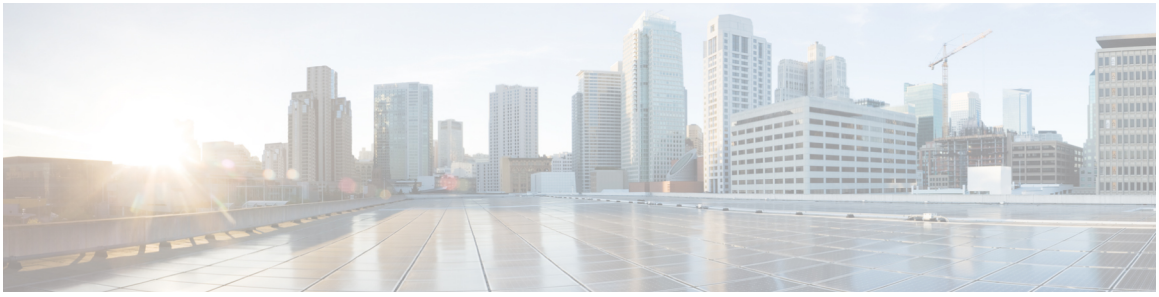
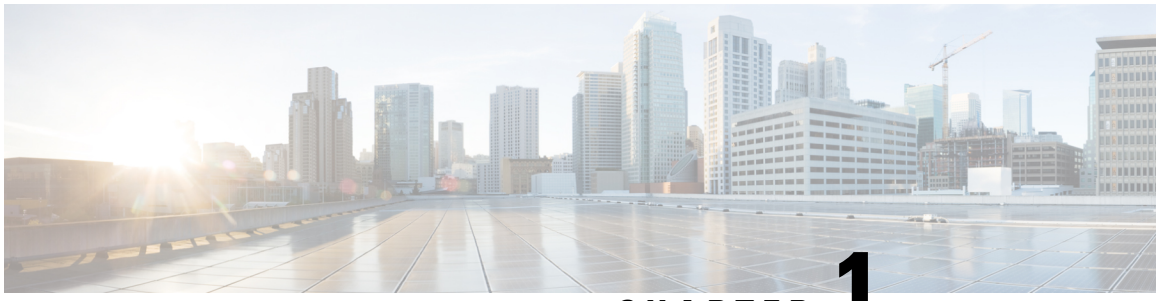
Cisco Catalyst IW9165E Rugged Access Point and Wireless Client Configuration Guide, Cisco IOS XE 17.15.x

First Published: 2024-08-14

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



CHAPTER 1

Introduction

Overview of the Access Point

The Cisco Catalyst IW9165E Rugged Access Point and Wireless Client (hereafter referred to as *IW9165E*) supports a 2x2 Wi-Fi 6E design with external antennas, and it is designed to add ultra-reliable wireless connectivity to moving vehicles and machines. Low power consumption, rugged IP30 design and small form factor make the Catalyst IW9165E very simple to integrate into industrial assets.

The IW9165E is designed to add ultrareliable wireless connectivity to moving vehicles and machines. The IW9165E can operate as Cisco Ultra-Reliable Wireless Backhaul (Cisco URWB) starting from Cisco Unified Industrial Wireless (UIW) software release 17.12.1, which delivers high availability, low latency, and zero packet loss with seamless handoffs.

Starting from Cisco Unified Industrial Wireless Software Release 17.13.1, the IW9165E can also operate as a Wi-Fi client in Workgroup Bridge (WGB) mode, which allows it to connect to a Cisco access point infrastructure, and Universal WGB (uWGB) mode, which allows it to connect to a third-party access point infrastructure. Both of these modes help bridge the wired clients that are behind the WGB to the access point on the infrastructure side.

From Cisco Unified Industrial Wireless Software Release 17.14.1, The Catalyst IW9167E can operate in Lightweight AP (control and provisioning of wireless access points (CAPWAP)) mode or Ultra-Reliable Wireless Backhaul (URWB) mode or WGB mode.

The IW9165E has the option to switch images by just updating the software to operate the IW9165E in CAPWAP or WGB or URWB mode without changing the hardware.

For CAPWAP mode, the access points can operate in the following modes:

- **Local mode:** This is the default mode for the AP. In this mode, the AP serves clients. In local mode, the AP creates two CAPWAP tunnels for the controller, one for management and the other for data traffic. This is known as central switching because the data traffic is switched (bridged) from the AP to the controller.

- **Flexconnect mode:** In FlexConnect mode, the data traffic is switched locally and is not sent to the controller. In this mode, the AP behaves like an autonomous AP, but is managed by the controller. Here, the AP continues to function even if the connection to the controller is lost.
- **Fabric mode:** The AP in a fabric mode has a VxLAN tunnel(Access-Tunnel) build to the fabric edge where the AP is attached. In cases where the AP is attached to an Extended Node(EN) or a Policy Extended Node(PEN). The access-tunnels are build between the Access Point (AP) and the respective fabric edge where the extended node is uplinked to. The VxLAN tunnel between an AP and a fabric edge is to preserve the segmentation till the access point. The access point is responsible to insert the SGT tag in the VxLAN tunnel to the fabric edge.
- **Sniffer mode:** In the wireless sniffer mode, the AP starts sniffing the air on a given channel. It captures and forwards all the clients' packets on that channel to a remote machine that runs Airopeek or Wireshark (packet analyzers for IEEE 802.11 wireless LANs). This includes information about the time stamp, signal strength, packet size, and so on.

Note In the sniffer mode, the server to which the data is sent should be on the same VLAN as the wireless controller management VLAN. Otherwise, an error message is displayed.

- **Monitor mode:** In the monitor mode, the AP is excluded from handling data traffic between clients and infrastructure. The AP acts as a dedicated sensor for location-based services (LBS), rogue AP detection, and Intrusion Detection System (IDS). When the AP is in monitor mode, it actively monitors the airwaves and typically does not serve clients.
- **Site Survey mode:** The AP GUI is enabled and is used for configuring the RF parameters for site survey investigation. For information, see the Access Points Survey Mode section in the *Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide*.

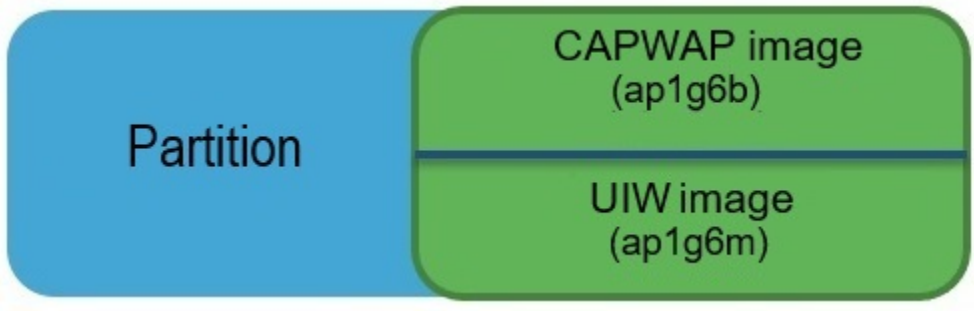
Unsupported Features

- 2.4G radio is not supported.
- Scan radio is not supported.

For more information about how to configure the AP on the Wireless Controller, See Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide.

Determining Image

Software images are stored under different folders on the same partition on IW9165E.



You need to choose the image to boot up with according to the mode your AP is runnning, CAPWAP, Cisco URWB, or WGB/uWGB. The following table provides the software images of each mode:

Table 1: IW9165E Software Images

IW9165E Mode	Software Image
CAPWAP	**ap1g6b-k9w8-xxx.tar**
URWB	Unified Industrial Wireless image **ap1g6m-k9c1-xxx.tar**
WGB/uWGB	

To determine the image that your IW9165E is running, use the **show version** command.

- If the **show version** output displays **Cisco AP Software, (ap1g6b)** as shown in the following example, it means that AP is running the CAPWAP image **ap1g6b-k9w8-xxx.tar**, which supports the CAPWAP mode.

```
Cisco AP Software, (ap1g6b), C9165, RELEASE SOFTWARE
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2024 by Cisco Systems, Inc.
Compiled Tue Feb 20 23:04:29 GMT 2024
```

- If the **show version** output displays **Cisco AP Software (ap1g6m)** as shown in the following example, it means that AP is running **ap1g6m-k9c1-xxx.tar** image, which supports the URWB mode or WGB/uWGB.

```
Cisco AP Software, (ap1g6m), C9165, RELEASE SOFTWARE
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2024 by Cisco Systems, Inc.
Compiled Tue Feb 20 23:04:29 GMT 2024
```

Catalyst IW9165E Lightweight Access Point supports three wireless technologies on a single hardware platform, such as CAPWAP, URWB, and WGB. The Catalyst IW9165E has the option to switch images by just updating the software to operate the Catalyst IW9165E in CAPWAP, WGB or URWB mode without changing the hardware.

Configuring Image Conversion

To convert the IW9165E access point either from Wi-Fi mode (CAPWAP AP) or URWB mode or WGB mode, follow these steps:

1. To convert from CAPWAP to URWB mode or from WGB/uWGB to URWB mode, use the following CLI command. The access point then reboots and starts up in URWB mode.

   ```
   configure boot mode urwb
   ```

2. To convert from URWB to CAPWAP mode or from WGB/uWGB to CAPWAP mode, use the following CLI command. The access point then reboots and starts up in CAPWAP mode.

   ```
   configure boot mode capwap
   ```

3. To convert from CAPWAP to WGB/uWGB mode or from URWB to WGB/uWGB mode, use the following CLI command:

   ```
   configure boot mode wgb
   ```

Note Image conversion performs a full factory reset which completely erases the configuration and data.

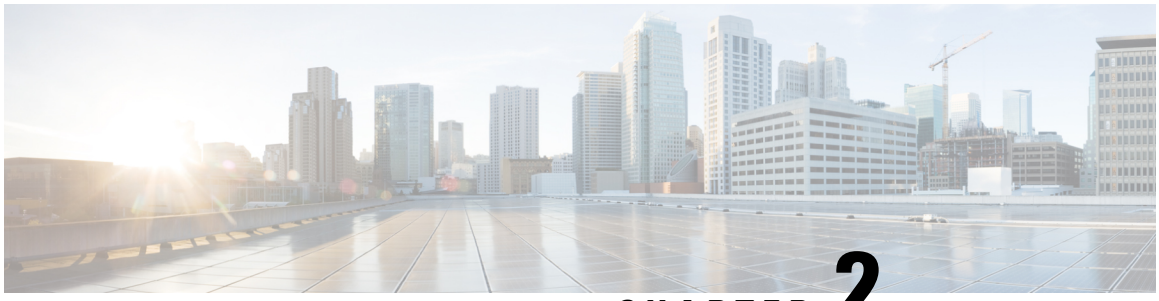
Related Documentation

To view all support information for the Cisco Catalyst IW9165 Rugged Series, see https://www.cisco.com/content/en/us/support/wireless/catalyst-iw9165-rugged-series/series.html.

In addition to the documentation available on the support page, you will need to refer to the following guides:

- For information about IW9165E hardware, see Cisco Catalyst IW9165E Rugged Access Point and Wireless Client Hardware Installation Guide.
- A full listing of the AP's features and specifications is provided in Cisco Catalyst IW9165 Series Data Sheet.
- For information about Cisco URWB mode configuration, see the relevant documents at:

  https://www.cisco.com/content/en/us/support/wireless/catalyst-iw9165-rugged-series/series.html.
- For more information about the configuration on Cisco Catalyst 9800 Series Wireless Controllers, see Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide.



CHAPTER 2

Workgroup Bridges

Overview

Workgroup Bridge Mode

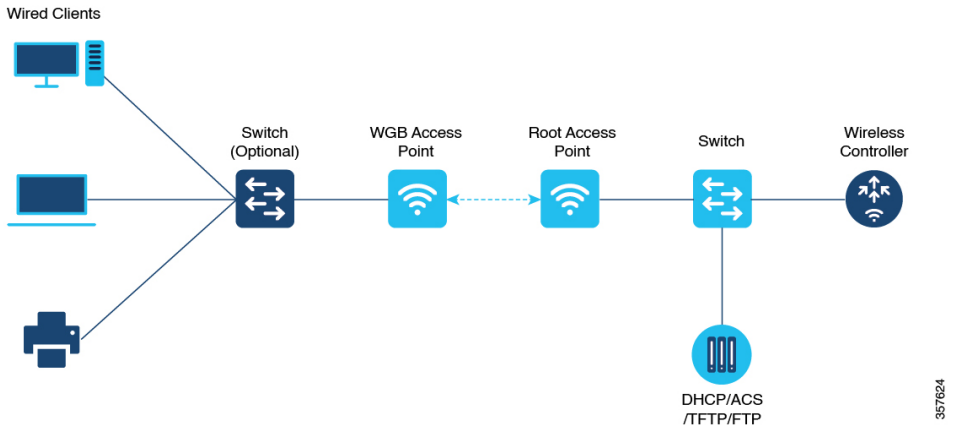
Workgroup Bridge Mode (WGB) is an Access Point (AP) mode that provides wireless connectivity to the wired clients. These clients are connected to the Ethernet port of the WGB AP. The WGB works like a bridge between the wired network and thesingle wireless segment. It does this by learning the MAC addresses of its

wired clients on the Ethernet. The WGB then shares these identifiers with the Wireless LAN Controller (WLC) through an infrastructure AP using the Internet Access Point Protocol (IAPP) messaging. The WGB establishes a single wireless connection to the root AP, which treats the WGB as a wireless client.

Universal Workgroup Bridge Mode

Universal Workgroup Bridge (uWGB) mode is a complementary mode of the WGB feature. It acts as a wireless bridge between the wired client connected to the uWGB and the wireless infrastructure. This infrastructure includes both Cisco and non-Cisco wireless networks. One of the wireless interfaces is used to connect with the access point. The radio MAC is used to associate with the AP.

Figure 1: Example of a WGB



Starting from Cisco Unified Industrial Wireless Software Release 17.13.1, WGB is supported on the Cisco Catalyst IW9165E Rugged Access Point and Wireless Client.

Limitations and Restrictions

This section provides limitations and restrictions for WGB and uWGB modes.

- The WGB can associate only with Cisco lightweight access points. The uWGB can associate to a third party access point.
- Speed and duplex are automatically negotiated based on the capabilities of the locally connected endpoint and cannot be manually configured on the AP's wired 0 and wired 1 interfaces.
- Per-VLAN Spanning Tree (PVST) and packets are used to detect and prevent loops in the wired and wireless switching networks. WGB transparently bridge STP packets. WGB can bridge STP packets between two wired segments. Incorrect or inconsistent configuration of STP in the wired segments can cause WGB wireless link to be blocked by the connected switch(es) to Access Point or WGB. This could cause WGB to disconnect from AP or AP disconnection to Controller to drop, and wired clients not receiving IP addresses, as STP begins to block switch port in the wired network. If administrator needs to disable bridging of STP between the wired segments by the WGB, we recommend disabling the STP on the directly connected switches in the wireless network.
- The following features are not supported for use with a WGB:
    - Idle timeout

    - Web authentication

- With Layer 3 roaming, if you plug a wired client into the WGB network after the WGB has roamed to another controller (for example, to a foreign controller), the wired client's IP address displays only on the anchor controller, not on the foreign controller.
- When you deauthenticate a WGB record from a controller, all of the WGB wired clients' entries are also deleted.
- These features are not supported for wired clients connected to a WGB:
    - MAC filtering
    - Link tests
    - Idle timeout
- Associating a WGB to a WLAN that is configured for Adaptive 802.11r is not supported.
- WGB supports IPv6 only when IPv4 is enable. But there is no impact on WGB wired clients IPv6 traffic.
- WGB management IPv6 does not work after WGB uplink association is completed. WGB can get an IPv6 address when the association is successful. But IPv6 ping will not be passed from or to WGB. SSH from wireless or wired client to WGB management IPv6 is not working. The workaround to bypass the pingable issue is to re-enable IPv6, even though IPv6 has already been enabled and the IPv6 address has been assigned.
- The uWGB mode does not support TFTP or SFTP. For software upgrade, you should perform it from WGB mode. For more information, see uWGB Image Upgrade, on page 9.
- From Cisco Unified Industrial Wireless Software Release 17.13.1, AP in uWGB mode supports to be managed by SSH, and image upgrade can be implemented when no wired client is detected.
    - When wired client is detected, AP in uWGB mode changes to uWGB state and AP cannot be managed.
    - When no wired client is detected, AP in uWGB mode changes to WGB state and AP can be managed.

Configuring Strong Password in Day0

It is required to set a strong password for WGB/uWGB after first login. The username and strong password should follow these rules:

1. Username length is between 1 and 32 characters.
2. Password length is between 8 to 120 characters.
3. Password must contain at least one uppercase character, one lowercase character, one digit, and one punctuation.
4. Password can contain alphanumeric characters and special characters (ASCII decimal code from 33 to 126), but the following special characters are not permitted: " (double quote), ' (single quote), ? (question mark).
5. Password cannot contain three sequential characters.

6. Password cannot contain three same characters consecutively.
7. Password cannot be the same as or reverse of the username.
8. New password must have at least four different characters compared to the current password.

For example, by default, the credential is

- username: Cisco
- password: Cisco
- enable password: Cisco

To reset the credential with the following strong password:

- username: demouser
- password: DemoP@ssw0rd
- enable password: DemoE^aP@ssw0rd

```
User Access Verification
Username: Cisco
Password: Cisco

% First Login: Please Reset Credentials

Current Password:Cisco
Current Enable Password:Cisco
New User Name:demouser
New Password:DemoP@ssw0rd
Confirm New Password:DemoP@ssw0rd
New Enable Password:DemoE^aP@ssw0rd
Confirm New Enable Password:DemoE^aP@ssw0rd

% Credentials changed, please re-login

[*04/18/2023 23:53:44.8926] chpasswd: password for user changed
[*04/18/2023 23:53:44.9074]
[*04/18/2023 23:53:44.9074] Management user configuration saved successfully
[*04/18/2023 23:53:44.9074]


User Access Verification
Username: demouser
Password: DemoP@ssw0rd
APFC58.9A15.C808>enable
Password:DemoE^aP@ssw0rd
APFC58.9A15.C808#
```

Note In above example, all passwords are displayed in plain text for demonstration purpose. In real case, they are hidden by asterisks (*).

Controller Configuration for WGB

For a WGB to join a wireless network, you need to configure specific settings on the WLAN and related policy profile on the controller.

Follow these steps to configure the Cisco Client Extensions option and set the support of Aironet IE in the WLAN:

1. Enter WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.

   ```
   #wlan profile-name
   ```

2. Configure the Cisco Client Extensions option and set the support of Aironet IE on the WLAN.

   ```
   #ccx aironet-iesupport
   ```

Note Without this configuration, WGB is not able to associate to AP.

Follow these steps to configure WLAN policy profile:

1. Enter wireless policy configuration mode.

   ```
   #wireless profile policy profile-policy
   ```

2. Assign the profile policy to the VLAN.

   ```
   #vlan vlan-id
   ```

3. Configure WGB VLAN client support.

   ```
   #wgb vlan
   ```

uWGB Image Upgrade

uWGB mode does not support TFTP or SFTP. To perform a software upgrade, follow these steps:

Step 1 Connect a TFTP or SFTP server to wired 0 port of uWGB.

Step 2 Turn radio interfaces into Administratively Down state.

configure Dot11Radio *slot_id* **disable**

Example:

```
#configure Dot11Radio 1 disable
```

Step 3 Convert uWGB to WGB mode.

configure Dot11Radio *slot_id* **mode wgb ssid-profile** *ssid_profile_name*

Example:

```
#configure Dot11Radio 1 mode wgb ssid-profile a_uwgb_demo_ssid

This command will reboot with downloaded configs.
Are you sure you want continue? <confirm>
```

Note *ssid_profile_name* can be any existing SSID profile configured by users.

Step 4 After rebooting, assign a static IP address to the WGB.

configure ap address ipv4 static *IPv4_address netmask Gateway_IPv4_address*

Example:

```
#configure ap address ipv4 static 192.168.1.101 255.255.255.0 192.168.1.1
```

Step 5 Verify the ICMP ping works.

ping *server_IP*

Example:

```
#ping 192.168.1.20
Sending 5, 100-byte ICMP Echos to 192.168.1.20, timeout is 2 seconds

PING 192.168.1.20
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0.858/0.932/1.001 ms
```

Step 6 Upgrade the software.

archive download/reload <**tftp | sftp | http**>*://server_ip/file_path*

Step 7 Convert WGB back to uWGB.

configure Dot11Radio *slot_id* **mode uwgb** *wired_client_mac_addr* **ssid-profile** *ssid_profile_name*

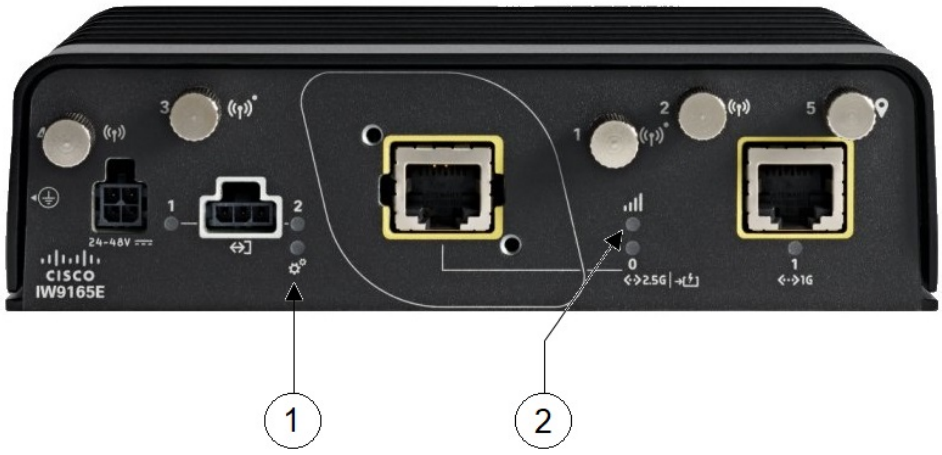
Example:

```
#configure Dot11Radio 1 mode uwgb 00b4.9e00.a891 ssid-profile a_uwgb_demo_ssid
```

LED Pattern

There are two LEDs located at the AP front panel, a system status LED and an RSSI LED, as shown in the following figure.

Figure 2: IW9165E LEDs



1	System Status LED • When WGB is in disassociated state, the System LED is blinking RED. • When WGB makes association to parent AP, the System LED turns to solid GREEN.	2	RSSI Status LED • When RSSI >= -71dBm, the RSSI LED is solid GREEN. • When -81 dBm <= RSSI < -70 dBm, the RSSI LED is blinking GREEN. • When -81 dBm < RSSI < -95 dBm, the RSSI LED is solid YELLOW. • Otherwise Off.

Configuring IP Address

Configuring IPv4 Address

Configure the IPv4 address of the AP by entering the following commands:

- To configure IPv4 address by DHCP, use the following command:

  #**configure ap address ipv4 dhcp**

- To configure the static IPv4 address, use the following command. By doing so, you can manage the device via wired interface without uplink connection.

  #**configure ap address ipv4 static** *ipv4_addr netmask gateway*

- To display current IP address configuration, use the following command:

  #**show ip interface brief**

Configuring IPv6 Address

Configure the IPv6 address of the AP by entering the following commands:

- To configure the static IPv6 address, use the following command. By doing so, you can manage the device via wired interface without uplink connection.

  #**configure ap address ipv6 static** *ipv6_addr prefixlen* [*gateway*]

- #**configure ap address ipv6 auto-config** {**enable**|**disable**}

Note The **configure ap address ipv6 auto-config enable** command is designed to enable IPv6 SLAAC. However, SLAAC is not applicable for cos WGB. This CLI will configure IPv6 address with DHCPv6 instead of SLAAC.

- To configure IPv6 address by DHCP, use the following command:

  #**configure ap address ipv6 dhcp**

- To display current IP address configuration, use the following command:

  #**show ipv6 interface brief**

WGB Configuration

The typical WGB configuration involves the following steps:

1. Create an SSID profile.
2. Configure radio as workgroup, and associate the SSID profile to the radio.
3. Turn on the radio.

WGB uplink supports various security methods, including:

- Open (unsecured)
- PSK
- Dot1x (LEAP, PEAP, FAST-EAP, TLS)

The following is an example of Dot1x FAST-EAP configuration:

```
configure dot1x credential demo-cred username demouser1 password Dem0Pass!@
configure eap-profile demo-eap-profile dot1x-credential demo-cred
configure eap-profile demo-eap-profile method fast
configure ssid-profile demo-FAST ssid demo-fast authentication eap profile demo-eap-profile
 key-management wpa2
configure dot11radio 1 mode wgb ssid-profile demo-FAST
configure dot11radio 1 enable
```

The following sections provide detailed information about WGB configuration:

SSID configuration

SSID configuration consists of the following two parts:

Creating an SSID Profile

Choose one of the following authentication protocols for the SSID profile:

Configuring an SSID profile with Open Authentication

Use the following command to configure an SSID profile with Open Authentication:

configure ssid-profile *ssid-profile-name* **ssid** *radio-serv-name* **authentication open**

Configuring an SSID profile with PSK Authentication

Use the following command to configure an SSID profile with PSK WPA2 Authentication:

configure ssid-profile *ssid-profile-name* **ssid** *SSID_name* **authentication psk** *preshared-key* **key-management wpa2**

Use the following command to configure an SSID profile with PSK Dot11r Authentication:

configure ssid-profile *ssid-profile-name* **ssid** *SSID_name* **authentication psk** *preshared-key* **key-management dot11r**

Use the following command to configure an SSID profile with PSK Dot11w Authentication:

configure ssid-profile *ssid-profile-name* **ssid** *SSID_name* **authentication psk** *preshared-key* **key-management dot11w**

Configuring an SSID Profile with Dot1x Authentication

Use the following commands to configure an SSID profile with Dot1x authentication:

configure ssid-profile *ssid-profile-name* **ssid** *radio-serv-name* **authentication eap profile** *eap-profile-name* **key-management** {**dot11r** | **wpa2** | **dot11w** {**optional | required**}}

The following example configures an SSID profile with Dot1x EAP-PEAP authentication:

```
configure dot1x credential c1 username wgbusr password cisco123456
configure eap-profile p1 dot1x-credential c1
configure eap-profile p1 method peap
configure ssid-profile iot-peap ssid iot-peap authentication eap profile p1 key-management
 wpa2
```

Configuring Radio Interface for Workgroup Bridges

IW9165E does not have 2.4 GHz radio. Only slot 1 (dot11radio 1) can be configured as uplink and operate in WGB mode.

- Map a radio interface to a WGB SSID profile by entering this command:

  #**configure dot11radio 1 mode wgb ssid-profile** *ssid-profile-name*

  Example

  ```
  #configure dot11radio 1 mode wgb ssid-profile psk_ssid
  ```

- Configure a radio interface by entering this command:

configure dot11radio 1 { **enable** | **disable** }

Example

```
#configure dot11radio 1 disable
```

Configuring a Dot1X Credential

Configure a dot1x credential by entering this command:

configure dot1x credential *profile-name* **username** *name* **password** *pwd*

View the WGB EAP dot1x profile summary by entering this command:

 # show wgb eap dot1x credential profile

Deauthenticating WGB Wired Client

Deauthenticate WGB wired client by entering this command:

clear wgb client {**all** |**single** *mac-addr*}

Configuring an EAP Profile

Follow these steps to configure the EAP profile:

1. Bind dot1x credential profile to EAP profile.
2. Bind EAP profile to SSID profile
3. Bind SSID profile to the radio.

Step 1 Configure the EAP profile method type by entering this command:

 # configure eap-profile *profile-name* **method** {**fast** | **leap** | **peap** | **tls**}

Step 2 Attaching the CA Trustpoint for TLS by entering the following command. With the default profile, WGB uses the internal MIC certificate for authentication.

 # configure eap-profile *profile-name* **trustpoint** {**default** | **name** *trustpoint-name*}

Step 3 Bind dot1x-credential profile by entering this command:

 # configure eap-profile *profile-name* **dot1x-credential** *profile-name*

Step 4 [Optional] Delete an EAP profile by entering this command:

 # configure eap-profile *profile-name* **delete**

Step 5 View summary of EAP and dot1x profiles by entering this command:

 # show wgb eap profile all

Configuring Manual Enrollment of a Trustpoint for Terminal

Step 1 Create a Trustpoint in WGB by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **enrollment terminal**

Step 2 Authenticate a Trustpoint manually by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **authenticate**

Enter the base 64 encoded CA certificate and end the certificate by entering **quit** in a new line.

Note User has to import complete certificate chains in the trustpoint if intermediate certificate is used.

Example:

```
#configure crypto pki trustpoint demotp authenticate

Enter the base 64 encoded CA certificate.
....And end with the word "quit" on a line by itself....

-----BEGIN CERTIFICATE-----
[base64 encoded root CA certificate]
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
[base64 encoded intermediate CA certificate]
-----END CERTIFICATE-----
quit
```

Step 3 Configure a private key size by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **key-size** *key-length*

Step 4 Configure the subject-name by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **subject-name** *name* `[Optional]` *2ltr-country-code state-name locality org-name org-unit email*

Step 5 Generate a private key and Certificate Signing Request (CSR) by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **enroll**

Create the digitally signed certificate using the CSR output in the CA server.

Step 6 Import the signed certificate in WGB by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **import certificate**

Enter the base 64 encoded CA certificate and end the certificate by entering **quit** in a new line.

Step 7 [Optional] Delete a Trustpoint by entering this command:

 # configure crypto pki trustpoint *trustpoint-name* **delete**

Step 8 View the Trustpoint summary by entering this command:

 # show crypto pki trustpoint

Step 9 View the content of the certificates that are created for a Trustpoint by entering this command:

 # show crypto pki trustpoint *trustpoint-name* **certificate**

Configuring Auto-Enrollment of a Trustpoint for Workgroup Bridge

Step 1 Enroll a Trustpoint in WGB using the server URL by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **enrollment url** *ca-server-url*

Step 2 Authenticate a Trustpoint by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **authenticate**

This command will fetch the CA certificate from CA server automatically.

Step 3 Configure a private key size by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **key-size** *key-length*

Step 4 Configure the subject-name by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **subject-name** *name* `[Optional]` *2ltr-country-code state-name locality org-name org-unit email*

Step 5 Enroll the Trust point by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **enroll**

Request the digitally signed certificate from the CA server.

Step 6 Enable auto-enroll by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **auto-enroll enable** *renew-percentage*

You can disable auto-enrolling by using the disable syntax in the command.

Step 7 [Optional] Delete a Trustpoint by entering this command:

 # configure crypto pki trustpoint *trustpoint-name* **delete**

Step 8 View the Trustpoint summary by entering this command:

 # show crypto pki trustpoint

Step 9 View the content of the certificates that are created for a Trustpoint by entering this command:

 # show crypto pki trustpoint *trustpoint-name* **certificate**

Step 10 View the PKI timer information by entering this command:

 # show crypto pki timers

Configuring Manual Certificate Enrollment Using TFTP Server

Step 1 Specify the enrollment method to retrieve the CA certificate and client certificate for a Trustpoint in WGB by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **enrollment tftp** *tftp-addr/file-name*

Step 2 Authenticate a Trustpoint manually by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **authenticate**

Retrieves the CA certificate and authenticates it from the specified TFTP server. If the file specification is included, the wgb will append the extension ".ca" to the specified filename.

Step 3 Configure a private key size by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **key-size** *key-length*

Step 4 Configure the subject-name by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **subject-name** *name* `[Optional]` *2ltr-country-code state-name locality org-name org-unit email*

Step 5 Generate a private key and Certificate Signing Request (CSR) by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **enroll**

Generates certificate request and writes the request out to the TFTP server. The filename to be written is appended with the extension ".req".

Step 6 Import the signed certificate in WGB by entering this command:

 # configure crypto pki trustpoint *ca-server-name* **import certificate**

Imports a certificate via TFTP at the console terminal, which retrieves the granted certificate. The WGB will attempt to retrieve the granted certificate via TFTP using the same filename and the file name append with ".crt" extension.

Step 7 View the Trustpoint summary by entering this command:

 # show crypto pki trustpoint

Step 8 View the content of the certificates that are created for a Trustpoint by entering this command:

 # show crypto pki trustpoint *trustpoint-name* **certificate**

Configuring WGB or uWGB Timer

The timer configuration CLIs are common for both WGB and uWGB. Use the following commands to configure timers:

- Configure the WGB association response timeout by entering this command:

  # **configure wgb association response timeout** *response-millisecs*

  The default value is 100 milliseconds. The valid range is between 100 and 5000 milliseconds.

- Configure the WGB authentication response timeout by entering this command:

  # **configure wgb authentication response timeout** *response-millisecs*

  The default value is 100 milliseconds. The valid range is between 100 and 5000 milliseconds.

- Configure the WGB EAP timeout by entering this command:

  # **configure wgb eap timeout** *timeout-secs*

  The default value is 3 seconds. The valid range is between 2 and 60 seconds.

- Configure the WGB bridge client response timeout by entering this command:

  # **configure wgb bridge client timeout** *timeout-secs*

  Default timeout value is 300 seconds. The valid range is between 10 and 1000000 seconds.

uWGB Configuration

The universal WGB is able to interoperate with non-Cisco access points using uplink radio MAC address, thus the universal workgroup bridge role supports only one wired client.

Most WGB configurations apply to uWGB. The only difference is that you configure wired client's MAC address with the following command:

configure dot11 <*slot_id*> **mode uwgb** <*uwgb_wired_client_mac_address*> **ssid-profile** <*ssid-profile*>

The following is an example of Dot1x FAST-EAP configuration:

```
configure dot1x credential demo-cred username demouser1 password Dem0Pass!@
configure eap-profile demo-eap-profile dot1x-credential demo-cred
configure eap-profile demo-eap-profile method fast
configure ssid-profile demo-FAST ssid demo-fast authentication eap profile demo-eap-profile
 key-management wpa2
configure dot11radio 1 mode uwgb fc58.220a.0704 ssid-profile demo-FAST
configure dot11radio 1 enable
```

The following sections provide detailed information about uWGB configuration:

- SSID configuration
- Configuring a Dot1X Credential
- Configuring an EAP Profile
- Configuring Manual Enrollment of a Trustpoint for Terminal
- Configuring Auto-Enrollment of a Trustpoint for Workgroup Bridge
- Configuring Manual Certificate Enrollment Using TFTP Server
- Configuring WGB or uWGB Timer

Converting Between WGB and uWGB

To convert from WGB to uWGB, use the following command:

#**configure dot11radio** <*radio_slot_id*> **mode uwgb** <*WIRED_CLIENT_MAC*> **ssid-profile** <*SSID_PROFILE_NAME*>

To convert from uWGB to WGB, use the following command. This conversion involves a reboot of the AP.

#**configure Dot11Radio 1 mode wgb ssid-profile** <*SSID_PROFILE_NAME*>

```
This command will reboot with downloaded configs.
Are you sure you want continue? [confirm]
```

Importing and Exporting WGB Configuration

You can upload the working configuration of an existing WGB to a server, and then download it to the new deployed WGBs.

To upload the configuration to a server, use the following command:

#**copy configuration upload** <**sftp:**|**tftp:**> *ip-address* [*directory*] [*file-name*]

To download a sample configuration to all WGBs in the deployment, use the following command:

#**copy configuration download** <**sftp:**|**tftp:**> *ip-address* [*directory*] [*file-name*]

The access point will reboot after the **copy configuration download** command is executed. The imported configuration will take effect after the rebooting.

Verifying the Configuration of WGB and uWGB

Use the **show run** command to check whether the AP is in WGB mode or uWGB mode.

- WGB:

```
#show run
AP Name              : APFC58.9A15.C808
AP Mode              : WorkGroupBridge
CDP State            : Enabled
Watchdog monitoring  : Enabled
SSH State            : Disabled
AP Username          : admin
Session Timeout      : 300


Radio and WLAN-Profile mapping:-
===================================
Radio ID    Radio Mode    SSID-Profile                    SSID
          Authentication
-------------------------------------------------------------------------------
--------------------------
1           WGB           myssid                          demo
          OPEN


Radio configurations:-
==============================
Radio Id             : NA
   Admin state       : NA
   Mode              : NA
Radio Id             : 1
   Admin state       : DISABLED
   Mode              : WGB
```

```
   Dot11 type        : 11ax
Radio Id             : NA
   Admin state       : NA
   Mode              : NA
```

- uWGB:

```
#show run
AP Name              : APFC58.9A15.C808
AP Mode              : WorkGroupBridge
CDP State            : Enabled
Watchdog monitoring  : Enabled
SSH State            : Disabled
AP Username          : admin
Session Timeout      : 300


Radio and WLAN-Profile mapping:-
===================================
Radio ID    Radio Mode    SSID-Profile                    SSID
          Authentication
-------------------------------------------------------------------------------------
--------------------------
1           UWGB          myssid                          demo
          OPEN


Radio configurations:-
==============================
Radio Id             : NA
   Admin state       : NA
   Mode              : NA
Radio Id             : 1
   Admin state       : DISABLED
   Mode              : UWGB
   Uclient mac       : 0009.0001.0001
   Current state     : WGB
   UClient timeout   : 0 Sec
   Dot11 type        : 11ax
Radio Id             : NA
   Admin state       : NA
   Mode              : NA
```

Use the **show wgb dot11 associations** command to verify the configuration of WGB and uWGB.

- WGB:

```
#show wgb dot11 associations
Uplink Radio ID : 1
Uplink Radio MAC : 00:99:9A:15:B4:91
SSID Name : roam-m44-open
Parent AP Name : APFC58.9A15.C964
Parent AP MAC : 00:99:9A:15:DE:4C
Uplink State : CONNECTED
Auth Type : OPEN
Dot11 type : 11ax
Channel : 100
Bandwidth : 20 MHz
Current Datarate (Tx/Rx) : 86/86 Mbps
Max Datarate : 143 Mbps
RSSI : 53
IP : 192.168.1.101/24
Default Gateway : 192.168.1.1
IPV6 : ::/128
```

```
Assoc timeout : 100 Msec
Auth timeout : 100 Msec
Dhcp timeout : 60 Sec
```

- uWGB:

```
#show wgb dot11 associations
Uplink Radio ID : 1
Uplink Radio MAC : 00:09:00:01:00:01
SSID Name : roam-m44-open
Parent AP MAC : FC:58:9A:15:DE:4C
Uplink State : CONNECTED
Auth Type : OPEN
Uclient mac : 00:09:00:01:00:01
Current state : UWGB
Uclient timeout : 60 Sec
Dot11 type : 11ax
Channel : 36
Bandwidth : 20 MHz
Current Datarate (Tx/Rx) : 77/0 Mbps
Max Datarate : 143 Mbps
RSSI : 60
IP : 0.0.0.0
IPV6 : ::/128
Assoc timeout : 100 Msec
Auth timeout : 100 Msec
Dhcp timeout : 60 Sec
```

Configuring Syslog

Syslog is a common protocol that the device uses to send event data logs to a central location for storing. Currently, only UDP mode is supported. Additional debug log will be collected if debug command is enabled in WGB. All collected log sent to syslog server will be in "kernel" facility and "warning" level.

- To enable WGB syslog, use the following command:

  # **logging host enable** <*server_ip*> **UDP**

- To disable WGB syslog (default), use the following command:

  # **logging host enable 0.0.0.0 UDP**

- To display current syslog configuration, use the following command:

  # **show running-config**

Configuring HT Speed Limit

In WGB field moving case deployment, you can manually set a transmission rate limit with High Throughput (HT) Modulation and Coding Scheme (MCS) .

The following is an example to configure WGB to transmit with 802.11n HT m4. m5. rate:

Config dot11radio [**1**|**2**] **802.11ax disable**

Config dot11radio [**1**|**2**] **802.11ac disable**

Config dot11radio [**1**|**2**] **speed ht-mcs m4. m5.**

Note WGB also supports to configure legacy rate:

Config dot11radio [1|2] speed legacy-rate basic-6.0 9.0 12.0 18.0 24.0

Legacy rate is used by 802.11 management frame and control frame. WGB legacy rates should match AP's legacy rates, or at least, having overlap between these two rate sets. Otherwise, WGB association will be rejected due to mismatched rates.

To check WGB Tx MCS rate, use the **debug wgb dot11 rate** command. The following example shows the output of this command.

```
JWGB1#debug wgb dot11 rate
[*10/14/2023 03:16:08.6175]
[*10/14/2023 03:16:08.6175]                MAC    Tx-Pkts    Rx-Pkts                Tx-Rate(Mbps)                Rx-Rate(Mbps)  RSSI Tx-Retries
[*10/14/2023 03:16:08.6175] 24:16:1B:F8:02:6E          0          0          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -70          0
JWGB1#[*10/14/2023 03:16:09.6179] 24:16:1B:F8:02:6E        330          3          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -71         15
[*10/14/2023 03:16:10.6183] 24:16:1B:F8:02:6E        332          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -71         25
[*10/14/2023 03:16:11.6187] 24:16:1B:F8:02:6E        327          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -71         18
[*10/14/2023 03:16:12.6190] 24:16:1B:F8:02:6E        330          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -70         13
[*10/14/2023 03:16:13.6194] 24:16:1B:F8:02:6E        333          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -71         21
[*10/14/2023 03:16:14.6198] 24:16:1B:F8:02:6E        331          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -70         16
[*10/14/2023 03:16:15.6202] 24:16:1B:F8:02:6E        328          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -70         24
[*10/14/2023 03:16:16.6206] 24:16:1B:F8:02:6E        330          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -70         21
[*10/14/2023 03:16:17.6210] 24:16:1B:F8:02:6E        332          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -70         22
[*10/14/2023 03:16:18.6214] 24:16:1B:F8:02:6E        327          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -71         22
[*10/14/2023 03:16:19.6218] 24:16:1B:F8:02:6E        333          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -71         18
[*10/14/2023 03:16:20.6221] 24:16:1B:F8:02:6E        330          2          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -71         17
[*10/14/2023 03:16:21.6258] 24:16:1B:F8:02:6E        328          3          HT-20,1SS,MCS5,(52)        HT-20,1SS,MCS5,SGI(57)   -70         16
```

802.11v Support

802.11v is the Wireless Network Management standard for the IEEE 802.11 family of standards. One enhancement of 802.11v 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.

By adding 802.11v support to WGB, WGB can be aware of imminent disconnection before disassociation happens, and then actively starts a roam and picks up an appropriate AP from a list of neighbor APs. WGB periodically queries for latest neighbor APs and associates to the optimal AP on next roam.

Since channel information of neighbor APs is included in Basic Service Set (BSS) Transition Request frame, roaming latency can be reduced for multiple channels deployment by scanning only the channels of neighboring APs.

The wireless controller can disassociate a client based on load balance, RSSI, and data rate on AP side. This disassociation can be notified to 802.11v client before it happens. Wireless controller can disassociate the client after a period of time, if the client does not re-associate to another AP within configurable period. To enable disassociating a client by network assisted roaming, the disassociation-imminent configuration can be turned on from wireless controller, which corresponds to the optional field (disassociation imminent) within BSS Transition Management Request frame.

For detailed information of 802.11v configuration on wireless controller, see https://www.cisco.com/c/en/us/td/docs/wireless/controller/9800/17-13/config-guide/b_wl_17_13_cg/m_802_11v_ewlc.html.

To configure 802.11v support on WGB, use the following command:

- To enable or disable 802.11v support on WGB, use the following command. By enabling 802.11v support, WGB scans only the channels learned from neighbor list.

configure wgb mobile station interface dot11Radio <*radio_slot_id*> **dot11v-bss-transition** [**enable**|**disable**]

- To configure the time interval that WGB sends BSS transition Query message to the parent AP, use the following command. Default value is 10 sec if not explicit configured. The timer is configured in seconds.

  # **configure wgb neighborlist-update-interval** <*1-900*>

- To check neighbor list received from associated AP, use the following command:

  # **show wgb dot11v bss-transition neighbour**

- To check channel list from dot11v neighbor, aux radio scanned, and residual channel scanned, use the following command:

  # **show wgb dot11v bss-transition channel**

- To clear neighbor list to provide error condition recover, use the following command:

  # **clear wgb dot11v bss-transition neighbor**

Configure Aux Scanning

The aux-scan mode can be configured as either scanning only or handoff mode on WGB radio 2 (5 GHz) to improve roaming performance.

Overview of Scanning-Only Mode

- The AP allows the radio to operate only for scanning purposes rather than providing client connectivity.
- The AP scans the wireless environment continuously to gather data on network performance, interference, rogue devices, and other critical metrics.

Configuring Scanning-Only Mode

When slot 2 radio is configured as scanning only mode, slot 1 (5G) radio will always be picked as uplink. Slot 2 (5G) radio will keep scanning configured SSID based on the channel list. By defualt, the channel list contains all supported 5G channels (based on reg domain). The scanning list can be configured manually or learned by 802.11v.

When a roaming is triggered, the algorithm looks for candidates from scanning table and skips scanning phase if the table is not empty. WGB then makes assocaition to that candidate AP.

To configure scanning only mode, use the following command:

configure dot11Radio 2 mode scan only

To manually configure the channel list, using the following command:

configure wgb mobile station interface dot11Radio 1 scan <*channel*> [**add**|**delete**]

By default, candidate AP entries in scanning table ages out in 1200 ms. You can adjust the timer by the following command:

#**configure wgb scan radio 2 timeout**

<1-5000> Scanning ap expire time

Note AP selection algorithm picks candidate with best RSSI from the scanning table. In some cases, the RSSI values are out-of-date. This can lead to a failed roaming.

Check the scanning table by using the **show wgb scan** command:

```
#show wgb scan
Best AP expire time: 5000 ms

************[ AP List ]***************
BSSID                RSSI   CHANNEL   Time
FC:58:9A:15:E2:4F     84     136       1531
FC:58:9A:15:DE:4F     37     136       41

***********[ Best AP ]****************
BSSID                RSSI   CHANNEL   Time
FC:58:9A:15:DE:4F    37     136       41
```

Configuring Aux-Scan Handoff Mode

When slot 2 radio is configured as handoff mode, both radio 1 and radio 2 are the uplink candidate. While one radio maintains wireless uplink, the other radio keeps scanning the channels. The scanning list can be configured manually or learned by 802.11v.

Radio 2 shares the same MAC address with radio 1, and supports the scanning function, association, and data serving. Both radios can work as **serving** or **scanning** role. When a roaming is triggered, the algorithm looks for the scanning database (internal tables), selects the best candidate AP and makes connection. The radio roles and traffic will dynamically switch between slot 1 and slot 2 after each roaming. WGB always uses the radio with operating role of **scanning** to complete the roaming association to a new AP. With this configuration, the roaming interruption time can be improved to 20-50 ms.

The following table is an example of aux-scan handoff radio mode configuration on IW9165E:

Slot 0 (2.4 G)	Slot 1 (5G)	Slot 2 (5G Only)	Slot 3 (Scanning radio)
N/A	WGB	Scan handoff	N/A

The following table compares roaming interruption time (3 channel case) in various mechanisms:

Roaming Interruption Time	Normal Channel Setting	Aux-scan Only	Aux-scan Handoff
Scanning	(40+20)*3=180 ms	0+40 ms	0 ms
Association	30-80 ms	30-80 ms	20-50 ms
Total	~210 ms	70-120 ms	20-50 ms

Use the following command to configure the WGB slot2 radio to aux-scan mode:

configure dot11Radio 2 mode scan handoff

Use the **show run** command to check your configuration:

```
#show run
...
Radio Id                    : 1
   Admin state              : ENABLED
   Mode                     : WGB
   Spatial Stream           : 1
   Guard Interval           : 800 ns
   Dot11 type               : 11n
   11v BSS-Neighbor         : Disabled
   A-MPDU priority          : 0x3f
   A-MPDU subframe number   : 12
   RTS Protection           : 2347(default)
   Rx-SOP Threshold         : AUTO
   Radio profile            : Default
   Encryption mode          : AES128
Radio Id                    : 2
   Admin state              : ENABLED
   Mode                     : SCAN - Handoff
   Spatial Stream           : 1
   Guard Interval           : 800 ns
   Dot11 type               : 11n
   11v BSS-Neighbor         : Disabled
   A-MPDU priority          : 0x3f
   A-MPDU subframe number   : 12
   RTS Protection           : 2347(default)
   Rx-SOP Threshold         : AUTO
   Radio profile            : Default
```

Use the **show wgb scan** command to display the current role of each radio and the aux scanning results:

```
APFC58.9A15.C808#show wgb scan
Best AP expire time: 2500 ms

Aux Scanning Radio Results (slot 2)
************[ AP List ]***************
BSSID                 RSSI   CHANNEL   Time
FC:58:9A:15:DE:4E     54     153       57
FC:58:9A:15:E2:4E     71     153       64

***********[ Best AP ]****************
BSSID                 RSSI   CHANNEL   Time
FC:58:9A:15:DE:4E    54     153       57

Aux Serving Radio Results
************[ AP List ]***************
BSSID                 RSSI   CHANNEL   Time
FC:58:9A:15:DE:4E     58     153       57
FC:58:9A:15:E2:4E     75     153       133

***********[ Best AP ]****************
BSSID                 RSSI   CHANNEL   Time
FC:58:9A:15:DE:4E    58     153       57
```

Optimized Roaming with Dual-Radio WGB

From the Cisco IOS-XE 17.15.1 release, devices with dual-radio configurations have improved roaming efficiency. Roaming is triggered due to continuous missing beacon frames or maximum packet retries. The second radio allows the WGB to skip the scanning phase and directly check the scanning table for potential APs. This process reduces service downtime.

Trigger Factors for Roaming

Roaming is triggered in these events:

- Low RSSI: Measures the power level that a wireless device, such as an AP, receives from a signal. Use RSSI values to determine the quality of the wireless connection to troubleshoot and optimize wireless networks.
- Beacon miss-count: Indicates the number of consecutive beacon frames that a client device has missed from an AP in a wireless network.
- Maximum packet retries: Specifies the maximum number of times a data packet can be retransmitted if the client device does not send an acknowledgement.

Dual-Radio Configuration

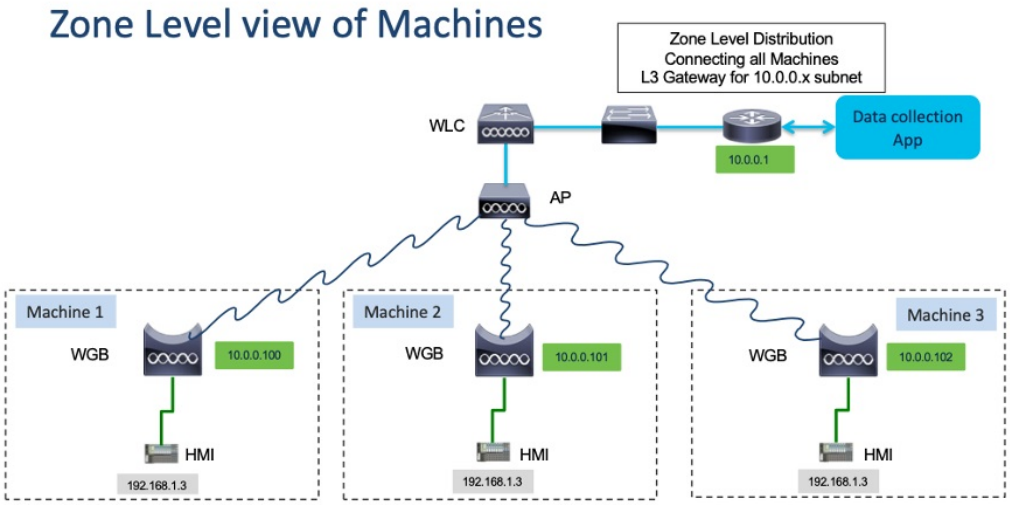
Here are the possible configurations for the IW9167E AP in a dual-radio setup:

Dual-Radio	AP
5 GHz radio 1 + radio 2 (scanning only mode)	IW9165E
5 GHz radio 1 + radio 2 (aux-scan handoff mode)	

Configuring Layer 2 NAT

One-to-one (1:1) Layer 2 NAT is a service that allows the assignment of a unique public IP address to an existing private IP address (end device), so that the end device can communicate with public network. Layer 2 NAT has two translation tables where private-to-public and public-to-private subnet translations can be defined.

In the industrial scenario where the same firmware is programmed to every HMI (customer machine, such as a Robot), firmware duplication across machines means IP address is reused across HMIs. This feature solves the problem of multiple end devices with the same duplicated IP addresses in the industrial network communicating with the public network.



The following table provides the commands to configure Layer 2 NAT:

Table 2: Layer 2 NAT Configuration Commands

Command	Description
#**configure l2nat** {**enable**|**disable**}	Enables or disables L2 NAT.
#**configure l2nat default-vlan** <*vlan_id*>	Specifies the default vlan where all NAT rules will be applied. If *vlan_id* is not specified, all NAT rules will be applied to vlan 0.
#**configure l2nat** {**add**|**delete**} **inside from host** <*original_ip_addr*> **to** <*translated_ip_addr*>	Adds or deletes a NAT rule which translates a private IP address to a public IP address. • *original_ip_addr*—Private IP address of the wired client connected to WGB Ethernet port. • *translated_ip_addr*—Public IP address that represents the wired client at public network.
#**configure l2nat** {**add**|**delete**} **outside from host** <*original_ip_addr*> **to** <*translated_ip_addr*>	Adds or deletes a NAT rule which translates a public IP address to a private IP address. • *original_ip_addr*—Public IP address of an outside network host. • *translated_ip_addr*—Private IP address which represents the outside network host at private network.

Command	Description
#**configure l2nat** {**add**|**delete**} **inside from network** <*original_nw_prefix*> **to** <*translated_nw_prefix*> <*subnet_mask*>	Adds or deletes a NAT rule which translates a private IP address subnet to a public IP address subnet. • *original_nw_prefix*—Private IP network prefix. • *translated_nw_prefix*—Public IP network prefix.
#**configure l2nat** {**add**|**delete**} **outside from network** <*original_nw_prefix*> **to** <*translated_nw_prefix*> <*subnet_mask*>	Adds or deletes a NAT rule which translates a public IP address subnet to a private IP address subnet. • *original_nw_prefix*—Public IP network prefix. • *translated_nw_prefix*—Private IP network prefix.

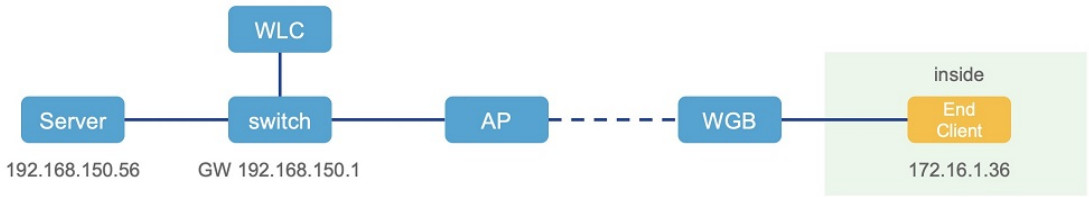
The following table provides the show and debug commands to verify and troubleshoot your Layer 2 NAT configuration:

Table 3: Layer 2 NAT Show and Debug Commands

Command	Description
#**show l2nat entry**	Displays the Layer 2 NAT running entries.
#**show l2nat config**	Displays the Layer 2 NAT configuration details.
#**show l2nat stats**	Displays the Layer 2 NAT packet translation statistics.
#**show l2nat rules**	Displays the Layer 2 NAT rules from the configuration.
#**clear l2nat statistics**	Clears packet translation statistics.
#**clear l2nat rule**	Clears Layer 2 NAT rules.
#**clear l2nat config**	Clears Layer 2 NAT configuration.
#**debug l2nat**	Enables debugging of packet translation process.
#**debug l2nat all**	Prints out the NAT entry match result when a packet arrives. **Caution** This debug command may create overwhelming log print in console. Console may lose response because of this command, especially when Syslog service is enabled with a broadcast address.
#**undebug l2nat**	Disables debugging of packet translation process.

Configuration Example of Host IP Address Translation

In this scenario, the end client (172.16.1.36) connected to WGB needs to communicate with the server (192.168.150.56) connected to the gateway. Layer 2 NAT is configured to provide an address for the end client on the outside network (192.168.150.36) and an address for the server on the inside network (172.16.1.56).



The following table shows the configuration tasks for this scenario.

Command	Purpose
#configure l2nat add inside from host 172.16.1.36 to 192.168.150.36 **#configure l2nat add outside from host 192.168.150.56 to 172.16.1.56**	Adds NAT rules to make inside client and outside server communicate with each other.
#configure l2nat add inside from host 172.16.1.1 to 192.168.150.1 **#configure l2nat add inside from host 172.16.1.255 to 192.168.150.255**	Adds NAT for gateway and broadcast address.

The following show commands display your configuration.

- The following command displays the Layer 2 NAT configuration details. In the output, I2O means "inside to outside", and O2I means "outside to inside".

```
#show l2nat config
L2NAT Configuration are:
================================
Status: enabled
Default Vlan: 0
The Number of L2nat Rules: 4
Dir      Inside                  Outside                 Vlan
O2I      172.16.1.56             192.168.150.56          0
I2O      172.16.1.36             192.168.150.36          0
I2O      172.16.1.255            192.168.150.255         0
I2O      172.16.1.1              192.168.150.1           0
```

- The following command displays the Layer 2 NAT rules.

```
#show l2nat rule
Dir      Inside                  Outside                 Vlan
O2I      172.16.1.56             192.168.150.56          0
I2O      172.16.1.36             192.168.150.36          0
I2O      172.16.1.255            192.168.150.255         0
I2O      172.16.1.1              192.168.150.1           0
```

- The following command displays Layer 2 NAT running entries.

```
#show l2nat entry
Direction           Original            Substitute            Age    Reversed
inside-to-outside   172.16.1.36@0       192.168.150. 36@0     -1     false
inside-to-outside   172.16.1.56@0       192.168.150. 56@0     -1     true
inside-to-outside   172.16.1.1@0        192.168.150. 1@0      -1     false
```

```
inside-to-outside    172.16.1.255@0       192.168.150. 255@0     -1     false
outside-to-inside    192.168.150.36@0     172.16.1.36@0          -1     true
outside-to-inside    192.168.150.56@0     172.16.1.56@0          -1     false
outside-to-inside    192.168.150.1@0      172.16.1.1@0           -1     true
outside-to-inside    192.168.150.255@0    172.16.1.255@0         -1     true
```

- The following command displays the WGB wired clients over the bridge.
  - Before Layer 2 NAT is enbled:

```
#show wgb bridge
    ***Client ip table entries***
                mac vap     port vlan_id          seen_ip  confirm_ago  fast_brg
B8:AE:ED:7E:46:EB   0   wired0       0      172.16.1.36     0.360000       true
24:16:1B:F8:05:0F   0 wbridge1       0           0.0.0.0  3420.560000       true
```

  - After Layer 2 NAT is enbled:

```
#show wgb bridge
    ***Client ip table entries***
                mac vap     port vlan_id          seen_ip  confirm_ago  fast_brg
B8:AE:ED:7E:46:EB   0   wired0       0   192.168.150.36     0.440000       true
24:16:1B:F8:05:0F   0 wbridge1       0           0.0.0.0  3502.220000       true
```

  If there are E2E traffic issues for wired client in NAT, restart the client register process by using the following command:

```
#clear wgb client single B8:AE:ED:7E:46:EB
```

- The following command displays the Layer 2 NAT packet translation statistics.

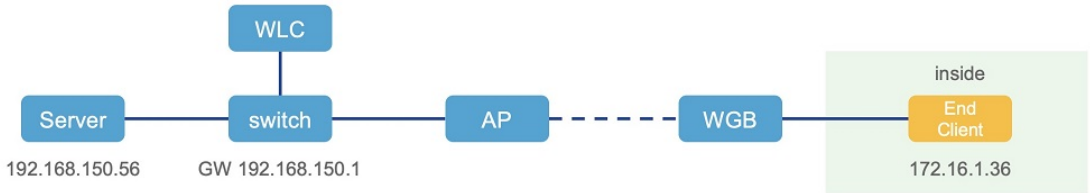
```
#show l2nat stats
Direction          Original              Substitute            ARP  IP   ICMP UDP  TCP
inside-to-outside  172.16.1.1@2660       192.168.150.1@2660    1    4    4    0    0
inside-to-outside  172.16.1.36@2660      192.168.150.36@2660   3    129  32   90   1
inside-to-outside  172.16.1.56@2660      192.168.150.56@2660   2    114  28   85   1
inside-to-outside  172.16.1.255@2660     192.168.150.255@2660  0    0    0    0    0
outside-to-inside  192.168.150.1@2660    172.16.1.1@2660       1    4    4    0    0
outside-to-inside  192.168.150.36@2660   172.16.1.36@2660      3    39   38   0    1
outside-to-inside  192.168.150.56@2660   172.16.1.56@2660      2    35   34   0    1
outside-to-inside  192.168.150.255@2660  172.16.1.255@2660     0    0    0    0    0
```

  To reset statistics number, use the following command:

```
#clear l2nat stats
```

Configuration Example of Network Address Translation

In this scenario, Layer 2 NAT is configured to translate the inside addresses from 172.16.1.0 255.255.255.0 subnet to addresses in the 192.168.150.0 255.255.255.0 subnet. Only the network prefix will be replaced during the translation. The host bits of the IP address remain the same.



The following command is configured for this scenario:

```
#configure l2nat add inside from network 172.16.1.0 to 192.168.150.0 255.255.255.0
```

Configuring Native VLAN on Ethernet Ports

A typical deployment of WGB is that a single wired client connects directly to the WGB Ethernet port. As a result, wired client traffic must be on the same VLAN as the WGB (or WLC/AP/WGB) management VLAN. If you need the wired client traffic to be on a different VLAN other than the WGB management VLAN, you should configure native VLAN on the Ethernet port.

Note Configuring native VLAN ID per Ethernet port is not supported. Both Ethernet ports share the same native VLAN configuration.

Note When WGB broadcast tagging is enabled and a single wired passive client connects directly to the WGB Ethernet port, it may hit the issue that infrastructure DS side client fails to ping this WGB behind the passive client. The workaround is to configure the following additional commands: **configure wgb ethport native-vlan enable** and **configure wgb ethport native-vlan id X**, where X is the same VLAN as the WGB (or WLC/AP/WGB) management VLAN.

The following table provides the commands to configure native VLAN:

Table 4: Native VLAN Configuration Commands

Command	Description
#**config wgb ethport native-vlan** {**enable**|**disable**} **Example:** `#config wgb ethport native-vlan enable`	Enables or disables native VLAN configuration.
#**config wgb ethport native-vlan id** <*vlan-id*> **Example:** `#config wgb ethport native-vlan id 2735`	Specifies native VLAN ID.

To verify your configuration, use the **show wgb ethport config** or **show running-config** command.

Low Latency Profile

IEEE 802.11 networks have a great role to play in supporting and deploying the Internet of Things (IoT) for the low latency and QoS requirement by applying the Enhanced Distributed Channel Access (EDCA), aggregated MAC protocol data unit (AMPDU), and aggregated or non-aggregated packet retry.

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

Configuring WGB optimized-video EDCA Profile

To configure optimized low latency profile for video use case, use the following command:

#**configure dot11Radio** <*radio_slot_id*> **profile optimized-video** {**enable** | **disable**}

Use the following command to verify the configuration:

```
WGB1#show controllers dot11Radio 1
EDCA profile: optimized-video
EDCA in use
=============
AC Type CwMin CwMax Aifs Txop ACM
AC_BE L 4 10 11 0 0
AC_BK L 6 10 11 0 0
AC_VI L 3 4 2 94 0
AC_VO L 2 3 1 47 0

Packet parameters in use
=============
wbridge1 A-MPDU Priority 0: Enabled
wbridge1 A-MPDU Priority 1: Enabled
wbridge1 A-MPDU Priority 2: Enabled
wbridge1 A-MPDU Priority 3: Enabled
wbridge1 A-MPDU Priority 4: Disabled
wbridge1 A-MPDU Priority 5: Disabled
wbridge1 A-MPDU Priority 6: Disabled
wbridge1 A-MPDU Priority 7: Disabled
wbridge1 A-MPDU subframe number: 3
wbridge1 Packet retries drop threshold: 16
```

Configuring WGB optimized-automation EDCA Profile

To configure optimized low latency profile for automation use case, use the following command:

#**configure dot11Radio** <*radio_slot_id*> **profile optimized-automation** {**enable** | **disable**}

Use the following command to verify the configuration:

```
WGB1#show controllers dot11Radio 1
EDCA profile: optimized-automation
EDCA in use
=============
AC Type CwMin CwMax Aifs Txop ACM
AC_BE L 7 10 12 0 0
AC_BK L 8 10 12 0 0
AC_VI L 7 7 3 0 0
AC_VO L 3 3 1 0 0

Packet parameters in use
=============
wbridge1 A-MPDU Priority 0: Enabled
wbridge1 A-MPDU Priority 1: Enabled
wbridge1 A-MPDU Priority 2: Enabled
wbridge1 A-MPDU Priority 3: Enabled
wbridge1 A-MPDU Priority 4: Disabled
wbridge1 A-MPDU Priority 5: Disabled
wbridge1 A-MPDU Priority 6: Disabled
wbridge1 A-MPDU Priority 7: Disabled
wbridge1 A-MPDU subframe number: 3
wbridge1 Packet retries drop threshold: 16
```

Configuring WGB customized-wmm EDCA profile

To configure customized Wi-Fi Multimedia (WMM) profile, use the following command:

#**configure dot11Radio** <*radio_slot_id*> **profile customized-wmm** {**enable** | **disable**}

To configure customized WMM profile parameters, use the following command:

#**configure dot11Radio** {**0**|**1**|**2**} **wmm** {**be** | **vi** | **vo** | **bk**} {**cwmin** <*cwmin_num*> | **cwmax** <*cwmax_num*> | **aifs** <*aifs_num*> | **txoplimit** <*txoplimit_num*>}

Parameter descriptions:

- be—best-effort traffic queue (CS0 and CS3)
- bk—background traffic queue (CS1 and CS2)
- vi—video traffic queue (CS4 and CS5)
- vo—voice traffic queue (CS6 and CS7)
- aifs—Arbitration Inter-Frame Spacing, <1-15> in units of slot time
- cwmin—Contention Window min, <0-15> 2^n-1, in units of slot time
- cwmax—Contention Window max, <0-15> 2^n-1, in units of slot time
- txoplimit—Transmission opportunity time, <0-255> integer number, in units of 32us

Configuring Low Latency Profile on WGB

Use the following command to configure low latency profile on WGB:

AP# **configure dot11Radio** <*radio_slot_id*> **profile low-latency** [**ampdu** <*length*>] [**sifs-burst** {**enable** | **disable**}] [**rts-cts** {**enable** | **disable**}] [**non-aggr** <*length*>] [**aggr** <*length*>]

Use the following command to display iot-low-latency profile EDCA detailed parameters:

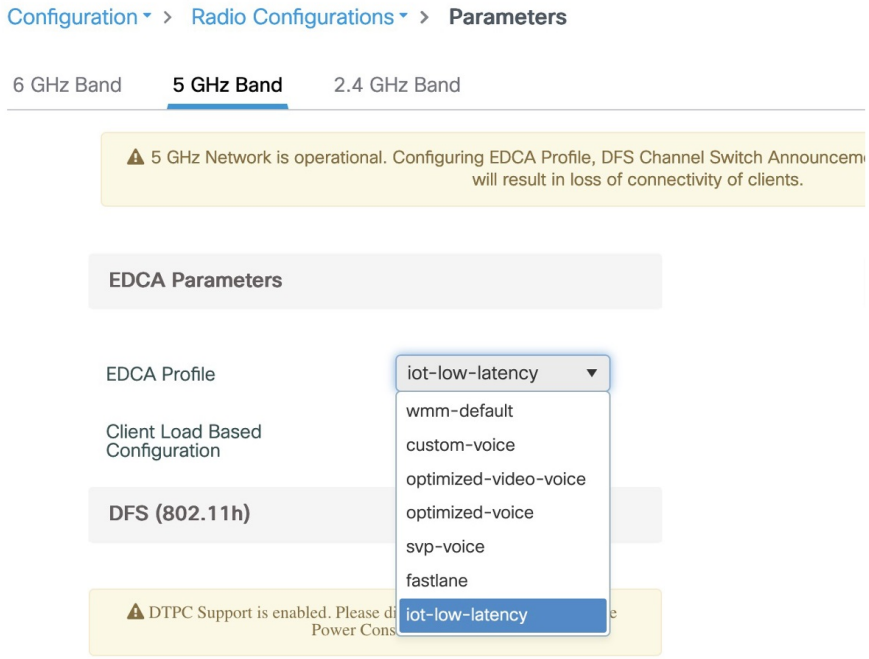
```
#show controllers dot11Radio 1 | beg EDCA
EDCA config
L: Local C:Cell A:Adaptive EDCA params
  AC   Type  CwMin  CwMax Aifs Txop ACM
AC_BE     L      4      6   11    0   0
AC_BK     L      6     10   11    0   0
AC_VI     L      3      4    1    0   0
AC_VO     L      0      2    0    0   1
AC_BE     C      4     10   11    0   0
AC_BK     C      6     10   11    0   0
AC_VI     C      3      4    2   94   0
AC_VO     C      2      3    1   47   1
```

Configuring EDCA Parameters (Wireless Controller GUI)

Step 1 Choose **Configuration > Radio Configurations > Parameters**. Using this page, you can configure global parameters for 6 GHz, 5 GHz, and 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 (Wireless Controller CLI)

Step 1 Enters global configuration mode.

configure terminal

Example:

```
Device# configure terminal
```

Step 2 Disables the radio network.

ap dot11 {**5ghz** | **24ghz** | **6ghz**} **shutdown**

Example:

```
Device(config)# ap dot11 5ghz shutdown
```

Step 3 Enables iot-low-latency EDCA profile for the 5 GHz, 2.4 GHz, or 6 GHz network.

ap dot11 {**5ghz** | **24ghz** | **6ghz**} **edca-parameters iot-low-latency**

Example:

```
Device(config)# ap dot11 5ghz edca-parameters iot-low-latency
```

Step 4 Enables the radio network.

no ap dot11 {**5ghz** | **24ghz** | **6ghz**} **shutdown**

Example:

```
Device(config)# no ap dot11 5ghz shutdown
```

Step 5 Returns to privileged EXEC mode.

end

Example:

```
Device(config)# end
```

Step 6 Displays the current configuration.

show ap dot11 {**5ghz** | **24ghz** | **6ghz**} **network**

Example:

```
Device(config)# show ap dot11 5ghz network
EDCA profile type check                  : iot-low-latency
```

Configuring A-MPDU

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).

The A-MPDU parameters define the size of an aggregated packet and define the proper spacing between aggregated packets so that the receive side WLAN station can decode the packet properly.

To configure profiled based A-MPDU under 2.4G, 5G and 6G radio, use the following commands:

WLC(config)# **ap dot11** {**5ghz** | **24ghz** | **6ghz**} **rf-profile** <*profile-name*>

WLC(config-rf-profile)# [**no**] **dot11n a-mpdu tx block-ack window-size** <*1-255*>

Global configuration is a special profile which can also be configured bu using the following command:

WLC(config)#[**no**] **ap dot11** {**5ghz** | **24ghz** | **6ghz**} **dot11n a-mpdu tx block-ack window-size** <*1-255*>

To bind different RF profiles with the radio RF tag, use the following command:

WLC(config)# **wireless tag rf** <*rf-tag-name*>

WLC (config-wireless-rf-tag)# **5ghz-rf-policy** <*rf-profile-name*>

Note RF profile level configured **a-mpdu tx block-ack window-size** value takes preference over globally configured value.

To display configured a-mpdu length value, use the following command:

show controllers dot11Radio *<radio_slot_id>*

```
AP# show controllers dot11Radio 1
Radio Aggregation Config:
=========================

TX A-MPDU Priority: 0x3f
TX A-MSDU Priority: 0x3f
TX A-MPDU Window:   0x7f
```

Configuring and Validating SNMP With WGB

Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language that is used for monitoring and managing devices in a network.

WGBs provide network administrators with an SNMP interface, allowing them to poll various states and counters. This enables administrators to easily monitor the health of their WGBs in the field.

By default, SNMP is disabled.

The SNMP framework has the following components, which are as follows.

- **SNMP Manager :** The Simple Network Management Protocol (SNMP) manager is a system that controls and monitors the activities of network hosts using SNMP. The most common managing system is a network management system (NMS). The term NMS can be applied either to a dedicated device used for network management or to the applications used on such a device.
- **SNMP Agent:** The Simple Network Management Protocol (SNMP) agent is the software component within a managed device that maintains the data for the device and reports this data, as needed, to managing systems.
- **SNMP MIB:** An SNMP agent contains MIB variables, whose values the SNMP manager can request or change through Get or Set operations. A manager can get a value from an agent or store a value in that agent. The agent gathers data from the SNMP MIB, the repository for information about device parameters and network data. The agent can also respond to manager requests to get or set data.

The following illustration shows the SNMP process. SNMP agent receives a request from SNMP client, and it passes the request to the subagent. The subagent then returns a response to the SNMP agent and the agent creates an SNMP response packet and sends the response to the remote network management station that initiated the request.

Figure 3: SNMP Process

SNMP Versions

Cisco IOS software supports the following versions of SNMP:

- SNMPv2c—The community-string-based administrative framework for SNMPv2. SNMPv2c is an update of the protocol operations and data types of SNMPv2p (SNMPv2 classic), and uses the community-based security model of SNMPv1.

- SNMPv3—Version 3 of SNMP. SNMPv3 uses the following security features to provide secure access to devices:
  - Message integrity—Ensuring that a packet has not been tampered with in transit.
  - Authentication—Determining that the message is from a valid source.
  - Encryption—Scrambling the contents of a packet to prevent it from being learned by an unauthorized source.

Supported SNMP MIB File

The Management Information Base (MIB) is a database of the objects that can be managed on a device. The managed objects, or variables, can be set or read to provide information on the network devices and interfaces and are organized hierarchically. The MIB consists of collections of managed objects identified by object identifiers. MIBs are accessed using a network management protocol such as SNMP.

The MIB module provides network management information on IEEE 802.11 wireless device association management and data packet forwarding configuration and statistics.

An Object Identifier (OID) uniquely identifies a MIB object on a managed network device. The OID identifies the MIB object's location in the MIB hierarchy, and provides a means of accessing the MIB object in a network of managed devices

Given below is a list of objects that are supported by the SNMP Management and Information Base (MIB): CISCO-DOT11-ASSOCIATION-MIB.

Table 5: Supported OIDs

OID Object Name	OID	OID Type	OID Description
cDot11ParentAddress	1.3.6.1.4.1.9.9.273.1.1.1	String	Provides the MAC address of the parent access point.
cDot11ActiveWirelessClients	1.3.6.1.4.1.9.9.273.1.1.2.1.1	Gauge	The device on this interface is currently associating with the number of wireless clients.
cDot11ActiveBridges	1.3.6.1.4.1.9.9.273.1.1.2.1.2	Gauge	The device on this interface is currently associating with the number of bridges.
cDot11ActiveRepeaters	1.3.6.1.4.1.9.9.273.1.1.2.1.3	Gauge	The device on the interface is currently associating with the number of repeaters.

OID Object Name	OID	OID Type	OID Description
cDot11AssStatsAssociated	1.3.6.1.4.1.9.9.273.1.1.3.1.1	Counter	When device restarts, the object counts the number of stations associated with the device on the interface.
cDot11AssStatsAuthenticated	1.3.6.1.4.1.9.9.273.1.1.3.1.2	Counter	When the device restarted, it currently counts the number of stations authenticated with the device on the interface.
cDot11AssStatsRoamedIn	1.3.6.1.4.1.9.9.273.1.1.3.1.3	Counter	When the device restarted, the object counts the number of stations roamed from another device to the device on the interface.
cDot11AssStatsRoamedAway	1.3.6.1.4.1.9.9.273.1.1.3.1.4	Counter	This object counts the number of stations roamed away from the device on the interface since device re-started.
cDot11AssStatsDeauthenticated	1.3.6.1.4.1.9.9.273.1.1.3.1.5	Counter	This object counts the number of stations deauthenticated with this device on the interface since device re-started
cDot11AssStatsDisassociated	1.3.6.1.4.1.9.9.273.1.1.3.1.6	Counter	This object counts the number of stations disassociated with this device on the interface since device re-started

OID Object Name	OID	OID Type	OID Description
cd11IfCipherMicFailClientAddress	1.3.6.1.4.1.9.9.273.1.1.4.1.1	String	This is MAC address of the client attached to the radio interface that caused the most recent MIC failure
cd11IfCipherTkipLocalMicFailures	1.3.6.1.4.1.9.9.273.1.1.4.1.2	Counter	When the device restarted, the object counts the number of MIC failures encountered on the radio interface.
cd11IfCipherTkipRemotMicFailures	1.3.6.1.4.1.9.9.273.1.1.4.1.3	Counter	When the device restarted, the object counts the number of MIC failures reported by clients on the radio interface.
cd11IfCipherTkipCounterMeasInvok	1.3.6.1.4.1.9.9.273.1.1.4.1.4	Counter	When the device restarted, the object counts the number of TKIP Counter Measures invoked on the interface.
cd11IfCipherCcmpReplaysDiscarded	1.3.6.1.4.1.9.9.273.1.1.4.1.5	Counter	When the device restarted, the object counts the number of received unicast fragments discarded by replay mechanism on the interface.
cd11IfCipherTkipReplaysDetected	1.3.6.1.4.1.9.9.273.1.1.4.1.6		When the device restarted, the object counts the number of TKIP replay errors detected on this interface.
cDot11ClientRoleClassType	1.3.6.1.4.1.9.9.273.1.2.1.1.3	Counter	The role classification of the client
cDot11ClientDevType	1.3.6.1.4.1.9.9.273.1.2.1.1.4	EnumVal	The device type of the client.

OID Object Name	OID	OID Type	OID Description
cDot11ClientRadioType	1.3.6.1.4.1.9.9.273.1.2.1.1.5	EnumVal	The radio classification of the client.
cDot11ClientWepEnabled	1.3.6.1.4.1.9.9.273.1.2.1.1.6	EnumVal	Whether WEP key mechanism is used for transmitting frames of data for the client
cDot11ClientWepKeyMixEnabled	1.3.6.1.4.1.9.9.273.1.2.1.1.7	EnumVal	Whether this client is using WEP key mixing
cDot11ClientMicEnabled	1.3.6.1.4.1.9.9.273.1.2.1.1.8	EnumVal	Whether the MIC is enabled for the client
cDot11ClientPowerSaveMode	1.3.6.1.4.1.9.9.273.1.2.1.1.9	EnumVal	The power management mode of the client.
cDot11ClientAid	1.3.6.1.4.1.9.9.273.1.2.1.1.10	Gauge	This is the association identification number of clients or multicast addresses associating with the device.
cDot11ClientDataRateSet	1.3.6.1.4.1.9.9.273.1.2.1.1.11	String	Is a set of data rates at which this client can transmit and receive data
cDot11ClientSoftwareVersion	1.3.6.1.4.1.9.9.273.1.2.1.1.12	String	Cisco IOS software version
cDot11ClientName	1.3.6.1.4.1.9.9.273.1.2.1.1.13	String	Cisco IOS device hostname
cDot11ClientAssociationState	1.3.6.1.4.1.9.9.273.1.2.1.1.14	EnumVal	The object indicates the state of the authentication and association process

OID Object Name	OID	OID Type	OID Description
cDot11ClientVlanId	1.3.6.1.4.1.9.9.273.1.2.1.1.17	Gauge	The VLAN which the wireless client is assigned to when it is successfully associated to the wireless station.
cDot11ClientSubIfIndex	1.3.6.1.4.1.9.9.273.1.2.1.1.18	Integer	This is the ifIndex of the sub-interface which this wireless client is assigned to when it is successfully associated to the wireless station.
cDot11ClientAuthenAlgorithm	1.3.6.1.4.1.9.9.273.1.2.1.1.19	EnumVal	The IEEE 802.1x authentication methods performed between the wireless station and this client during association
cDot11ClientDot1xAuthenAlgorithm	1.3.6.1.4.1.9.9.273.1.2.1.1.21	Octet String	The IEEE 802.1x authentication methods performed between the wireless client and the authentication server.
cDot11ClientUpTime	1.3.6.1.4.1.9.9.273.1.3.1.1.2	Gauge	The time in seconds that this client has been associated with this device
cDot11ClientSignalStrength	1.3.6.1.4.1.9.9.273.1.3.1.1.3	Integer	The device-dependent measure the signal strength of the most recently received packet from the client.

OID Object Name	OID	OID Type	OID Description
cDot11ClientSigQuality	1.3.6.1.4.1.9.9.273.1.3.1.1.4	Gauge	The device-dependent measure the signal quality of the most recently received packet from the client.
cDot11ClientPacketsReceived	1.3.6.1.4.1.9.9.273.1.3.1.1.6	Counter	The number of packets received from this client.
cDot11ClientBytesReceived	1.3.6.1.4.1.9.9.273.1.3.1.1.7	Counter	The number of bytes received from the client.
cDot11ClientPacketsSent	1.3.6.1.4.1.9.9.273.1.3.1.1.8	Counter	The number of packets sent to the client.
cDot11ClientBytesSent	1.3.6.1.4.1.9.9.273.1.3.1.1.9	Counter	The number of bytes sent to the client.
cDot11ClientMsduRetries	1.3.6.1.4.1.9.9.273.1.3.1.1.11	Counter	The counter increases when it successfully transmits an MSDU after one or more retransmissions.
cDot11ClientMsduFails	1.3.6.1.4.1.9.9.273.1.3.1.1.12	Counter	The counter increments when the client fails to transmit an MSDU successfully because the number of transmit attempts exceeds a certain limit.

Configuring SNMP from the WGB CLI

The following CLI commands are used for SNMP configuration.

Note

- SNMP CLI logic modified for SNMP configuration, all parameters of SNMP are required to be configured before enable SNMP feature by CLI: configure snmp enabled.
- All the related configurations of SNMP will be removed automatically when disable SNMP feature.

Step 1 Enter the **SNMP v2c community ID** number (SNMP v2c only).

Device#**configure snmp v2c community-id** <**length 1-64**>

Step 2 Specify the **SNMP protocol version**.

Device#**configure snmp version** {**v2c** | **v3**}

Step 3 Specify the **SNMP v3 authentication** protocol (SNMP v3 only).

Device#**configure snmp auth-method** <**md5** | **sha**>

Step 4 Enter the **SNMP v3 username** (SNMP v3 only).

Device#**configure snmp v3 username** <**length 32**>

Step 5 Enter the **SNMP v3 user password** (SNMP v3 only).

Device#**configure snmp v3 password** <**length 8-64**>

Step 6 Specify the **SNMP v3 encryption protocol** (SNMP v3 only).

Device#**configure snmp encryption** {**des** | **aes** | **none**}

Note Possible encryption values are des or aes. Alternatively, enter none if a v3 encryption protocol is not needed.

Step 7 Enter the **SNMP v3 encryption passphrase** (SNMP v3 only).

Device#**configure snmp secret** <**length 8-64**>

Step 8 **Enable SNMP** functionality in WGB.

Device#**configure snmp enabled**

To configure SNMP **v2c**, repeat Step 1 through Step 2 and Step 8.

To configure SNMP **v3**, repeat Step 2 through Step 8.

Step 9 **Disable SNMP configuration**.

Device#**configure snmp disabled**

When SNMP is disabled, all related configuration is removed.

Example

Example of SNMP configuration.

- **CLI for configuring SNMP v2c:**

```
Device#configure snmp v2 community-id <length 1-64>
Device#configure snmp version v2c
Device#configure snmp enabled
```

- **CLI for configuring SNMP v3 (security level AuthPriv):**

```
Device#configure snmp auth-method <md5|sha>
Device#configure snmp v3 username <length 32>
```

```
Device#configure snmp v3 password <length 8-64>
Device#configure snmp secret <length 8-64>
Device#configure snmp encryption <aes|des>
Device#configure snmp version v3
Device#configure snmp enabled
```

- **CLI for configuring SNMP v3 (security level AuthNoPriv):**

```
Device#configure snmp auth-method <md5|sha>
Device#configure snmp v3 username <length 32>
Device#configure snmp v3 password <length 8-64>
Device#configure snmp encryption none
Device#configure snmp version v3
Device#configure snmp enabled
```

Verifying SNMP from WGB CLI

Use the following show command to verify the SNMP configuration.

- **Show output of SNMP version v3:**

```
Device# show snmp
SNMP: enabled
Version: v3
Community ID: test
Username: username
Password: password
Authentication method: SHA
Encryption: AES
Encryption Passphrase: passphrase
Engine ID: 0x8000000903c0f87fe5f314
```

- **Show output of SNMP version v2c:**

```
Device# show snmp
SNMP: enabled
Version: v2c
Community ID: test
Username: username
Password: password
Authentication method: SHA
Encryption: AES
Encryption Passphrase: passphrase
Engine ID: 0x8000000903c0f87fe5f314
```

Support for QoS ACL Classification and Marking

Starting from Cisco Unified Industrial Wireless Software Release 17.14.1, WGB allows you to classify different packets from two wired ports and mark them to the different access control driver queues according to the user configuration.

In addition to TCP or UDP, WGB also supports ethertype-based and DSCP-based classification. To meet the jitter and latency requirement, the WGB must classify packets and assign them to different access control queues based on the field environment.

Overview

WGB allows you to create custom rules to map incoming packets from an Ethernet port to specific priority queues on the wireless side. WGB offers the functionality to map upstream data traffic based on either IEEE 802.1p (dot1p) or Differentiated Services Code Point (DSCP).

You can configure the rules based on Ethernet type (for example, Profinet), transport layer port numbers or port range, and DSCP. It ensures forwarding packets to the different access control queues on the wireless network, facilitating efficient QoS enforcement.

As incoming packets arrive at the Ethernet port, it directs them to a specific access control queue on the wireless side using a customized rule-based mapping.

The customized rule dictates the classification and assignment of packets to different access control queues based on predetermined criteria such as source/destination IP addresses, port numbers, or protocol types. Once defined, the rules identify critical services or traffic within the incoming packets. Matching these critical services using the defined rules enables mapping them to higher priority queues within the network infrastructure.

Using rule-based traffic classification and mapping on the WGB, you can effectively manage and prioritize network traffic to meet the specific demands of critical applications and services. This approach enables you to enforce QoS policies effectively within your network to maintain optimal network performance, minimizes latency for critical services, and enhances overall user experience.

Traffic Classification Based on QoS and ACL

Classification is the process of distinguishing one traffic from another by examining the fields in the packet. The device enables classification only when QoS is enabled.

During classification, the device performs a lookup and assigns a QoS label to the packet. The QoS label indicates all QoS actions to perform on the packet and identifies the queue from which the packet is sent.

Layer 2 ethernet frames use the Ethertype field to carry classification information. The ethertype field, typically 2 bytes in size, normally indicates the type of data encapsulated in the frames

Layer 3 IP packets carry the classification information in the type of service (ToS) field that has 8 bits. The ToS field carries either an IP precedence value or a Differentiated Services Code Point (DSCP) value. IP precedence values range 0–7. DSCP values range 0–63.

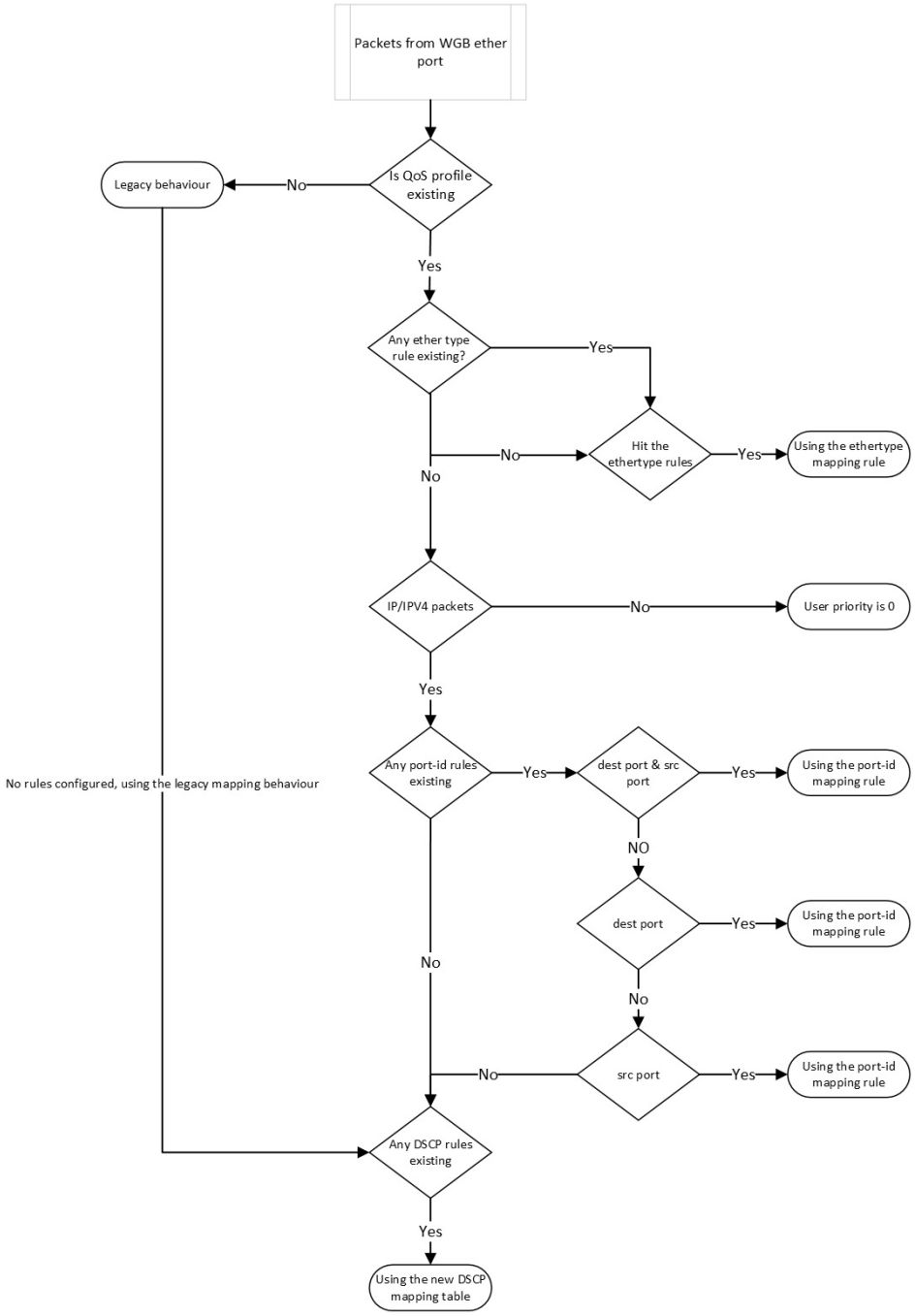
Layer 4 TCP segments or UDP datagrams carry the classification information in the source or destination port field. These port fields specify the port numbers associated with the sender and receiver of the data, enabling networking devices to classify traffic based on predetermined criteria.

The system assigns traffic to a specific service class based on ether type, DSCP, or UDP/TCP port (or port range), treating packets within the service class consistently. The WGB help to classify different packets from the two wired ports and map them to the different driver queues according to the user config.

The data plane statistics provide counts of how many times each rule hit by network traffic. These counters are essential for network administrators to analyse the effectiveness of their rules and policies, and optimize network performance.

The control plane is a part of a network architecture responsible for managing and configuring how data is forwarded though the network.

Figure 4: Flowchart of traffic flows from WGB ethernet port



When QoS is disabled, access points follows the legacy mapping behavior and perform the following:

1. Retrieve the Tag Control Information (TCI) priority from the VLAN element for the specified ethertype 0x8100.
2. For ethertype 0x8892 (profinet) QoS mapping, assigns the TCI priority as 6.

3. For ethertype 0x0800 (IP) and 0x86DD (IPv6), the DSCP priority is set according to the default dscp2dot1p mapping table.

```
======= dscp mapping =======
Default dscp2dot1p Table Value:
[0]->0 [1]->0 [2]->0 [3]->0 [4]->0 [5]->0 [6]->0 [7]->0
[8]->1 [9]->1 [10]->1 [11]->1 [12]->1 [13]->1 [14]->1 [15]->1
[16]->2 [17]->2 [18]->2 [19]->2 [20]->2 [21]->2 [22]->2 [23]->2
[24]->3 [25]->3 [26]->3 [27]->3 [28]->3 [29]->3 [30]->3 [31]->3
[32]->4 [33]->4 [34]->4 [35]->4 [36]->4 [37]->4 [38]->4 [39]->4
[40]->5 [41]->5 [42]->5 [43]->5 [44]->5 [45]->5 [46]->5 [47]->5
[48]->6 [49]->6 [50]->6 [51]->6 [52]->6 [53]->6 [54]->6 [55]->6
[56]->7 [57]->7 [58]->7 [59]->7 [60]->7 [61]->7 [62]->7 [63]->7
```

When QoS is enabled, access points perform the following:

1. The priority for an ethertype QoS mapping 0x8892 (profinet) is based on the configuration setting.
2. For ethertype 0x0800 (IP) and 0x86DD (IPv6), the priority is based on mapping rules that consider port or DSCP.
   - Check the UDP/TCP port (or port range) rule.
   - Check the DSCP rule.
3. Assigns the user priority value 0 to non-IPv4/IPv6 packets.
4. If there is no rule configuration, the QoS profile follows the legacy mapping behavior.

Note if 802.1p priority exists, it overrides any customised rule.

Configuring Quality of Service Mapping Profile

The following commands allow users to define the different classification rules for configuring WGB QoS mapping.

Step 1 Enable the QoS mapping profile.

Device#**config wgb qos-mapping** <**profile-name**> **enable**

Example:

```
Device#configure wgb qos-mapping demo-profile enable
```

Step 2 WGB QoS mapping profile rules based on **ethernet type**.

The below command is used to set the rules based on ethernet frame type.

- Add rules based on ethernet type.

  Device#**config wgb qos-mapping** <**profile-name**> **add ethtype hex** <**number**> **priority** <**0-7**>

Example:

```
Device#configure wgb qos-mapping demo-profile add ethtype hex 8892 priority 5
```

If the command specify a profile that does not exist, the command will create a new empty profile and then add mapping rule to it.

- Delete rules based on ethernet type

  Device#**config wgb qos-mapping** <**profile-name**> **delete ethtype hex** <**number**>

Example:

```
Device#configure wgb qos-mapping demo-profile delete ethtype hex 8892
```

The command will issue a warning message if it specifies a profile that does not exist. Furthermore, if deleting the specified mapping rule leaves the profile empty, it will be automatically removed.

Step 3 Rules based on **port-id/range**.

The below command is used to set the rules based on L4 port id/range.

- Add rules based on port-id/range.

  Device#**config wgb qos-mapping** <**profile-name**> add **srcport** <**number**> | <**range** <**start-number**> <**end-number**>> [**dstport** <**number**> | <**range** <**start-number**> <**end-number**>>] **priority** <**0-7**>

Example:

```
Device#configure wgb qos-mapping demo-profile add srcport range 5050 5070 dstport 8000 priority 3
```

If the command specify a profile that does not exist, the command will create a new empty profile and then add mapping rule to it.

- Delete rules based on port-id/range.

  Device#**config wgb qos-mapping** <**profile-name**> delete [**srcport** <**number**> | <**range** <**start-number**> <**end-number**>> [**dstport** <**number**> | <**range** <**start-number**> <**end-number**>>]]

Example:

```
Device#configure wgb qos-mapping demo-profile delete srcport range 5050 5070 dstport 8000
```

The command will issue a warning message if it specifies a profile that does not exist. Furthermore, if deleting the specified mapping rule leaves the profile empty, it will be automatically removed.

Step 4 Rules based on **DSCP**.

The below command is used to set the rules based on IPv4/IPv6 packet DSCP value.

- Add

  Device#**config wgb qos-mapping** <**profile-name**> add dscp <**number**> priority < **0-7**>

Example:

```
Device#configure wgb qos-mapping demo-profile add dscp 63 priority 4
```

If the command specify a profile that does not exist, the command will create a new empty profile and then add mapping rule to it.

- Delete

  Device#**config wgb qos-mapping** <**profile-name**> delete dscp <**number**> priority < **0-7**>

Example:

```
Device#configure wgb qos-mapping demo-profile delete dscp 63
```

The command will issue a warning message if it specifies a profile that does not exist. Furthermore, if deleting the specified mapping rule leaves the profile empty, it will be automatically removed.

Note After deleting the DSCP mapping rule, the rules are reset to the default values of the DSCP mapping.

Step 5 Disable the QoS mapping profile.

Device#**config wgb qos-mapping** <**profile-name**> disable

Example:

```
Device#configure wgb qos-mapping demo-profile disable
```

When disabled, the command clear the profile from the datapath and retain it in the WGB configuration file. If the specified profile does not exist, the command issue a warning message and will not create a new empty profile.

Step 6 Delete the QoS mapping profile.

Device#**config wgb qos-mapping** <**profile-name**> delete

Example:

```
Device#configureure wgb qos-mapping demo-profile delete
```

When deleted, the profile is removed from data path and WGB configuration.

Verifying WGB Quality of Service Mapping

To verify the WGB QoS mapping configuration on the Control Plane, run the **show wgb qos-mapping**.

```
Device# show wgb qos-mapping

Number of QoS Mapping Profiles: 2
====================================
Profile name : qos1
Profile status : active
Number of Rules: 8
Rules:
L4 srcport : 31000-31100, dstport : 6666-7777, priority : 7
L4 srcport : 23000, dstport : N/A, priority : 3
L4 srcport : N/A, dstport : 20000-20100, priority : 5
L4 srcport : N/A, dstport : 2222, priority : 2
L4 srcport : 12300-12500, dstport : N/A, priority : 6
IPv4/IPv6 dscp: 43, priority : 1
Ethernet type : 0x8892, priority : 0
L4 srcport : 8888, dstport : 9999, priority : 4

Profile name : qos2
Profile status : inactive
Number of Rules: 8
Rules:
L4 srcport : 31000-31100, dstport : 6666-7777, priority : 2
L4 srcport : 23000, dstport : N/A, priority : 6
L4 srcport : N/A, dstport : 20000-20100, priority : 4
L4 srcport : N/A, dstport : 2222, priority : 7
L4 srcport : 12300-12500, dstport : N/A, priority : 3
IPv4/IPv6 dscp: 43, priority : 0
Ethernet type : 0x8892, priority : 1
L4 srcport : 8888, dstport : 9999, priority : 5
```

To verify the WGB QoS mapping configuration on the Data Plane, run the **show datapath qos-mapping rule**.

```
Device# show datapath qos-mapping rule

Status: active
QoS Mapping entries
======= dscp mapping =======
Default dscp2dot1p Table Value:
[0]->0 [1]->0 [2]->0 [3]->0 [4]->0 [5]->0 [6]->0 [7]->0
[8]->1 [9]->1 [10]->1 [11]->1 [12]->1 [13]->1 [14]->1 [15]->1
[16]->2 [17]->2 [18]->2 [19]->2 [20]->2 [21]->2 [22]->2 [23]->2
[24]->3 [25]->3 [26]->3 [27]->3 [28]->3 [29]->3 [30]->3 [31]->3
[32]->4 [33]->4 [34]->4 [35]->4 [36]->4 [37]->4 [38]->4 [39]->4
[40]->5 [41]->5 [42]->5 [43]->5 [44]->5 [45]->5 [46]->5 [47]->5
[48]->6 [49]->6 [50]->6 [51]->6 [52]->6 [53]->6 [54]->6 [55]->6
[56]->7 [57]->7 [58]->7 [59]->7 [60]->7 [61]->7 [62]->7 [63]->7

active dscp2dot1p Table Value:
[0]->0 [1]->0 [2]->0 [3]->0 [4]->0 [5]->0 [6]->0 [7]->0
[8]->1 [9]->1 [10]->1 [11]->1 [12]->1 [13]->1 [14]->1 [15]->1
[16]->7 [17]->2 [18]->2 [19]->2 [20]->2 [21]->2 [22]->2 [23]->2
[24]->3 [25]->3 [26]->3 [27]->3 [28]->3 [29]->3 [30]->3 [31]->3
[32]->4 [33]->4 [34]->4 [35]->4 [36]->4 [37]->4 [38]->4 [39]->4
[40]->5 [41]->5 [42]->5 [43]->5 [44]->5 [45]->5 [46]->5 [47]->5
[48]->6 [49]->6 [50]->6 [51]->6 [52]->6 [53]->6 [54]->6 [55]->6
[56]->7 [57]->7 [58]->7 [59]->7 [60]->7 [61]->7 [62]->7 [63]->7
```

To verify the WGB QoS mapping statistics on Data Plane, run the **show datapath qos-mapping statistics** command.

```
Device# show datapath qos-mapping statistics

======= pkt stats per dscp-mapping rule =======
dscp up pkt_cnt
16 7 0
```

To clear the WGB QoS mapping statistics on Data Plane, run the **clear datapath qos-mapping statistics** command.

Note The command clears packet count statistics per rule on data-plane.

Packet Capture: TCP Dump on WGB

TCP Dump on WGB

The TCP dump utility is a network packet analyzer commonly used for network monitoring and data acquisition. When applied to a WGB, the TCP dump can capture, display, and save the packets transmitted over the wired interfaces of the WGB.

TCP Dump on WGB chapter provides information on how to enable TCP dump through the WGB wired interface on the Catalyst IW9165E.

Purpose of TCP Dump Utility

TCP dump on a WGB monitors and troubleshoots network communications, ensuring the WGB relays frames correctly between the wired clients and the wireless networks.

Functions of TCP Dump Utility

- display captured packets in real time on the WGB terminal, and
- capture packets to storage.

Note The TCP dump utility does not support the simultaneous capture of packets to storage and printing them on the WGB terminal.

Packet Capture Modes

- Default: Displays captured packets with header in the real time on the WGB terminal
- Verbose: Parses and prints real-time packets on the WGB terminal, displaying the headers and prints the data of each packet, including its link-level header, in hexadecimal format.

Note Reformat the verbose output for text2pcap compatibility.

In default or verbose mode, the WGB terminal can print a maximum of 1000 packet entries.

- Capture: Captures packets to a file storage instead of printing them in real time. Use the **show pcap** command to view the captured internal wired packets.

Note Every round of Packet Capture (PCAP) clears the existing PCAP file.

Before any new PCAP session, transfer the current PCAP file to an external server to prevent it from being overwritten.

PCAP stops automatically when the PCAP file reaches a size of 100 MB.

Protocol Packet Capture Capabilities on WGB

You can capture packets from an AP either using a default or custom filter through the WGB wired port and then upload them to an external server.

The default filter captures three main protocol packets such as IP, TCP, or UDP.

A custom filter captures specific packets that are relevant for troubleshooting specific issues or monitoring certain types of network activity.

You can use different protocol filters to capture packets for debugging. For instance, include the given protocols in your filter expression:

- Transmission Control Protocol
- Internet Control Message Protocol (ICMP) and ICMPv6
- Profinet with IP proto 0x8892
- Address Resolution Protocol (ARP)
- Internet Group Management Protocol (IGMP)
- User Datagram Protocol
- Dynamic Host Configuration Protocol (DHCP) with port 67 or port 68 and DHCPv6 with port 546 or port 547
- Common Industrial Protocol (CIP) with TCP port 44818
- Domain Name System (DNS) with port 53
- Simple Network Management Protocol with port 161 or port 162.

Note The protocols listed represent only a portion of the PCAP capabilities.

Filter expressions for packet captures

The filter expression for a PCAP comprises at least one primitive. Primitives usually consist of qualifiers followed by an identifier. The identifier can be a name or a number.

There are three kinds of qualifiers.

- Type: Specifies the type of the identifier. The type can be a port, a host, a network, or a range of ports.

  For example: **port** 20

- Dir: Specifies that the capture is for only packets with a given transfer direction.

  For example: **src x.x.x.x and port** ftp-data or **dst x.x.x.x and port** ftp

- Proto: Limits the capture to a specific protocol.

  For example: **tcp port** 21.

The filter expressions can be combined using the logical operators AND, OR, and NOT to create more specific and complex filters.

Note When constructing filter expressions, it is important to understand the order of operations and use parentheses to group expressions when necessary to ensure the correct interpretation.

Enable Wired Packet Capture on WGB

Step 1 To enable PCAP, choose one of the options given here:

a. PCAP using default filter:

Device#**debug traffic wired** [*0* | *1*] {**ip** | **tcp** | **udp**} [**verbose** | **capture**]

[0-1]: Specifies the wired interface number. If not selected, capture packets from all the wired interface.

This table lists examples of PCAP in default, verbose, and capture modes:

Mode	Example
Default: Captures IP protocol header packets.	Device#**debug traffic wired 1 ip** APXXXX.XXXX.XXXX#reading from file /dev/click_wired_log, link-type EN10MB (Ethernet) 1 08:35:50.529851 IP 209.165.200.213 > 209.165.200.1: ICMP echo request, id 13721, seq 1, length 64 2 08:35:50.534813 IP 209.165.200.1 > 209.165.200.213: ICMP echo reply, id 13721, seq 1, length 64
Verbose: Captures detailed information of the UDP protocol packets.	Device#**debug traffic wired 1 udp verbose** APXXXX.XXXX.XXXX#reading from file /dev/click_wired_log, link-type EN10MB (Ethernet) 1 08:25:59.696990 IP6 fe80::322c:712c:5787:f246.dhcpv6-client > ff02::1:2.dhcpv6-server: dhcp6 solicit 0x0000: 3333 0001 0002 fc58 9a16 e428 86dd 6001 0x0010: 7b92 006d 1101 fe80 0000 0000 0000 322c 0x0020: 712c 5787 f246 ff02 0000 0000 0000 0000 0x0030: 0000 0001 0002 0222 0223 006d 00a6 010c 0x0040: d064 0008 0002 ffff 0006 001e 0034 0011 0x0050: 0015 0016 0017 0018 001f 0038 0040 0043 0x0060: 0052 0053 005e 005f 0060 0001 000a 0003 0x0070: 0001 fc58 9a16 e428 0014 0000 0027 0013 0x0080: 0006 4150 4643 3538 0439 4131 3604 4534 0x0090: 3238 0000 0300 0c00 0000 0100 0000 0000 0x00a0: 0000 00
Capture: Writes TCP packet information to the PCAP file.	Device#**debug traffic wired 1 tcp capture** % Writing packets to "/pcap/APXXXX.XXXX.XXXX_capture.pcap0" APXXXX.XXXX.XXXX#reading from file /dev/click_wired_log, link-type EN10MB (Ethernet)

b. PCAP using custom filter:

Note Enable only one PCAP process at a time. Do not use unsupported characters like " ` $ ^ & | \ > < ? ; and ~ in the filter expressions.

Device#**debug traffic wired** [*0* | *1*] **filter** *expression* [**verbose** | **capture**]

This table lists examples of PCAP in default, verbose, and capture modes:

Mode	Example
Default: Captures IP protocol header packets.	Device#**debug traffic wired 0 filter icmp** APXXXX.XXXX.XXXX#reading from file /dev/click_wired_log, link-type EN10MB (Ethernet) 1 10:38:59.948729 IP 209.165.200.213 > 209.165.200.1: ICMP echo request, id 16204, seq 1, length 64 2 10:38:59.954308 IP 209.165.200.1 > 209.165.200.213: ICMP echo reply, id 16204, seq 1, length 64

Mode	Example
Verbose: Captures detailed information of the UDP protocol packets.	Device#**debug traffic wired 1 filter icmp verbose** APXXXX.XXXX.XXXX##reading from file /dev/click_wired_log, link-type EN10MB (Ethernet) 17:13:30.706493 IP 209.165.200.213 > 209.165.200.1: ICMP echo request, id 986, seq 1, length 64 0x0000: fc58 9a17 afd4 f8e4 3b9d 7322 0800 4500 0x0010: 0054 57a0 4000 4001 889e c0a8 6cc8 c0a8 0x0020: 6c51 0800 940c 03da 0001 7f3d 5365 0000 0x0030: 0000 cea2 0000 0000 0000 1011 1213 1415 0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 0x0060: 3637 17:13:30.710567 IP 209.165.200.1 > 209.165.200.213: ICMP echo reply, id 986, seq 1, length 64 0x0000: f8e4 3b9d 7322 fc58 9a17 afd4 0800 4500 0x0010: 0054 9102 0000 4001 8f3c c0a8 6c51 c0a8 0x0020: 6cc8 0000 9c0c 03da 0001 7f3d 5365 0000 0x0030: 0000 cea2 0000 0000 0000 1011 1213 1415 0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 0x0060: 3637
Capture: Writes TCP packet information to the PCAP file.	Device#**ddebug traffic wired 1 filter icmp capture** % Writing packets to "/tmp/pcap/APXXXX.XXXX.XXXX_capture.pcap0" APXXXX.XXXX.XXXX#reading from file /dev/click_wired_log, link-type EN10MB (Ethernet)

For more information on filter expressions, see *TCP dump pcap-filter* documentation.

c. PCAP in multiple vlan using custom filter:

Note Some custom filters miss traffic in non-native VLANs. For example, the custom filter command **#debug traffic wired 0 filter icmp** fails to capture downlink ICMP traffic in non-native VLANs.

To capture downlink traffic in non-native VLANs, you have two options:

- Include the VLAN in the filter expression to capture bidirectional traffic of the wired client in a non-native VLAN

```
Device#debug traffic wired 0 filter "icmp or (vlan and icmp)"
1  12:27:40.833815 IP 209.165.200.102 > 209.165.200.1: ICMP echo request, id 27279, seq 1, length 64
2  12:27:40.841331 IP 209.165.200.1 > 209.165.200.102: ICMP echo reply, id 27279, seq 1, length 64
```

- To capture all IP traffic including native vlan and non-native vlan, use the default IP filter.

```
Device#debug traffic wired 0 ip
1  12:27:40.833815 IP 209.165.200.102 > 209.165.200.1: ICMP echo request, id 27279, seq 1, length 64
2  12:27:40.841331 IP 209.165.200.1 > 209.165.200.102: ICMP echo reply, id 27279, seq 1, length 64
```

To disable wired PCAP, see Disable Wired Packet Capture on WGB.

Step 2 To upload the packets to an external server, use the command given here:

Note Before uploading the packets, complete the PCAP process and save the packets to file.

Use TFTP, SFTP, or SCP server to upload the PCAP file to an external server.

Device#**copy pcap APxxxx.xxxx.xxxx_capture.pcap0 <tftp|sftp>://A.B.C.D[/dir][/filename]**

copy pcap APxxxx.xxxx.xxxx_capture.pcap0 scp://username@A.B.C.D[:port]:/dir[/filename]

Example:

```
Device#copy pcap APXXXX.XXXX.XXXX_capture.pcap0 scp://iot@209.165.200.213:/capture/wgb_sniffer.pcap
copy ""/pcap/APXXXX.XXXX.XXXX_capture.pcap0"" to
"scp://iot@209.165.200.213:/capture/wgb_dhcp_sniffer_0_46_29.pcap" (Y/N)Y
iot@209.165.200.213 password:
APXXXX.XXXX.XXXX_capture.pcap0                 0%    0     0.0KB/s   --:-- ETA
APXXXX.XXXX.XXXX_capture.pcap0               100% 2530   916.5KB/s   00:00
```

Disable Wired Packet Capture on WGB

To disable PCAP, use the command given here:

a. Default filter:

Device#**no debug traffic wired** [*0-3*] {**ip** | **tcp** | **udp**} [**verbose** | **capture**]

b. Custom filter:

Device#**no debug traffic wired** [*0-3*] **filter** *expression* [**verbose** | **capture**]

Note Use either the **no debug** or **undebug all** command to terminate the capture process.

Verify Wired Packet Capture on WGB

- To verify the debug status, use the **show debug** command.

```
Device#show debug
traffic:
  wired tcp debugging is enabled
```

- To view the captured internal wired packets stored in the file, use the **show pcap** command.

Note After capturing packets to the file, use the **show pcap** command to view them.

```
Device#show pcap
reading from file /pcap/APXXXX.XXXX.XXXX_capture.pcap0, link-type EN10MB (Ethernet)
1  00:00:00.000000 IP 0.0.0.0 > 224.0.0.1: igmp query v2
2  09:41:48.903670 IP 209.165.200.189 > 209.165.200.1: ICMP echo request, id 29920, seq
 1, length 64
3  09:41:48.908927 IP 209.165.200.1 > 209.165.200.189: ICMP echo reply, id 29920, seq
1, length 64
4  09:41:49.904914 IP 209.165.200.102 > 209.165.200.1: ICMP echo request, id 29920, seq
 2, length 64
5  09:41:49.909009 IP 209.165.200.1 > 209.165.200.102: ICMP echo reply, id 29920, seq
2, length 64
```

- To filter and view the basic content of captured packets sequentially, run the **show pcap [filter** *expression***]** command.

```
Device#show pcap filter "src 209.165.200.189”
reading from file /pcap/APXXXX.XXXX.XXXX_capture.pcap0, link-type EN10MB (Ethernet)
 1  09:41:48.903670 IP 209.165.200.189 > 209.165.200.1: ICMP echo request, id 29920,
seq 1, length 64
 2  09:41:48.908927 IP 209.165.200.1 > 209.165.200.189: ICMP echo reply, id 29920, seq
 1, length 64
```

- To filter and view the detailed content of a specific packet, run the **show pcap [filter** *expression***][detail** *no***]** command.

```
Device#show pcap filter "src 209.165.200.189" detail 2
2024-04-25 09:41:49.904914
000000 18 59 f5 96 af 74 00 50 56 85 8a 0a 08 00 45 00
000010 00 54 14 6c 40 00 40 01 b7 9d 64 16 53 72 64 16
000020 53 01 08 00 70 81 74 e0 00 02 d4 3e 2b 66 00 00
000030 00 00 50 24 04 00 00 00 00 00 10 11 12 13 14 15
000040 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25
000050 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35
000060 36 37
```

AAA User Authentication Support

Information About AAA User Authentication Support

This chapter provides information on how to use AAA to control the use of network resources (via authentication) and define permissible actions (via authorization). From Release 17.15.1, AAA-based user management and authentication are supported on IW9165E WGB.

The AAA server assigns a privilege level from 0-15 to clients using an Authorization-Reply message. Only levels 1 (view user) and 15 (management user) are currently supported, with levels 2-14 reserved. Privilege levels 0 and 2-14 must not be used when adding users to the AAA server. If a user is added without a privilege level, WGB will assign the lowest privilege level to that user.

Features of AAA-based user management and authentication are as follows:

- Provides multiple-user support
- Stores usernames and passwords on the AAA server
- Utilizes AAA for user authentication
- Supports differentiated user privileges
- Restricts CLI access based on user privileges

Note Similar to a Cisco Router or Switch, the Workgroup Bridge (WGB) can also create and store usernames and passwords locally.

Configuring AAA Server

Before you begin

- You can add a secondary AAA server (RADIUS or TACACS+) before adding a primary AAA server. Once the primary AAA server is added, clients connect to the primary AAA server.
- When both primary and secondary RADIUS servers are configured, the WGB attempts to connect with the primary RADIUS server three times before switching to the secondary RADIUS server.
- For the TACACS+ server, the connection attempt is done only once with the primary TACACS+ server. If the primary TACACS+ server fails to respond, the secondary TACACS+ server is used.

Note The WGB AAA RADIUS server configuration command is officially supported starting from the 17.15.1 release.

When you downgrade the image from the 17.15.1 release or later to the 17.14.1 release or earlier, or upgrade from the 17.14.1 release or earlier to the 17.15.1 release or later, the originally configured RADIUS server port is reset to zero. You need to reconfigure the RADIUS server port again.

Step 1 Configure a AAA server (RADIUS or TACACS+) using the following command:

Device# **config** {**radius** | **tacplus**} **authentication** {**primary** | **secondary**} **address** {**ipv4** | **ipv6**} *ip-address* **port** *port-number* **secret** *secret-string*

Step 2 (Optional) To remove a AAA server (RADIUS or TACACS+), use the following command:

Device# **config** {**radius** | **tacplus**} **authentication** {**primary** | **secondary**} **delete**

Enable or Disable RADIUS Authentication for Login User

Step 1 Run the following command to enable AAA RADIUS authentication for the login user:

Device# **config ap management aaa radius enable**

Step 2 (Optional) Run the following command to disable AAA RADIUS authentication for the login user:

Device# **config ap management aaa radius disable**

Enable or Disable TACACS+ Authentication for Login User

Before you begin

Step 1 Run the following command to enable AAA TACACS+ authentication for the login user:

Device# **config ap management aaa tacplus enable**

Step 2 (Optional) Run the following command to disable AAA TACACS+ authentication for the login user:
Device# **config ap management aaa tacplus disable**

Verify the AAA Authentication Configuration

To verify the AAA server (RADIUS or TACACS+) configuration, use the **show running-configuration** command.

The following is a sample output when AAA RADIUS authentication is enabled:

```
Device# show running-config

AAA server configuration:-
==============================
Status: Enabled
AAA server type : radius
Primary RADIUS IP address : 192.0.2.0
Primary RADIUS port : 1812
.
.
.
```

The following is a sample output when AAA tacplus authentication is enabled:

```
Device# show running-config

AAA server configuration:-
==============================
Status: Enabled
AAA server type : tacplus
Primary TACPLUS IP address : 192.0.2.0
Primary TACPLUS port : 49
.
.
.
```

Radio Statistics Commands

To help troubleshooting radio connection issues, use the following commands:

- #**debug wgb dot11 rate**

```
#debug wgb dot11 rate
[*03/13/2023 18:00:08.7814]                MAC    Tx-Pkts    Rx-Pkts
  Tx-Rate(Mbps)                    Rx-Rate(Mbps)  RSSI   SNR Tx-Retries
[*03/13/2023 18:00:08.7814] FC:58:9A:17:C2:51          0          0
HE-20,2SS,MCS6,GI0.8 (154)       HE-20,3SS,MCS4,GI0.8 (154)   -30    62          0
[*03/13/2023 18:00:09.7818] FC:58:9A:17:C2:51          0          0
HE-20,2SS,MCS6,GI0.8 (154)       HE-20,3SS,MCS4,GI0.8 (154)   -30    62          0
```

In this example, FC:58:9A:17:C2:51 is the parent AP radio MAC.

- #**show interfaces dot11Radio** <*slot-id*> **statistics**

```
#show interfaces dot11Radio 1 statistics
Dot11Radio Statistics:
        DOT11 Statistics (Cumulative Total/Last 5 Seconds):
RECEIVER                                TRANSMITTER
Host Rx K Bytes:        965570/0        Host Tx K Bytes:       1611903/0
Unicasts Rx:            379274/0        Unicasts Tx:           2688665/0
Broadcasts Rx:         3166311/0        Broadcasts Tx:               0/0
Beacons Rx:          722130099/1631     Beacons Tx:          367240960/784
Probes Rx:           588627347/2224     Probes Tx:            78934926/80
Multicasts Rx:         3231513/0        Multicasts Tx:           53355/0
Mgmt Packets Rx:     764747086/1769     Mgmt Packets Tx:     446292853/864
Ctrl Frames Rx:        7316214/5        Ctrl Frames Tx:              0/0
RTS received:                0/0        RTS transmitted:             0/0
Duplicate frames:            0/0        CTS not received:            0/0
MIC errors:                  0/0        WEP errors:            2279546/0
FCS errors:                  0/0        Retries:                896973/0
Key Index errors:            0/0        Tx Failures:              8871/0
                                        Tx Drops:                    0/0

Rate Statistics for Radio::
[Legacy]:
6 Mbps:
 Rx Packets:     159053/0           Tx Packets:      88650/0
                                    Tx Retries:       2382/0
9 Mbps:
 Rx Packets:         43/0           Tx Packets:         23/0
                                    Tx Retries:         71/0
12 Mbps:
 Rx Packets:          1/0           Tx Packets:        119/0
                                    Tx Retries:        185/0
18 Mbps:
 Rx Packets:          0/0           Tx Packets:          5/0
                                    Tx Retries:        134/0
24 Mbps:
 Rx Packets:        235/0           Tx Packets:      20993/0
                                    Tx Retries:       5048/0
36 Mbps:
 Rx Packets:          0/0           Tx Packets:        781/0
                                    Tx Retries:        227/0
54 Mbps:
 Rx Packets:        133/0           Tx Packets:       9347/0
                                    Tx Retries:       1792/0

[SU]:
M0:
 Rx Packets:          7/0           Tx Packets:          0/0
                                    Tx Retries:          6/0
M1:
 Rx Packets:       1615/0           Tx Packets:      35035/0
                                    Tx Retries:       3751/0
M2:
 Rx Packets:      15277/0           Tx Packets:     133738/0
                                    Tx Retries:      22654/0
M3:
 Rx Packets:      10232/0           Tx Packets:       1580/0
                                    Tx Retries:      21271/0
M4:
 Rx Packets:     218143/0           Tx Packets:     190408/0
                                    Tx Retries:      36444/0
M5:
 Rx Packets:     399283/0           Tx Packets:     542491/0
                                    Tx Retries:     164048/0
M6:
 Rx Packets:    3136519/0           Tx Packets:     821537/0
```

```
                                    Tx Retries:     329003/0
M7:
 Rx Packets:    1171128/0           Tx Packets:     303414/0
                                    Tx Retries:     154014/0



Beacons missed: 0-30s 31-60s 61-90s 90s+
                   2      0      0    0
```

- **#show wgb dot11 uplink latency**

```
AP4C42.1E51.A050#show wgb dot11 uplink latency
Latency Group Total Packets Total Latency Excellent(0-8) Very Good(8-16) Good (16-32
ms) Medium (32-64ms) Poor (64-256 ms) Very Poor (256+ ms)
        AC_BK             0             0              0               0
 0                0                0                 0
        AC_BE          1840       4243793           1809              10
14                7                0                 0
        AC_VI             0             0              0               0
 0                0                0                 0
        AC_VO            24         54134             24               0
 0                0                0                 0
```

- **#show wgb dot11 uplink**

```
AP4C42.1E51.A050#show wgb dot11 uplink

HE Rates: 1SS:M0-11 2SS:M0-11
Additional info for client 8C:84:42:92:FF:CF
RSSI: -24
PS  : Legacy (Awake)
Tx Rate: 278730 Kbps
Rx Rate: 410220 Kbps
VHT_TXMAP: 65530
CCX Ver: 5
Rx Key-Index Errs: 0
             mac     intf TxData TxUC TxBytes TxFail TxDcrd TxCumRetries MultiRetries
 MaxRetriesFail RxData RxBytes RxErr               TxRt(Mbps)               RxRt(Mbps)
   LER PER stats_ago
8C:84:42:92:FF:CF wbridge1   1341 1341  184032      0      0          543           96
             0    317   33523     0 HE-40,2SS,MCS6,GI0.8 (309) HE-40,2SS,MCS9,GI0.8
(458) 27272   0  1.370000
Per TID packet statistics for client 8C:84:42:92:FF:CF
Priority Rx Pkts Tx Pkts Rx(last 5 s) Tx (last 5 s)
       0      35    1314            0             8
       1       0       0            0             0
       2       0       0            0             0
       3       0       0            0             0
       4       0       0            0             0
       5       0       0            0             0
       6     182      24            1             0
       7       3       3            0             0
Rate Statistics:
Rate-Index    Rx-Pkts    Tx-Pkts Tx-Retries
         0         99          3          0
         4          1          1          9
         5         21         39         35
         6         31        185         64
         7         26        124         68
         8         28        293         82
         9         77        401        151
        10         32        140         97
        11          2        156         37
```

Event Logging

For WGB field deployment, event logging will collect useful information (such as WGB state change and packets rx/tx) to analyze and provide log history to present context of problem, especially in roaming cases.

You can configure WGB trace filter for all management packet types, including probe, auth, assoc, eap, dhcp, icmp, and arp. To enable or disable WGB trace, use the following command:

#**config wgb event trace** {**enable**|**disable**}

Four kinds of event types are supported:

- **Basic event**: covers most WGB basic level info message
- **Detail event**: covers basic event and additional debug level message
- **Trace event**: recording wgb trace event if enabled
- **All event**: bundle trace event and detail event

The log format is [*timestamp*] *module*:*level* <*event log string*>.

When abnormal situations happen, the eventlog messages can be dumped manually to memory by using the following show command which also displays WGB logging:

#**show wgb event** [**basic**|**detail**|**trace**|**all**]

The following example shows the output of **show wgb event all**:

```
APC0F8.7FE5.F3C0#show wgb event all
[*08/16/2023 08:18:25.167578] UP_EVT:4 R1 IFC:58:9A:17:B3:E7] parent_rssi: -42 threshold:
-70
[*08/16/2023 08:18:25.329223] UP_EVT:4 R1 State CONNECTED to SCAN_START
[*08/16/2023 08:18:25.329539] UP_EVT:4 R1 State SCAN_START to STOPPED
[*08/16/2023 08:18:25.330002] UP_DRV:1 R1 WGB UPLINK mode stopped
[*08/16/2023 08:18:25.629405] UP_DRV:1 R1 Delete client FC:58:9A:17:B3:E7
[*08/16/2023 08:18:25.736718] UP_CFG:8 R1 configured for standard: 7
[*08/16/2023 08:18:25.989936] UP_CFG:4 R1 band 1 current power level: 1
[*08/16/2023 08:18:25.996692] UP_CFG:4 R1 band 1 set tx power level: 1
[*08/16/2023 08:18:26.003904] UP_DRV:1 R1 WGB uplink mode started
[*08/16/2023 08:18:26.872086] UP_EVT:4 Reset aux scan
[*08/16/2023 08:18:26.872096] UP_EVT:4 Pause aux scan on slot 2
[*08/16/2023 08:18:26.872100] SC_MST:4 R2 reset uplink scan state to idle
[*08/16/2023 08:18:26.872104] UP_EVT:4 Aux bring down vap - scan
[*08/16/2023 08:18:26.872123] UP_EVT:4 Aux bring up vap - serv
[*08/16/2023 08:18:26.872514] UP_EVT:4 R1 State STOPPED to SCAN_START
[*08/16/2023 08:18:26.8727091 SC_MST:4 R1 Uplink Scan Started.
[*08/16/2023 08:18:26.884054] UP_EVT:8 R1 CH event 149
```

Note It might take a long time to display the **show wgb event** command output in console. Using *ctrl+c* to interrupt the printing will not affect log dump to memory.

The following clear command erases WGB events in memory:

#**clear wgb event** [**basic**|**detail**|**trace**|**all**]

To save all event logs to WGB flash, use the following command:

#**copy event-logging flash**

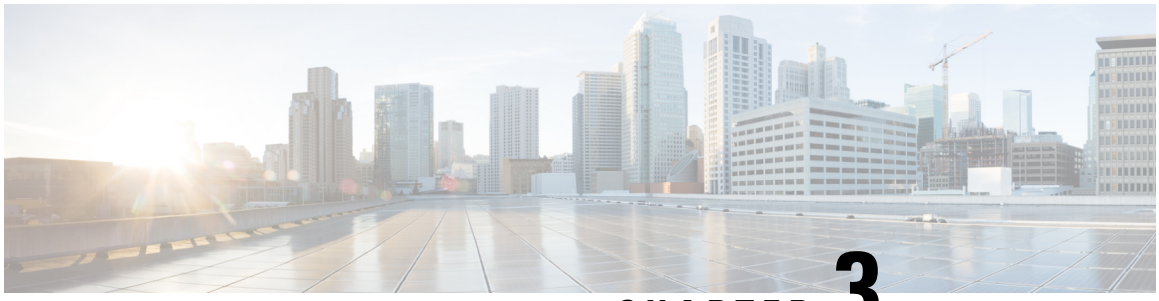
The package file consists of four separate log files for different log levels.

You can also save event log to a remote server by using the following command:

#**copy event-logging upload** <**tftp**|**sftp**|**scp**>://A.B.C.D[/*dir*][/*filename.tar.gz*]

The following example saves event log to a TFTP server:

```
APC0F8.7FE5.F3C0#copy event-logging upload
tftp://192.168.100.100/tftpuser/evtlog-2023-05-31_11:45:49.tar.gz
Starting upload of WGB config
tftp://192.168.100.100/tftpuser/evtlog-2023-05-31_11:45:49.tar.gz ...
It may take a few seconds. If longer, please cancel command, check network and try again.
########################################################################## 100.0%
Config upload completed.
```



CHAPTER 3

Control and Provisioning of Wireless Access Points

Overview

CAPWAP is an IEEE standard protocol that enables a wireless LAN controller to manage multiple APs and Wireless LAN Controllers (WLCs) to exchange control and data plane information over a secure communication tunnel.

CAPWAP only operates in Layer 3 and requires IP addresses to be present on both the APs and WLC. CAPWAP establishes tunnels on the UDP ports 5246 and 5247 for IPv4 and IPv6 respectively. It adds extra security with Datagram Transport Layer Security (DTLS) encryption.

DTLS serves as a protocol ensuring security between the AP and WLC, facilitating encrypted communication to prevent eavesdropping or tampering in potential man-in-the-middle attacks.

By default, DTLS secures the control channel for CAPWAP, encrypting all CAPWAP management and control traffic between the AP and WLC.

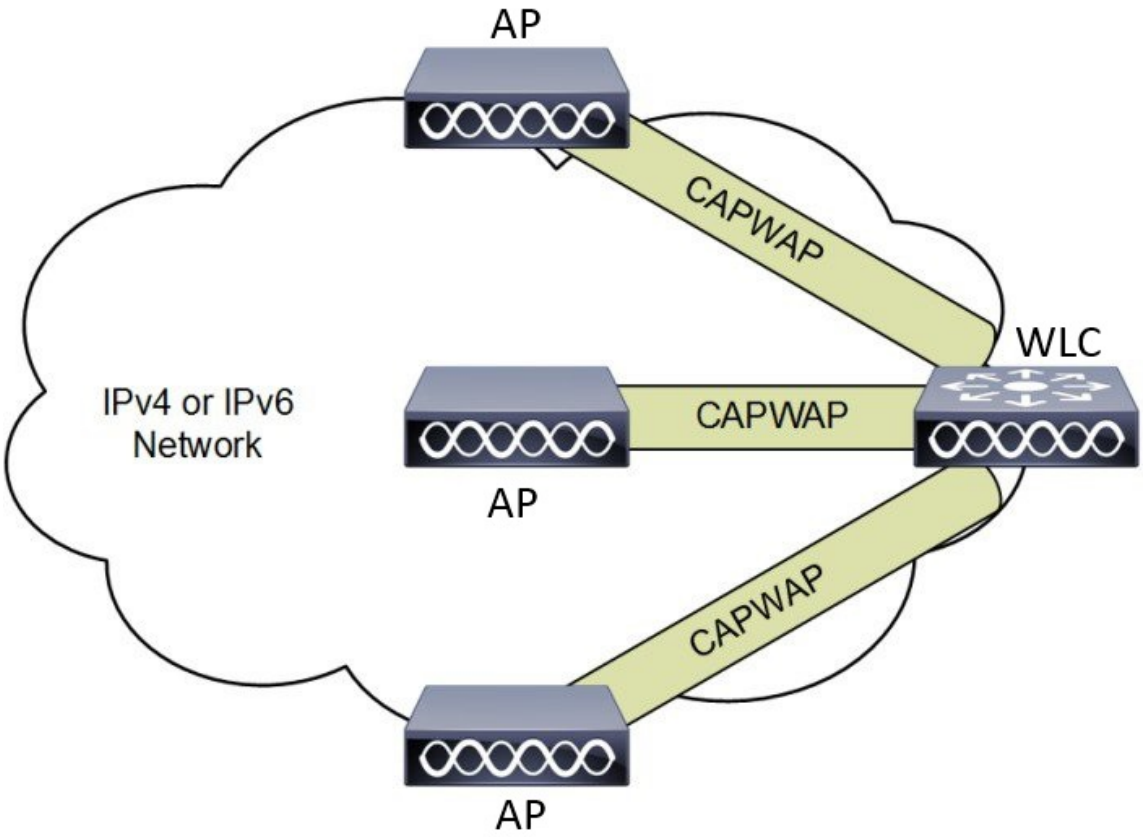
The data channel remains disabled by default, leaving client data moving between an AP and WLC unencrypted. Enabling CAPWAP data encryption is discretionary, and it necessitates the installation of a DTLS license on the WLC before activation on an AP.

If an AP does not support DTLS data encryption, DTLS is enabled only for the control plane, and a DTLS session for the data plane is not established

If an AP supports Data DTLS, it enables data DTLS after receiving the new configuration from the controller. The AP performs a DTLS handshake on port 5247 and after successfully establishing the DTLS session. All the data traffic (from the AP to the controller and the controller to the AP) is encrypted.

CAPWAP allows administrators to manage the entire wireless network from a central location. The IW9165E use the Internet Engineering Task Force (IETF) standard CAPWAP to communicate between the controller and other AP on the network.

Figure 5: CAPWAP APs connected to a WLC



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.

Understanding CAPWAP Connectivity On AP

When CAPWAP is enabled, the first function is to initiate a discovery phase. Wireless APs search for a controller by sending discovery request messages. Upon receiving a discovery request, the controller replies with a discovery response. At this point, the two devices establish a secure connection using the Datagram Transport Layer Security (DTLS) protocol to exchange CAPWAP control and data messages.

By using CAPWAP discovery mechanisms, then AP sends a CAPWAP join request to the controller. The controller sends a CAPWAP join response to the AP that allows the AP to join the controller. When the AP joins the wireless controller, the wireless controller manages its configuration, firmware, control transactions, and data transactions.

The CAPWAP has two channels, namely control and data. The AP uses the control channel to send configuration messages, download images and client keys, or receive the context. The control channel has a single window in the current implementation. The APs must acknowledge every message sent from the controller in a single window. The APs does not transmit the next control packet until it acknowledges the earlier one.

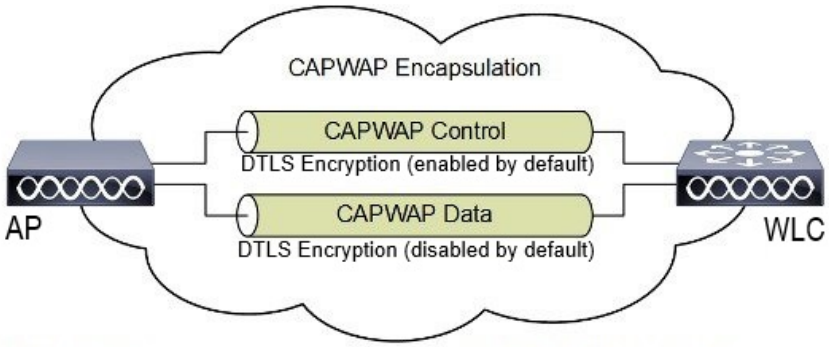
CAPWAP data channel facilitates the encapsulation and tunneling of user data traffic between APs and WLCs. This provide centralized management of user data flow, enabling the WLC to enforce policies, apply Quality of Service (QoS), and ensure security measures consistently across the wireless network. The user data is encapsulated within CAPWAP frames, allowing it to be transported between the APs and WLCs.

According to IETF, CAPWAP supports two modes of operation:

- **Split Media Access Control (MAC)**: A key component of CAPWAP is the concept of a split MAC, where part of the 802.11 protocol operation is managed by the CAPWAP AP, while the remaining parts are managed by the WLC.

  In split MAC mode, the CAPWAP protocol encapsulates all Layer 2 wireless data and management frames, which are then exchanged between the WLC and AP.

  Figure 6: Split MAC Architecture



Access Point MAC Functions:

- 802.11: Beacons, Probe Responses.
- 802.11 Control: Packet Acknowledgements and Transmission.
- 802.11e: Frame Queuing and Packet Prioritization.
- 802.11i: MAC Layer Data Encryption and Decryption.

Controller MAC Functions:

- 802.11 MAC: Management: Association Requests and Actions.
- 802.11e: Resource Reservation.
- 802.11i: Authentication and Key Management.

- **Local MAC**: Local MAC mode enables data frames to be locally bridged or tunneled as Ethernet frames.

Local MAC where all the wireless MAC functions are performed at the AP. The complete 802.11 MAC functions, including management and control frame processing, are resident on the APs.

In either mode, the AP processes Layer 2 wireless management frames locally and then forwards them to the controller.

Reset Button Settings

The following reset actions are performed in the IW9165E when the LED turns to blinking red (after the boot loader gets the reset signal). Ensure you to press the device's reset button before the device is powering on.

- Keep the button pressed for < 20 seconds for full reset.
- Keep the button pressed for > 20 seconds and < 60 seconds for full factory reset (clear fips flag).

Ethernet Port Usage On CAPWAP Mode

The Catalyst IW9165E supports up to a 3.6 Gbps PHY data rate with two 2x2 multiple input and multiple output (MIMO) and two ethernet ports (2.5G mGig and 1G).

Catalyst IW9165E have below internal port mapping rules:

- Wired0 – One mGig (2.5 Gbps) ethernet ports with 802.3af, 802.3at, 802.3bt PoE support.

Note The wired0 port is used as CAPWAP uplink port in the AP local/Flexconnect mode.

- Wired1 – 1Gig ethernet Lan Port.

Note Starting from 17.14.1 release, RLAN feature is not supported in the wired1 port.

Configuring Indoor Deployment

The IW9165E supports indoor and outdoor deployment for the regulatory domain -B (USA), -E (EU), -A (Canada), -Z (Australia, New Zealand).

By default, AP deployment mode is indoor.

Outdoor and indoor frequencies are the same for the -B domain.

Table 6: Radio 6G power mode support table

AP Deployment Mode	6G Deployment Mode	Low Power Indoor support	Standard Power support
Indoor AP	Outdoor	No	Yes

Note Outdoor mode can be used indoors, but indoor mode cannot be used outdoors because 5150–5350 MHz channels are indoor-only in -E countries.

For more information about Configuring AP Deployment Mode on the Wireless Controller, See Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide.

The command triggers an AP reboot. After AP registers to the wireless controller after rebooting, you need to assign corresponding country code to the AP.

Verifying Indoor Deployment

To verify whether the indoor deployment is enabled or not on the WLC.

Run the **#show ap name <AP_Name> config general | inc Indoor** command.

- When indoor mode is enabled, the show command provides the following output:

```
#show ap name <AP_Name> config general | inc Indoor
    AP Indoor Mode                                  : Enabled
```

- When indoor mode is disabled, the show command provides the following output:

```
#show ap name <AP_Name> config general | inc Indoor
    AP Indoor Mode                                  : Disabled
```

To check the status of the indoor deployement on AP, run the **show controllers Dot11Radio [1|2]** command.

- When indoor mode is enabled, the show command provides the following output:

```
Device#show controllers Dot11Radio [1|2]
…
Radio Info Summary:
=======================
Radio: 5.0GHz
Carrier Set: (-Ei)   ( GB )
Base radio MAC: FC:58:9A:15:B7:C0
Supported Channels:
36 40 44 48 52 56 60 64 100 104 108 112 116 120 124 128 132 136 140
```

Note In the command output, "**-Ei**" indicates the indoor mode is enabled

- When indoor mode is disabled, the show command provides the following output:

```
Device#show controllers Dot11Radio [1|2]
…
Radio Info Summary:
=======================
Radio: 5.0GHz
Carrier Set: (-E)   ( GB )
Base radio MAC: FC:58:9A:15:B7:C0
Supported Channels:
100 104 108 112 116 120 124 128 132 136 140
```

Note In the command output, "**-E**" indicates that indoor mode is disabled.

The CLI output also shows the supported channels.

AP Radio Slot

Starting from Cisco Unified Industrial Wireless Software Release 17.14.1, Cisco Catalyst IW9165E now has one dedicated 2x2 5GHz Wi-Fi radio and Dual Band (XOR) radio serving for 5 GHz and 6 GHz 2x2 radios bands.

The Catalyst IW9165E has the option to switch between 5G and 6G band. To switch between 5G and 6G band use the following CLI command.

```
ap name <ap-name> dot11 dual-band band 6ghz/5ghz
```

Note By default, admin state is disabled.

Slot 2 XOR radio is fixed to 5G.

Table 7: AP Wi-Fi radio architecture modes

Mode	**5 GHz** **Slot 1**	**5/6 GHz** **Slot 2**
5G + 5G	5GHz 2x2:2SS (20/40/80 MHz)	5G 2x2:2SS (20/40/80/160 MHz)
5G + 6G	5GHz 2x2:2SS (20/40/80 MHz)	6G 2x2:2SS (20/40/80/160 MHz)

Supporting Fixed Domains and Country Codes

The ROW regulatory domain simplifies the domain management of the manufacturing process for all the country codes that do not have a specific domain mapped. The fixed domain and country code support for the Catalysts IW9165E access points are described in this section.

Supported Fixed Domains

Domain	**Country Codes**
A	CA (Canada)
B	US (United States of America)

Domain	Country Codes
E	

Domain	Country Codes
	• AT (Austria) • AT (Austria) • BE (Belgium) • BG (Bulgaria) • HR (Croatia) • CY (Cyprus) • CZ (Czech Republic) • DK (Denmark) • EE (Estonia) • FI (Finland) • FR (France) • DE (Germany) • GR (Greece) • HU Hungary) • IS (Iceland) • IE (Ireland) • IT (Italy) • LV (Latvia) • LI (Liechtenstein) • LT (Lithuania) • LU (Luxembourg) • MT (Malta) • NL (Netherlands) • NO (Norway) • PL (Poland) • PT (Portugal) • RO (Romania) • SK (Slovak Republic) • SI (Slovenia) • ES (Spain) • SE (Sweden), and

Domain	Country Codes
	• CH (Switzerland).
F	ID (Indonesia)
Q	JP (Japan)
Z	• AU (Australia) and • NZ (New Zealand).

Catalyst IW9165 Supported Country Codes (ROW)

Domain	Country Codes
ROW	• CL (Chile) • KR (Korea, Republic of) • GB (United Kingdom), and • VN (Vietnam).

You are responsible for ensuring APs approval for use in your country. To verify approval and to identify the regulatory domain associated with a particular country. For more information, see Cisco Product Approval Status.

Configuring Radio Antenna Settings

The Catalyst IW9165E supports four external antennas with RP-SMA (f) connectors. Radio 1 connects to antenna ports 1 and 2. Radio 2 connects to antenna ports 3 and 4.

The IW9165E is compatible with Self Identifiable Antenna (SIA) antennas for the 6G band. Antenna ports 1 and 3 can support SIA antennas. For more information on antennas, see the Cisco Catalyst IW9165E Rugged Access Point and Wireless Client Hardware Installation Guide.

Note A power cycle is mandatory after the first installation of the SIA antenna.

SIA supports antenna IW-ANT-OMV-2567-N and IW-ANT-OMH-2567-N only.

Table 8: Antenna Gain (dBm)

5 GHz Slot 1	5 GHz Slot 2	6 GHz Slot 2
3 4 7 8 10 13 15	3 4 7 8 10 13 15	7

The following sections describe CLI commands to verify the SIA test.

To verify the SIA status on the controller, run **show ap config slots <AP>** command.

```
Device#show ap config slot ap_name
```

```
show ap config slots AP2CF8.9B1C.CE78
Cisco AP Name : AP4C42.1E51.A144
Attributes for Slot 2
SIA Status       : Present(RPTNC)
SIA Product ID   : IW-ANT-OMV-2567-N
```

AFC Support for 6G Standard Power Mode

The Cisco Catalyst IW9165E supports the Automated Frequency Coordination (AFC) 6 GHz Standard Power mode. A standard power AP joins the system. Before enabling standard power, the AP must get the available frequencies and the power in each frequency range from the AFC system.

The AFC system computes the available frequencies and maximum allowable power based on the information provided by the regulatory body (FCC for United States). The response is sent back to controller, which may assign a standard power channel to the AP based on the allowed channel list returned by the AFC system.

Standard Power AP coordinate through an AFC service. The AFC accesses information and, along with the AP's geographical location and antenna characteristics, creates a topographical propagation map modeling the AP's interference radius. This map allows you to assign maximum transmission power and coordinate/configure the channel settings to avoid interference.

Table 9: Radio 6 GHz power mode support

AP Deployment Mode	6G Deployment Mode	Low-power Indoor Support	Standard Power Support
Indoor AP	Outdoor	No	Yes

The transmission power is limited to a maximum of 36 dB Effective Isotropic Radiated Power (EIRP), and APs must be coordinated through an AFC service. These APs are allowed to operate in the UNII-5 (5.925-6.425 GHz) and UNII-7 (6.525-7.125 GHz) in the -B (U.S) domain.

Table 10: 6 GHz Target Power

Conducter Per Path Power		Antenna Gain	Tx x Rx Chains	Max EIRP	Max EIRP (SP/AFC)
20-80Mhz	160Mhz				
17 dBm	17 dBm	7 dBi*	2x2	27 dBm*	36 dBm

Verifying AFC Status on AP

To verify the AFC request and response data on AP, run the **show rrm afc** command.

```
Device#show rrm afc
Location Type: 1
Deployment Type: 2
Height: 129
Uncertainty: 5
Height Type: 0
Request Status: 5
Request Status Timestamp: 2023-08-31T06:20:17Z
Request Id Sent: 5546388983266789933
```

```
Ellipse 1: longitude: -121.935066 latitude: 37.512830 major axis: 43 minor axis:
 9 orientation: 36.818100
AFC Response Request ID: 5546388983266789933
AFC Response Ruleset ID: US_47_CFR_PART_15_SUBPART_E
```

To verify the current operating power mode, run the **show controllers dot11Radio 2 | i Radio** command.

```
Device#show controllers dot11Radio 2 | i Radio
Dot11Radio2     Link encap:Ethernet  HWaddr 24:16:1B:F8:06:C0
Radio Info Summary:
Radio: 6.0GHz (SP)
```

GNSS Support

Global Navigation Satellite System (GNSS) is supported on IW9165E. The AP tracks GPS information for devices deployed in the outdoor environment and sends the GNSS information to the wireless controller.

Use the following command to display the GNSS information on the AP:

```
ap# show gnss info
```

Use the following commands to display the GPS location of the AP:

```
controller# show ap geolocation summary
```

```
controller# show ap name <Cisco AP> geolocation detail
```

Information About Antenna Disconnection Detection

Having multiple antennas on the transmitter and receiver of an access point (AP) results in better performance and reliability. Multiple antennas improve reception through the selection of the stronger signal or a combination of individual signals at the receiver. Therefore, detection of an impaired antenna or physical breakage of an antenna is critical to the reliability of APs.

The Antenna Disconnection Detection feature is based on the signal strength delta across the antennas on the receiver. If the delta is more than the defined limit for a specific duration, the antenna is considered to have issues.

For every detection time period that you configure, the AP sends an Inter-Access Point Protocol (IAPP) message that carries the antenna condition. This message is sent only once when the issue is detected and is displayed in the controller trap messages, SNMP traps, and controller debug logs.

Configuration Workflow

1. Configure APs.
2. Configure an AP profile.
3. Enable the feature in AP profile.
4. Configure feature parameters.
5. Verify the configuration.

For more information about Configuring Antenna Disconnection Detection on the Wireless Controller, See Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide.

Verifying Antenna Disconnection Detection

To verify the Antenna Disconnection Detection feature configuration on an AP, use the following command:

```
9800-Conroller#sh ap name AP4C42.1E51.A144 config general

Cisco AP Name   : AP4C42.1E51.A144
================================================

Cisco AP Identifier                              : 8c84.4292.f840
Country Code                                   : Multiple Countries : US,CN,GB,HK,DE,IN,CZ,NZ
Regulatory Domain Allowed by Country             : 802.11bg:-ACE^   802.11a:-ABCDEHNSZ^
802.11 6GHz:-BEZ^
Radio Authority IDs                              : None
AP Country Code                                  : CZ  - Czech Republic
AP Regulatory Domain
  802.11bg                                       : -E
  802.11a                                        : -E
MAC Address                                      : 8c84.4292.f840
IP Address Configuration                         : DHCP
IP Address                                       : 9.9.33.3
IP Netmask                                       : 255.255.255.0
Gateway IP Address                               : 9.9.33.1
Fallback IP Address Being Used                   :
Domain                                           :
Name Server                                      :
CAPWAP Path MTU                                  : 1485
Capwap Active Window Size                        : 1
```

To verify the Antenna Disconnection Detection feature configuration on an AP profile, use the following command:

```
9800-Controller#show ap profile name ap-profile detailed

AP Profile Name: ap-profile
.
.
.
AP broken antenna detection:
  Status                                                 : ENABLED
  RSSI threshold                                         : 40
  Weak RSSI                                              : -80
  Detection Time                                         : 120
```

Troubleshooting

The document provides use cases to understand the reason for the Control and Provisioning of Wireless Access Points (CAPWAP)/Lightweight Access Point Protocol (LWAPP) tunnel break between Access Points (APs) and the Wireless Controller. For more information, see Troubleshoot Access Point Disassociation from Controller

Note There could be some changes in the software or hardware that can cause commands to stop working, the syntax to change, or GUIs and CLIs to look different from one release to another.

Feedback Request

Your input helps. A key aspect to improving these support documents is customer feedback. Note that these documents are owned and maintained by multiple teams within Cisco. If you find an issue specific to the document (unclear, confusing, information missing, etc):

- Provide feedback using the Feedback button located at the right panel of the corresponding article. The document owner will be notified, and will either update the article, or flag it for removal.
- Include information regarding the section, area, or issue you had with the document and what could be improved. Provide as much detail as possible.

Disclaimer and Caution

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.