   : 1
  Flex AP Leave                  : 0
  Local AP Join                  : 0
  Local AP leave                 : 0
  Tunnel status (rx)             : 4
  Domain status (rx)             : 1
  IAPP stats msg (rx)            : 3
  Client count (rx)              : 6
  VAP Payload msg (tx)           : 4
  Domain config (tx)             : 1
  Global config (tx)             : 1
  Client delete (tx)             : 1
  Client delete per domain (tx)  : 3
  DHCP option 82 (tx)            : 4

Client events
  Add-mobile                     : 2
  Run-State                      : 3
  Delete                         : 1
  Cleanup                        : 0
  Join                           : 2
  Plumb                          : 0
  Join Errors                    : 0
  HandOff                        : 0
  MsPayload                      : 2
  FT Recover                     : 0
  Zombie GW counter increase     : 0
  Zombie GW counter decrease     : 0
  Tunnel Profile reset           : 88
  Client deauth                  : 0
  HA reconciliation              : 0

Client Join Events
  Generic Error                  : 0
  MSPayload Fail                 : 0
  Invalid VLAN                   : 0
  Invalid Domain                 : 0
  No GWs in Domain               : 0
  Domain Shut                    : 0
  Invalid GWs                    : 0
  GWs Down                       : 0
  Rule Match Error               : 0
  AAA-override                   : 0
  Flex No Active GW              : 0
  Open Auth join attempt         : 2
  Dot1x join attempt             : 2
```

```
  Mobility join attempt          : 0
  Tunnel Profile not valid       : 2
  Tunnel Profile valid           : 2
  No rule match                  : 0
  Rule match                     : 2
  AAA proxy                      : 0
  AAA proxy accounting           : 0
  AAA eogre attributes           : 0
  Has aaa override               : 0
  Error in handoff payload       : 0
  Handoff AAA override           : 0
  Handoff no AAA override        : 0
  Handoff payload received       : 0
  Handoff payload sent           : 0

SNMP Traps
  Client                         : 0
  Tunnel                         : 2
  Domain                         : 0

IPC
  IOSd TX messages               : 0

Zombie Client
  Entries                        : 0
```

To view the tunnel manager statistics of a specific process instance in the local mode, use the following command:

```
Device# show tunnel eogre manager stats instance instance-number

Tunnel Manager statistics for process instance : 0
Last Updated                     : 02/18/2019 23:50:35
EoGRE Objects
  Gateways                       : 6
  Domains                        : 2

EoGRE Flex Objects
  AP Gateways                    : 2
  AP Domains                     : 1
  AP Gateways HA inconsistencies : 0
  AP Domains HA inconsistencies  : 0

Config events
  IOS Tunnel updates             : 102
  IOS Domain updates             : 11
  Global updates                 : 6
  Tunnel Profile updates         : 15
  Tunnel Rule updates            : 2
  AAA proxy key updates          : 0

AP events
  Flex AP Join                   : 1
  Flex AP Leave                  : 0
  Local AP Join                  : 0
  Local AP leave                 : 0
  Tunnel status (rx)             : 4
  Domain status (rx)             : 1
  IAPP stats msg (rx)            : 3
  Client count (rx)              : 6
  VAP Payload msg (tx)           : 4
  Domain config (tx)             : 1
  Global config (tx)             : 1
```

```
  Client delete (tx)              : 1
  Client delete per domain (tx)   : 3
  DHCP option 82 (tx)             : 4

Client events
  Add-mobile                      : 2
  Run-State                       : 3
  Delete                          : 1
  Cleanup                         : 0
  Join                            : 2
  Plumb                           : 0
  Join Errors                     : 0
  HandOff                         : 0
  MsPayload                       : 2
  FT Recover                      : 0
  Zombie GW counter increase      : 0
  Zombie GW counter decrease      : 0
  Tunnel Profile reset            : 11
  Client deauth                   : 0
  HA reconciliation               : 0

Client Join Events
  Generic Error                   : 0
  MSPayload Fail                  : 0
  Invalid VLAN                    : 0
  Invalid Domain                  : 0
  No GWs in Domain                : 0
  Domain Shut                     : 0
  Invalid GWs                     : 0
  GWs Down                        : 0
  Rule Match Error                : 0
  AAA-override                    : 0
  Flex No Active GW               : 0
  Open Auth join attempt          : 2
  Dot1x join attempt              : 2
  Mobility join attempt           : 0
  Tunnel Profile not valid        : 2
  Tunnel Profile valid            : 2
  No rule match                   : 0
  Rule match                      : 2
  AAA proxy                       : 0
  AAA proxy accounting            : 0
  AAA eogre attributes            : 0
  Has aaa override                : 0
  Error in handoff payload        : 0
  Handoff AAA override            : 0
  Handoff no AAA override         : 0
  Handoff payload received        : 0
  Handoff payload sent            : 0

SNMP Traps
  Client                          : 0
  Tunnel                          : 2
  Domain                          : 0

IPC
  IOSd TX messages                : 0

Zombie Client
  Entries                         : 0
```

The `show ap tunnel eogre` command displays the tunnel domain information, EoGRE events, and the tunnel gateway status on the APs, in the flex mode.

To view the summary information of an EoGRE tunnel gateway in the flex mode, use the following command:

```
Device# show ap tunnel eogre domain summary

AP MAC          Domain                          Active Gateway
-----------------------------------------------------------------------------
80e8.6fd4.9520  eogre_domain                    Tunnel1
```

To view the wireless tunnel profile summary, use the following command:

```
Device# show wireless profile tunnel summary

Profile Name                    AAA-Override AAA-Proxy DHCP Opt82 Enabled
------------------------------- ------------ --------- ---------- ---------
eogre_tunnel                             No        No        Yes       Yes
eogre_tunnel_set                         No        No        Yes        No
eogre_tunnel_snmp                        No        No         No        No
```

To view a wireless tunnel profile's details, use the following command:

```
Device# show wireless profile tunnel detailed profile-name

Profile Name : eogre_tunnel
Status : Enabled
 AAA-Proxy/Accounting-Proxy: Disabled / Disabled
 AAA-Override : Disabled
 DHCP Option82 : Enabled
  Circuit-ID : ap-mac,ap-ethmac,ap-location,vlan
  Remote-ID : ssid-name,ssid-type,client-mac,ap-name

Tunnel Rules

Priority Realm                Vlan Domain (Status/Primary GW/Secondary GW)
-------- -------------------- ---- ------------------------------------------
1        *                    2121 eogre_domain (Enabled/Tunnel1/Tunnel2)
```

To view detailed information about an EoGRE tunnel domain's status, use the following command:

```
Device# show ap tunnel eogre domain detailed

Domain     : eogre_domain
AP MAC     : 80e8.6fd4.9520
Active GW  : Tunnel1
```

To view the EoGRE events on an AP, use the following command:

```
Device# show ap tunnel eogre events

AP 80e8.6fd4.9520  Event history
Timestamp               #Times   Event               RC Context
----------------------- -------- ------------------- --
----------------------------------------
02/18/2019 23:50:26.341 6        IAPP_STATS           0 GW Tunnel2 uptime:0s

02/18/2019 23:49:40.222 2        CLIENT_JOIN         0 74da.3828.88b0, (eogre_domain/2121)

02/18/2019 23:48:43.549 1        CLIENT_LEAVE        0 74da.3828.88b0, (eogre_domain/2121)

02/18/2019 23:47:33.127 1        DOMAIN_STATUS        0 eogre_domain Active GW: Tunnel1

02/18/2019 23:47:33.124 4        AP_TUNNEL_STATUS     0 Tunnel2 Dn
```

```
02/18/2019 23:47:33.124 1        MSG_CLIENT_DEL       0 GW Tunnel2 (IP: 9.51.1.12)

02/18/2019 23:47:33.124 2        TUNNEL_ADD           0 GW Tunnel2

02/18/2019 23:47:33.120 3        MSG_CLIENT_DEL_PD    0 GW Tunnel1 (IP: 9.51.1.11)

02/18/2019 23:47:31.763 2        AP_DOMAIN_PUSH       0 Delete:eogre_domain_set, 0 GWs

02/18/2019 23:47:31.753 4        AP_VAP_PUSH          0 profile:'eogre_tunnel',
wlan:pyats_eogre
```

To view the summary information of the EoGRE tunnel gateway, use the following command:

```
Device# show ap tunnel eogre gateway summary

AP MAC          Gateway         Type  IP                                        State  Clients
-----------------------------------------------------------------------------------------------
80e8.6fd4.9520  Tunnel1          IPv4  9.51.1.11                                   Up     1

80e8.6fd4.9520  Tunnel2          IPv4  9.51.1.12                                   Down   0
```

To view detailed information about an EoGRE tunnel gateway, use the following command:

```
Device# show ap tunnel eogre gateway detailed gateway-name

Gateway : Tunnel1
 Mode   : IPv4
 IP     : 9.51.1.11
 State  : Up
 MTU    : 1476
 Up Time: 14 hours 25 minutes 2 seconds
 AP MAC : 80e8.6fd4.9520

 Clients
  Total Number of Wireless Clients      : 1
 Traffic
  Total Number of Received Packets      : 6
  Total Number of Received Bytes        : 2643
  Total Number of Transmitted Packets   : 94
  Total Number of Transmitted Bytes     : 20629
  Total Number of Lost Keepalive        : 3
```

To view summary information about the EoGRE tunnel gateway status, use the following command:

```
Device# show ap tunnel eogre domain summary

AP MAC          Domain                          Active Gateway
-------------------------------------------------------------------------------
80e8.6fd4.9520  eogre_domain                    Tunnel1
```

To view information about EoGRE events on an AP, use the following command:

```
Device# show ap name ap-name tunnel eogre events

AP 80e8.6fd4.9520  Event history
Timestamp               #Times   Event               RC Context
----------------------- -------- ------------------- --
----------------------------------------
02/18/2019 23:50:26.341 6        IAPP_STATS           0 GW Tunnel2 uptime:0s

02/18/2019 23:49:40.222 2        CLIENT_JOIN         0 74da.3828.88b0, (eogre_domain/2121)

02/18/2019 23:48:43.549 1        CLIENT_LEAVE        0 74da.3828.88b0, (eogre_domain/2121)
```

```
02/18/2019 23:47:33.127 1        DOMAIN_STATUS        0 eogre_domain Active GW: Tunnel1

02/18/2019 23:47:33.124 4        AP_TUNNEL_STATUS     0 Tunnel2 Dn

02/18/2019 23:47:33.124 1        MSG_CLIENT_DEL       0 GW Tunnel2 (IP: 9.51.1.12)

02/18/2019 23:47:33.124 2        TUNNEL_ADD           0 GW Tunnel2

02/18/2019 23:47:33.120 3        MSG_CLIENT_DEL_PD    0 GW Tunnel1 (IP: 9.51.1.11)

02/18/2019 23:47:31.763 2        AP_DOMAIN_PUSH       0 Delete:eogre_domain_set, 0 GWs

02/18/2019 23:47:31.753 4        AP_VAP_PUSH          0 profile:'eogre_tunnel',
wlan:pyats_eogre
```

To view the summary information about EoGRE tunnel domain's status on an AP, use the following command:

```
Device# show ap name ap-name tunnel eogre domain summary

AP MAC          Domain                            Active Gateway
-------------------------------------------------------------------------------
80e8.6fd4.9520  eogre_domain
```

To view the detailed information about EoGRE tunnel domain on an AP, use the following command:

```
Device# show ap name ap-name tunnel eogre domain detailed

Domain Name    : eogre_domain
 Primary GW    : Tunnel1
 Secondary GW  : Tunnel2
 Active GW     : Tunnel1
 Redundancy    : Non-Revertive
 AdminState    : Up
```

To view the summary information about EoGRE tunnel gateways on an AP, use the following command:

```
Device# show ap name ap-name tunnel eogre gateway summary

AP MAC          Gateway          Type  IP                                      State  Clients
-------------------------------------------------------------------------------------------
80e8.6fd4.9520  Tunnel1          IPv4  9.51.1.11                                 Up     1

80e8.6fd4.9520  Tunnel2          IPv4  9.51.1.12                                 Down   0
```

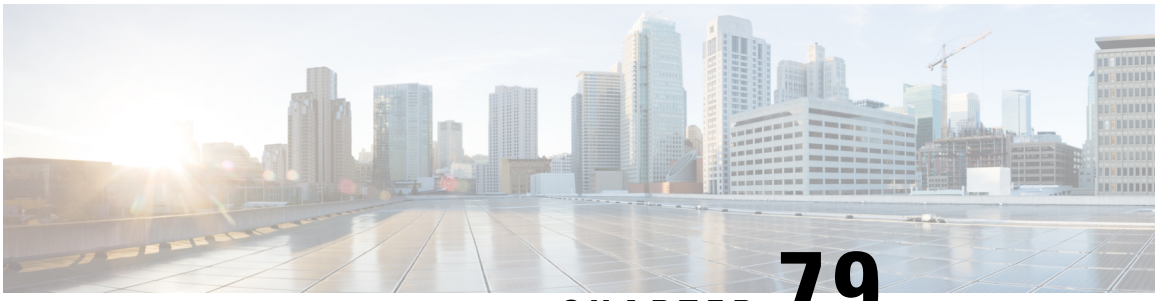
To view detailed information about an EoGRE tunnel gateway's status on an AP, use the following command:

```
Device# show ap name ap-name tunnel eogre gateway detailed gateway-name

Gateway : Tunnel2
 Mode   : IPv4
 IP     : 9.51.1.12
 State  : Down
 MTU    : 0
 AP MAC : 80e8.6fd4.9520

 Clients
  Total Number of Wireless Clients      : 0
 Traffic
  Total Number of Received Packets      : 0
  Total Number of Received Bytes        : 0
  Total Number of Transmitted Packets   : 0
```

```
Total Number of Transmitted Bytes     : 0
Total Number of Lost Keepalive        : 151
```

PART XII

Cisco DNA Service for Bonjour



CHAPTER 79

Cisco Catalyst Center Service for Bonjour Solution Overview

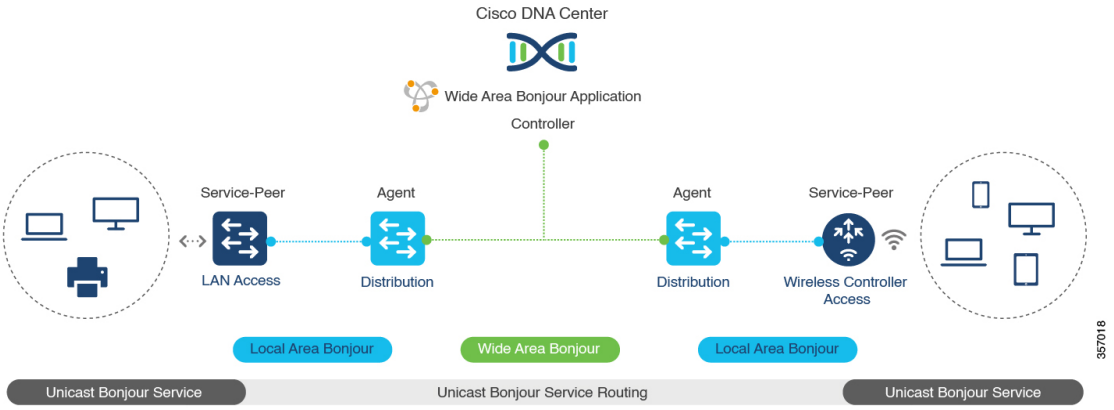
About the Cisco Catalyst Center Service for Bonjour Solution

The Apple Bonjour protocol is a zero-configuration solution that simplifies rich services and enables intuitive experience between connected devices, services, and applications. Using Bonjour, you can discover and use IT-managed, peer-to-peer, audio and video, or Internet of Things (IoT) services with minimal intervention and technical knowledge. Bonjour is originally designed for single Layer 2 small to mid-size networks, such as home or branch networks. The Cisco Catalyst Center Service for Bonjour solution eliminates the single Layer 2 domain constraint and expands the matrix to enterprise-grade traditional wired and wireless networks, including overlay networks such as Cisco Software-Defined Access (SD-Access) and industry-standard BGP EVPN with VXLAN. The Cisco Catalyst 9000 Series LAN switches, Cisco Nexus 9300 Series Switches, and Cisco Catalyst 9800 Series Wireless Controller follow the industry standard, RFC 6762-based multicast DNS (mDNS) specification to support interoperability with various compatible wired and wireless consumer products in enterprise networks.

The Cisco Wide Area Bonjour application on Catalyst Center enables mDNS service routing to advertise and discover services across enterprise-grade wired and wireless networks. The new-distributed architecture is designed to eliminate mDNS flood boundaries and transition to unicast-based service routing, providing policy enforcement points and enabling the management of Bonjour services.

The following figure illustrates how the Cisco Wide Area Bonjour application operates across two integrated service-routing domains.

Figure 30: Cisco Wide Area Bonjour Solution Architecture



- **Local Area Service Discovery Gateway Domain - Unicast Mode**: The new enhanced Layer 2 unicast policy-based deployment model. The new mDNS service discovery and distribution using the Layer 2 unicast address enables flood-free LAN and wireless networks. Cisco Catalyst 9000 Series Switches and Cisco Catalyst 9800 Series Wireless Controller in Layer 2 mode introduce a new service-peer role, replacing the classic flood-n-learn, for new unicast-based service routing support in the network. The service-peer switch and wireless controller also replace mDNS flood-n-learn with unicast-based communication with any RFC 6762 mDNS-compatible wired and wireless endpoints.
- **Wide-Area Service Discovery Gateway Domain**: The Wide Area Bonjour domain is a controller-based solution. The Bonjour gateway role and responsibilities of Cisco Catalyst and Cisco Nexus 9300 Series Switches are extended from a single SDG switch to an SDG agent, enabling Wide Area Bonjour service routing beyond a single IP gateway. The network-wide distributed SDG agent devices establish a lightweight, stateful, and reliable communication channel with a centralized Catalyst Center controller running the Cisco Wide Area Bonjour application. The SDG agents route locally discovered services based on the export policy.

Note The classic Layer 2 multicast flood-n-learn continues to be supported on wired and wireless networks with certain restrictions to support enhanced security and location-based policy enforcement. The Cisco Catalyst and Cisco Nexus 9300 Series Switches at Layer 3 boundary function as an SDG to discover and distribute services between local wired or wireless VLANs based on applied policies.

Solution Components

The Cisco Catalyst Center Service for Bonjour solution is an end-to-end solution that includes the following key components and system roles to enable unicast-based service routing across the local area and Wide Area Bonjour domain:

- **Cisco Service Peer**: Cisco Catalyst Switches and Cisco Wireless Controllers in Layer 2 access function in service peer mode to support unicast-based communication with local attached endpoints and export service information to the upstream Cisco Catalyst SDG agent in the distribution layer.

Note Cisco Nexus 9300 Series Switches don't support unicast-based service routing with downstream Layer 2 access network devices.

- **Cisco SDG Agent**: Cisco Catalyst and Cisco Nexus 9300 Series Switches function as an SDG agent and communicate with the Bonjour service endpoints in Layer 3 access mode. At the distribution layer, the SDG agent aggregates information from the downstream Cisco service peer switch and wireless controller, or local Layer 2 networks, and exports information to the central Catalyst Center controller.

Note Cisco Nexus 9300 Series Switches don't support multilayer LAN-unicast deployment mode.

- **Catalyst Center controller**: The Catalyst Center controller builds the Wide Area Bonjour domain with network-wide and distributed trusted SDG agents using a secure communication channel for centralized services management and controlled service routing.
- **Endpoints**: A Bonjour endpoint is any device that advertises or queries Bonjour services conforming to RFC 6762. The Bonjour endpoints can be in either LANs or WLANs. The Cisco Wide Area Bonjour application is designed to integrate with RFC 6762-compliant Bonjour services, including AirPlay, Google Chrome cast, AirPrint, and so on.

Supported Platforms

The following table lists the supported controllers, along with the supported hardware and software versions.

Table 38: Supported Controllers with Supported Hardware and Software Versions

Supported Controller	Hardware	Software Version
Catalyst Center appliance	DN2-HW-APL DN2-HW-APL-L DN2-HW-APL-XL	Catalyst Center, Release 2.3.6
Cisco Wide Area Bonjour application	—	2.4.660.75403

The following table lists the supported SDG agents along with their licenses and software requirements.

Table 39: Supported SDG Agents with Supported License and Software Requirements

Supported Platform	Supported Role	Local Area SDG	Wide Area SDG	Minimum Software
Cisco Catalyst 9200 Series Switches	SDG agent Service peer	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1

Supported Platform	Supported Role	Local Area SDG	Wide Area SDG	Minimum Software
Cisco Catalyst 9200L Series Switches	SDG agent Service peer	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9300 and 9300-X Series Switches	Service peer SDG agent	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9400 and 9400-X Series Switches	Service peer SDG agent	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9500 and 9500-X Series Switches	Service peer SDG agent	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9500 High Performance Series Switches	Service peer SDG agent	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9600 and 9600-X Series Switches	Service peer SDG agent	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9800 Wireless Controller	Service peer	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9800-L Wireless Controller	Service peer	Catalyst Center Advantage	Catalyst Center Advantage	Cisco IOS XE Release 17.11.1
Cisco Nexus 9300 Series Switches	SDG agent	Catalyst Center Advantage	Catalyst Center Advantage	Cisco NX-OS Release 10.2(3)F

Supported Network Design

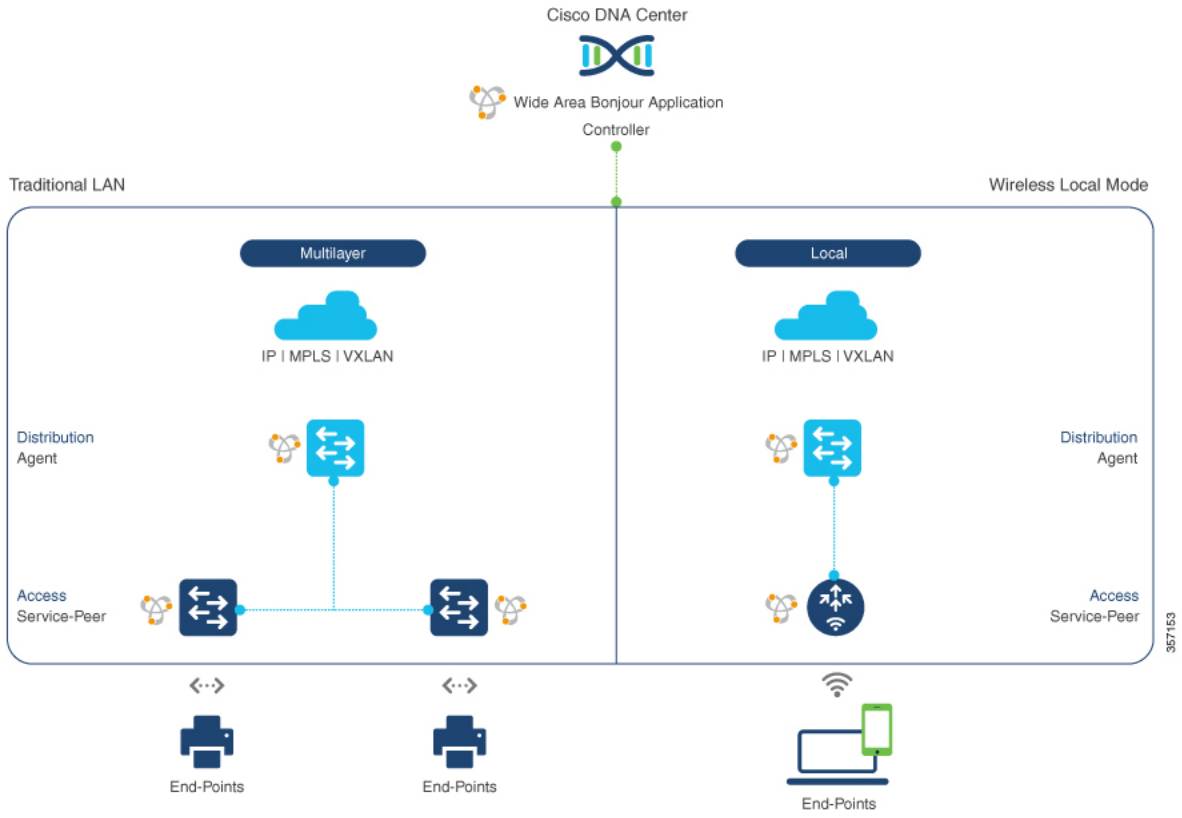
The Cisco Catalyst Center Service for Bonjour supports a broad range of enterprise-grade networks. The end-to-end unicast-based Bonjour service routing is supported on traditional, Cisco SD-Access, and BGP EVPN-enabled wired and wireless networks.

Traditional Wired and Wireless Networks

Traditional networks are classic Layer 2 or Layer 3 networks for wired and wireless modes deployed in enterprise networks. Cisco Catalyst Center Service for Bonjour supports a broad range of network designs to enable end-to-end service routing and replace flood-n-learn-based deployment with a unicast mode-based solution.

The following figure illustrates traditional LAN and central-switching wireless local mode network designs that are commonly deployed in an enterprise.

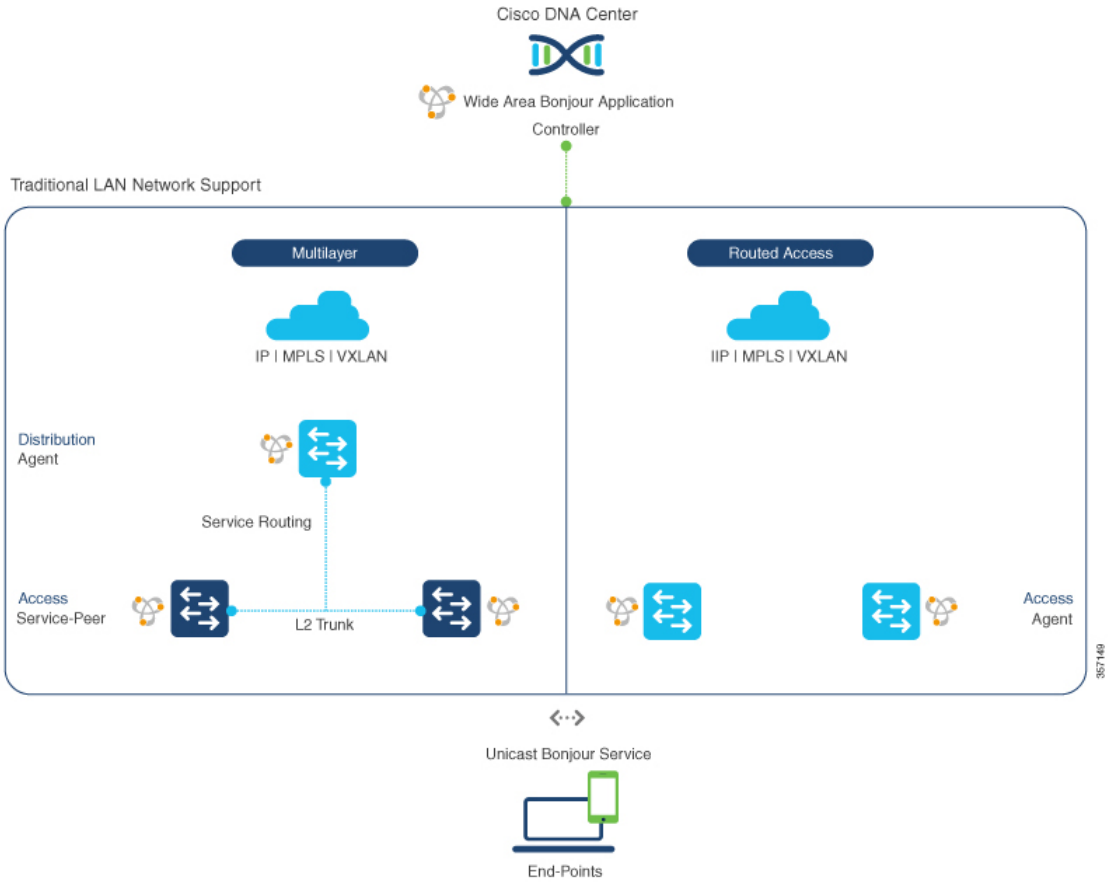
Figure 31: Enterprise Traditional LAN and Wireless Local Mode Network Design



Wired Networks

The following figure shows the supported traditional LAN network designs that are commonly deployed in an enterprise.

Figure 32: Enterprise Wired Multilayer and Routed Access Network Design



The Cisco Catalyst or Cisco Nexus 9300 Series Switches in SDG agent role that provide Bonjour gateway functions are typically IP gateways for wired endpoints that could reside in the distribution layer in multilayer network designs, or in the access layer in Layer 3 routed access network designs:

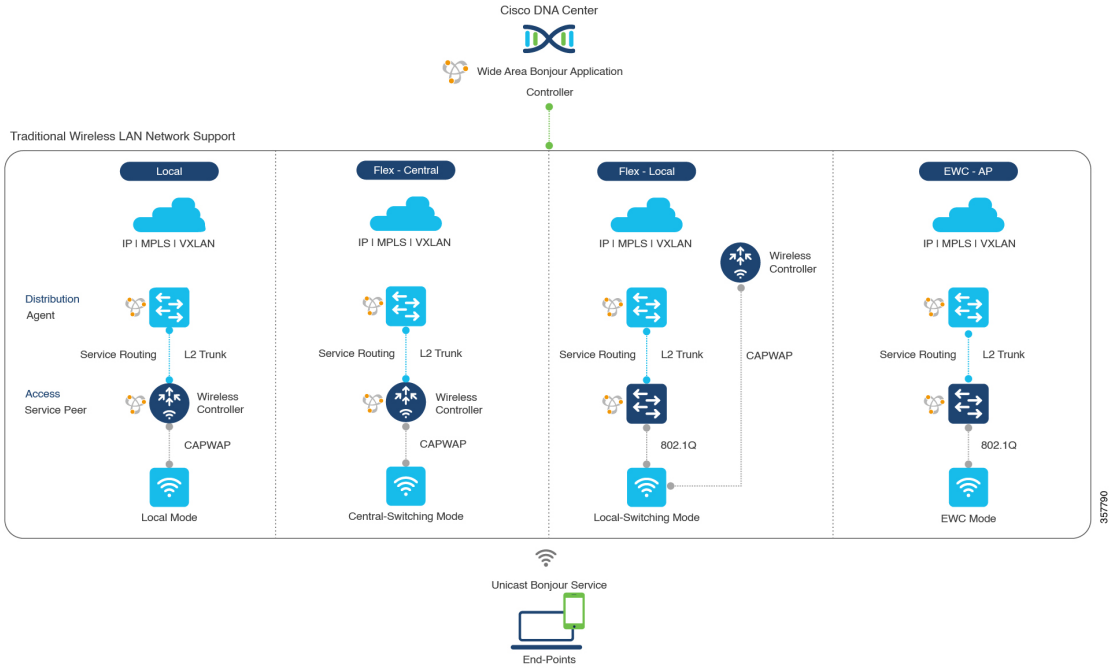
- **Multilayer LAN—Unicast Mode**: In this deployment mode, the Layer 2 access switch provides the first-hop mDNS gateway function to locally attached wired endpoints. In unicast mode, the mDNS services are routed to the distribution layer systems providing IP gateway and SDG agent mode. The policy-based service routing between the SDG agents is performed by the Catalyst Center controller.
- **Multilayer LAN—Flood-n-Learn Mode**: In this deployment mode, the Layer 2 access switch or wireless controller are in mDNS passthrough modes with the Cisco Catalyst or Cisco Nexus 9300 Series Switches operating in the SDG agent mode. The mDNS gateway function at distribution layer in a network enables inter-VLAN mDNS local proxy. It also builds stateful Wide Area Bonjour unicast service routing with the Catalyst Center to discover or distribute mDNS services beyond a single IP gateway.
- **Routed Access**: In this deployment mode, the first-hop Cisco Catalyst or Cisco Nexus 9300 Series Switch is an IP gateway boundary and, therefore, it must also perform the SDG agent role. The policy-based service routing between the SDG agents is performed by the Catalyst Center controller.

Wireless Networks

The Cisco Catalyst Center Service for Bonjour extends the single wireless controller mDNS gateway function into the Wide Area Bonjour solution. The mDNS gateway on Cisco Catalyst 9800 Series Wireless Controller can be deployed in an enhanced mode as a service peer. In this mode, the wireless controller builds unicast service routing with an upstream Cisco Catalyst gateway switch for end-to-end mDNS service discovery. It replaces the classic flood-n-learn mDNS services from wired network using mDNS AP or other methods.

The following figure shows the supported traditional wireless LAN network designs that are commonly deployed in an enterprise. Based on the wireless network design, the mDNS gateway function may be on the wireless controller, or first-hop Layer 2 or Layer 3 Ethernet switch of an Access Point in local-switching mode.

Figure 33: Enterprise Traditional Wireless LAN Network Design



The Cisco Catalyst Center Service for Bonjour supports the following modes for wireless LAN networks:

- **Local Mode**: In the central switching wireless deployment mode, the m-DNS traffic from local mode Cisco access points is terminated on the Cisco Catalyst 9800 Series Wireless Controller. The Cisco Catalyst 9800 Series Wireless Controller extends the mDNS gateway function to the new service peer mode. The wireless controller can discover and distribute services to local wireless users and perform unicast service routing over a wireless management interface to the upstream Cisco Catalyst Switch in the distribution layer, which acts as the IP gateway and the SDG agent.
- **FlexConnect—Central**: The mDNS gateway function for Cisco access point in FlexConnect central switch SSID functions consistently as described in **Local Mode**. The new extended mDNS gateway mode on the Cisco Wireless Controller and upstream service routing with SDG agent operate consistently to discover services across network based on policies and locations.
- **FlexConnect—Local**: In FlexConnect local switching mode, the Layer 2 access switch in mDNS gateway service peer mode provides the policy-based mDNS gateway function to locally attached wired and wireless users. The Cisco Catalyst Switches in the distribution layer function as SDG agents and enable

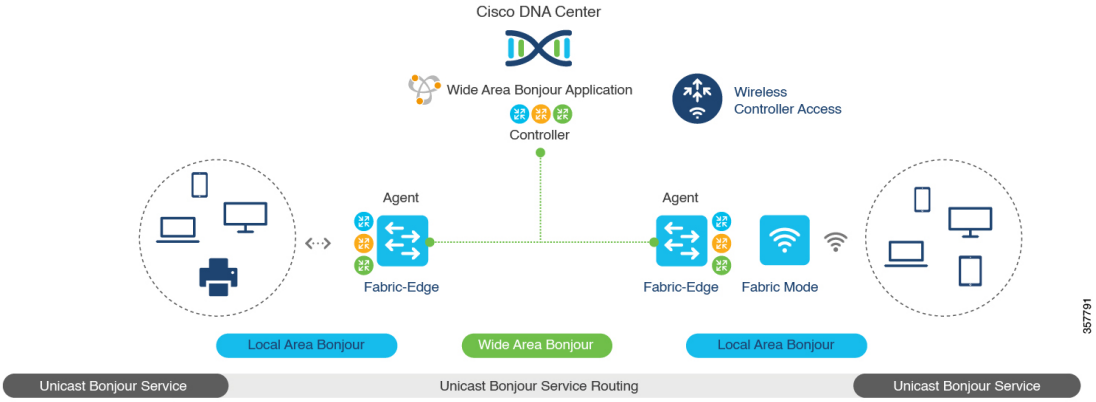
mDNS service-routing across all Layer 2 ethernet switches to support unicast-based service routing to LAN and wireless LAN user groups.

- **Embedded Wireless Controller—Access Point**: The Layer 2 access switch in service peer mode provides unified mDNS gateway function to wired and wireless endpoints associated with Cisco Embedded Wireless Controller on Cisco Catalyst 9100 Series Access Points. The SDG agent in the distribution layer provides unicast service routing across all Layer 2 service peer switches in the Layer 2 network block without any mDNS flooding.

Cisco SD-Access Wired and Wireless Networks

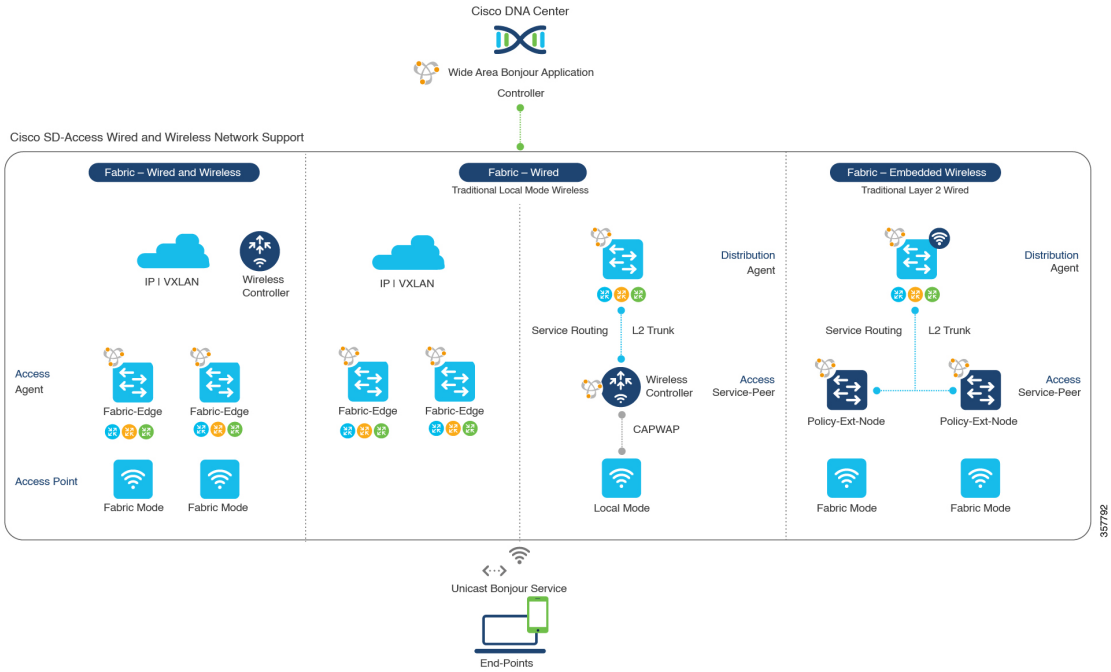
Cisco SD-Access-enabled wired and wireless networks support Cisco Catalyst Center Service for Bonjour across fabric networks. The Cisco Catalyst 9000 Series Switches support VRF-aware Wide Area Bonjour service routing to provide secure and segmented mDNS service discovery and distribution management for virtual networks. The VRF-aware unicast service routing eliminates the need to extend Layer 2 flooding, and improves the scale and performance of the fabric core network and endpoints.

Figure 34: Cisco SD-Access Wired and Wireless Network Design



Cisco SD-Access supports flexible wired and wireless network design alternatives to manage fully distributed, integrated, and backward-compatible traditional network infrastructure. Wide Area Bonjour service routing is supported in all network designs providing intuitive user experience. The following figure illustrates the various SD-Access enabled wired and wireless network design alternatives.

Figure 35: Cisco SD-Access Wired and Wireless Network Design Alternatives



The Cisco Catalyst Center Service for Bonjour for SD-Access enabled wired and fabric, or traditional mode-wireless networks use two-tier service routing providing end-to-end unicast-based mDNS solution. Based on the network design, each solution component is enabled in a unique role to support the Wide Area Bonjour domain:

- **Fabric Edge SDG Agent**: The Layer 3 Cisco Catalyst Fabric Edge switch in the access layer configured as SDG agent provides unicast-based mDNS gateway function to the locally attached wired and wireless endpoints. The VRF-aware mDNS service policy provides network service security and segmentation in a virtual network environment. The mDNS services can be locally distributed and routed through centralized Catalyst Center.
- **Policy Extended Node**: The Layer 2 Cisco Catalyst access layer switch enables first-hop mDNS gateway function without flooding across the Layer 2 broadcast domain. The unicast-based service routing with upstream Fabric Edge switch in the distribution layer enables mDNS service routing within the same Layer 2 network block. It can also perform remote service discovery and distribution from centralized Catalyst Center.
- **Cisco Wireless Controller**: Based on the following wireless deployment modes, Cisco Wireless Controller supports unique function to enable mDNS service routing in Cisco SD-Access enabled network:
  - **Fabric-Enabled Wireless**: Cisco Wireless Controller doesn't require any mDNS gateway capability to be enabled in distributed fabric-enabled wireless deployments.
  - **Local Mode Wireless**: As Cisco Wireless Controller provides central control and data plane termination, it provides mDNS gateway in service peer mode for wireless endpoints. The wireless controller provides mDNS gateway between locally associated wireless clients. The wireless controller builds service routing with upstream SDG agent Catalyst switch providing IP gateway and service routing function for wireless endpoints.

- **Embedded Wireless Controller—Switch**: The Cisco Embedded Wireless Controller solution enables the lightweight integrated wireless controller function within the Cisco Catalyst 9300 Series Switch. The Cisco Catalyst switches in the distribution layer function as SDG agents to the wired and wireless endpoints. The SDG agent in the distribution layer provides unicast service routing across all wireless access points and Layer 2 service peer switches without mDNS flooding.

- **Catalyst Center Controller**: The Cisco Wide Area Bonjour application on Catalyst Center supports policy and location-based service discovery, and distribution between network-wide distributed Fabric Edge switches in SDG agent mode.

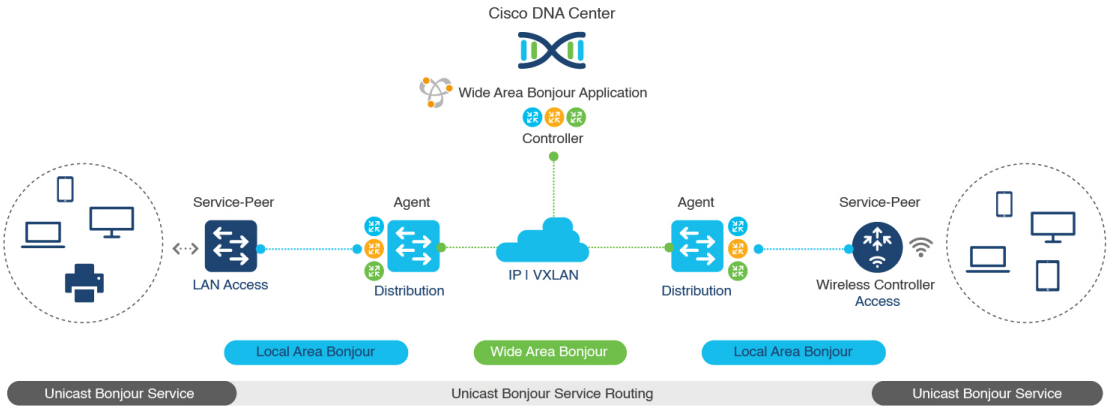
The Wide Area Bonjour communication between the SDG agent and controller takes place through the network underlay. Based on policies, the SDG agent forwards the endpoint announcements or queries to the Catalyst Center. After discovering a service, the endpoints can establish direct unicast communication through the fabric overlay in the same virtual network. The inter-virtual network unicast communication takes place through the Fusion router or external Firewall system. This communication is subject to the configured overlay IP routing and Security Group Tag (SGT) policies.

BGP EVPN Networks

The BGP EVPN-based technology provides a flexible Layer 3 segmentation and Layer 2 extension overlay network. The VRF and EVPN VXLAN-aware Wide Area Bonjour service routing provides secure and segmented mDNS service solution. The overlay networks eliminate mDNS flooding over EVPN-enabled Layer 2 extended networks and solve the service reachability challenges for Layer 3 segmented routed networks in the fabric.

The following figure shows the BGP EVPN leaf switch in the distribution layer, supporting overlay Bonjour service routing for a BGP EVPN-enabled traditional Layer 2 wired access switch and traditional wireless local mode enterprise network interconnected through various types of Layer 2 networks and Layer 3 segmented VRF-enabled networks.

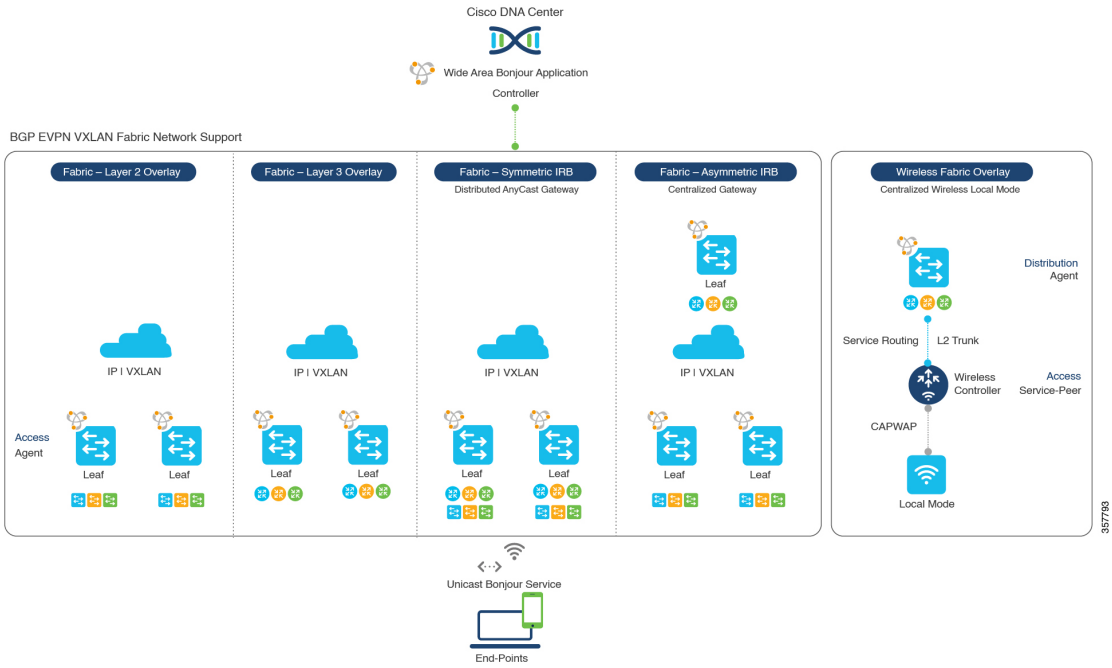
Figure 36: Overlay Bonjour Service for a BGP EVPN-Enabled Enterprise Network



Cisco Catalyst Center Service for Bonjour supports all the industry-standard overlay network designs enabling end-to-end unicast-based mDNS service routing, and preventing flooding and service boundary limitation across wired and wireless networks.

The following figure illustrates the various BGP EVPN VXLAN reference overlay network design alternatives. This network design enables end-to-end mDNS service discovery and distribution based on overlay network policies.

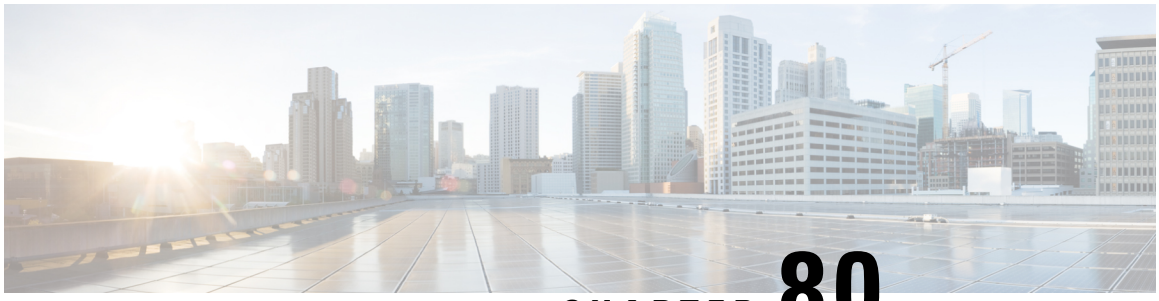
Figure 37: BGP EVPN VXLAN Wired and Wireless Design Alternatives



The Cisco Catalyst and Cisco Nexus 9000 Series Switches can be deployed in Layer 2 or Layer 3 leaf roles supporting mDNS service routing for a broad range of overlay networks. In any role, the mDNS communication is limited locally and supports end-to-end unicast-based service routing across Wide Area Bonjour domain:

- **Layer 2 Leaf SDG Agent**: The Cisco Catalyst or Cisco Nexus switches can be deployed as Layer 2 leaf supporting end-to-end bridged network with IP gateway within or beyond BGP EVPN VXLAN fabric network. By default, the mDNS is flooded as Broadcast, Unknown Unicast, Multicast (BUM) over the fabric-enabled core network. This mDNS flooding may impact network performance and security. The Layer 2 leaf, enabled as SDG agent, prevents mDNS flooding over VXLAN and supports unicast-based service routing.
- **Layer 3 Leaf SDG Agent**: The Cisco Catalyst or Cisco Nexus switches can be deployed as SDG agent supporting Layer 3 overlay network in BGP EVPN VXLAN fabric. The IP gateway and mDNS service boundary is terminated at the SDG agent switches and remote services can be discovered or distributed through centralized Catalyst Center.
- **Local Mode Wireless**: The centralized wireless local mode network can be terminated within or outside the EVPN VXLAN fabric domain to retain network segmentation and service discovery for wireless endpoints. The Cisco Catalyst 9800 Series Wireless Controller in service peer mode can build unicast service routing with distribution layer IP and SDG agent Cisco Catalyst switch to discover services from BGP EVPN VXLAN fabric overlay network.
- **Catalyst Center**: Catalyst Center supports Wide Area Bonjour capability to dynamically discover and distribute mDNS services based on Layer 2 or Layer 3 Virtual Network ID (VNID) policies to route the mDNS services between SDG agent switches in the network.

For more information about BGP EVPN networks, see *Cisco Catalyst Center Service for Bonjour Configuration Guide, Cisco IOS XE Bengaluru 17.6.x (Catalyst 9600 Switches)*.



CHAPTER 80

Configuring Local Area Bonjour for Embedded Wireless Controller Access Point Mode

Overview of Local Area Bonjour for Embedded Wireless Controller - Access Point Mode

The Cisco Embedded Wireless Controller on Catalyst Access Points introduces unicast mode function in Local Area Bonjour network domain. The enhanced gateway function at the first hop of wired and wireless networks communicate directly with any industry standard RFC 6762 compliant multicast DNS (mDNS) end point in Layer 2 unicast mode.

The Cisco Catalyst 9100 series Access Points (AP) support distributed wireless forwarding with Embedded Wireless Controller (EWC) in Local-Switching mode. The Catalyst 9000 series LAN switch introduces new Service-Peer mode to support mDNS gateway for locally attached wired and wireless endpoints in Unicast mode. The mDNS service discovery and distribution boundary is expanded from single-gateway to end-to-end service-routing with upstream SDG Agent switch to enable unicast-mode, increased scale, performance, and resiliency in the network.

Restrictions for Local Area Bonjour for Embedded Wireless Controller - Access Point Mode

- The mDNS gateway on EWC Cisco Catalyst 9100 series Access Points does not support service-peer mode to enable service-routing and unicast mode mDNS communication.
- The mDNS gateway on EWC Catalyst 9100 series Access Points must be in disabled state.
- The mDNS bridging is required, allowing mDNS service discovery and distribution from locally attached mDNS gateway Layer 2 access switch in Service-Peer mode.
- The Catalyst 9000 series switches in Service-Peer mode supports per Layer 2 access switch level Location-Based service for wireless users connected to EWC mode Access Point and Wired endpoints.

Prerequisites for Local Area Bonjour for Embedded Wireless Controller - Access Point Mode

The EWC mode Cisco Catalyst 9100 series Access Points must be successfully configured and operational before implementing Cisco Local Area Bonjour for EWC AP mode wireless networks.

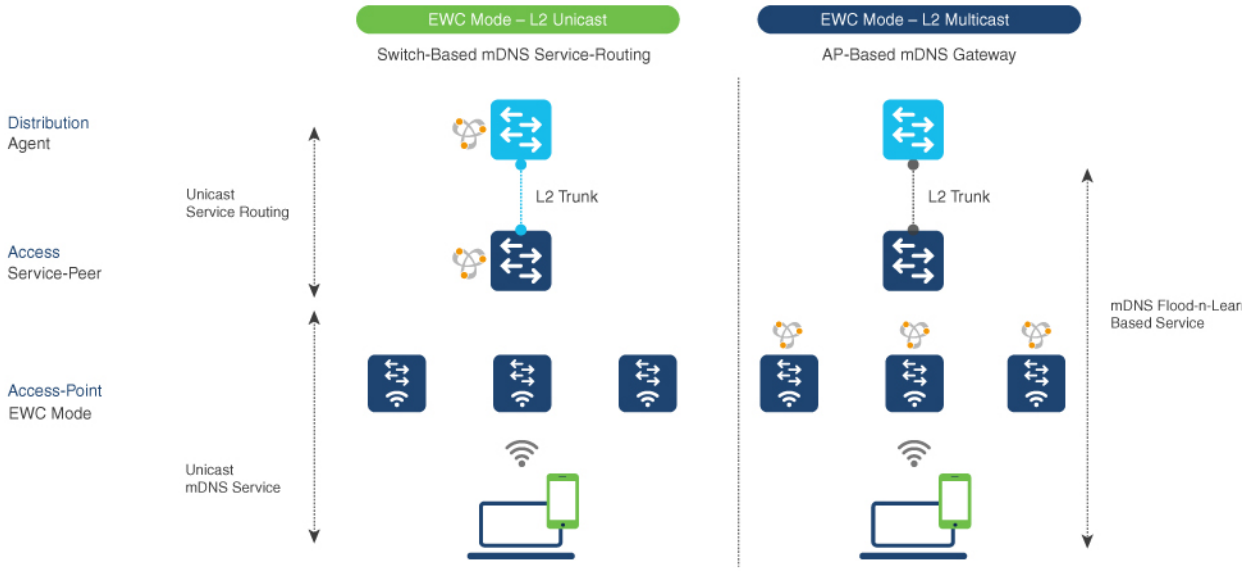
The following are the pre-requisites verified on EWC mode AP and the Layer 2 access Cisco Catalyst 9000 series switches deployed in Service-Peer mode supporting mDNS gateway for wired and wireless users:

- The EWC mode Cisco Catalyst 9100 series Access Point must be pre-configured to implement wireless network and other advanced parameters. For more information, see the Cisco Embedded Wireless Controller on Catalyst Access Points Configuration Guide .
- The EWC mode Cisco Catalyst 9100 series Access Point may run operate recommended IOS-XE software version. There are no mDNS requirements and software version dependency on EWC mode AP to enable Local Area Bonjour gateway.
- Ensure that the targeted controller for Service-Peer role has the required Cisco IOS XE software version.
- Ensure that the controller runs a valid Cisco DNA-Advantage license.
- Ensure that the upstream distribution-layer Cisco Catalyst switch in SDG-Agent mode runs a valid Cisco DNA-Advantage license.
- Ensure that the controller is interconnected as Layer 2 trunk in multi-layer network, when Layer 2 Unicast service-routing is running between SDG-Agent in distribution-layer and the controller service-peer.
- Ensure the Catalyst 9000 access layer switches have IP reachability to upstream Cisco Catalyst 9000 series switches in SDG Agent mode over IPv4 subnet. that is, switch management IP network

Understanding EWC Mode mDNS Gateway Alternatives

The Cisco Catalyst controllers continue to innovate mDNS gateway function to address evolving business and technical requirements in enterprise networks. The EWC mode Access Point based wireless networks can implement mDNS gateway using following two methods as displayed in figure below:

Figure 38: EWC Mode Access Point mDNS Gateway Alternatives



The mDNS gateway for EWC mode Access Point wireless network can be implemented using in either mode to address service discovery and distribution based on operating network environment:

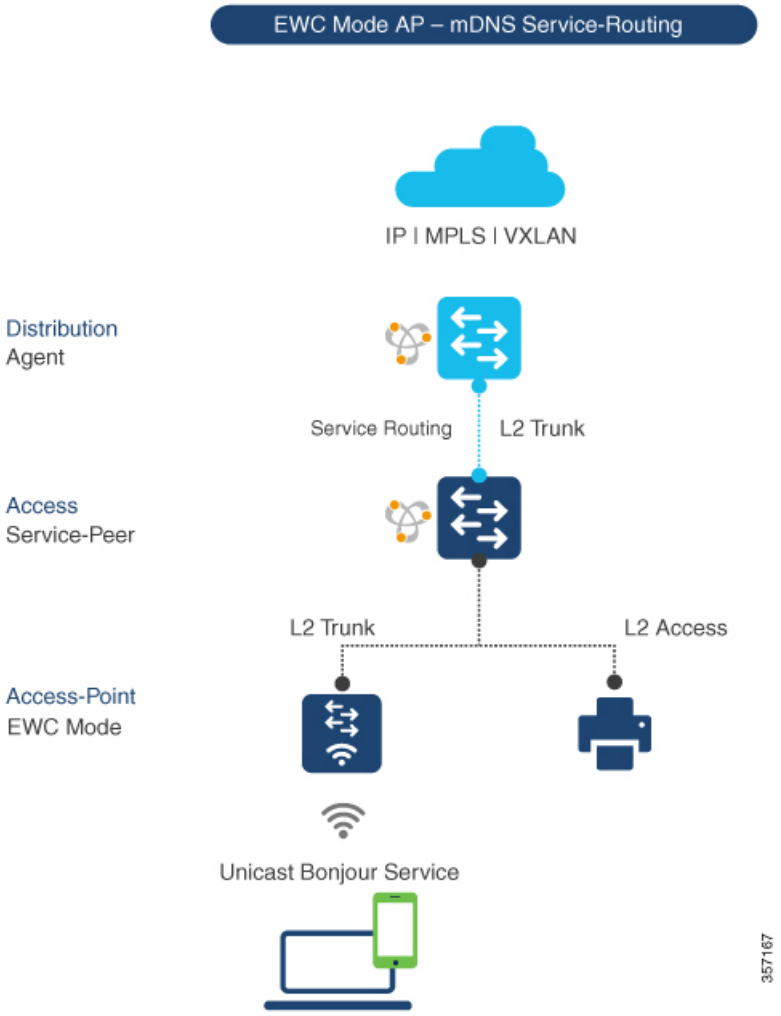
- **Switch Based mDNS Gateway**– Catalyst 9000 series switch in Layer 2 access can be implemented as mDNS gateway in Service-Peer role supporting following key benefits:
  - Replaces flood-n-learn with new enhanced Unicast-based mDNS communication with locally attached wired and EWC mode Access Point wireless users.
  - The Catalyst 9000 eliminates mDNS flood with Unicast service-routing to LAN distribution. The Unicast service-routing between LAN distribution and Layer 2 access layer switches forms Local Area Bonjour domain to enable policy and location-based service discovery and distribution. The unicast based service-routing over Layer 2 trunk eliminates mDNS flood-free and enables service-oriented wired and wireless networks.
  - The switch-based mDNS gateway solution eliminates requirement to forward wired network traffic to wireless APs improving wireless scale, performance and network reliability.
- **AP Based mDNS Gateway** – The Cisco EWC mode Access Point can alternatively be implemented as mDNS gateway in case if connected to unsupported LAN access switch. In this classic method the mDNS service discovery and distribution follows flood-n-learn mechanic over the Layer 2 wired and wireless network. Refer to Multicast Domain Name System chapter module for Cisco Embedded Wireless Controller Configuration Guide, Release 17.3.1 to implement AP based mDNS gateway.

Understanding Local Area Bonjour for Embedded Wireless Controller Access Point Mode

The Cisco Catalyst LAN switches and WLC supported mDNS gateway function with various advancements for broad range of Wired and Wireless network types. As the enterprise requirements expands it drives IT organization to introduce new network deployment models, supporting mobile devices and distributed zero-configuration services following increased scale, granular security control and resiliency for mission critical networks. The common unified Cisco IOS-XE operating system across Catalyst 9000 series LAN switches and EWC mode Catalyst 9100 series Access Points enables distributed Bonjour gateway function at network edge and with end-to-end Wide Area Bonjour service-routing the new solution enables service-oriented enterprise networks with intuitive user-experience.

The figure below displays the Cisco Catalyst 9000 series switches connected to EWC mode Access Points that supports the mDNS gateway function to the locally attached EWC mode wireless users and wired users.

Figure 39: Cisco Catalyst Switch and EWC Mode Access Point



The Cisco Catalyst 9000 series switches in Layer 2 access layer and at Layer 3 distribution layer must be configured in following mDNS gateway mode to enable Unicast-based mDNS service-routing between wired and EWC mode Access Points mode wireless users within same Layer 2 network block.

- **Service-Peer**- The Layer 2 access switch connecting Wireless Access-Point in EWC mode must be configured with mDNS gateway in Service-Peer mode. Each Layer 2 access switch provides mDNS gateway function between locally attached wired and EWC mode Access Point wireless users. The Unicast-based mDNS service discovery and distribution within same or different VLANs is supported with bi-directional mDNS policies on single Layer 2 access switch.
- **SDG Agent**- The mDNS flood-n-learn based method in Layer 2 network is replaced with simple Unicast based service-routing between Layer 2 access switch in Service-Peer mode and upstream distribution-layer in mDNS gateway SDG Agent mode. The Unicast based mDNS service-routing eliminates mDNS flood over Layer 2 trunk ports providing increase bandwidth, enhanced security, location-based services and flood control management in wired and EWC mode Access Point wireless network.

Configuring Local Area Bonjour for Embedded Wireless Controller Access Point Mode

This topic describes the configuration steps to implement Layer 2 access layer Cisco Catalyst 9000 series switch as mDNS gateway and enable Service-Peer on Layer 2 access layer switch and the SDG Agent mode. To enable mDNS service discovery and distribution between multiple Layer 2 access switches, service-routing must be enabled with upstream distribution-layer Cisco Catalyst 9000 series switch in SDG Agent mode to build Local Area Bonjour service-routing domain.

Note mDNS gateway must be globally disabled on Catalyst 9100 series Access Point in EWC mode.

Configuring mDNS Gateway Mode (CLI)

To enable mDNS gateway and service-peer mode on Layer 2 access switch and SDG Agent mode on Layer 3 distribution layer switch, follow the procedure given below:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables Privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose
Step 3	**mdns-sd gateway** **Example:** `Device(config)# mdns-sd gateway`	Enables mDNS on the Layer 2 Catalyst switch and enters the mDNS gateway configuration mode. (Optional) You can configure the following additional parameters: • **air-print-helper**: Enables communication between Apple iOS devices like iPhone or iPad to discover and use older printers that does not support driverless AirPrint function. • **cache-memory-max**: Configures the percentage memory for cache. • **ingress-client**: Configures Ingress client packet tuners. • **rate-limit**: Enables rate limiting of incoming mDNS packets. • **service-announcement-count**: Configures maximum advertisements. • **service-announcement-timer**: Configures advertisements announcement timer periodicity. • **service-query-count**: Configures maximum queries. • **service-query-timer**: Configures query forward timer periodicity. • **service-type-enumeration**: Configures service enumeration. **Note** For **cache-memory-max**, **ingress-client**, **rate-limit**, **service-announcement-count**, **service-announcement-timer**, **service-query-count**, **service-query-timer**, and **service-type-enumeration** commands, you can retain the default value of the respective parameter for general deployments. Configure a different value, if required, for a specific deployment.

	Command or Action	Purpose
Step 4	**mode** {**service-peer** | **sdg-agent**} **Example:** `Device(config-mdns-sd)# mode service-peer`	Configures mDNS gateway in one of the following modes based on the system settings: • **Service-Peer**– Enables Layer 2 Catalyst access switch in mDNS Service-Peer mode. • **SDG Agent**– Default. Enables Layer 3 distribution layer Catalyst switch in SDG Agent mode to peer with central Cisco DNA Center controller for Wide Area Bonjour service routing.
Step 5	**exit** **Example:** `Device(config-mdns-sd)# exit`	Exits mDNS gateway configuration mode.

Configuring mDNS Service Policy (CLI)

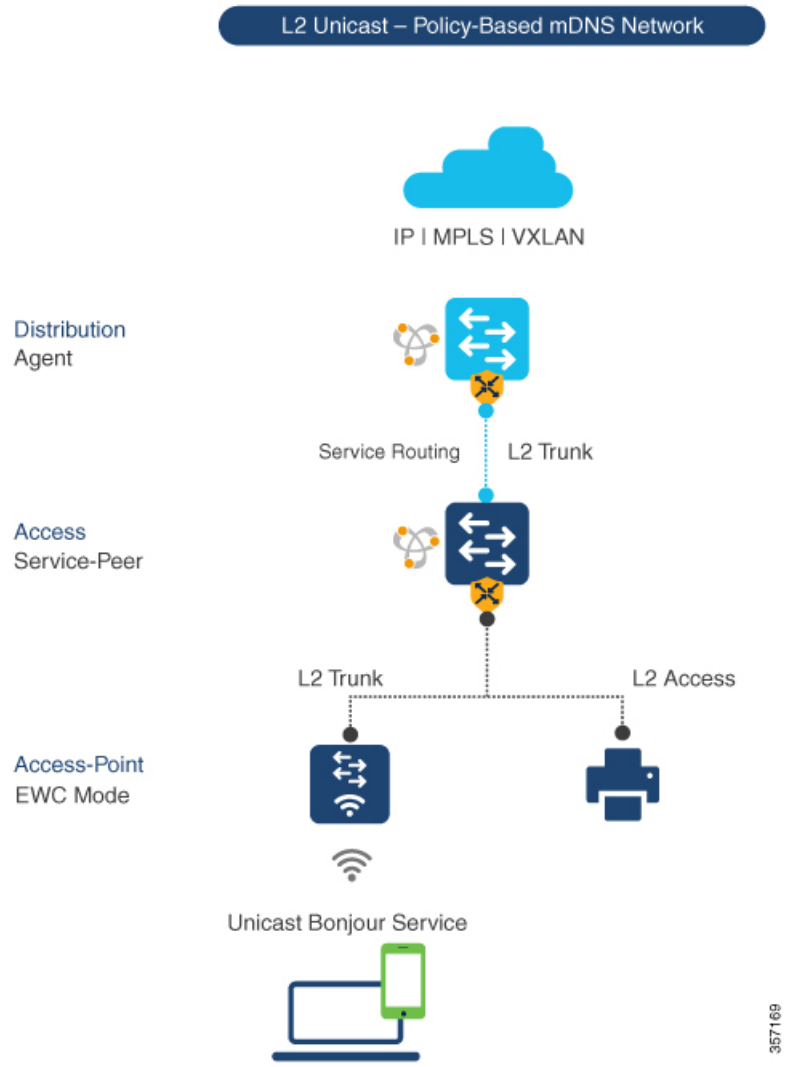
To configure an mDNS service policy, follow the steps given below:

1. Create service-list to permit built-in or user-defined custom service types.
2. Associate service-list to a service-policy to enforce ingress or egress direction.
3. Apply the service policy to the new VLAN configuration mode.

Note You will need this configuration in service-peer mode for Layer 2 Catalyst switch and SDG agent mode for Layer 3 Catalyst switch.

The figure given below displays how to configure mDNS policies on the Catalyst switch in service-peer and SDG agent modes:

Figure 40: Catalyst Service-Peer and SDG Agent mDNS Service Policy Configuration



To build and apply service-policies on target VLAN in service-peer and SDG agent modes, follow the procedure given below:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables Privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.

	Command or Action	Purpose

Step 3	**mdns-sd service-list** *service-list-name* {**in** | **out**}
	Example:
	`Device(config)# mdns-sd service-list VLAN100-LIST-IN in`
	`Device(config)# mdns-sd service-list VLAN100-LIST-OUT out`
		Configure mDNS service-list to classify one or more service types. Unique service-list is required to process incoming mDNS message and outbound response to request locally locally connected wired or EWC mode Access Point wireless end points.

Step 4	**match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}]
	Example:
	`Device(config)# mdns-sd service-list VLAN100-LIST-IN in`
	`Device(config-mdns-sl-in)# match APPLE-TV`
	`Device(config-mdns-sl-in)# match PRINTER-IPPS message-type announcement`
		Matches inbound service-list.
		The Catalyst switch validates to accept or drop incoming mDNS service-type (such as, Apple TV) advertisement or query matching message type from locally connected wired or EWC mode Access Point wireless end points. The service-list contains implicit deny at the end.
		The default message-type used is **any**.

Step 5	**match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}]
	Example:
	`Device(config)# mdns-sd service-list VLAN100-LIST-OUT out`
	`Device(config-mdns-sl-in)# match APPLE-TV`
	`Device(config-mdns-sl-in)# match PRINTER-IPPS`
		Matches outbound service-list.
		The Catalyst switch provides local service proxy function by responding matching service-type to the requesting end point(s). For example, the Apple-TV and Printer learnt from VLAN 100 will be distributed to EWC mode Access Point wireless in same VLAN 100. The service-list contains implicit deny at the end.
		The message-type for outbound service-list is not required.

Step 6	**mdns-sd service-policy** *service-policy-name* **Example:** `Device(config)# mdns-sd service-policy VLAN100-POLICY`	Creates a unique mDNS service-policy in the global configuration mode.

Step 7	**service-list** *service-list-name* {**in** | **out**}
	Example:
	`Device(config)# mdns-sd service-policy VLAN100-POLICY`
	`Device(config-mdns-ser-policy)# service-list VLAN100-LIST-IN in`
	`Device(config-mdns-ser-policy)# service-list VLAN100-LIST-OUT out`
		Configures an mDNS service-policy to associate service-list for each direction.

Step 8	**vlan configuration** *ID* **Example:** `Device(config)# vlan configuration 100`	Enables wired or wireless EWC mode Access Point user VLAN configuration for advanced service parameters. One or more VLANs can be created for the same settings.

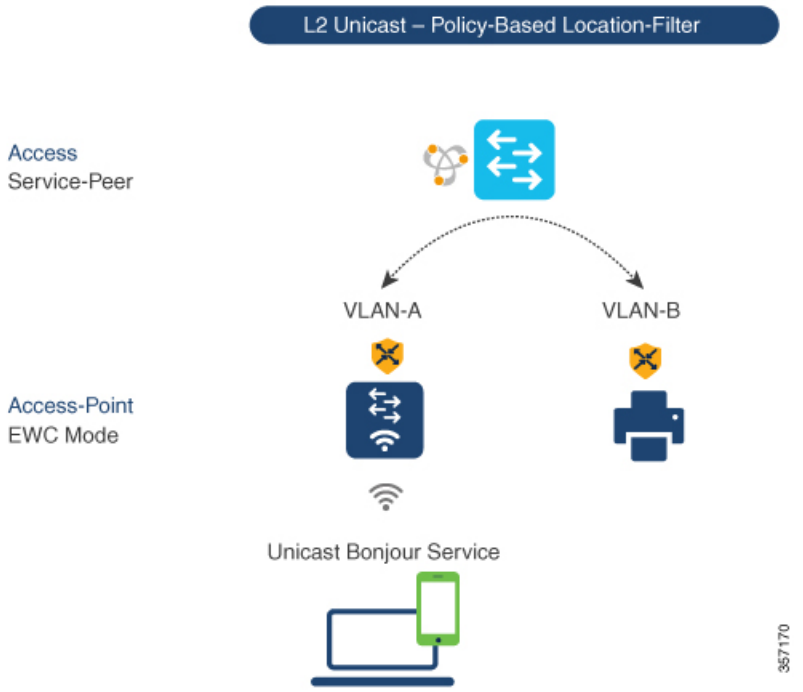
	Command or Action	Purpose
		Here, *ID* refers to the VLAN configuration ID. For example, *vlan configuration 101-110,200* range allows you to configure consecutive and non-consecutive VLAN ID(s) range.
Step 9	**mdns-sd gateway** **Example:** `Device(config-vlan)# mdns-sd gateway`	Enables mDNS gateway on configured wired or EWC mode Access Point wireless user VLAN ID(s).
Step 10	**service-policy** *service-policy-name* **Example:** `Device(config-vlan-mdns)# service-policy VLAN100-POLICY`	Associates mDNS service-policy to the configured wired or EWC mode Access Point wireless user VLAN ID(s).
Step 11	**exit** **Example:** `Device(config-vlan-mdns)# exit`	Exits the mDNS gateway configuration mode.

Configuring mDNS Location-Filter (CLI)

The Layer 2 Cisco Catalyst access-layer switch in the service-peer mode, by default provides local service proxy between mDNS service provider and receiver connected in the same Layer 2 VLAN associated to wired or EWC mode Access Point wireless user networks. Optionally, you can configure mDNS location-filter to allow service discovery and distribution between locally configured VLAN IDs associated to wired or EWC mode Access Point wireless user networks.

The following figure displays and references location-filter policy on Catalyst switch in service-peer mode permitting discovery and distribution of mDNS services between wired and EWC mode Access Point wireless user VLANs.

Figure 41: Catalyst Service-Peer mDNS Location-Filter Configuration



To enable local service proxy on Cisco Catalyst switch in service-peer mode and to discover mDNS services between local wired and wireless EWC mode Access Point user VLANs, follow the procedure given below:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables Privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 3	**mdns-sd location-filter** *location-filter-name* **Example:** `Device(config)# mdns-sd location-filter LOCAL-PROXY`	Configures a unique location-filter in the global configuration mode.
Step 4	**match location-group** {**all** | **default** | **ID**} **vlan** [*ID*] **Example:**	Configures the match criteria to mutually distribute the permitted services between grouped VLANs. For example, mDNS services can be discovered and distributed using the

	Command or Action	Purpose
	`Device(config-mdns-loc-filter)# match location-group default vlan 100` `Device(config-mdns-loc-filter)# match location-group default vlan 101`	unicast-mode between Wireless EWC mode Access Point user VLAN ID 100 and wired user VLAN ID 101.
Step 5	**mdns-sd service-list** *service-list-name* {**in** | **out**} **Example:** `Device(config)# mdns-sd service-list VLAN100-LIST-OUT out`	Configures the mDNS service-list to classify one or more service types. Unique service-list is required to process incoming mDNS message and response outbound requesting wired or EWC mode Access Point user end points.

Step 6	**match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}]
	Example:
	`Device(config)# mdns-sd service-list VLAN100-LIST-OUT out`
	`Device(config-mdns-sl-out)# match APPLE-TV location-filter LOCAL-PROXY`
		Associates location-filter to one or more service types to enable local proxy between local VLANs. For example, the Apple-TV learnt from VLAN 100 and VLAN 101 will be distributed to receiver in VLAN 100.
		Note The service-list contains implicit deny at the end.
		You do not require a **message-type** for the outbound service-list.

Step 7	**mdns-sd service-policy** *service-policy-name* **Example:** `Device(config)# mdns-sd service-policy VLAN100-POLICY`	Creates a unique mDNS service-policy in the global configuration mode.

Step 8	**service-list** *service-list-name* {**in** | **out**}
	Example:
	`Device(config)# mdns-sd service-policy VLAN100-POLICY`
	`Device(config-mdns-ser-policy)# service-list VLAN100-LIST-OUT out`
		Configures an mDNS service-policy to associate the service-list for each direction.

Step 9	**vlan configuration** *ID* **Example:** `Device(config)# vlan configuration 100`	Enables VLAN configuration for advanced service parameters. You can create one or more VLANs with the same settings. Here, *ID* refers to the VLAN configuration ID. For example, *vlan configuration 101-110,200* range allows you to configure consecutive and non-consecutive VLAN ID range.
Step 10	**mdns-sd gateway** **Example:** `Device(config-vlan-config)# mdns-sd gateway`	Enables the mDNS gateway on the configured VLAN ID(s).

	Command or Action	Purpose
Step 11	**service-policy** *service-policy-name* **Example:** `Device(config-vlan-mdns-sd)# service-policy VLAN100-POLICY`	Associates mDNS service-policy to the configured VLAN ID(s).
Step 12	**exit** **Example:** `Device(config-vlan-mdns-sd)# exit`	Exits the mDNS gateway configuration mode.

Configuring Custom Service Definition (CLI)

The Cisco IOS-XE supports various built-in mDNS service-definition types that map to key mDNS PTR records and user-friendly names. For example, built-in Apple-TV service-type is associated with _airplay._tcp.local and _raop._tcp.local PTR records to successfully enable service in the network. Network administrators create custom service-definition with matching mDNS PTR records to enable end mDNS service-routing in the network.

To associate the custom service-definition to the service-list, follow the procedure given below:

Procedure

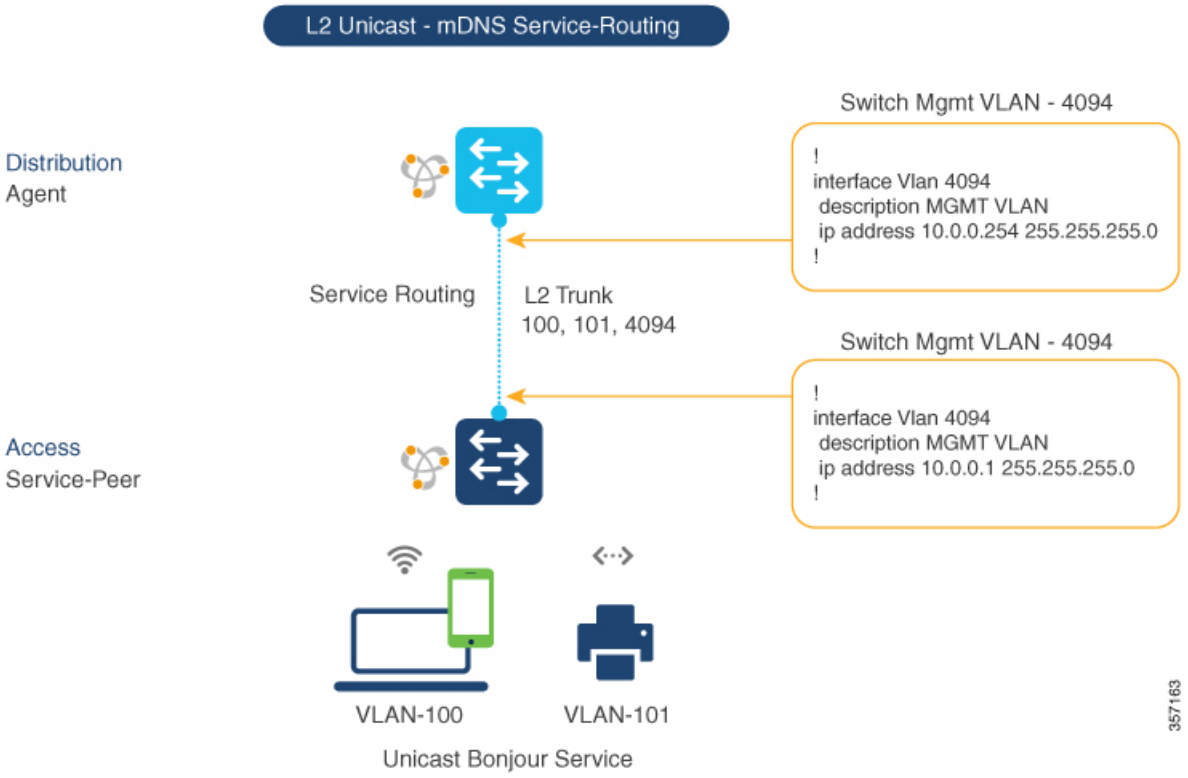
	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables Privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 3	**mdns-sd service-definition** *service-definition-name* **Example:** `Device(config)# mdns-sd service-definition APPLE-CLASSROOM`	Creates a unique service-definition name for custom service-types.
Step 4	**service-type** *custom-mDNS-PTR* **Example:** `Device(config-mdns-ser-def)# service-type _classroom._tcp.local`	Configures a regular-expression string for custom mDNS PoinTeR(PTR) record.
Step 5	**exit** **Example:** `Device(config-mdns-ser-def)# exit`	Exits the mDNS gateway configuration mode.

Configuring Service-Routing on Service-Peer (CLI)

The Layer 2 Cisco Catalyst switch in service-peer mode builds a service-routing with an upstream distribution-layer switch in the SDG Agent mode. To build service-routing, the Layer 2 Cisco Catalyst switch requires at least one interface with valid IP address to reach the upstream SDG Agent Catalyst switch. The switch management port is unsupported.

The following figure displays the topology to enable unicast-based service-routing over Layer 2 trunk between access-layer Catalyst switch in the service-peer mode and distribution-layer Catalyst switch in SDG Agent mode.

Figure 42: Catalyst Service-Peer Service-Routing Configuration



To enable service-routing on Cisco Catalyst switch in service-peer mode and setup mDNS trust interface settings, follow the procedure given below:

Procedure

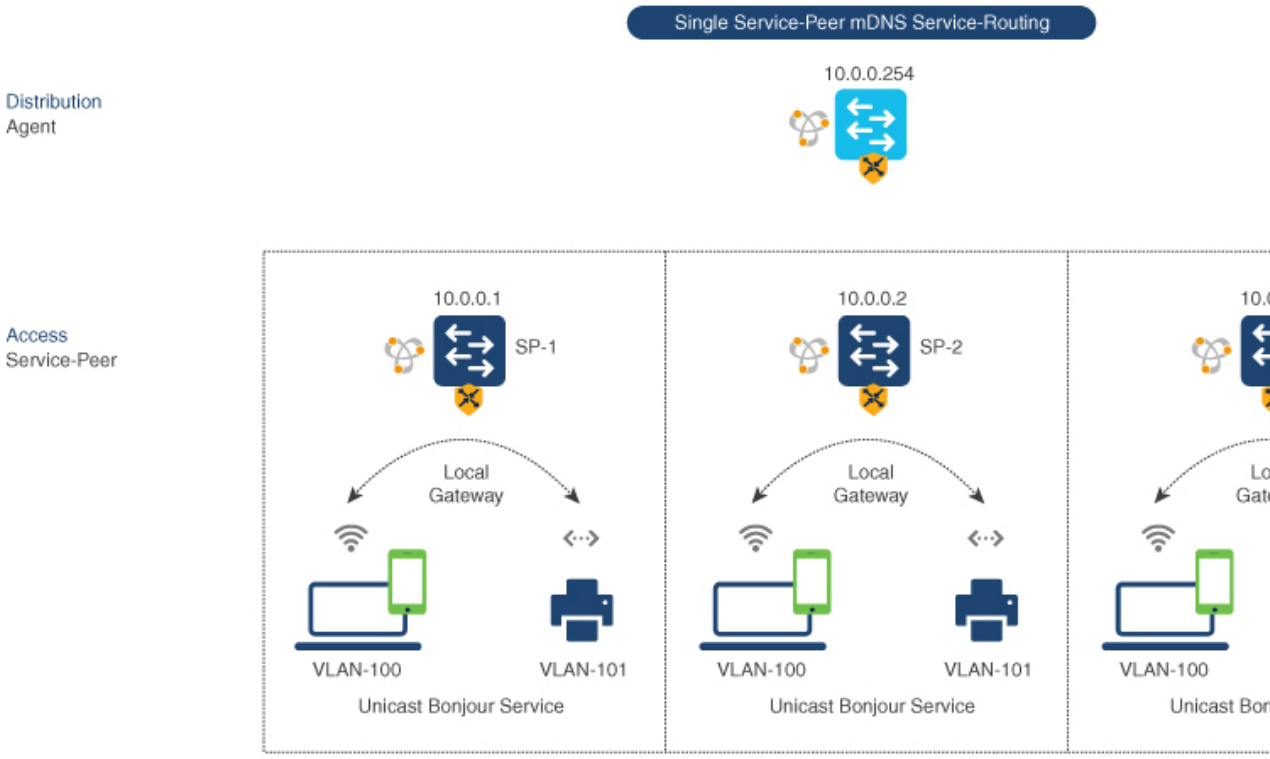
	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables Privileged EXEC mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:**	Enters global configuration mode.

	Command or Action	Purpose
	`Device# configure terminal`	
Step 3	**vlan configuration** *ID* **Example:** `Device(config)# vlan configuration 100`	Enables wired and EWC mode AP wireless user VLAN configuration for advanced service parameters. One or more VLANs can be created for the same settings. Here, *ID* refers to the VLAN configuration ID. For example, *vlan configuration 101-110, 200* range, allows to configure consecutive and non-consecutive VLAN ID(s).
Step 4	**mdns-sd gateway** **Example:** `Device(config-vlan-config)# mdns-sd gateway`	Enables mDNS gateway on configured VLAN ID(s). To enable the respective functionalities, enter the following commands in the mDNS gateway configuration mode: • **active-query timer [sec]**: Configure to enable refresh discovered services and their records with periodic mDNS Query message for permitted service types. The valid range is from 60 to 3600 seconds. The recommended value is 3600 seconds. • **service-mdns-query {ptr | srv | txt}**: Permits processing specific Query type. The default query type is PTR. • **transport {ipv4 | ipv6 | both}**: Permits processing for IPv4, IPv6, or both. It is recommended to use one network type to reduce redundant processing and respond with the same information over two network types. The default network type is IPv4.
Step 5	**source-interface** *ID* **Example:** `Device(config-vlan-mdns-sd)# source-interface vlan 4094`	Selects the interface with a valid IP address to source service-routing session with the upstream Cisco Catalyst SDG Agent switch. Typically, the management VLAN interface can be used.
Step 6	**sdg-agent** [*IPv4_address*] **Example:** `Device(config-vlan-mdns-sd)# sdg-agent 10.0.0.254`	Configures the SDG Agent IPv4 address, typically, the management VLAN gateway address. If FHRP mode, then use the FHRP virtual IP address of the management VLAN.
Step 7	**exit** **Example:** `Device(config-vlan-mdns-sd)# exit`	Exits the mDNS gateway configuration mode.

Configuring Location-Based mDNS

By default, the Layer 2 Catalyst switch in the service-peer mode enables per-switch mDNS discovery and distribution between wired and EWC mode Access Point wireless users attached locally to the switch. This default per-switch location-based mDNS is supported even when wired and EWC mode Access Point wireless users VLANs may be extended between multiple Layer 2 Catalyst switches for user mobility purpose. The mDNS service-policy configuration SDG Agent is required to accept policy-based mDNS service provider and receiver information from downstream service-peer access-layer switch.

Figure 43: Per-Switch Location-Based Wired and EWC Mode Access Point Configuration



Note Configure the mDNS service policy on the distribution layer SDG Agent switch before proceeding to the next configuration step. For more information, see the Configuring mDNS Service Policy (CLI), on page 827 section.

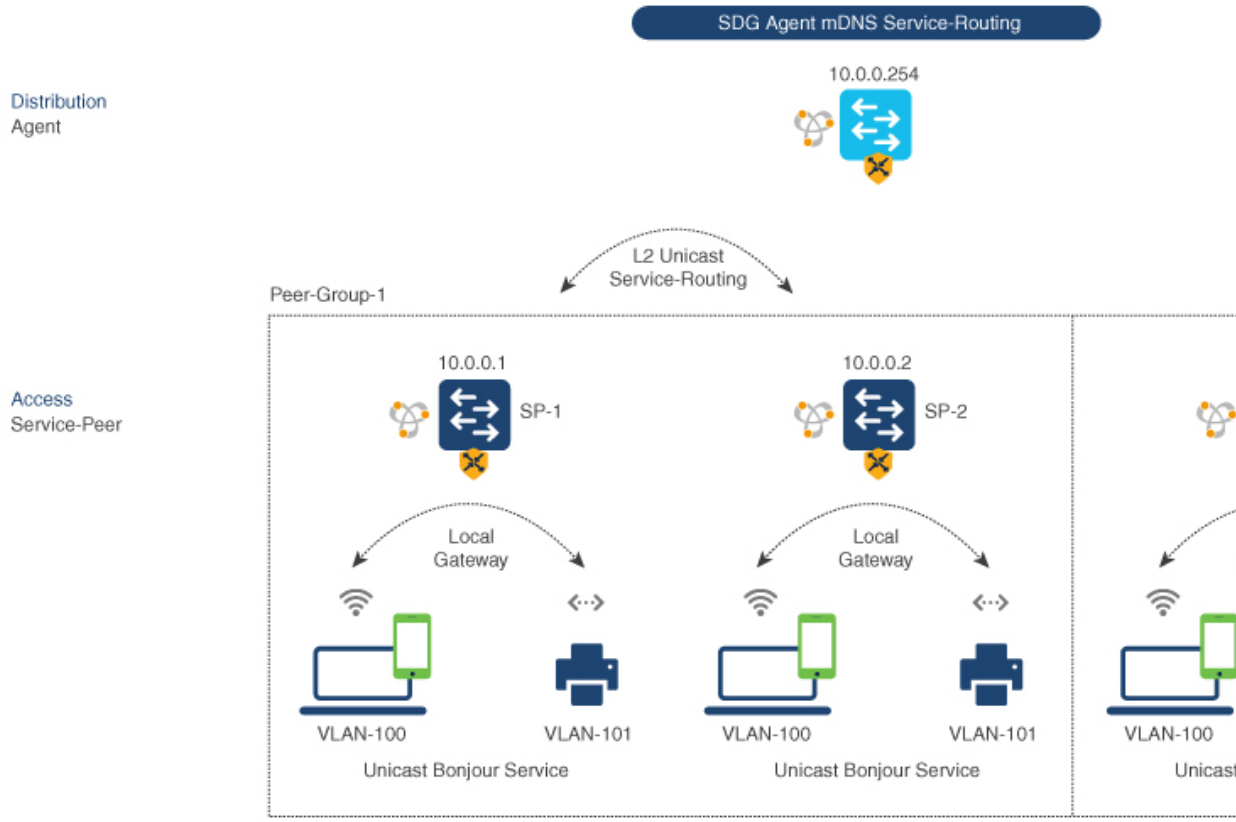
Configuring Service-Routing on SDG Agent (CLI)

The Cisco Catalyst 9000 series switches support SDG Agent mode automatically at the distribution layer and enables Unicast mode Bonjour service-routing with the downstream Layer 2 access-layer Ethernet switches connected to wired and EWC mode Access Point wireless users. The SDG Agent must be configured with mDNS service-policy on wired or EWC mode Access Point wireless user VLAN to accept mDNS service cache from downstream service-peer switches.

This section provides the step-by-step configuration to enable policy-based service discovery and distribution between locally paired Layer 2 access network switches in the service-peer mode.

The following figure displays the unicast service-routing on SDG Agent and downstream Layer 2 access network switches in the service-peer mode:

Figure 44: Catalyst SDG Agent Service-Routing Configuration



Note Configure the mDNS service policy on the distribution layer SDG Agent switch before proceeding to the next configuration step. For more information, see the Configuring mDNS Service Policy (CLI), on page 827 section.

To enable the mDNS service policy and peer-group on SDG Agent switch, and enable unicast mode service-routing with Layer 2 access network switches in Service-Peer mode, follow the steps given below:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enables Privileged EXEC mode. Enter your password, if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters the global configuration mode.
Step 3	**mdns-sd service-peer group** *service-peer-group-name* **Example:** `Device(config)# mdns-sd service-peer group service-peer-group-name`	Configures a unique service-peer group in the global configuration mode.
Step 4	**peer-group** [*ID*] **Example:** `Device(config-mdns-svc-peer)# peer-group 1`	Assigns a unique peer-group ID to the service-peers pair permitting mDNS service discovery and distribution within the assigned group list. The valid peer-group range is from 1 to 1000 for each SDG Agent switch.
Step 5	**service-policy** *service-policy-name* **Example:** `Device(config-mdns-svc-peer-grp)# service-policy VLAN100-POLICY`	Associates an mDNS service policy to accept service advertisements and query from the paired service-peers.

Step 6	**service-peer** [*IPv4_address*] **location-group** {**all** | **default** | **id**}
	Example:
	`Device(config-mdns-svc-peer-grp)# service-peer 10.0.0.1 location-group default`
	`Device(config-mdns-svc-peer-grp)# service-peer 10.0.0.2 location-group default`
		Configures atleast one service-peer to accept the mDNS service advertisement or query message. When a group has more than one service-peers, the SDG Agent provides Layer 2 unicast mode routing between the configured peers.
		For example, the SDG Agent provides unicast based service gateway function between three (10.0.0.1 and 10.0.0.2) Layer 2 service-peer switches matching the associated service-policy.
		The mDNS service information from the unpaired Layer 2 service-peer (10.0.0.3) cannot announce or receive mDNS services with the other grouped service-peers (10.0.0.1 and 10.0.0.2).

Step 7	**exit** **Example:** `Device(config-mdns-svc-peer-grp)# exit`	Exits the mDNS gateway configuration mode.

Verifying Local Area Bonjour in Service-Peer Mode

This section provides guidelines to verify various Local Area Bonjour domain mDNS service configuration parameters, cache records, statistics and more on the controller in service-peer mode

Table 40:

Command or Action	Purpose
show mdns-sd cache {all | interface | mac | name | service-peer | static | type | vlan}	Displays available mDNS cache records supporting multiple variables providing granular source details received from wired or EWC mode AP wireless user VLANs. The variables are as follows: • all – Displays all available cache records discovered from multiple source connections of a system. • interface – Displays available cache records discovered from the specified Layer 3 interface. • mac - Displays available cache records discovered from the specified MAC address. • name - Displays available cache records based on the service provider announced name. • service-peer - Displays available cache records discovered from the specified Layer 2 Service-Peer. • static – Displays locally configured static mDNS cache entry. • type – Displays available cache records based on the specific mDNS record type, such as, PTR, SRV, TXT, A or AAAA. • vlan - Displays available cache records discovered from the specified Layer 2 VLAN ID in the Unicast mode.
show mdns-sd service-definition {name | type}	Displays built-in and user-defined custom service-definition that maps service name to the mDNS PTR records. The service-definition can be filtered by name or type.
show mdns-sd service-list {direction | name}	Displays inbound or outbound direction list of configured service-list to classify matching service-types for service-policy. The list can be filtered by name or specific direction.

Command or Action	Purpose
show mdns-sd service-policy {interface | name}	Displays list of mDNS service-policy mapped with inbound or outbound service-list. The service-policy list can be filtered by an associated specified interface or name.
show mdns-sd statistics {all | cache | debug | interface | service-list | service-policy | services | vlan}	Displays detailed mDNS statistics processed bi-directionally by the system on each mDNS gateway enabled VLAN configured mDNS in Unicast mode. The expanded keyword for mDNS statistics can provide detailed view on interface, policy, service-list, and services.
show mdns-sd summary {interface | vlan}	Displays brief information about mDNS gateway and key configuration status on all wired and EWC mode AP wireless user VLANs, and interfaces of the system.

Verifying Local Area Bonjour in SDG Agent Mode

This section provides guidelines to verify various Local Area Bonjour domain mDNS service configuration parameters, cache records, statistics and more on the controller in SDG Agent mode.

Table 41:

Command or Action	Purpose
show mdns-sd cache {all | interface | mac | name | service-peer | static | type | vlan | vrf}	Displays available mDNS cache records supporting multiple variables providing granular source details. The variables are as follows: • all – Displays all available cache records discovered from multiple source connections of a system. • interface – Displays available cache records discovered from the specified Layer 3 interface. • mac - Displays available cache records discovered from the specified MAC address. • name - Displays available cache records based on the service provider announced name. • service-peer - Displays available cache records discovered from the specified Layer 2 Service-Peer. • static – Displays locally configured static mDNS cache entry. • type – Displays available cache records based on the specific mDNS record type, such as, PTR, SRV, TXT, A or AAAA. • vlan - Displays available cache records discovered from the specified Layer 2 VLAN ID in the Unicast mode. • vrf - Displays per-VRF available cache records based on specific mDNS record type, that is, PTR, SRV, TXT, A or AAAA.
show mdns-sd service-definition {name | type}	Displays built-in and user-defined custom service-definition that maps service name to the mDNS PTR records. The service-definition can be filtered by name or type.
show mdns-sd service-list {direction | name}	Displays inbound or outbound direction list of the configured service-list to classify matching service-types for service-policy. The list can be filtered by name or specific direction.
show mdns-sd service-policy {interface | name}	Displays list of mDNS service-policy mapped with inbound or outbound service-list. The service-policy list can be filtered by an associated specified interface or name.

Command or Action	Purpose
show mdns-sd statistics {all | cache | debug | interface | service-list | service-policy | services | vlan}	Displays detailed mDNS statistics processed bi-directionally by the system on each mDNS gateway enabled VLAN configured mDNS in Unicast mode. The expanded keyword for mDNS statistics can provide detailed view on interface, policy, service-list, and services.
show mdns-sd summary {interface | vlan}	Displays brief information about mDNS gateway and key configuration status on all VLANs and interfaces of the system.

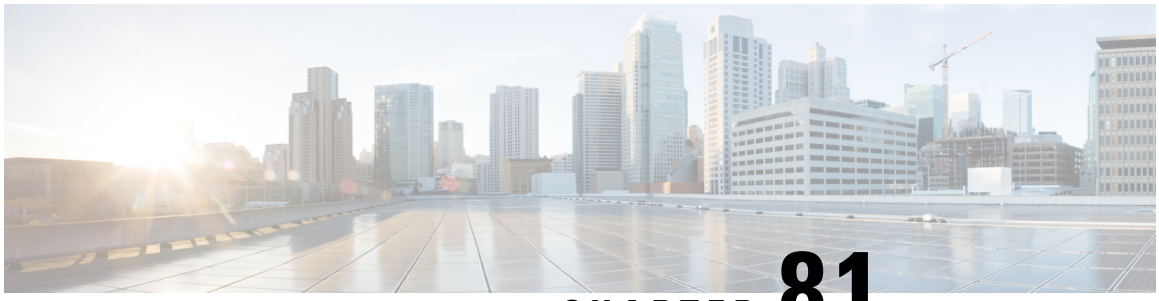
Reference

Table 42: Reference

Related Topic	Document Title
Cisco Embedded Wireless Controller on Catalyst Access Points CCO Configuration Guide	Cisco Embedded Wireless Controller on Catalyst Access Points Configuration Guide, IOS XE Bengaluru 17.5.x
DNA Service for Bonjour Deployment on Cisco Catalyst 9600 Switch	Cisco Catalyst 9600 Series Switch Software Configuration Guide, Release 17.4.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9500 Switch	Cisco Catalyst 9500 Series Switch Software Configuration Guide, Release 17.4.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9400 Switch	Cisco Catalyst 9400 Series Switch Software Configuration Guide, Release 17.4.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9300 Switch	Cisco Catalyst 9300 Series Switch Software Configuration Guide, Release 17.4.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9800 Wireless LAN Controller	Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide, Cisco IOS XE Bengaluru 17.5.x
Cisco Wide Area Bonjour Application on Cisco DNA Center User Guide	Cisco Wide Area Bonjour Application on Cisco DNA Center User Guide, Release 2.2.x

PART XIII

Multicast Domain Name System



CHAPTER 81

Multicast Domain Name System

Introduction to mDNS Gateway

Multicast Domain Name System (mDNS) is an Apple service discovery protocol which locates devices and services on a local network with the use of mDNS service records.

The Bonjour protocol operates on service announcements and queries. Each query or advertisement is sent to the Bonjour multicast address ipv4 224.0.0.251 (ipv6 FF02::FB). This protocol uses mDNS on UDP port 5353.

The address used by the Bonjour protocol is link-local multicast address and therefore is only forwarded to the local L2 network. As, multicast DNS is limited to an L2 domain for a client to discover a service it has to be part of the same L2 domain, This is not always possible in any large scale deployment or enterprise.

In order to address this issue, the Cisco Catalyst 9800 Series Wireless Controller acts as a Bonjour Gateway. The controller then listens for Bonjour services, caches these Bonjour advertisements (AirPlay, AirPrint, and so on) from the source or host. For example, Apple TV responds back to Bonjour clients when asked or requested for a service. This way you can have sources and clients in different subnets.

By default, the mDNS gateway is disabled on the controller. To enable mDNS gateway functionality, you must explicitly configure mDNS gateway using CLI or Web UI.

Prerequisite

Since the Cisco Catalyst 9800 Series Wireless Controller will respond and advertise for services cached when acting as a Bonjour Gateway, it must have an SVI interface with a valid IP address on every VLAN where mDNS is allowed or used. This will be the source IP address of those mDNS packets that are coming out from the controller acting as mDNS Gateway.

Enabling mDNS Gateway (GUI)

Procedure

Step 1 Choose **Configuration > Services > mDNS**.

Step 2 In the **Global** section, toggle the slider to enable or disable the **mDNS Gateway**.

Step 3 From the **Transport** drop-down list, choose one of the following types:

- **ipv4**
- **ipv6**
- **both**

Step 4 Enter an appropriate timer value in **Active-Query Timer**. The valid range is between 15 to 120 minutes. The default is 30 minutes.

Step 5 From the **mDNS-AP Service Policy** drop-down list, choose an mDNS service policy.

Note Service policy is optional only if mDNS-AP is configured. If mDNS-AP is not configured, the system uses default-service-policy.

Step 6 Click **Apply**.

Enabling or Disabling mDNS Gateway (CLI)

Note

- mDNS gateway is disabled by default globally on the controller.
- You need both global and WLAN configurations to enable mDNS gateway.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged **EXEC** mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**mdns-sd gateway** **Example:** `Device(config)# mdns-sd gateway`	Enables mDNS gateway.
Step 4	**transport** {**ipv4** | **ipv6** | **both**} **Example:** `Device(config-mdns-sd)# transport ipv4`	Processes mDNS message on a specific transport. Here, **ipv4** signifies that the IPv4 mDNS message processing is enabled. This is the default value. **ipv6** signifies that the IPv6 mDNS message processing is enabled. **both** signifies that the IPv4 and IPv6 mDNS message is enabled for each network.
Step 5	**active-query timer** *active-query-periodicity* **Example:** `Device(config-mdns-sd)# active-query timer 15`	Changes the periodicity of mDNS multicast active query. **Note** An active query is a periodic mDNS query to refresh dynamic cache. Here, *active-query-periodicity* refers to the active query periodicity in Minutes. The valid range is from 15 to 120 minutes. Active query runs with a default periodicity of 30 minutes.
Step 6	**exit** **Example:** `Device(config-mdns-sd)# exit`	Returns to global configuration mode.

Creating Custom Service Definition (GUI)

Procedure

Step 1 Choose **Configuration** > **Services** > **mDNS**.

Step 2 In the **Service Definition** section, click **Add**.

Step 3 In the **Quick Setup: Service Definition** page that is displayed, enter a name and description for the service definition.

Step 4 Enter a service type and click **+** to add the service type.

Step 5 Click **Apply to Device**.

Creating Custom Service Definition

Service definition is a construct that provides an admin friendly name to one or more mDNS service types or A pointer (PTR) Resource Record Name.

By default, few built-in service definitions are already predefined and available for admin to use.

In addition to built-in service definitions, admin can also define custom service definitions.

You can execute the following command to view the list of all the service definitions (built-in and custom):

```
Device# show mdns-sd master-service-list
```

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged **EXEC** mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**mdns-sd service-definition** *service-definition-name* **Example:** `Device(config)# mdns-sd service-definition CUSTOM1`	Configures mDNS service definition. **Note** • All the created custom service definitions are added to the primary service list. • Primary service list comprises of a list of custom and built-in service definitions.

	Command or Action	Purpose
Step 4	**service-type** *string* **Example:** `Device(config-mdns-ser-def)# service-type _custom1._tcp.local`	Configures mDNS service type.
Step 5	**exit** **Example:** `Device(config-mdns-ser-def)# exit`	Returns to global configuration mode.

Creating Service List (GUI)

Procedure

Step 1 Choose **Configuration > Services > mDNS**.

Step 2 In the **Service List** section, click **Add**.

Step 3 In the **Quick Setup: Service List** page that is displayed, enter a name for the service list.

Step 4 From the **Direction** drop-down list, choose **IN** for inbound filtering or **OUT** for outbound filtering.

Step 5 From the **Available Services** drop-down list, choose a service type to match the service list.

Note To allow all services, choose the **all** option.

Step 6 Click **Add Services**.

Step 7 From the **Message Type** drop-down list, choose the message type to match from the following options:

- **any**—To allow all messages.
- **announcement**—To allow only service advertisements or announcements for the device.
- **query**—To allow only a query from the client for a service in the network.

Step 8 Click **Save** to add services.

Step 9 Click **Apply to Device**.

Creating Service List

mDNS service list is a collection of service definitions.

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged **EXEC** mode.

	Command or Action	Purpose
	Example: `Device> enable`	Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 3	**mdns-sd service-list** *service-list-name* {**IN** | **OUT**}
	Example:
	`Device(config)# mdns-sd service-list Basic-In IN`
	`Device(config)# mdns-sd service-list Basic-Out OUT`
		Configures mDNS service list.
		• **IN**: Provides inbound filtering.
		• **Out**: Provides outbound filtering.

Step 4	**match** *service-definition-name* **message-type** {**announcement** | **any** | **query**} **Example:** `Device(config-mdns-sl-in)# match CUSTOM1 message-type query`	Matches the service to the message type. Here, *service-definition-name* refers to the names of services, such as, airplay, airserver, airtunes, and so on. **Note** To add a service, the service name must be part of the primary service list. If the mDNS service list is set to IN, you get to view the following command: **match service-definition-name message-type {announcement | any | query}**. If the mDNS service list is set to Out, you get to view the following command: **match service-definition-name**.
Step 5	**show mdns-sd service-list {direction | name }**	Displays inbound or outbound direction list of the configured service-list to classify matching service-types for service-policy. The list can be filtered by name or specific direction.
Step 6	**exit** **Example:** `Device(config-mdns-sl-in)# exit`	Returns to global configuration mode.

Creating Service Policy (GUI)

Procedure

Step 1 Choose **Configuration > Services > mDNS**.
Step 2 In the **Service Policy** section, click **Add**.
Step 3 In the **Quick Setup: Service Policy** page that is displayed, enter a name for the service policy.
Step 4 From the **Service List Input** drop-down list, choose one of the types.
Step 5 From the **Service List Output** drop-down list, choose one of the types.
Step 6 From the **Location** drop-down list, choose the location you want to associate with the service list.
Step 7 Click **Apply to Device**.

Creating Service Policy

mDNS service policy is used for service filtering while learning services or responding to queries.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged **EXEC** mode. Enter your password, if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**mdns-sd service-policy** *service-policy-name* **Example:** `Device(config)# mdns-sd service-policy mdns-policy1`	Enables mDNS service policy.
Step 4	**location** {**lss** | **site-tag**} **Example:**	Filters mDNS service types based on LSS or site-tag.

	Command or Action	Purpose
	`Device(config-mdns-ser-pol)# location lss`	**Note** In Location Specific Services (LSS) based filtering, the mDNS gateway responds with the service instances learnt from the neighboring APs of the querying client AP. Other service instances for the rest of APs are filtered. In Site tag based filtering, the mDNS gateway responds with the service instances that belong to the same site-tag as that of querying client. The mDNS gateway responds back with wired services even if the location based filtering is configured.
Step 5	**service-list** *service-list-name* {**IN** | **OUT**} **Example:** `Device(config-mdns-ser-pol)# service-list VLAN100-list IN`	Configures various service-list names for IN and OUT directions. **Note** If an administrator decides to create or use a custom service policy, then the custom service policy must be configured with service-lists for both directions (IN and OUT); otherwise, the mDNS Gateway will not work (will not learn services if there is no IN service-list, or will not reply or announce services learned if there is no OUT service-list).
Step 6	**exit** **Example:** `Device(config-mdns-ser-pol)# exit`	Returns to global configuration mode.

Configuring a Local or Native Profile for an mDNS Policy

When an administrator configures local authentication and authorization and does not expect to get any mDNS policy from the AAA server, the administrator can configure a local or native profile to select a mDNS policy based on user, role, or device type. When this local or native profile is mapped to the wireless profile policy, mDNS service policy is applied on the mDNS packets that are processed on that WLAN.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**service-template** *template-name* **Example:** `Device(config)# service-template mdns`	Configures the service-template or identity policy.
Step 3	**mdns-service-policy** *mdns-policy-name* **Example:** `Device(config-service-template)# mdns-service-policy mdnsTV`	Configures the mDNS policy.
Step 4	**exit** **Example:** `Device(config-service-template)# exit`	Returns to global configuration mode.

Configuring an mDNS Flex Profile (GUI)

Procedure

Step 1 Choose **Configuration > Services > mDNS**.

Step 2 In the **mDNS Flex Profile** section, click **Add**.
The **Add mDNS Flex Profile** window is displayed.

Step 3 In the **Profile Name** field, enter the flex mDNS profile name.

Step 4 In the **Service Cache Update Timer** field, specify the service cache update time. The default value is 1 minute. The valid range is from 1 to 100 minutes.

Step 5 In the **Statistics Update Timer** field, specify the statistics update timer. The default value is 1 minute. The valid range is from 1 to 100 minutes.

Step 6 In the **VLANs** field, specify the VLAN ID. You can enter multiple VLAN IDs separated by commas, or enter a range of VLAN IDs. Maximum number of VLANs allowed is 16.

Step 7 Click **Apply to Device**.

Configuring an mDNS Flex Profile (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**mdns-sd flex-profile** *mdns-flex-profile-name* **Example:** `Device(config)# mdns-sd flex-profile mdns-flex-profile-name`	Enters the mDNS Flex Profile mode.
Step 3	**update-timer service-cache** *service-cache timer-value <1-100>* **Example:** `Device(config-mdns-flex-profile)# update-timer service-cache 60`	Configures the mDNS update service cache timer for the flex profile. The default value is 1 minute. Value range is between 1 minute and 100 minutes.
Step 4	**update-timer statistics** *statistics timer-value <1-100>* **Example:** `Device(config-mdns-flex-profile)# update-timer statistics 65`	Configures the mDNS update statistics timer for the flex profile. The default value is 1 minute. The valid range is from 1 to 100 minutes.
Step 5	**wired-vlan-range** *wired-vlan-range value* **Example:** `Device(config-mdns-flex-profile)# wired-vlan-range 10 - 20`	Configures the mDNS wired VLAN range for the flex profile. The default value is 1 minute. The valid range is from 1 minute to 100 minutes.

Applying an mDNS Flex Profile to a Wireless Flex Connect Profile (GUI)

Procedure

Step 1 Choose **Configuration > Tags & Profiles > Flex**.

Step 2 Click **Add**.
The **Add Flex Profile** window is displayed.

Step 3 Under the **General** tab, from the **mDNS Flex Profile** drop-down list, choose a flex profile name from the list.

Step 4 Click **Apply to Device**.

Applying an mDNS Flex Profile to a Wireless Flex Connect Profile (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile flex** *wireless-flex-profile-name* **Example:** `Device# wireless profile flex wireless-flex-profile-name`	Enters wireless flex profile configuration mode.
Step 3	**mdns-sd** *mdns-flex-profile* **Example:** `Device(config-wireless-flex-profile)# mdns-sd mdns-flex-profile-name`	Enables the mDNS features for all the APs in the profile

Location-Based Service Filtering

Prerequisite for Location-Based Service Filtering

You need to create the Service Definition and Service Policy. For more information, see Creating Custom Service Definition section and Creating Service Policy section.

Configuring mDNS Location-Based Filtering Using SSID

When a service policy is configured with the SSID as the location name, the response to the query will be the services that were learnt on that SSID.

Procedure

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

	Command or Action	Purpose
Step 2	**mdns-sd service-policy** *service-policy-name* **Example:** `Device(config)# mdns-sd service-policy mdns-policy1`	Configures the service policy.
Step 3	**location ssid** **Example:** `Device(config-mdns-ser-pol)# location ssid`	Configures location-based filtering using SSID.
Step 4	**end** **Example:** `Device(config-mdns-ser-pol)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring mDNS Location-Based Filtering Using AP Name

When a service policy is configured with the AP name as the location, the response to the query will be the services that were learnt on that AP.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**mdns-sd service-policy** *service-policy-name* **Example:** `Device(config)# mdns-sd service-policy mdns-policy1`	Configures the service policy.
Step 3	**location ap-name** **Example:** `Device(config-mdns-ser-pol)# location ap-name`	Configures location-based filtering using an AP name.
Step 4	**end** **Example:** `Device(config-mdns-ser-pol)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring mDNS Location-Based Filtering Using AP Location

When a service policy is configured with location as the AP-location, the response to the query will be the services that were learnt on all the APs using the same AP "location" name (not to be confused with "site-tag").

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**mdns-sd service-policy** *service-policy-name* **Example:** `Device(config)# mdns-sd service-policy mdns-policy1`	Configures the service policy.
Step 3	**location ap-location** **Example:** `Device(config-mdns-ser-pol)# location ap-location`	Configures location-based filtering using the AP location.
Step 4	**end** **Example:** `Device(config-mdns-ser-pol)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring mDNS Location-Based Filtering Using Regular Expression

- When a service policy is configured with the location as a regular expression that matches the corresponding AP name, the response to the query will be the services that were learnt on a group of APs based on the AP name.
- When a service policy is configured with the location as a regular expression that matches the corresponding AP location, the response to the query will be the services that were learnt on a group of APs based on the AP location.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**mdns-sd service-policy** *service-policy-name* **Example:** `Device(config)# mdns-sd service-policy mdns-policy1`	Configures the service policy.
Step 3	**location regex** {**ap-location** *regular-expression* | **ap-name** *regular-expression*} **Example:**	Configures location-based filtering using regular expression.

	Command or Action	Purpose
	`Device(config-mdns-ser-pol)# location regex ap-location dns_location` `Device(config-mdns-ser-pol)# location regex ap-name dns_name`	
Step 4	**end** **Example:** `Device(config-mdns-ser-pol)# end` **Note** To filter the services for which AP names have the specific keyword such as *AP-2FLR-SJC-123*, you can use the regex AP name as *AP-2FLR-* to match the services that are learnt from the set of access points.	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring mDNS AP

In most of the deployments, the services may be available in VLANs that the APs can hear in the wired side (allowed in the switchport where the AP is directly connected: its own VLAN, or even more VLANs if switchport is a trunk).

The following procedure shows how to configure mDNS AP:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**mdns-sd gateway** **Example:** `Device(config)# mdns-sd gateway`	Configures the mDNS gateway.
Step 3	**ap name** *ap-name* **mdns-ap enable vlan** *vlan-id* **Example:** `Device# ap name ap1 mdns-ap enable vlan 22`	Enables mDNS on the AP, and configures a VLAN for the mDNS AP.
Step 4	**ap name** *ap-name* **mdns-ap vlan add** *vlan-id* **Example:** `Device# ap name ap1 mdns-ap vlan add 200`	Adds a VLAN to the mDNS AP. *vlan-id* ranges from 1 to 4096.

	Command or Action	Purpose
Step 5	**ap name** *ap-name* **mdns-ap vlan del** *vlan-id* **Example:** `Device# ap name ap1 mdns-ap vlan del 2`	Deletes a VLAN from the mDNS AP.
Step 6	**ap name** *ap-name* **mdns-ap disable** **Example:** `Device# ap name ap1 mdns-ap disable`	(Optional) Disables the mDNS AP.
Step 7	**end** **Example:** `Device# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode. **Note** You can configure a maximum of 10 VLANs per AP.

Associating mDNS Service Policy with Wireless Profile Policy (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **Policy**.

Step 2 Click the **policy profile** name.

Step 3 In the **Advanced** tab, choose the mDNS service policy from the **mDNS Service Policy** drop-down list.

Step 4 Click **Update & Apply to Device**.

Associating mDNS Service Policy with Wireless Profile Policy

Note You must globally configure the mDNS service policy before associating it with the wireless profile policy.

A default mDNS service policy is already attached once the wireless profile policy is created. You can use the following commands to override the default mDNS service policy with any of your service policy:

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless profile policy** *profile-policy* **Example:** `Device(config)# wireless profile policy default-policy-profile`	Configures wireless profile policy. Here, *profile-policy* refers to the name of the WLAN policy profile.
Step 3	**mdns-sd service-policy** *custom-mdns-service-policy* **Example:** `Device(config-wireless-policy)# mdns-sd service-policy custom-mdns-service-policy`	Associates an mDNS service policy with the wireless profile policy. The default mDNS service policy name is **default-mdns-service-policy**. **Note** The **default-mdns-profile-policy** uses **default-mdns-service-list** configuration for filtering mDNS service announcement and queries. In wireless network, the mDNS packets are consumed by the mDNS gateway and clients or device is deprived of learning this service. To share the service with the device and provide ease of configuration to the administrator, a list of few standard service types are shared by default on the wireless network. The list of such standard service types is termed as default service policy that comprises a set of service types. The table covers a sample service list in the default service policy.

Table 43: Default Name and mDNS Service Type

Default Name	mDNS Service Type
Apple HomeSharing	_home-sharing._tcp.local
Printer-IPPS	_ipps._tcp.local
Google-chromecast	_googlecast._tcp.local

	Command or Action	Purpose
		Note • Location would be disabled on mDNS default service policy. • You cannot change the contents of the mDNS default service policy. However, you can create separate mDNS service policies and associate them under the wireless policy profile.
Step 4	**exit** **Example:** `Device(config-wireless-policy)# exit`	Returns to global configuration mode.

Enabling or Disabling mDNS Gateway for WLAN (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **WLANs**.
Step 2 Click on the WLAN.
Step 3 In the **Advanced** tab, choose the mode in **mDNS Mode** drop-down list.
Step 4 Click **Update & Apply to Device**.

Enabling or Disabling mDNS Gateway for WLAN

Note Bridging is the default behaviour. This means that the mDNS packets are always bridged.

Procedure

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

	Command or Action	Purpose
Step 2	**wlan** *profile-name wlan-id ssid-name* **Example:** `Device(config)# wlan test 24 ssid1`	Specifies the WLAN name and ID. • *profile-name* is the WLAN name which can contain 32 alphanumeric characters • *wlan-id* is the wireless LAN identifier. The valid range is from 1 to 512. • *ssid-name* is the SSID which can contain 32 alphanumeric characters. **Note** Global configuration must be in place for mDNS gateway to work.

Step 3	**mdns-sd-interface** {**gateway** | **drop**}
	Example:
	`Device(config-wlan)# mdns-sd gateway`
	`Device(config-wlan)# mdns-sd drop`
		Enables or disables mDNS gateway and bridge functions on WLAN.

Step 4	**exit** **Example:** `Device(config-wlan)# exit`	Returns to global configuration mode.
Step 5	**show wlan name** *wlan-name* | **show wlan all** **Example:** `Device# show wlan name test | show wlan all`	Verifies the status of mDNS on WLAN.
Step 6	**show wireless profile policy** **Example:** `Device# show wireless profile policy`	Verifies the service policy configured in WLAN.

Verifying mDNS Gateway Configurations

To verify the mDNS summary, use the following command:

```
Device# show mdns-sd summary
mDNS Gateway: Enabled
Active Query: Enabled
  Periodicity (in minutes): 30
Transport Type: IPv4
```

To verify the mDNS cache, use the following command:

```
Device# show mdns-sd cache
------------------------------------------------------------ PTR Records
----------------------------------------
RECORD-NAME                     TTL      WLAN   CLIENT-MAC       RR-RECORD-DATA
```

```
-------------------------------------------------------------------------------------------------------------
_airplay._tcp.local             4500     30     07c5.a4f2.dc01   CUST1._airplay._tcp.local

_ipp._tcp.local                 4500     30     04c5.a4f2.dc01   CUST3._ipp._tcp.local2

_ipp._tcp.local                 4500     15     04c5.a4f2.dc01   CUST3._ipp._tcp.local4

_ipp._tcp.local                 4500     10     04c5.a4f2.dc01   CUST3._ipp._tcp.local6

_veer_custom._tcp.local         4500     10     05c5.a4f2.dc01
CUST2._veer_custom._tcp.local8
```

To verify the mDNS cache from wired service provider, use the following command:

```
Device# show mdns-sd cache wired
------------------------------------------------------------ PTR Records
---------------------------------------
RECORD-NAME                             TTL      VLAN      CLIENT-MAC       RR-RECORD-DATA

-------------------------------------------------------------------------------------------------------------
_airplay._tcp.local                       4500     16        0866.98ec.97af
wiredapple._airplay._tcp.local
_raop._tcp.local                          4500     16        0866.98ec.97af
086698EC97AF@wiredapple._raop._tcp.local

------------------------------------------------------------ SRV Records
-----------------------------------------
RECORD-NAME                             TTL      VLAN      CLIENT-MAC       RR-RECORD-DATA

-------------------------------------------------------------------------------------------------------------
wiredapple._airplay._tcp.local            4500     16        0866.98ec.97af   0 0 7000
wiredapple.local
086698EC97AF@wiredapple._raop._tcp.local   4500     16        0866.98ec.97af   0 0 7000
wiredapple.local

------------------------------------------------------------ A/AAAA Records
---------------------------------------
RECORD-NAME                             TTL      VLAN      CLIENT-MAC       RR-RECORD-DATA

-------------------------------------------------------------------------------------------------------------
wiredapple.local                          4500     16        0866.98ec.97af
2001:8:16:16:e5:c446:3218:7437

------------------------------------------------------------ TXT Records
-----------------------------------------
RECORD-NAME                             TTL      VLAN      CLIENT-MAC       RR-RECORD-DATA

-------------------------------------------------------------------------------------------------------------
wiredapple._airplay._tcp.local            4500     16        0866.98ec.97af
[343]'acl=0''deviceid=08:66:98:EC:97:AF''features=
086698EC97AF@wiredapple._raop._tcp.local   4500     16        0866.98ec.97af
[193]'cn=0,1,2,3''da=true''et=0,3,5''ft=0x5A7FFFF7
```

To verify the mdns-sd type PTR, use the following command:

```
Device# show mdns-sd cache type {PTR | SRV | A-AAA | TXT}
RECORD-NAME                                TTL      WLAN      CLIENT-MAC
RR-Record-Data
-------------------------------------------------------------------------------------------------------------
_custom1._tcp.local                        4500     2         c869.cda8.77d6
service_t1._custom1._tcp.local
_custom1._tcp.local                        4500     2         c869.cda8.77d6
vk11._custom1._tcp.local
_ipp._tcp.local                            4500     2         c869.cda8.77d6
service-4._ipp._tcp.local
```

To verify the mdns-sd cache for a client MAC, use the following command:

```
Device# show mdns-sd cache {ap-mac <ap-mac> | client-mac <client-mac> | wlan-id <wlan-id>
| wired}
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
RR-Record-Data
-------------------------------------------------------------------------------------------
_custom1._tcp.local                            4500     2         c869.cda8.77d6
service_t1._custom1._tcp.local
_custom1._tcp.local                            4500     2         c869.cda8.77d6
vk11._custom1._tcp.local
_ipp._tcp.local                                4500     2         c869.cda8.77d6
service-4._ipp._tcp.local

------------------------------------------------------------ SRV Records
-------------------------------------------------------------
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
RR-Record-Data
-------------------------------------------------------------------------------------------
service-4._ipp._tcp.local                     4500     2         c869.cda8.77d6   0 0 1212
 mDNS-Client1s-275.local
vk11._custom1._tcp.local                       4500     2         c869.cda8.77d6   0 0 987
 mDNS-Client1s-275.local
service_t1._custom1._tcp.local                 4500     2         c869.cda8.77d6   0 0 197
 mDNS-Client1s-275.local

------------------------------------------------------------ A/AAAA Records
-------------------------------------------------------------
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
RR-Record-Data
-------------------------------------------------------------------------------------------
mDNS-Client1s-275.local                       4500     2         c869.cda8.77d6   120.1.1.33


------------------------------------------------------------ TXT Records
-------------------------------------------------------------
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
RR-Record-Data
-------------------------------------------------------------------------------------------
service-4._ipp._tcp.local                     4500     2         c869.cda8.77d6   'CLient1'

vk11._custom1._tcp.local                       4500     2         c869.cda8.77d6
'txtvers=11'
service_t1._custom1._tcp.local                 4500     2         c869.cda8.77d6
'txtvers=12'
```

To verify the mdns-sd cache in detail, use the following command:

```
Device# show mdns-sd cache detail

Name: _custom1._tcp.local
  Type: PTR
  TTL: 4500
  WLAN: 2
  WLAN Name: mdns120
  VLAN: 120
  Client MAC: c869.cda8.77d6
  AP Ethernet MAC: 7069.5ab8.33d0
  Expiry-Time: 09/09/18 21:50:47
  Site-Tag: default-site-tag
  Rdata: service_t1._custom1._tcp.local
```

To verify the mdns-sd statistics, use the following command:

```
Device# show mdns-sd statistics

------------------------------------------------------
Consolidated mDNS Packet Statistics
------------------------------------------------------
mDNS stats last reset time: 03/11/19 04:17:35
mDNS packets sent: 61045
  IPv4 sent: 30790
    IPv4 advertisements sent: 234
    IPv4 queries sent: 30556
  IPv6 sent: 30255
    IPv6 advertisements sent: 17
    IPv6 queries sent: 30238
  Multicast sent: 57558
    IPv4 sent: 28938
    IPv6 sent: 28620
mDNS packets received: 72796
  advertisements received: 13604
  queries received: 59192
  IPv4 received: 40600
    IPv4 advertisements received: 6542
    IPv4 queries received: 34058
  IPv6 received: 32196
    IPv6 advertisements received: 7062
    IPv6 queries received: 25134
mDNS packets dropped: 87

------------------------------------------------------
Wired mDNS Packet Statistics
------------------------------------------------------
mDNS stats last reset time: 03/11/19 04:17:35
mDNS packets sent: 61033
  IPv4 sent: 30778
    IPv4 advertisements sent: 222
    IPv4 queries sent: 30556
  IPv6 sent: 30255
    IPv6 advertisements sent: 17
    IPv6 queries sent: 30238
  Multicast sent: 57558
    IPv4 sent: 28938
    IPv6 sent: 28620
mDNS packets received: 52623
  advertisements received: 1247
  queries received: 51376
  IPv4 received: 32276
    IPv4 advertisements received: 727
    IPv4 queries received: 31549
  IPv6 received: 20347
    IPv6 advertisements received: 520
    IPv6 queries received: 19827
mDNS packets dropped: 63

------------------------------------------------------
mDNS Packet Statistics, for WLAN: 2
------------------------------------------------------
mDNS stats last reset time: 03/11/19 04:17:35
mDNS packets sent: 12
  IPv4 sent: 12
    IPv4 advertisements sent: 12
    IPv4 queries sent: 0
  IPv6 sent: 0
    IPv6 advertisements sent: 0
    IPv6 queries sent: 0
  Multicast sent: 0
```

```
    IPv4 sent: 0
    IPv6 sent: 0
mDNS packets received: 20173
  advertisements received: 12357
  queries received: 7816
  IPv4 received: 8324
    IPv4 advertisements received: 5815
    IPv4 queries received: 2509
  IPv6 received: 11849
    IPv6 advertisements received: 6542
    IPv6 queries received: 5307
mDNS packets dropped: 24
```

To verify the default service list details, use the following command:

```
Device# show mdns-sd default-service-list

-----------------------------------------
        mDNS Default Service List
-----------------------------------------

Service Definition: airplay
Service Names: _airplay._tcp.local

Service Definition: airtunes
Service Names: _raop._tcp.local

Service Definition: homesharing
Service Names: _home-sharing._tcp.local

Service Definition: printer-ipp
Service Names: _ipp._tcp.local

Service Definition: printer-lpd
Service Names: _printer._tcp.local

Service Definition: printer-ipps
Service Names: _ipps._tcp.local

Service Definition: printer-socket
Service Names: _pdl-datastream._tcp.local

Service Definition: google-chromecast
Service Names: _googlecast._tcp.local

Service Definition: itune-wireless-devicesharing2
Service Names: _apple-mobdev2._tcp.local
```

To verify the primary service list details, use the following command:

```
Device# show mdns-sd master-service-list

-----------------------------------------
        mDNS Master Service List
-----------------------------------------

Service Definition: fax
Service Names: _fax-ipp._tcp.local

Service Definition: roku
Service Names: _rsp._tcp.local

Service Definition: airplay
Service Names: _airplay._tcp.local
```

```
Service Definition: scanner
Service Names: _scanner._tcp.local

Service Definition: spotify
Service Names: _spotify-connect._tcp.local

Service Definition: airtunes
Service Names: _raop._tcp.local

Service Definition: airserver
Service Names: _airplay._tcp.local
               _airserver._tcp.local

.
.
.

Service Definition: itune-wireless-devicesharing2
Service Names: _apple-mobdev2._tcp.local
```

To verify the mDNS-AP configured on the controller and VLAN(s) associated with it, use the following command:

```
Device# show mdns-sd ap

Number of mDNS APs.............................. 1

AP Name   Ethernet MAC   Number of Vlans   Vlanidentifiers
-----------------------------------------------------------------------------------------------------
AP3600-1   7069.5ab8.33d0          1      300
```

Further Debug

To debug mDNS further, use the following procedure:

1. Run this command at the controller:

   ```
   set platform software trace wncd <0-7> chassis active R0 mdns debug
   ```

2. Reproduce the issue.

3. Run this command to gather the traces enabled:

```
show wireless loadbalance ap affinity wncd 0

AP MAC   Discovery Timestamp  Join Timestamp                Tag    Vlanidentifiers
-----------------------------------------------------------------------------------------
0cd0.f894.0600      06/30/21 12:39:48    06/30/21 12:40:021 default-site-tag     300
```