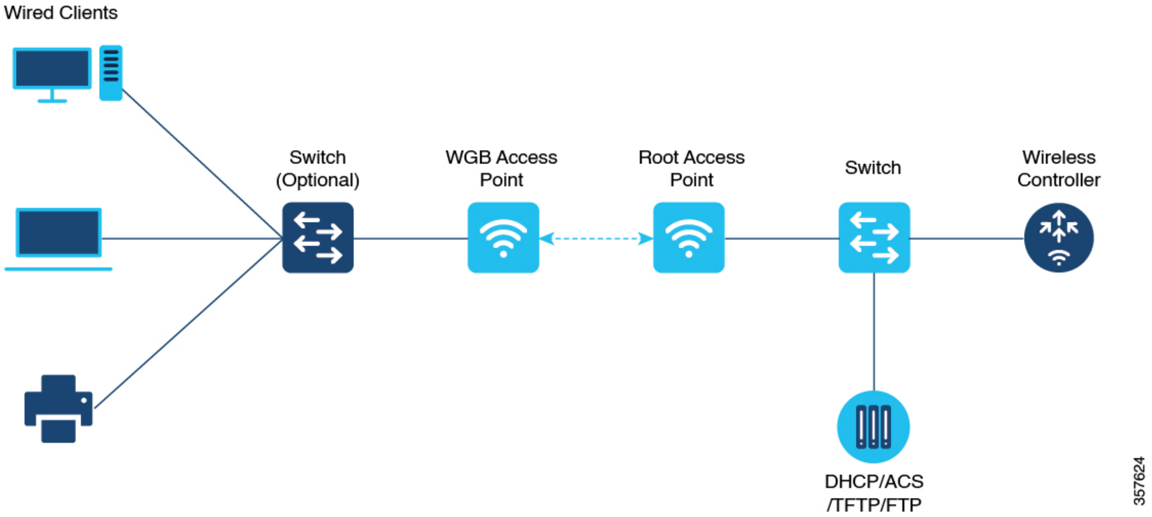
Workgroup Bridges

Cisco Workgroup Bridges

A workgroup bridge (WGB) is an Access Point (AP) mode to provide wireless connectivity to wired clients that are connected to the Ethernet port of the WGB AP. A WGB connects a wired network over a single wireless segment by learning the MAC addresses of its wired clients on the Ethernet interface and reporting them to the WLC through infrastructure AP using Internet Access Point Protocol (IAPP) messaging. The WGB establishes a single wireless connection to the root AP, which in turn, treats the WGB as a wireless client.

Figure 1: Example of a WGB



Starting from Cisco IOS XE Cupertino 17.8.1, WGB is supported on the following Cisco Catalyst 9100 Series Access Points.

- Cisco Catalyst 9105
- Cisco Catalyst 9115
- Cisco Catalyst 9120

The following features are supported for use with a WGB:

Table 1: WGB Feature Matrix

Feature	Cisco Wave 1 APs	Cisco Wave 2 and 11AX APs
802.11r	Supported	Supported
QOS	Supported	Supported
UWGB mode	Supported	Supported on Wave 2 APs Not supported on 11AX APs
IGMP Snooping or Multicast	Supported	Supported
802.11w	Supported	Supported
PI support (without SNMP)	Supported	Not supported
IPv6	Supported	Supported
VLAN	Supported	Supported
802.11i (WPAv2)	Supported	Supported
Broadcast tagging/replicate	Supported	Supported
Unified VLAN client	Implicitly supported (No CLI required)	Supported
WGB client	Supported	Supported
802.1x – PEAP, EAP-FAST, EAP-TLS	Supported	Supported
NTP	Supported	Supported
Wired client support on all LAN ports	Supported in Wired-0 and Wired-1 interfaces	Supported in all Wired-0, 1 and LAN ports 1, 2, and 3

The following table shows the supported and unsupported authentication and switching modes for Cisco APs when connecting to a WGB.

Note Workgroup Bridge mode is supported on the WiFi6 Pluggable Module from Cisco IOS XE Bengaluru 17.6.1.

Table 2: Supported Access Points and Requirements

Access Points	Requirements
Cisco Aironet 2700, 3700, and 1572 Series	Requires autonomous image.
Cisco Aironet 2800, 3800, 4800, 1562, and Cisco Catalyst 9105, 9115, IW6300 and ESW6300 Series	CAPWAP image starting from Cisco AireOS 8.8 release.

Table 3: WGB Support on APs

WGB WLAN Support	Cisco Wave 2 APs	Cisco Catalyst 9100 Series APs
Central Authentication	Supported	Supported
Central Switching	Supported	Supported
Local Authentication	Not Supported	Not Supported
Local Switching	Supported	Supported

- MAC filtering is not supported for wired clients.
- Idle timeout is not supported for both WGB and wired clients.
- Session timeout is not applicable for wired clients.
- Web authentication is not supported.
- WGB supports only up to 20 clients.
- If you want to use a chain of certificates, copy all the CA certificates to a file and install it under a trust point on the WGB, else server certificate validation may fail.
- Wired clients connected to the WGB are not authenticated for security. Instead, the WGB is authenticated against the access point to which it associates. Therefore, we recommend that you physically secure the wired side of the WGB.
- Wired clients connected to a WGB inherit the WGB's QoS and AAA override attributes.
- To enable the WGB to communicate with the root AP, create a WLAN and make sure that Aironet IE is enabled under the Advanced settings.

Configuring Workgroup Bridge on a WLAN

Follow the procedure given below to configure a WGB on a WLAN:

For WGB to join a wireless network there are specific settings on the WLAN and on the related policy profile.

Note For the configuration given below, it is assumed that the WLAN security is already configured.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wlan** *profile-name* **Example:** `Device(config)# wlan WGB_Test`	Enters WLAN configuration submode. The *profile-name* is the profile name of the configured WLAN.
Step 3	**ccx aironet-iesupport** **Example:** `Device(config-wlan)# ccx aironet-iesupport`	Configures the Cisco Client Extensions option and sets the support of Aironet IE on the WLAN.
Step 4	**exit** **Example:** `Device(config-wlan)# exit`	Exits the WLAN configuration submode.
Step 5	**wireless profile policy** *profile-policy* **Example:** `Device(config)# wireless profile policy test-wgb`	Configures WLAN policy profile and enters the wireless policy configuration mode.
Step 6	**description** *description* **Example:** `Device(config-wireless-policy)# description "test-wgb"`	Adds a description for the policy profile.
Step 7	**vlan** *vlan-no* **Example:** `Device(config-wireless-policy)# vlan 48`	Assigns the profile policy to the VLAN.
Step 8	**wgb vlan** **Example:** `Device(config-wireless-policy)# wgb vlan`	Configures WGB VLAN client support.
Step 9	**wgb broadcast-tagging** **Example:** `Device(config-wireless-policy)# wgb broadcast-tagging`	Configures WGB broadcast tagging on a WLAN.
Step 10	**no shutdown** **Example:** `Device(config-wireless-policy)# no shutdown`	Restarts the policy profile.

	Command or Action	Purpose
Step 11	**exit** **Example:** `Device(config-wireless-policy)# exit`	Exits the wireless policy configuration mode.
Step 12	**wireless tag policy** *policy-tag* **Example:** `Device(config)# wireless tag policy WGB_Policy`	Configures policy tag and enters policy tag configuration mode.
Step 13	**wlan** *profile-name* **policy** *profile-policy* **Example:** `Device(config-policy-tag)# wlan WGB_Test policy test-wgb`	Maps a policy profile to a WLAN profile.

Step 14	**end**
	Example:
	`Device(config-policy-tag)# end`
		Exits policy tag configuration mode, and returns to privileged EXEC mode.

Verifying the Status of a Workgroup Bridge on the Controller

Use the following commands to verify the status of a WGB.

To display the wireless-specific configuration of active clients, use the following command:

```
Device# show wireless client summary
```

To display the WGBs on your network, use the following command:

```
Device# show wireless wgb summary
```

To display the details of wired clients that are connected to a particular WGB, use the following command:

```
Device# show wireless wgb mac-address 00:0d:ed:dd:25:82 detail
```

Configuring Access Points as Workgroup Bridge

Turning Cisco Aironet 2700/3700/1572 Series AP into Autonomous Mode

Before you begin

Download the autonomous image for the specific access point from software.cisco.com and place it on a TFTP server.

Procedure

	Command or Action	Purpose
Step 1	**debug capwap console cli** **Example:** `Device# debug capwap console cli`	Enables the console CLI.
Step 2	**archive download-sw force-reload overwrite tftp:***ipaddress filepath filename* **Example:** `Device(config)# archive download-sw force-reload overwrite tftp://10.10.10.1/tftp/c1800.tar`	Downloads the autonomous image to the access point.

Configuring Cisco Wave 2 APs or 11AX APs in Workgroup Bridge or CAPWAP AP Mode (CLI)

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device# enable`	Enters in to the privileged mode of the AP.
Step 2	**ap-type workgroup-bridge** **Example:** `Device# ap-type workgroup-bridge`	Moves the AP in to the Workgroup Bridge mode.

Step 3	**configure ap address ipv4 dhcp** or **configure ap address ipv4 static***ip-address netmask gateway-ipaddress*
	Example:
	DHCP IP Address
	`Device# configure ap address ipv4 dhcp`
	Static IP Address
	`Device# configure ap address ipv4 static 10.10.10.2 255.255.255.234 192.168.4.1`
		Configures DHCP or Static IP address.

Step 4	**configure ap management add username** *username* **password** *password* **secret** *secret* **Example:** `Device# configure ap management add username xyz-user password ****** secret cisco`	Configures an username for the AP management.

	Command or Action	Purpose
Step 5	**configure ap hostname***host-name* **Example:** `Device# configure ap hostname xyz-host`	Configures the AP hostname.

Configure an SSID Profile for Cisco Wave 2 and 11AX APs (CLI)

This procedure is an AP procedure. The CLIs listed in the procedure given below work only on the AP console and not on the controller.

Procedure

	Command or Action	Purpose

Step 1	**configure ssid-profile** *ssid-profile-name* **ssid** *radio-serv-name* **authentication** {**open** | **psk** *preshared-key* **key-management** {**dot11r** | **wpa2** | **dot11w** |{**optional** | **required** }}| **eap profile** *eap-profile-name* **key-management** {**dot11r** | **wpa2** | **dot11w**|{**optional** | **required**}}
	Example:
	SSID profile with open authentication.
	`Device# configure ssid-profile test WRT s1 authentication open`
	SSID profile with PSK authentication.
	`Device# configure ssid-profile test WRT s1 authentication psk 1234 key-management dot11r optional`
	SSID profile with EAP authentication.
	`Device# configure ssid-profile test WRT s1 authentication eap profile test2 key-management dot11r optional`
		Choose an authentication protocol (Open, PSK, or EAP) for the SSID profile.

Step 2	**configure dot11radio** *radio-interface* **mode wgb ssid-profile** *profle-name* **Example:** `Device# configure dot11radio r1 mode wgb ssid-profile doc-test`	Attaches an SSID profile to a radio interface.
Step 3	**configure ssid-profile** *profle-name* **delete** **Example:** `Device# configure ssid-profile doc-test delete`	(Optional) Deletes an SSID profile.
Step 4	**show wgb ssid** **Example:**	(Optional) Displays summary of configured and connected SSIDs.

	Command or Action	Purpose
	`Device# show wgb ssid`	
Step 5	**show wgb packet statistics** **Example:** `Device# show wgb packet statistics`	(Optional) Displays management, control, and data packet statistics.

Configuring a Dot1X Credential (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure dot1x credential** *profile-name* **username** *name* **password** *password* **Example:** `Device# configure dot1x credential test1 username XYZ password *****`	Configures a dot1x credential.
Step 2	**configure dot1x credential** *profile-name* **delete** **Example:** `Device# configure dot1x credential test1 delete`	Removes a dot1x profile.

Step 3	**clear wgb client**{**all** | **single** *mac-addr* }
	Example:
	`Device# clear wgb client single xxxx.xxxx.xxxx.xxxx`
		Deauthenticates a WGB client.

Configuring an EAP Profile (CLI)

Procedure

	Command or Action	Purpose

Step 1	**configure eap-profile** *profile-name* **method** {**fast** | **leap** | **peap** | **tls**}
	Example:
	`Device# configure eap-profile test-eap method fast`
		Configures an EAP profile.

Step 2	**configure eap-profile** *profile-name* **trustpoint default** or **configure eap-profile** *profile-name* **trustpoint name** *trustpoint-name* **Example:** EAP Profile to Trustpoint with MIC Certificate.	Configures an EAP profile with a trustpoint.

	Command or Action	Purpose

	`Device# configure eap-profile test-eap trustpoint default`
	EAP Profile to Trustpoint with CA Certificate.
	`Device# configure eap-profile test-eap trustpoint cisco`	

Step 3	**configure eap-profile** *profile-name* **trustpoint** {**default** | **name** *trustpoint-name*}
	Example:
	`Device# configure eap-profile test-eap trustpoint default`
		Attaches the CA trustpoint.
		Note With the default profile, WGB uses the internal MIC certificate for authentication.

Step 4	**configure eap-profile** *profile-name* **dot1x-credential** *profile-name* **Example:** `Device# configure eap-profile test-eap dot1x-credential test-profile`	Configures the 802.1X credential profile.
Step 5	**configure eap-profile** *profile-name* **delete** **Example:** `Device# configure eap-profile test-eap delete`	(Optional) Deletes an EAP profile.
Step 6	**show wgb eap dot1x credential profile** **Example:** `Device# show wgb eap dot1x credential profile`	(Optional) Displays the WGB EAP dot1x profile summary.
Step 7	**show wgb eap profile** **Example:** `Device# show wgb eap profile`	(Optional) Displays the EAP profile summary.
Step 8	**show wgb eap profile all** **Example:** `Device# show wgb eap profile all`	(Optional) Displays the EAP and dot1x profiles.

Configuring Manual-Enrollment of a Trustpoint for Workgroup Bridge (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure crypto pki trustpoint** *ca-server-name* **enrollment terminal** **Example:**	Configures a trustpoint in WGB.

	Command or Action	Purpose

	`Device# configure crypto pki trustpoint`
	`ca-server-US enrollment terminal`	

Step 2	**configure crypto pki trustpoint** *ca-server-name* **authenticate**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US authenticate`
		Authenticates a trustpoint manually.
		Enter the base 64 encoded CA certificate and end the certificate by entering **quit** in a new line.

Step 3	**configure crypto pki trustpoint** *ca-server-name* **key-size** *key-length*
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-Us key-size 60`
		Configures a private key size.

Step 4	**configure crypto pki trustpoint** *ca-server-name* **subject-name** *name* [*2ltr-country-code* |*state-name* |*locality* |*org-name* |*org-unit* |*email*]
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US subject-name test US CA abc cisco AP test@cisco.com`
		Configures the subject name.

Step 5	**configure crypto pki trustpoint** *ca-server-name* **enrol**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US enroll`
		Generates a private key and Certificate Signing Request (CSR).
		Afterwards, create the digitally signed certificate using the CSR output in the CA server.

Step 6	**configure crypto pki trustpoint** *ca-server-name* **import certificate**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US import certificate`
		Import the signed certificate in WGB.
		Enter the base 64 encoded CA certificate and end the certificate by using **quit** command in a new line.

Step 7	**configure crypto pki trustpoint** *ca-server-name* **delete**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US delete`
		(Optional) Delete a trustpoint.

	Command or Action	Purpose
Step 8	**show crypto pki trustpoint** **Example:** `Device# show crypto pki trustpoint`	(Optional) Displays the trustpoint summary.
Step 9	**show crypto pki trustpoint** *trustpoint-name* **certificate** **Example:** `Device# show crypto pki trustpoint ca-server-US certificate`	(Optional) Displays the content of the certificates that are created for a trustpoint.

Configuring Auto-Enrollment of a Trustpoint for Workgroup Bridge (CLI)

Procedure

	Command or Action	Purpose

Step 1	**configure crypto pki trustpoint** *ca-server-name* **enrollment url** *ca-server-url*
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US enrollment url https://cisco/certsrv`
		Enrolls a trustpoint in WGB using the server URL.

Step 2	**configure crypto pki trustpoint** *ca-server-name* **authenticate**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US authenticate`
		Authenticates a trustpoint by fetching the CA certificate from CA server automatically.

Step 3	**configure crypto pki trustpoint** *ca-server-name* **key-size** *key-length*
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-Us key-size 60`
		Configures a private key size.

Step 4	**configure crypto pki trustpoint** *ca-server-name* **subject-name** *name* [*2ltr-country-code* |*state-name* |*locality* |*org-name* |*org-unit* |*email*]
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US subject-name test US CA abc cisco AP test@cisco.com`
		Configures the subject name.

	Command or Action	Purpose

Step 5	**configure crypto pki trustpoint** *ca-server-name* **enrol l**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US enroll`
		Enrolls the trustpoint.
		Request the digitally signed certificate from the CA server.

Step 6	**configure crypto pki trustpoint** *ca-server-name* **auto-enroll enable** *renew-percentage*
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US auto-enroll enable  10`
		Enables auto-enroll of the trustpoint.
		You can disable auto-enrolling by using the **disable** option in the command.

Step 7	**configure crypto pki trustpoint***trustpoint-name* **delete**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US delete`
		(Optional) Deletes a trustpoint.

Step 8	**show crypto pki trustpoint** **Example:** `Device# show crypto pki trustpoint`	(Optional) Displays the trustpoint summary.
Step 9	**show crypto pki trustpoint***trustpoint-name* **certificate** **Example:** `Device# show crypto pki trustpoint ca-server-US certificate`	(Optional) Displays the content of the certificates that are created for a trustpoint.
Step 10	**show crypto pki timers** **Example:** `Device# show crypto pki timers`	(Optional) Displays the PKI timer information.

Configuring Manual Certificate Enrolment Using TFTP Server (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure crypto pki trustpoint** *ca-server-name* **enrollment tftp** *addr/file-name* **Example:**	Specifies the enrolment method to retrieve the CA certificate and client certificate for a trustpoint in WGB.

	Command or Action	Purpose

	`Device# configure crypto pki trustpoint`
	`ca-server-US enrollment`
	`tftp://10.8.0.6/all_cert.txt`	

Step 2	**configure crypto pki trustpoint** *ca-server-name* **authenticate**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US 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 crypto pki trustpoint** *ca-server-name* **key-size** *key-length*
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-Us key-size 60`
		Configures a private key size.

Step 4	**configure crypto pki trustpoint** *ca-server-name* **subject-name** *name* [*2ltr-country-code* |*state-name* |*locality* |*org-name* |*org-unit* |*email*]
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US subject-name test US CA abc`
	` cisco AP test@cisco.com`
		Configures the subject name.

Step 5	**configure crypto pki trustpoint** *ca-server-name* **enrol**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US enroll`
		Generate a private key and Certificate Signing Request (CSR) and writes the request out to the TFTP server. The filename to be written is appended with the extension “.req”.

Step 6	**configure crypto pki trustpoint** *ca-server-name* **import certificate**
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US import certificate`
		Import the signed certificate in WGB using TFTP at the console terminal, which retrieves the granted certificate.
		The WGB will attempt to retrieve the granted certificate using TFTP using the same filename and the file name append with “.crt” extension.

Step 7	**show crypto pki trustpoint** **Example:** `Device# show crypto pki trustpoint`	(Optional) Displays the trustpoint summary.
Step 8	**show crypto pki trustpoint** *trustpoint-name* **certificate** **Example:**	(Optional) Displays the content of the certificates that are created for a trustpoint.

	Command or Action	Purpose
	`Device# show crypto pki trustpoint ca-server-US certificate`	

Importing the PKCS12 Format Certificates from the TFTP Server (CLI)

Procedure

	Command or Action	Purpose

Step 1	**configure crypto pki trustpoint** *ca-server-name* **import pkcs12 tftp** *addr/file-name* **password** *pwd*
	Example:
	`Device# configure crypto pki trustpoint`
	`ca-server-US enrollment tftp://10.8.0.6/all_cert.txt password ******`
		Imports PKCS12 format certificate from the TFTP server.

Step 2	**show crypto pki trustpoint** **Example:** `Device# show crypto pki trustpoint`	(Optional) Displays the trustpoint summary.
Step 3	**show crypto pki trustpoint** *trustpoint-name* **certificate** **Example:** `Device# show crypto pki trustpoint ca-server-US certificate`	(Optional) Displays the content of the certificates that are created for a trustpoint.

Configuring Radio Interface for Workgroup Bridges (CLI)

From the available two radio interfaces, before configuring WGB or UWGB mode on one radio interface, configure the other radio interface to root AP mode.

Procedure

	Command or Action	Purpose
Step 1	**configure dot11radio** *radio-int* **mode root-ap** **Example:** `Device# configure dot11Radio 0/3/0 mode root-ap`	Maps a radio interface as root AP. **Note** When an active SSID or EAP profile is modified, you need to reassociate the profile to the radio interface for the updated profile to be active.

	Command or Action	Purpose
Step 2	**configure dot11Radio <0|1> beacon-period** *beacon-interval* **Example:** `Device# configure dot11radio 1 beacon-period 120`	Configures the periodic beacon interval in milli-seconds. The value range is between 2 and 2000 milli-seconds.
Step 3	**configure dot11Radio** *radio-int* **mode wgb ssid-profile** *ssid-profile-name* **Example:** `Device# configure dot11Radio 0/3/0 mode wgb ssid-profile bgl18`	Maps a radio interface to a WGB SSID profile.
Step 4	**configure dot11Radio** *radio-int* **mode uwgb** *mac-addr* **ssid-profile** *ssid-profile-name* **Example:** `Device# configure dot11Radio 0/3/0 mode uwgb 0042.5AB6.0EF0 ssid-profile bgl18`	Maps a radio interface to a WGB SSID profile.

Step 5	**configure dot11Radio** *radio-int* {**enable**| **disable**}
	Example:
	`Device# configure dot11Radio 0/3/0 mode enable`
		Configures a radio interface.
		Note After configuring the uplink to the SSID profile, we recommend that you disable and enable the radio for the changes to be active.

Step 6	**configure dot11Radio** *radio-int* **antenna** {**a-antenna** | **ab-antenna** | **abc-antenna** | **abcd-antenna**}
	Example:
	`Device# configure dot11Radio 0/3/0 antenna a-antenna`
		Configures a radio antenna.

Step 7	**configure dot11Radio** *radio-int* **encryption mode ciphers aes-ccm** {
	Example:
	`Device# configure dot11Radio radio-int encryption mode ciphers aes-ccm`
		Configures the radio interface.

Step 8	**configure wgb mobile rate** {**basic 6 9 18 24 36 48 54** | **mcs** *mcs-rate*} **Example:** `Device# configure wgb mobile rate basic 6 9 18 24 36 48 54`	Configures the device channel rate.
Step 9	**configure wgb mobile period** *secondsthres-signal* **Example:**	Configure the threshold duration and signal strength to trigger scanning.

	Command or Action	Purpose
	`Device# configure wgb mobile period 30 -50`	

Step 10	**configure wgb mobile station interface dot11Radio** *radio-int* **scan** *channel-number* **add**
	Example:
	`Device# configure wgb mobile station interface`
	`dot11Radio 0/3/0 scan 2 add`
		Configures the static roaming channel.

Step 11	**configure wgb mobile station interface dot11Radio** *radio-int* **scan** *channel-number* **delete**
	Example:
	`Device# configure wgb mobile station interface`
	`dot11Radio 0/3/0 scan 2 delete`
		(Optional) Delete the mobile channel.

Step 12	**configure wgb mobile station interface dot11Radio** *radio-int* **scan disable**
	Example:
	`Device# configure wgb mobile station interface`
	`dot11Radio 0/3/0 scan disable`
		(Optional) Disable the mobile channel.

Step 13	**configure wgb beacon miss-count** *value* **Example:** `Device# configure wgb beacon miss-count 12`	(Optional) Configure the beacon miss-count. By default, this is set to disabled. **Note** When you set the beacon miss-count value to 10 or lower, then the beacon miss-count gets disabled. Set the value to 11 or higher to enable this function.
Step 14	**show wgb wifi** *wifi-interface* **stats** **Example:** `Device# show wgb wifi 0/3/0 stats`	(Optional) Displays the Wi-Fi station statistics.
Step 15	**show controllers dot11Radio** *radio-interface* **antenna** **Example:** `Device# show controllers dot11Radio 0/3/0 antenna`	(Optional) Displays the radio antenna statistics.
Step 16	**show wgb mobile scan channel** **Example:** `Device# show wgb mobile scan channel`	(Optional) Displays the mobile station channels scan configuration.

	Command or Action	Purpose
Step 17	**show configuration** **Example:** `Device# show configuration`	(Optional) Displays the configuration that is stored in the NV memory.
Step 18	**show running-config** **Example:** `Device# show running-config`	(Optional) Displays the running configuration in the device.

Configuring Workgroup Bridge Timeouts (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure wgb association response timeout** *response-millisecs* **Example:** `Device# configure wgb association response timeout 4000`	Configures the WGB association response timeout. The default value is 5000 milliseconds. The valid range is between 300 and 5000 milliseconds.
Step 2	**configure wgb authentication response timeout** *response-millisecs* **Example:** `Device# configure wgb authentication response timeout 4000`	Configures the WGB authentication response timeout. The default value is 5000 milliseconds. The valid range is between 300 and 5000 milliseconds.
Step 3	**configure wgb uclient timeout** *timeout-secs* **Example:** `Device# configure wgb uclient timeout 70`	Configure the Universal WGB client response timeout. The default timeout value is 60 seconds. The valid range is between 1 and 65535 seconds..
Step 4	**configure wgb eap timeout** *timeout-secs* **Example:** `Device# configure wgb eap timeout 20`	Configures the WGB EAP timeout. The default timeout value is 3 seconds. The valid range is between 2 and 60 seconds.
Step 5	**configure wgb channel scan timeout** {**fast**| **medium** | **slow**} **Example:** `Device# configure wgb channel scan timeout slow`	Configures the WGB channel scan timeout.
Step 6	**configure wgb dhcp response timeout** *timeout-secs* **Example:** `Device# configure wgb dhcp response timeout 70`	Configures the WGB DHCP response timeout. The default value is 60 seconds. The valid range is between 1000 and 60000 milliseconds.

	Command or Action	Purpose
Step 7	**show wgb dot11 association** **Example:** `Device# show wgb dot11 association`	Displays the WGB association summary.

Configuring Bridge Forwarding for Workgroup Bridge (CLI)

Before you begin

The Cisco Wave 2 and 11AX APs as Workgroup Bridge recognizes the Ethernet clients only when the traffic has the bridging tag.

We recommend setting the WGB bridge client timeout value to default value of 300 seconds, or less in environment where change is expected, such as:

- Ethernet cable is unplugged and plugged back.
- Endpoint is changed.
- Endpoint IP is changed (static to DHCP and vice versa).

If you need to retain the client entry in the WGB table for a longer duration, we recommend you increase the client WGB bridge timeout duration.

Procedure

	Command or Action	Purpose
Step 1	**configure wgb bridge client add** *mac-address* **Example:** `Device# configure wgb bridge client add F866.F267.7DFB-`	Adds a WGB client using the MAC address.
Step 2	**configure wgb bridge client timeout** *timeout-secs* **Example:** `Device# configure wgb bridge client timeout 400`	Configures the WGB bridge client timeout. Default timeout value is 300 seconds. The valid range is between 10 and 1000000 seconds.
Step 3	**show wgb bridge** **Example:** `Device# show wgb bridge`	Displays the WGB wired clients over the bridge.
Step 4	**show wgb bridge wired gigabitEthernet** *interface* **Example:** `Device# show wgb bridge wired gigabitEthernet 0/1`	Displays the WGB Gigabit wired clients over the bridge.

	Command or Action	Purpose
Step 5	**show wgb bridge dot11Radio** *interface-number* **Example:** `Device# show wgb bridge dot11Radio 0/3/1`	Displays the WGB bridge radio interface summary.

Information About Simplifying WGB Configuration

From Cisco IOS XE Cupertino 17.8.1, it is possible to configure WGB in multiple Cisco access points (APs) simultaneously. By importing a running configuration, you can deploy multiple WGBs in a network and make them operational quicker. When new Cisco APs are added to the network, you can transfer an existing or working configuration to the new Cisco APs to make them operational. This enhancement eliminates the need to configure multiple Cisco APs using CLIs, after logging into them.

A network administrator can onboard Cisco APs using either of the following methods:

- Upload the working configuration from an existing Cisco AP to a server and download it to the newly deployed Cisco APs.
- Send a sample configuration to all the Cisco APs in the deployment.

This feature is supported only on the following Cisco APs:

- Cisco Aironet 1562 Access Points
- Cisco Aironet 2800 Access Points
- Cisco Aironet 3800 Access Points
- Cisco Catalyst 9105 Access Points
- Cisco Catalyst 9115 Access Points
- Cisco Catalyst 9120 Access Points
- Cisco Catalyst IW6300 Series Heavy Duty Access Points

For latest support information on various features in Cisco Wave 2 and 802.11ax (Wi-Fi 6) Access Points in Cisco IOS XE releases, see the Feature Matrix for Wave 2 and 802.11ax (Wi-Fi 6) Access Points document.

Configuring Multiple WGBs (CLI)

Perform the following procedure on the APs in WGB mode.

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:**	Enters privileged EXEC mode.

	Command or Action	Purpose
	`Device# enable`	
Step 2	**copy configuration upload**{**sftp:**| **tftp:**} *ip-address* [*directory*] [*file-name*] **Example:** `Device# copy configuration upload sftp: 10.10.10.1 C:sample.txt`	Creates upload configuration file and uploads to the SFTP or TFTP server using the specified path.
Step 3	**copy configuration download**{**sftp:**| **tftp:**} *ip-address* [*directory*] [*file-name*] **Example:** `Device# copy configuration download sftp: 10.10.10.1 C:sample.txt`	Downloads the configuration file and replaces the old configuration in the AP and reboots the WGB. When the device restarts, new configuration is applied.
Step 4	**show wgb dot11 association** **Example:** `Device# show wgb dot11 association`	Lists the WGB uplink information.
Step 5	**show version** **Example:** `Device# show version`	Displays the AP software information.

Verifying WGB Configuration

After completing the configuration download and reboot of the AP, the WGB rejoins the network. Use the **show logging** command to list and verify the download events that are captured in the debug logs:

```
Device# show logging

Jan 13 18:19:17 kernel: [*01/13/2022 18:19:17.4880] WGB - Applying download config...
Jan 13 18:19:18 download_config: configure clock timezone UTC
Jan 13 18:19:18 download_config: configure dot1x credential dot1x_profile username wifiuser
 password U2FsdGVkX1+8PWmAOnFO8BXyk5EAphMy2PmhPPhWV0w=
Jan 13 18:19:18 download_config: configure eap-profile eap_profile method PEAP
Jan 13 18:19:18 download_config: configure eap-profile eap_profile dot1x-credential
dot1x_profile
Jan 13 18:19:18 chpasswd: password for user changed
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7260] chpasswd: password for user changed
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7610]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7610]  Management user configuration saved
successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7610]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650] Warning!!! Attach SSID profile with the
 radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650]  Dot1x credential configuration has
been saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740] Warning!!! Attach SSID profile with the
 radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740]  EAP profile configuration has been
```

```
saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790] Warning!!! Attach SSID profile with the
 radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790]  EAP profile configuration has been
saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7830] Warning!!! Attach SSID profile with the
 radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7830]
Jan 13 18:19:18 download_config: configure ssid-profile psk ssid alpha_psk authentication
psk U2FsdGVkX18meBfFFeiC4sgkEmbGPNH/ul1dne6h/m8= key-management wpa2
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930] Warning!!! Attach SSID profile with the
 radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930]  EAP profile configuration has been
saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930]
Jan 13 18:19:18 download_config: configure ssid-profile open ssid alpha_open authentication
 open
Jan 13 18:19:18 download_config: configure ssid-profile openax ssid alpha_open_ax
authentication open
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.8650]  SSID-Profile dot1xpeap has been saved
 successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.8650]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.9270]  SSID-Profile psk has been saved
successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.9270]
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]  SSID-Profile open has been saved
successfully
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]  SSID-Profile openax has been saved
successfully
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]
Jan 13 18:19:22 download_config: configure wgb broadcast tagging disable
Jan 13 18:19:22 download_config: configure wgb packet retries 64 drop
Jan 13 18:19:22 kernel: [*01/13/2022 18:19:22.9710] Broadcast tagging 0 successfully
Jan 13 18:19:22 kernel: [*01/13/2022 18:19:22.9710]
Jan 13 18:19:23 download_config: configure dot11Radio 1 mode wgb ssid-profile open
Jan 13 18:19:23 download_config: configure dot11Radio 1 enable
Jan 13 18:19:23 download_config: configure ap address ipv6 disable
```