Configure Point-to-Point Mesh Link with Ethernet Bridging on Embedded Wireless Controller with C9124 Access Points

Contents

Introduction

This document describes how to configure P2P Mesh Link with Ethernet Bridging on Embedded Wireless Controller (eWC) with C9124 Access Points.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Cisco Wireless Lan Controllers (WLC) 9800.
- Cisco Catalyst Access Points (APs).
- Embedded Wireless Controller on Catalyst Access Points.
- Mesh technology.

Components Used

The information in this document is based on these software and hardware versions:

- EWC IOS® XE 17.12.2.
- 2x APs C9124.
- 2x Power Injectors AIR-PWRINJ-60RGD1.
- 2x switches;
- 2x laptops;
- 1x AP C9115.

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.

Background Information

Ethernet Bridging

The mesh network solution, which is part of the Cisco unified wireless network solution, enables two or more Cisco Mesh Access Points (hereafter called mesh access points) to communicate with each other over one or more wireless hops to join multiple LANs or to extend WiFi coverage.

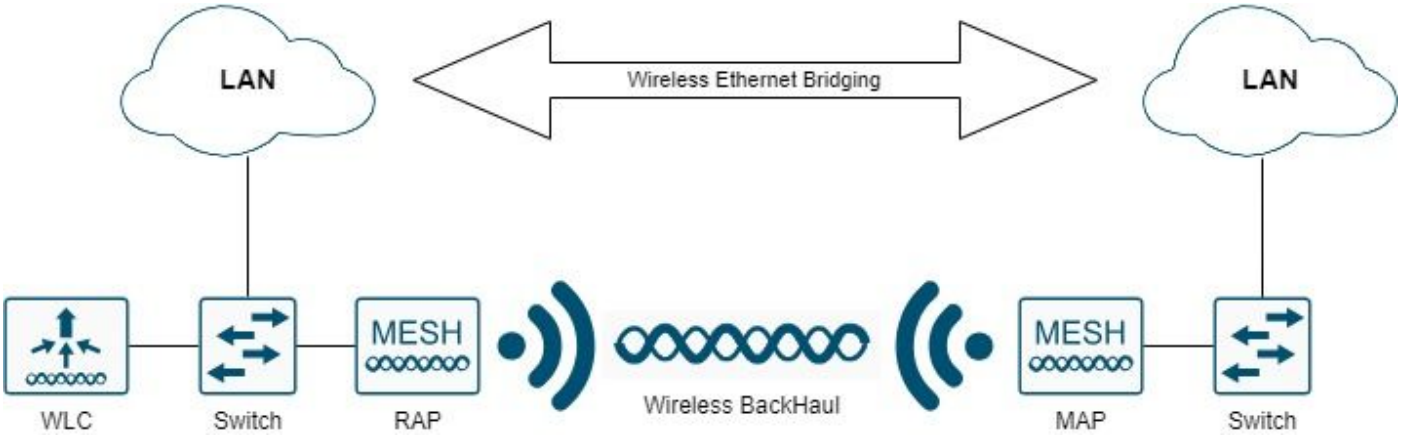
Cisco mesh access points are configured, monitored, and operated from and through any Cisco Wireless LAN controller that is deployed in the mesh networking solution.

Supported mesh networking solution deployments are of one of three general types:

- Point-to-point deployment
- Point-to-multipoint deployment
- Mesh deployment

This document focuses on how to configure point-to-point mesh deployment and Ethernet bridging on the same.

In point-to-point mesh deployment, the mesh access points provide wireless access and backhaul to wireless clients, and can simultaneously support bridging between one LAN and a termination to a remote Ethernet device or another Ethernet LAN.



Refer toMesh Deployment Guide for Cisco Catalyst 9800 Series Wireless Controllersfor detailed information on each of these deployment types.

The Cisco Catalyst 9124 Series outdoor mesh AP is a wireless device designed for wireless client access and point-to-point bridging, point-to-multipoint bridging, and point-to-multipoint mesh wireless connectivity.

The outdoor access point is a standalone unit that can be mounted on a wall or overhang, on a rooftop pole, or on a street light pole.

You can operate the C9124 in one of these mesh roles:

- Roof-top Access Point (RAP)
- Mesh Access Point (MAP)

RAPs have a wired connection to a Cisco wireless LAN controller. They use the backhaul wireless interface to communicate with nearby MAPs. RAPs are the parent node to any bridging or mesh network and connect a bridge or mesh network to the wired network, so there can be only one RAP for any bridged or mesh network segment.

MAPs have no wired connection to a Cisco Wireless LAN controller. They can be completely wireless and support clients that communicate with other MAPs or RAPs, or they can be used to connect to peripheral devices or a wired network.

Embedded Wireless Controller on Catalyst Access Point

The Cisco Embedded Wireless Controller (EWC) on Catalyst Access Points is a software-based controller integrated into Cisco Catalyst 9100 Access Points.

In a Cisco EWC network, an Access Point (AP) that runs the wireless controller function is designated as the active AP.

The other access points, which are managed by this active AP, are referred to as subordinate APs.

The active EWC has two roles:

- It functions and operates as a Wireless LAN Controller (WLC) to manage and control the subordinate APs. The subordinate APs operate as lightweight access points to serve clients.
- It operates as an access point to serve clients.

To have a product overview about EWC on APs, please visit the Cisco Embedded Wireless Controller on Catalyst Access Points Data Sheet.

To know how to deploy EWC on your network please visit the Cisco Embedded Wireless Controller on Catalyst Access Points (EWC) White Paper.

This document focuses on C9124 as EWC and assumes there is already an AP 9124 in EWC mode.

Configure

Network Diagram

All the devices in this network are located inside the 192.168.100.0/24 subnet except the laptops that are in VLAN 101 with subnet 192.168.101.0/25.

The EWC AP (WLC) has its management interface untagged, and the native VLAN on switchports is set to VLAN 100.

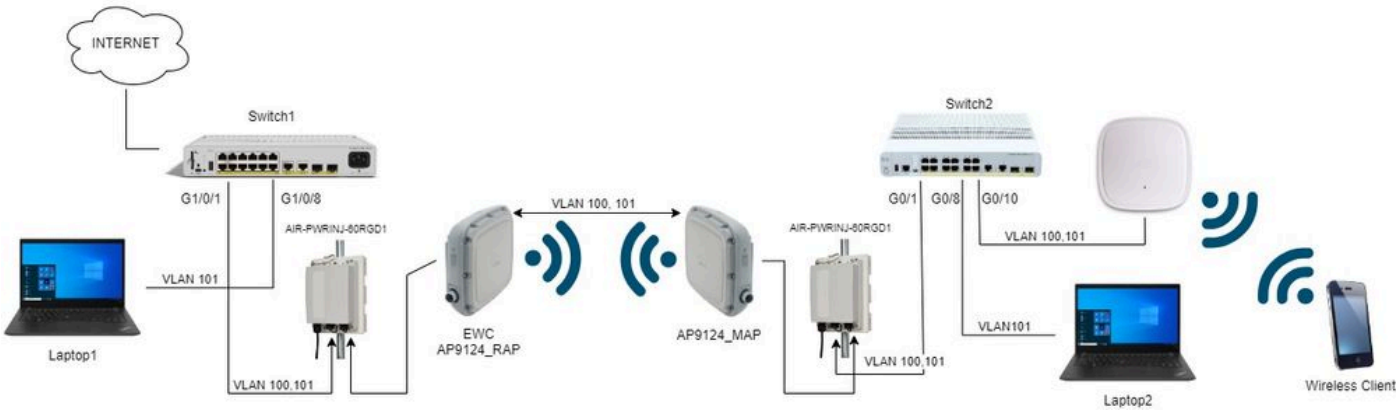
AP AP9124_RAP has the role of a eWC and Root Access Point (RAP), while AP9124_MAP takes the role of Mesh Access Point (MAP).

In this lab an AP C9115 is also placed behind the MAP to show that we can have APs to join a WLC over a Mesh link.

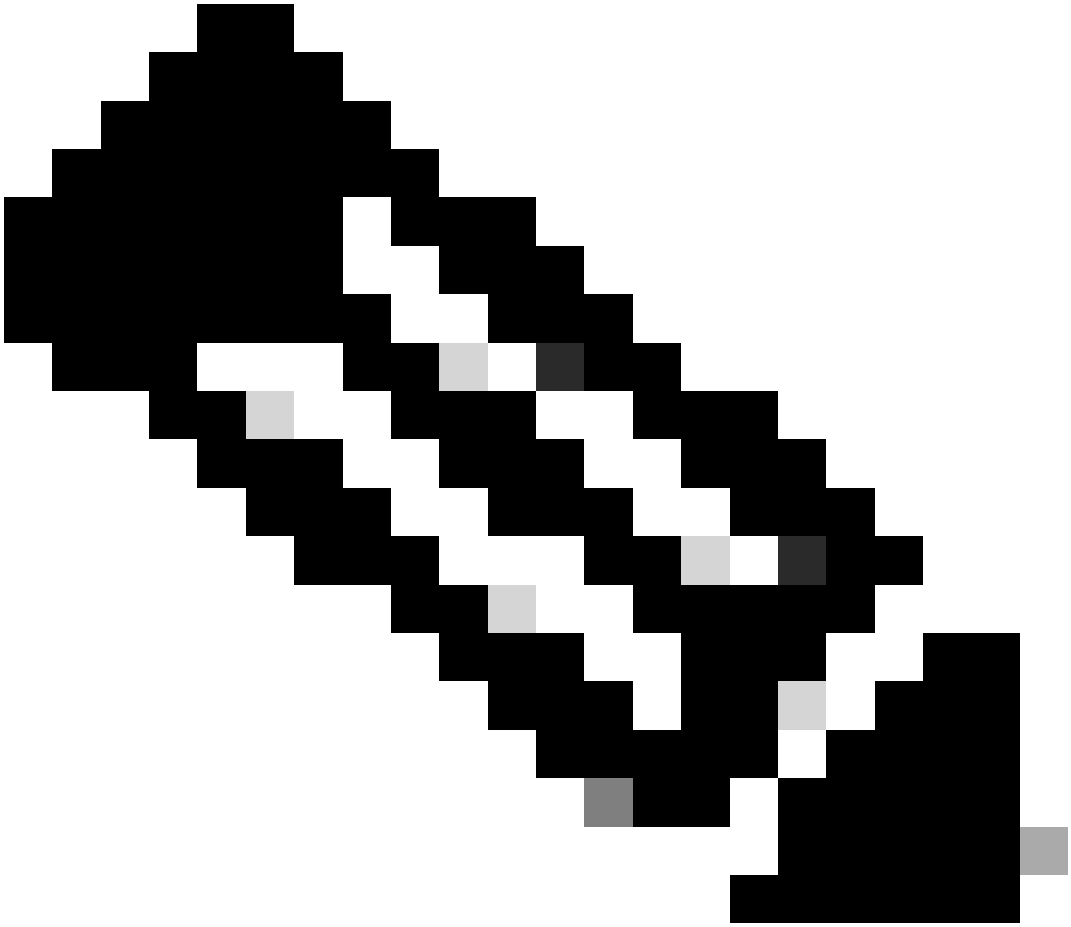
This table contains the IP addresses of all devices in the network:

Note: Tagging the management interface can cause issues with the AP joining the internal WLC process. If you decide to tag the management interface, ensure that the wired infrastructure part is configured accordingly.

Device	IP Address
Default Gateway	Static on VLAN 100: 192.168.100.1
Laptop1	DHCP on VLAN 101
Laptop2	DHCP on VLAN 101
Switch1 (DHCP server)	VLAN 100 SVI: Static on VLAN 100: 192.168.100.1 (DHCP server)
Switch1 (DHCP server)	VLAN 101 SVI: Static on VLAN 101: 192.168.101.1 (DHCP server)
Switch2	VLAN 100 SVI: DHCP on VLAN 100
Switch2	VLAN 101 SVI: DHCP on VLAN 101
9124EWC	Static on VLAN 100: 192.168.100.40
AP9124_RAP	DHCP on VLAN 100
AP9124_MAP	DHCP on VLAN 100
AP9115	DHCP on VLAN 100



Network Diagram



Note: The C9124 APs are powered using AIR-PWRINJ-60RGD1 with the guidelines in the Cisco Catalyst 9124AX Series Outdoor Access Point Hardware Installation Guide.

Configurations

This document assumes there is already an AP 9124 running EWC with initial deployment done as per Cisco Embedded Wireless Controller on Catalyst Access Points (EWC) White Paper.

For other Tips and Tricks regarding conversion process the please check the Convert Catalyst 9100 Access Points to Embedded Wireless Controller document.

Switch Configurations

Here are the switches relevant configurations.

Switch ports where APs are connected are in trunk mode with the native VLAN set to 100 and allowing VLAN 101.

During staging of the APs, you need to configure the MAP as MAP, therefore you need to make the AP join

the eWC via ethernet. Here we use Switch1 port G1/0/2 for staging the MAP. After staging the MAP is moved to Switch2.

Switchports where laptops are connected are configured as access ports on VLAN 101.

Switch1:

```
ip dhcp excluded-address 192.168.101.1 192.168.101.10
ip dhcp excluded-address 192.168.100.1 192.168.100.10
!
ip dhcp pool AP_VLAN100
network 192.168.100.0 255.255.255.0
default-router 192.168.100.1
dns-server 192.168.1.254
!
ip dhcp pool VLAN101
network 192.168.101.0 255.255.255.0
default-router 192.168.101.1
dns-server 192.168.1.254
!
interface GigabitEthernet1/0/1
 description AP9124_RAP (EWC)
 switchport trunk native vlan 100
 switchport trunk allowed vlan 100,101
 switchport mode trunk
 end
interface GigabitEthernet1/0/2
 description AP9124_MAP_Staging
 switchport trunk native vlan 100
 switchport trunk allowed vlan 100,101
 switchport mode trunk
 end
interface GigabitEthernet1/0/8
 description laptop1
 switchport access vlan 101
 switchport mode access
 spanning-tree portfast edge
 end
```

Switch2:

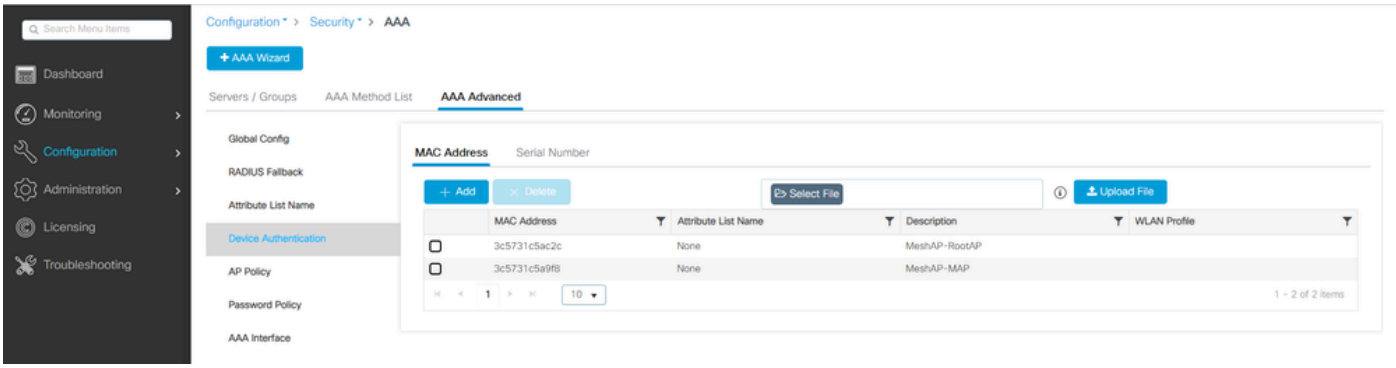
```
interface GigabitEthernet0/1
 description AP9124_MAP
 switchport trunk native vlan 100
 switchport trunk allowed vlan 100,101
 switchport mode trunk
 end
interface GigabitEthernet0/8
 description laptop2
 switchport access vlan 101
 switchport mode access
 spanning-tree portfast edge
 end
interface GigabitEthernet0/1
 description AP9115
 switchport trunk native vlan 100
```

```
 switchport trunk allowed vlan 100,101
 switchport mode trunk
 end
```

EWC and RAP Configuration

After Day0 configuration of the EWC AP, the embedded AP needs to joins itself.

1. Add the Ethernet mac addresses of Root AP and Mesh AP to Device Authentication. Go to **Configuration** > **Security** > **AAA** > **AAA Advanced** > **Device Authentication,** click button **+Add:**



MAC Addresses in Device Authentication

CLI commands:

```
9124EWC(config)#username 3c5731c5ac2c mac description MeshAP-RootAP
9124EWC(config)#username 3c5731c5a9f8 mac description MeshAP-MAP
```

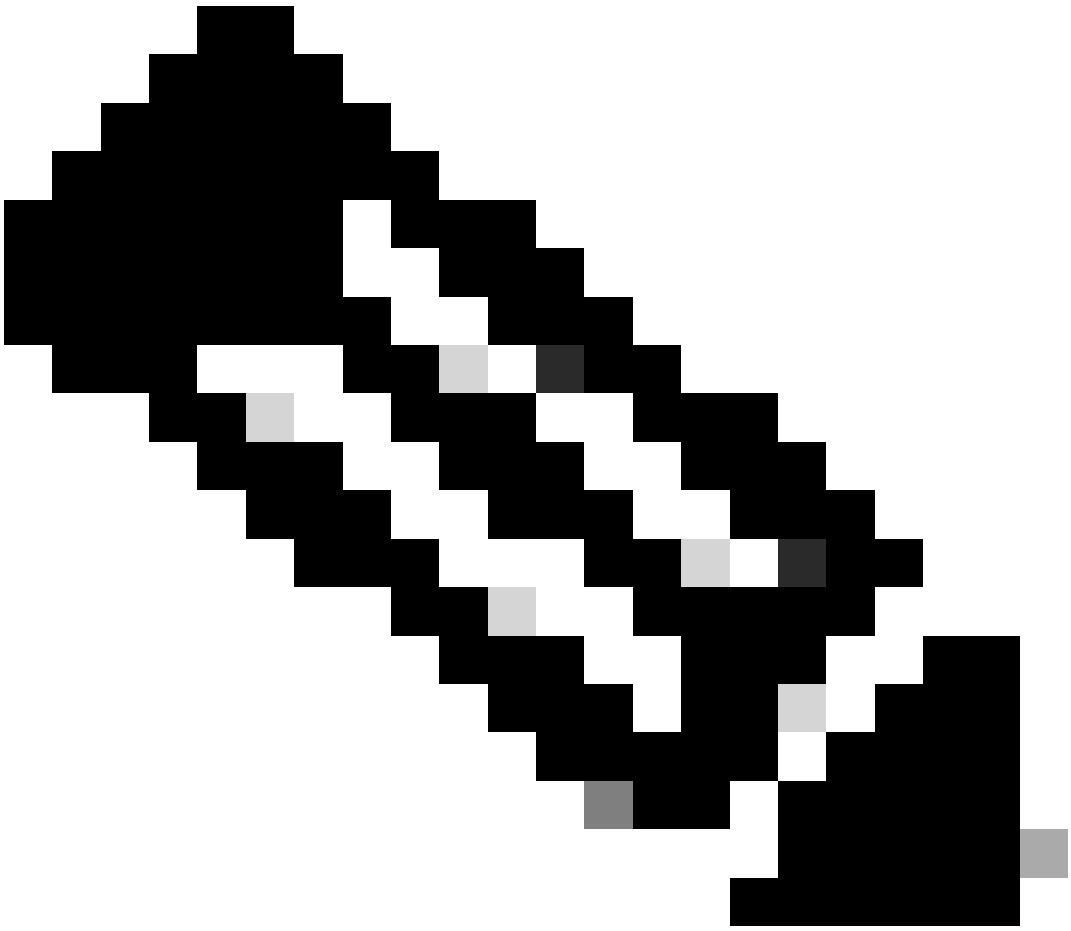
The Ethernet mac address can be confirmed by running the "show controllers wired 0" from the AP CLI. Example from root AP:

```
AP3C57.31C5.AC2C#show controllers wired 0
wired0 Link encap:Ethernet HWaddr 3C:57:31:C5:AC:2C
```

Access to the underlying AP shell can be completed with the command "*wireless ewc-ap ap shell username x*" as exemplified:

```
9124EWC#wireless ewc-ap ap shell username admin
[...]
admin@192.168.255.253's password:
AP3C57.31C5.AC2C>en
Password:
AP3C57.31C5.AC2C#
AP3C57.31C5.AC2C#logout
Connection to 192.168.255.253 closed.
```

```
9124EWC#
```

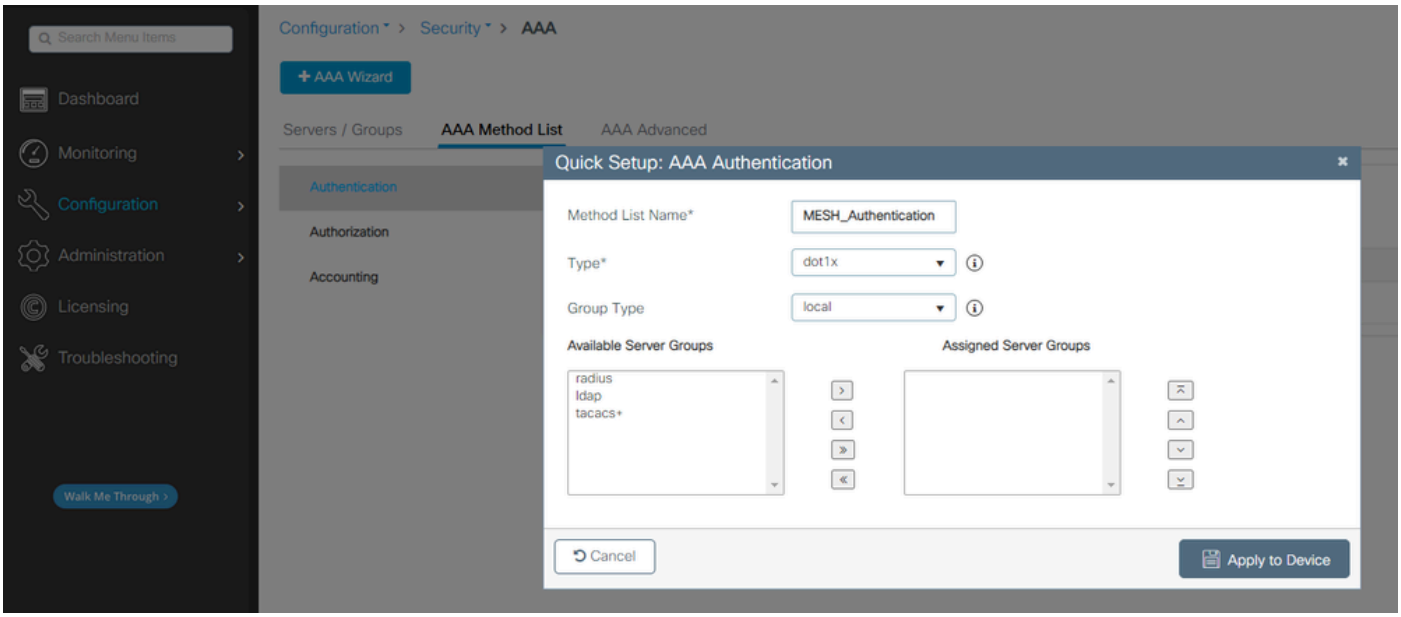


Note: This command is equivalent to ***apciscoshell*** that was previously available in Mobility Express controllers.

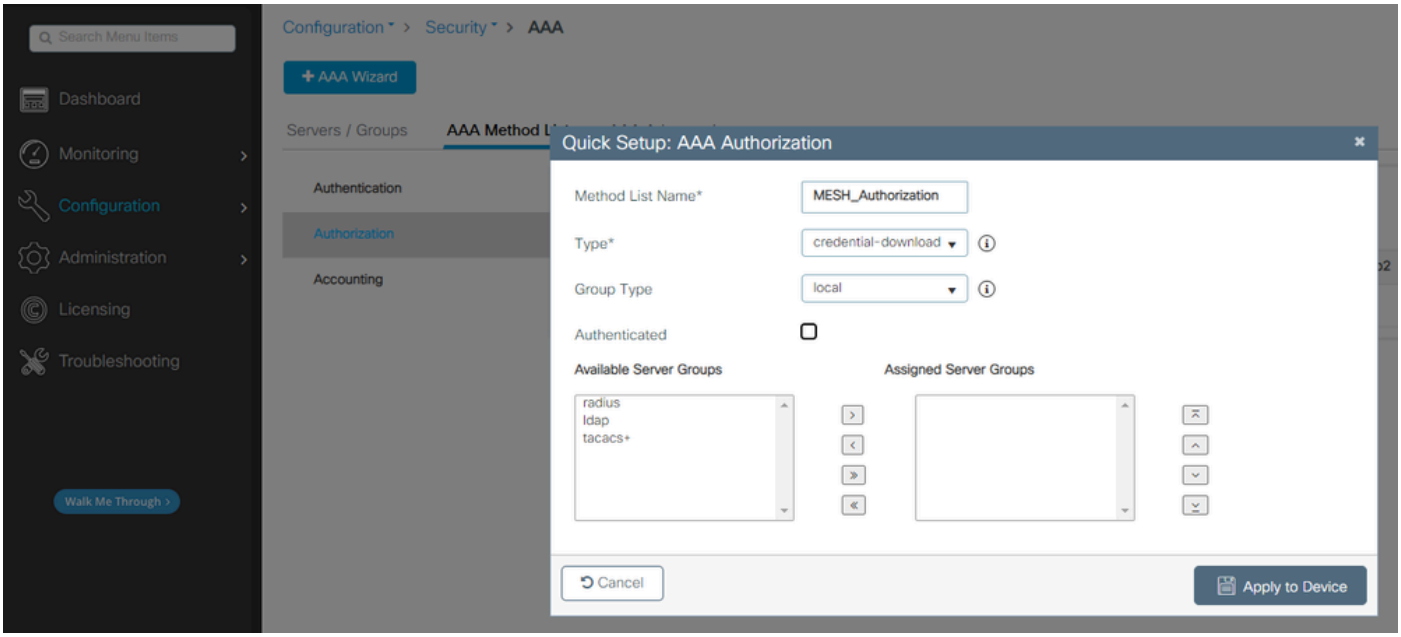
If the AP management username and password are not specified in the AP profile, use the default username **Cisco** and password **Cisco** instead.

---

2. Add Authentication and Authorization Methods:



Authentication Method List

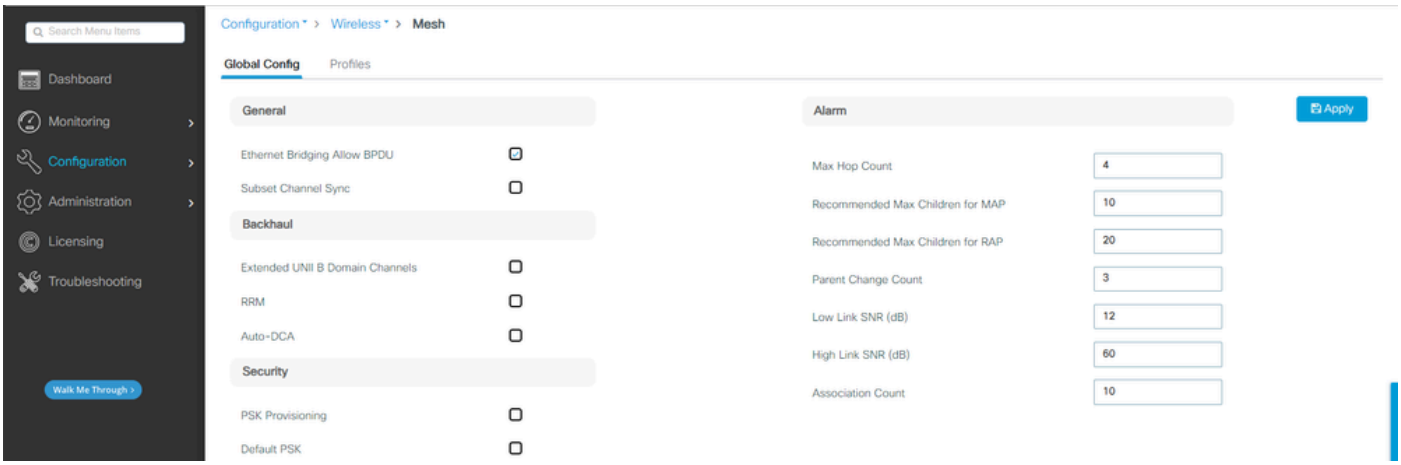


Authorization Method List

CLI commands:

```
9124EWC(config)#aaa authentication dot1x MESH_Authentication local
9124EWC(config)#aaa authorization credential-download MESH_Authorization local
```

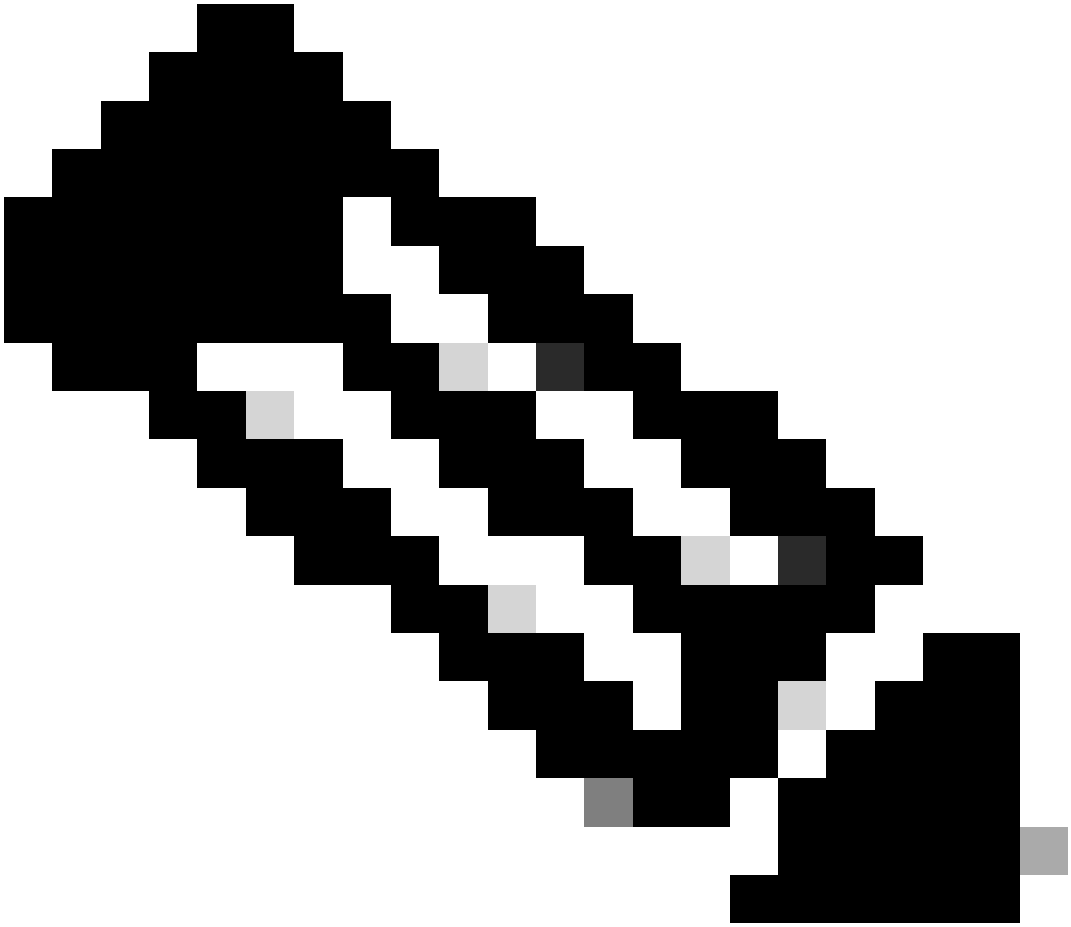
3. Go to **Configuration** > **Wireless** > **Mesh**. As the setup in this document requires Ethernet bridging, enable **Ethernet Bridging Allow BPDUs**:



Ethernet Bridging Allow BPDU

CLI commands:

```
9124EWC(config)#wireless mesh ethernet-bridging allow-bdpu
```



Note: By default, the mesh APs are not forwarding BPDUs over the mesh link.

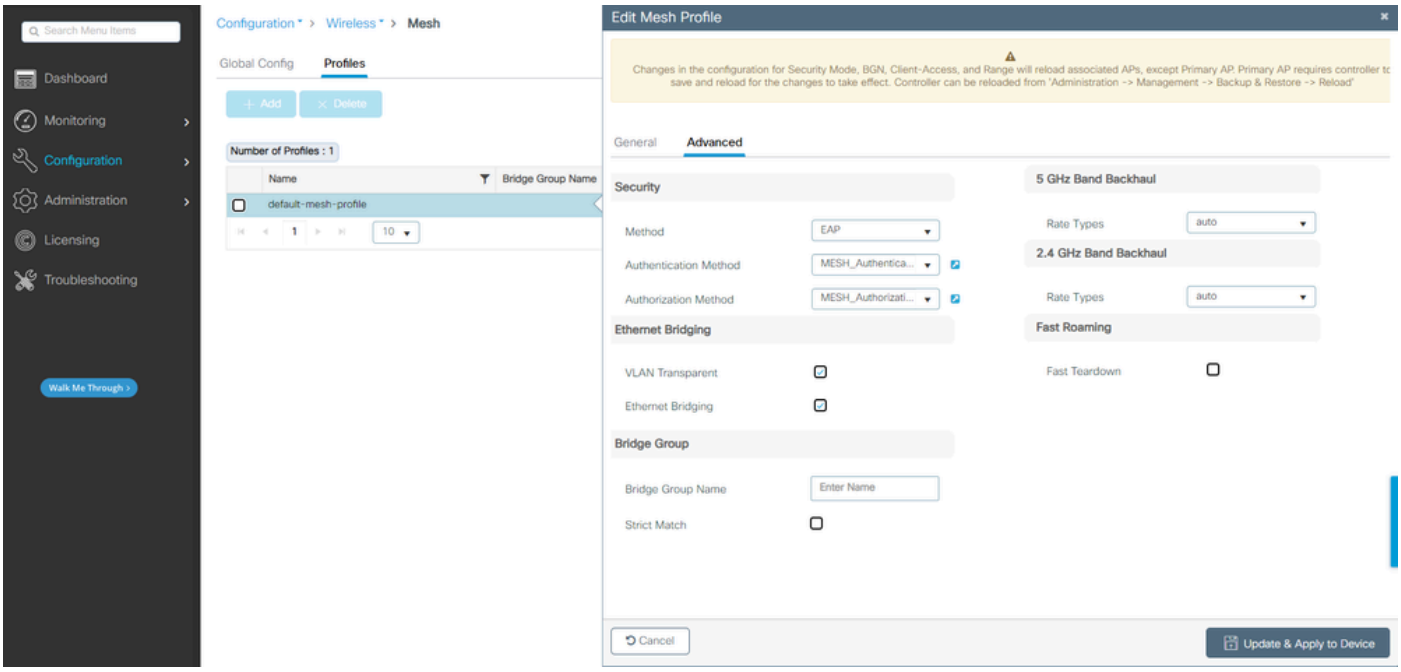
If you do not have any redundant link between the 2 sites, then it is not needed.

If there are redundant links then you need to allow BPDUs. If this is not done, you risk creating a STP loop in the network.

---

4. Configure the **default-mesh-profile** where you select the previously configured AAA Authentication and Authorization methods. **Click** and edit the **default-mesh-profile**.

Go to the **Advanced** tab and select the **Authentication** and **Authorization** methods. Enable option **Ethernet Bridging**.



Edit default-mesh-profile

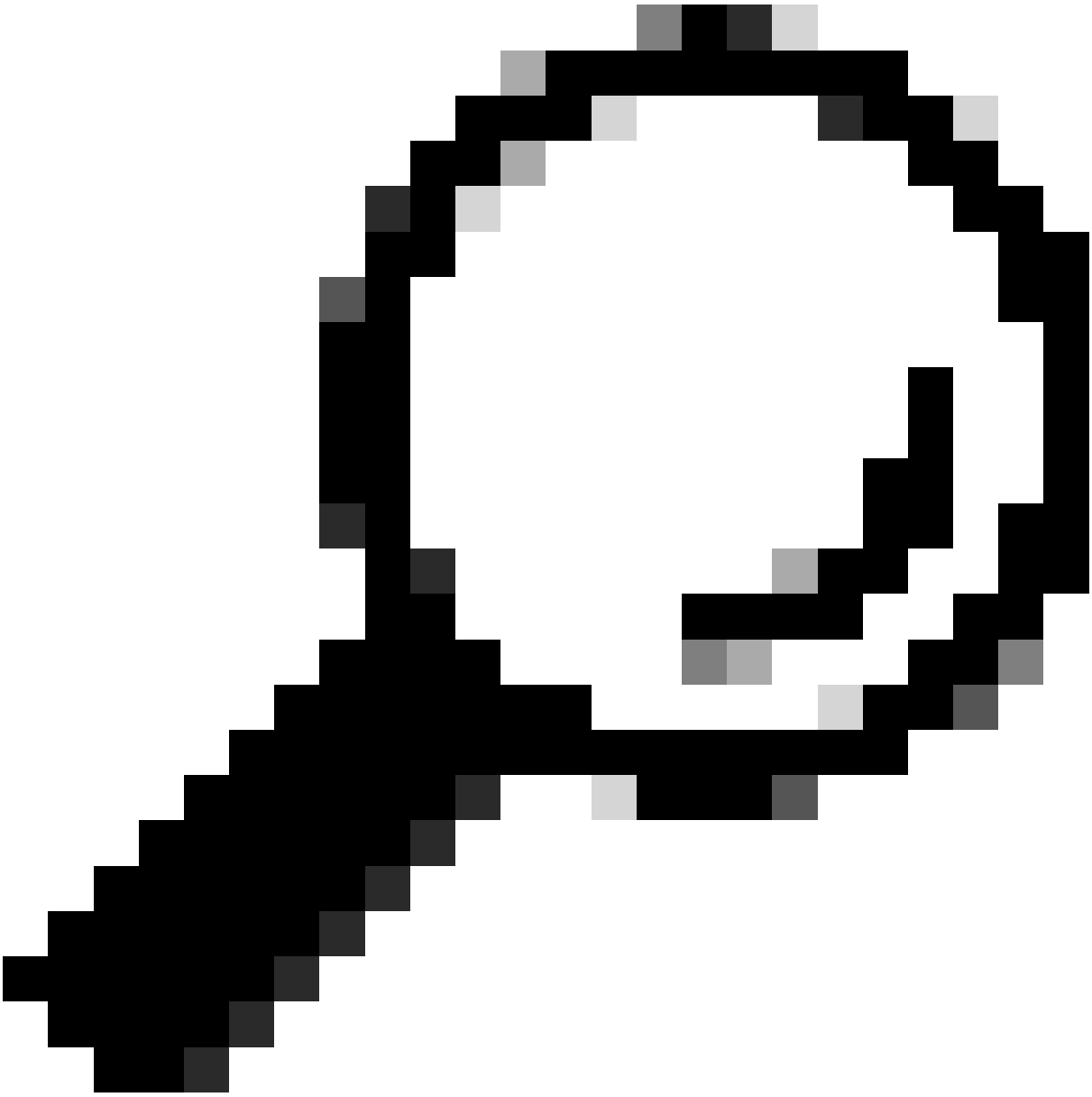
CLI commands:

```
9124EWC(config)#wireless profile mesh default-mesh-profile
9124EWC(config-wireless-mesh-profile)#description "default mesh profile"
9124EWC(config-wireless-mesh-profile)#ethernet-bridging
9124EWC(config-wireless-mesh-profile)#ethernet-vlan-transparent
9124EWC(config-wireless-mesh-profile)#method authentication MESH_Authentication
9124EWC(config-wireless-mesh-profile)#method authorization MESH_Authorization
```

Special callout to the option VLAN Transparent:

This feature determines how a mesh access point handles VLAN tags for Ethernet bridged traffic:

- If VLAN Transparent is enabled, then VLAN tags are not handled and packets are bridged as untagged packets.
  - No configuration of Ethernet ports is required when VLAN transparent is enabled. The Ethernet port passes both tagged and untagged frames without interpreting the frames.
- If VLAN Transparent is disabled, then all packets are handled according to the VLAN configuration on the port (trunk, access, or normal mode).
  - If the Ethernet port is set to Trunk mode, then Ethernet VLAN tagging must be configured.



Tip: To use AP VLAN tagging, you must uncheck the VLAN Transparent check box.

If you do not use VLAN tagging, it means the RAP and MAP are on Native VLAN configured on the Trunk ports. In this condition, if you want other devices behind MAP to be on the Native VLAN (here VLAN 100), then you need to enable VLAN Transparent.

5. The internal AP joins the EWC and you can verify the AP join state using the command "show ap summary":

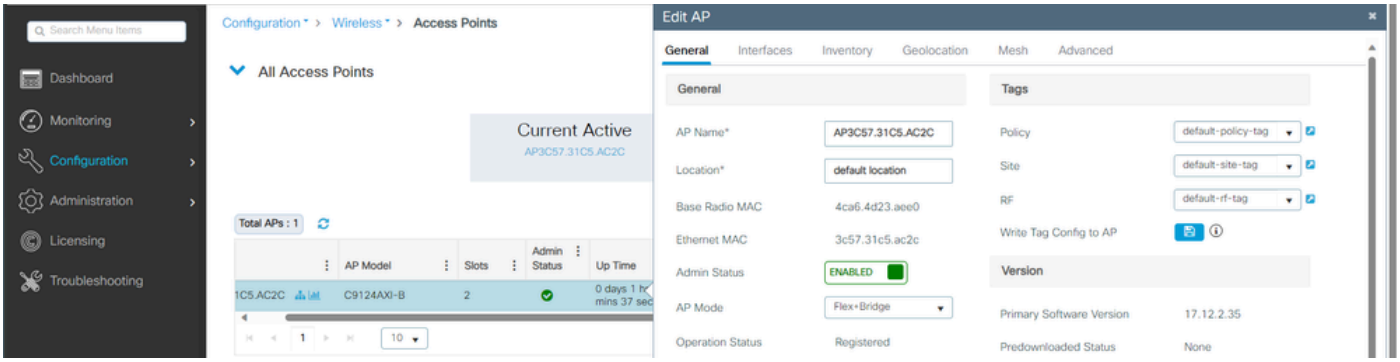
```
9124EWC#show ap summary
Number of APs: 1

CC = Country Code
RD = Regulatory Domain

AP Name                        Slots AP Model            Ethernet MAC   Radio MAC      CC   RD   IP Address                                State        Location
-------------------------------------------------------------------------------------------------------------------------------------------------------------------
AP3C57.31C5.AC2C               2     C9124AXI-B          3c57.31c5.ac2c 4ca6.4d23.aee0 US   -B   192.168.100.11                            Registered   default location
```

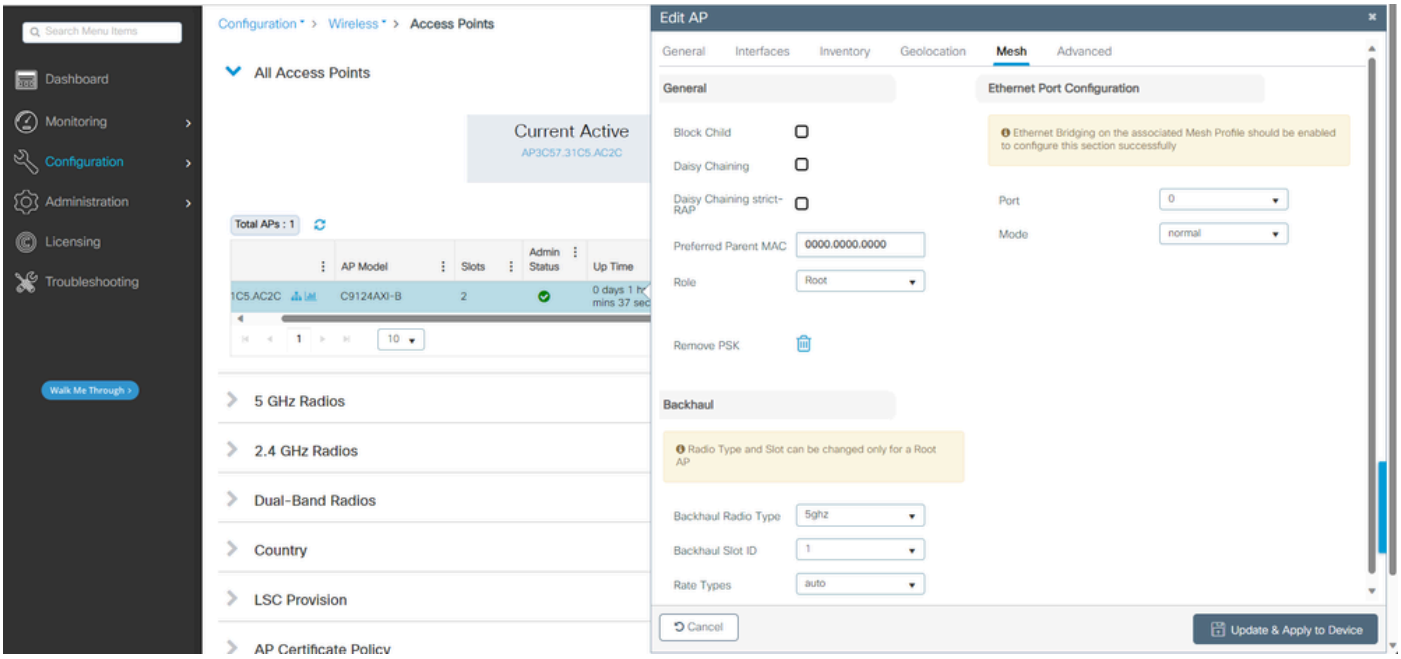
show ap summary

You can also see the AP joined via the GUI where the AP show up as Flex+Bridge mode. For convenience you can change the name of the AP now. In this setup it is used the name AP9124_RAP:

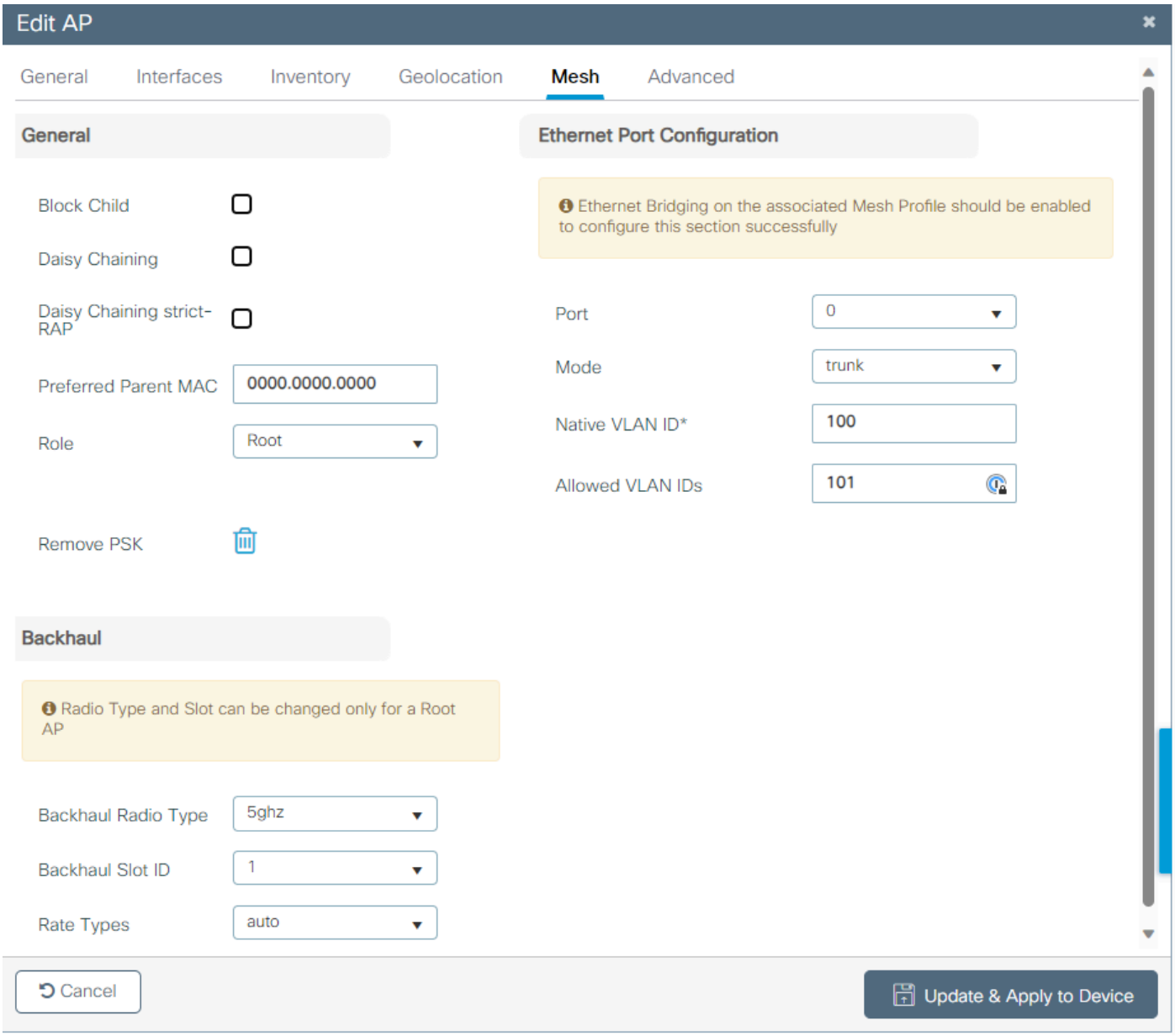


AP General details

You can edit the **Geolocation** and then in the **Mesh** tab, make sure its **Role** is configured as **Root** AP and the **Ethernet Port Configuration** is set to **trunk** with corresponding VLAN IDs:



Mesh Role Root

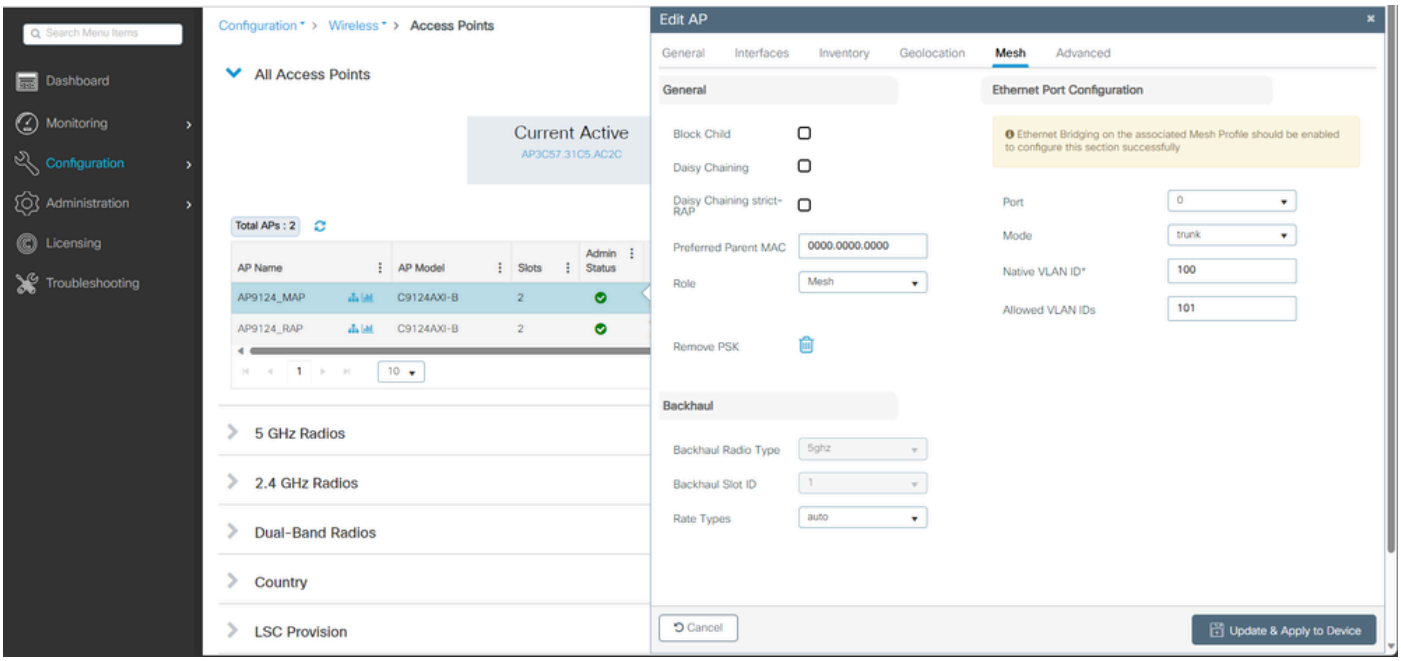


Ethernet Port Configuration

Configure MAP

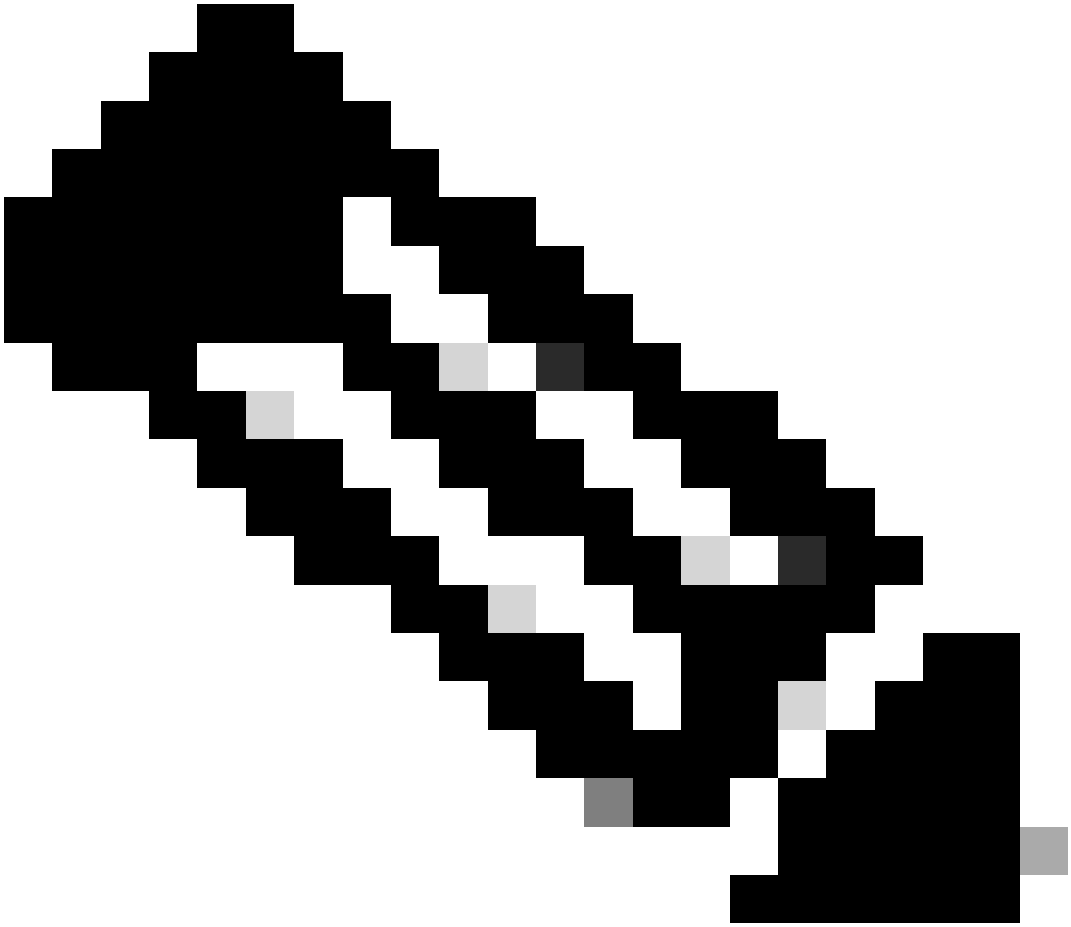
It is now time to join the 9124 MAP.

1. Connect the MAP AP to the Switch1 for staging. The AP joins the EWC and shows in the AP list. Change its name to something like AP9124_MAP and configure it as **Mesh Role** in the **Mesh** tab. Click **Update & Apply to Device**:



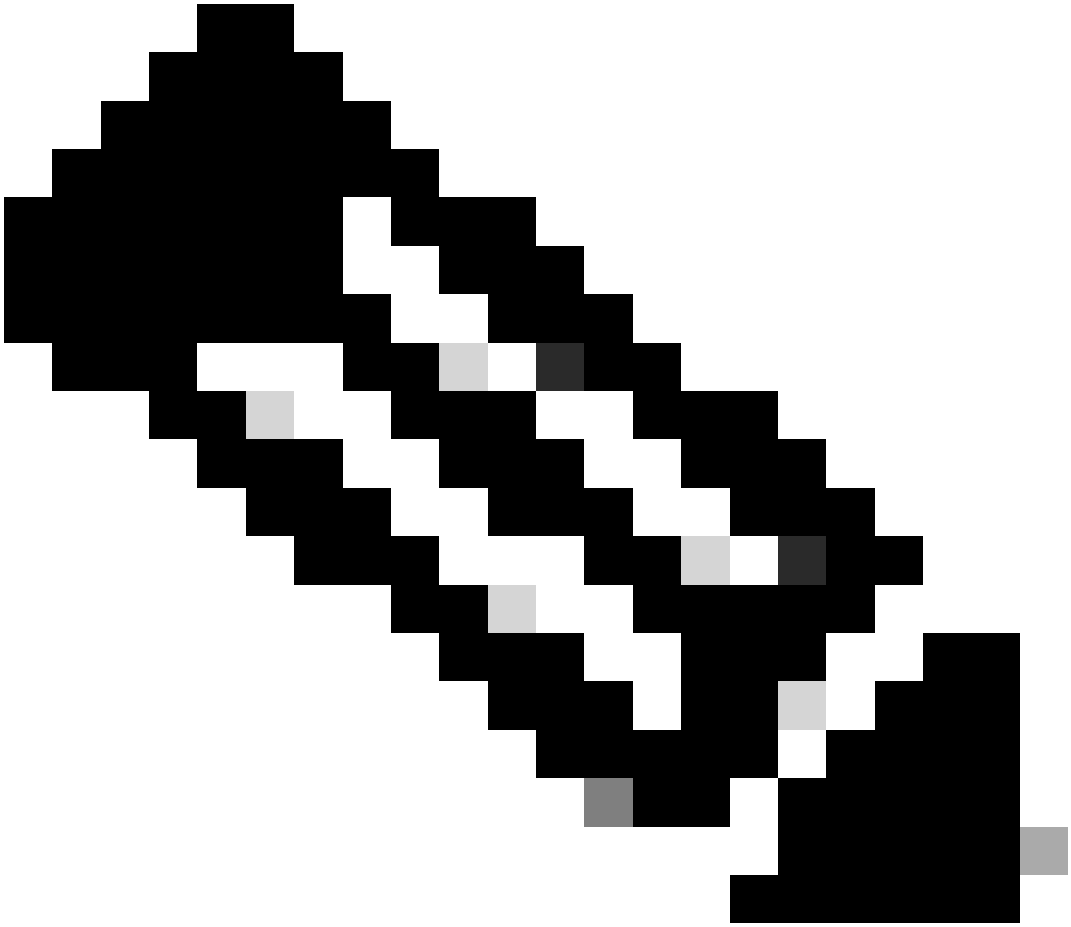
MAP configuration

2. Disconnect the AP from Switch1 and connect to Switch2 as per the Network Diagram. The MAP joins the EWC via wireless interface through the RAP.



Note: As the APs are powered via power injector, the AP does not go down, and as the setup is in a controlled environment, the Switch2 is physically close and we can simply move the cable from one switch to the other.

You can connect a console cable to the AP and see what happens via console. Here are some important messages seen.



Note: From release 17.12.1, the default console baud rate of the 802.11AX APs is changed from 9600 bps to 115200 bps.

MAP looses connectivity to EWC:

```
AP9124_MAP#
[*01/11/2024 14:08:23.0214] chatter: Device wired0 notify state change link DO
[*01/11/2024 14:08:28.1474] Re-Tx Count=1, Max Re-Tx Value=5, SendSeqNum=83, N
[*01/11/2024 14:08:28.1474]
[*01/11/2024 14:08:31.1485] Re-Tx Count=2, Max Re-Tx Value=5, SendSeqNum=83, N
[*01/11/2024 14:08:31.1486]
[*01/11/2024 14:08:33.4214] chatter: Device wired0 notify state change link UP
[*01/11/2024 14:08:34.1495] Re-Tx Count=3, Max Re-Tx Value=5, SendSeqNum=83, N
[*01/11/2024 14:08:34.1495]
[*01/11/2024 14:08:37.1505] Re-Tx Count=4, Max Re-Tx Value=5, SendSeqNum=84, N
[*01/11/2024 14:08:37.1505]
[*01/11/2024 14:08:40.1515] Re-Tx Count=5, Max Re-Tx Value=5, SendSeqNum=84, N
[*01/11/2024 14:08:40.1515]
[*01/11/2024 14:08:43.1524] Max retransmission count exceeded, going back to D
```

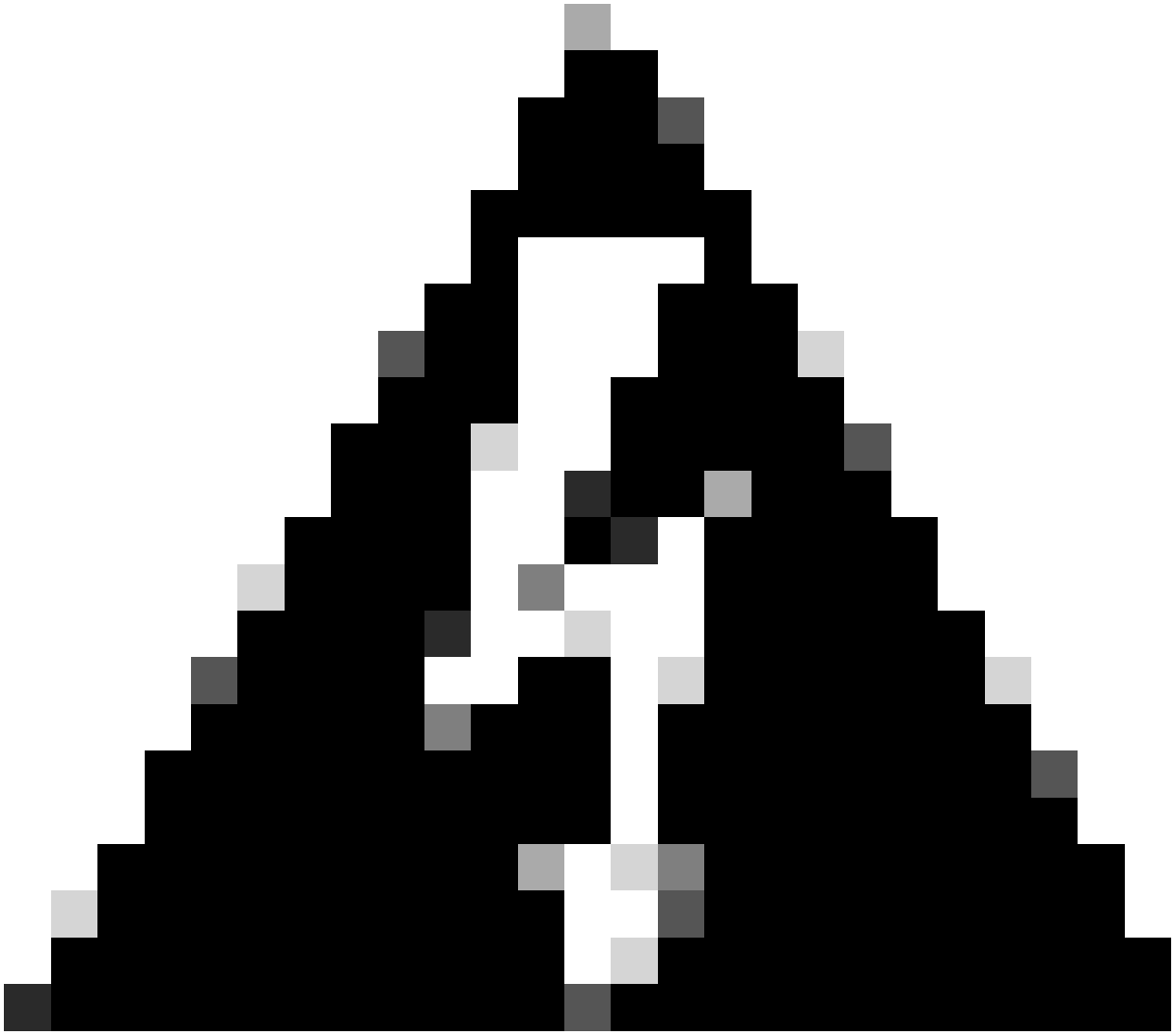
```
[...]
[*01/11/2024 14:08:48.1537] CRIT-MeshWiredAdj[0][3C:57:31:C5:A9:F8]: Blocklist
[*01/11/2024 14:08:48.1538] CRIT-MeshWiredAdj[0][3C:57:31:C5:A9:F8]: Remove as
[*01/11/2024 14:08:48.1539] CRIT-MeshLink: Link Down Block Root port Mac: 3C:5
[*01/11/2024 14:08:48.1542] CRIT-MeshWiredBackhaul[0]: Remove as uplink
```

MAP moves to discovery mode via wireless and finds the RAP via Radio Backhaul on channel 36, finds EWC and joins it:

```
[*01/11/2024 14:08:51.3893] CRIT-MeshRadioBackhaul[1]: Set as uplink
[*01/11/2024 14:08:51.3894] CRIT-MeshAwppAdj[1][4C:A6:4D:23:AE:F1]: Set as Par
[*01/11/2024 14:08:51.3915] wlan: [0:I:CMN_MLME] mlme_ext_vap_down: VAP (mon0)
[*01/11/2024 14:08:51.3926] wlan: [0:I:CMN_MLME] mlme_ext_vap_down: VAP (apbhr
[*01/11/2024 14:08:51.4045] wlan: [0:I:CMN_MLME] mlme_ext_vap_up: VAP (apbhr0)
[*01/11/2024 14:08:51.4053] wlan: [0:I:CMN_MLME] mlme_ext_vap_up: VAP (mon0) t
[*01/11/2024 14:08:53.3898] CRIT-MeshLink: Set Root port Mac: 4C:A6:4D:23:AE:F
[*01/11/2024 14:08:53.3904] Mesh Reconfiguring DHCP.
[*01/11/2024 14:08:53.8680] DOT11_UPLINK_EV: wgb_uplink_set_port_authorized: c
[*01/11/2024 14:08:53.9232] CRIT-MeshSecurity: Mesh Security successful auther
[...]
[*01/11/2024 14:09:48.4388] Discovery Response from 192.168.100.40
[*01/11/2024 14:09:59.0000] Started wait dtls timer (60 sec)
[*01/11/2024 14:09:59.0106]
[*01/11/2024 14:09:59.0106] CAPWAP State: DTLS Setup
[*01/11/2024 14:09:59.0987] dtls_verify_server_cert: Controller certificate ve
[*01/11/2024 14:09:59.8466]
[*01/11/2024 14:09:59.8466] CAPWAP State: Join
[*01/11/2024 14:09:59.8769] Sending Join request to 192.168.100.40 through por
[*01/11/2024 14:10:04.7842] Sending Join request to 192.168.100.40 through por
[*01/11/2024 14:10:04.7953] Join Response from 192.168.100.40, packet size 139
[...]
[*01/11/2024 14:10:06.6919] CAPWAP State: Run
[*01/11/2024 14:10:06.8506] AP has joined controller 9124EWC
[*01/11/2024 14:10:06.8848] Flexconnect Switching to Connected Mode!
[...]
```

MAP is now joined to EWC via RAP.

AP C9115 can now get an IP address on VLAN 100 and then join the EWC:



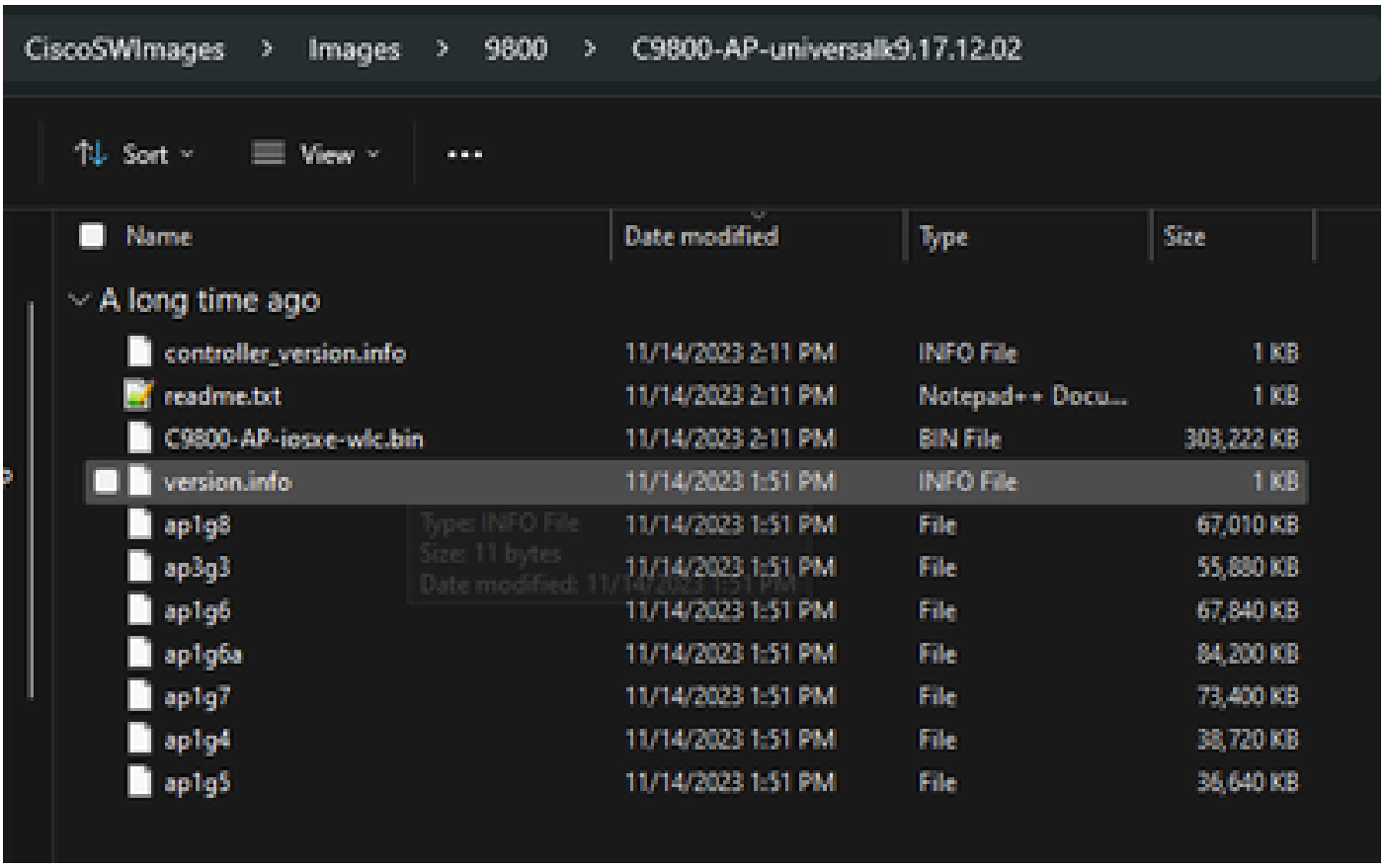
Warning: Keep in mind that VLAN 100 is the switchports trunk Native VLAN. In order for the traffic from the AP on VLAN 100 to reach the WLC on VLAN 100, the mesh link must have **VLAN Transparent enabled**. This is done in the mesh profile Ethernet Bridging section.

```
[*01/19/2024 11:40:55.0710] ethernet_port wired0, ip 192.168.100.14, netmask 2
[*01/19/2024 11:40:58.2070]
[*01/19/2024 11:40:58.2070] CAPWAP State: Init
[*01/19/2024 11:40:58.2150]
[*01/19/2024 11:40:58.2150] CAPWAP State: Discovery
[*01/19/2024 11:40:58.2400] Discovery Request sent to 192.168.100.40, discover
[*01/19/2024 11:40:58.2530] Discovery Request sent to 255.255.255.255, discove
[*01/19/2024 11:40:58.2600]
[*01/19/2024 11:40:58.2600] CAPWAP State: Discovery
[*01/19/2024 11:40:58.2670] Discovery Response from 192.168.100.40
[*01/19/2024 11:40:58.2670] Found Configured MWAR '9124EWC' (respIdx 1).
[*01/19/2024 15:13:56.0000] Started wait dtls timer (60 sec)
[*01/19/2024 15:13:56.0070]
[*01/19/2024 15:13:56.0070] CAPWAP State: DTLS Setup
[...]
[*01/19/2024 15:13:56.1660] dtls_verify_server_cert: Controller certificate ve
```

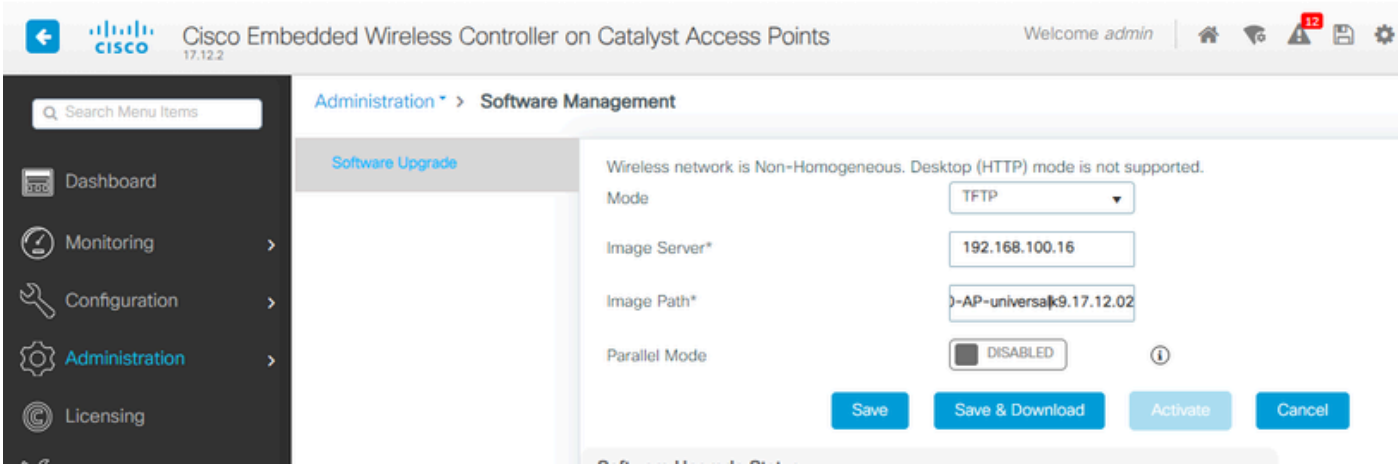
```
[*01/19/2024 15:13:56.9000] sudi99_request_check_and_load: Use HARSA SUDI cert
[*01/19/2024 15:13:57.2980]
[*01/19/2024 15:13:57.2980] CAPWAP State: Join
[*01/19/2024 15:13:57.3170] shared_setenv PART_BOOTCNT 0 &> /dev/null
[*01/19/2024 15:13:57.8620] Sending Join request to 192.168.100.40 through por
[*01/19/2024 15:14:02.8070] Sending Join request to 192.168.100.40 through por
[*01/19/2024 15:14:02.8200] Join Response from 192.168.100.40, packet size 139
[*01/19/2024 15:14:02.8200] AC accepted previous sent request with result code
[*01/19/2024 15:14:03.3700] Received wlcType 2, timer 30
[*01/19/2024 15:14:03.4440]
[*01/19/2024 15:14:03.4440] CAPWAP State: Image Data
[*01/19/2024 15:14:03.4440] AP image version 17.12.2.35 backup 17.9.4.27, Cont
[*01/19/2024 15:14:03.4440] Version is the same, do not need update.
[*01/19/2024 15:14:03.4880] status 'upgrade.sh: Script called with args:[NO_UP
[*01/19/2024 15:14:03.5330] do NO_UPGRADE, part2 is active part
[*01/19/2024 15:14:03.5520]
[*01/19/2024 15:14:03.5520] CAPWAP State: Configure
[*01/19/2024 15:14:03.5600] Telnet is not supported by AP, should not encode t
[*01/19/2024 15:14:03.6880] Radio [1] Administrative state DISABLED change to
[*01/19/2024 15:14:03.6890] Radio [0] Administrative state DISABLED change to
[*01/19/2024 15:14:03.8670]
[*01/19/2024 15:14:03.8670] CAPWAP State: Run
[*01/19/2024 15:14:03.9290] AP has joined controller 9124EWC
[*01/19/2024 15:14:03.9310] Flexconnect Switching to Connected Mode!
```

As this is an EWC AP, it contains only the AP image that corresponds to its own model (here a C9124 runs ap1g6a). When you join a different model of AP you have a Non-Homogeneous network.

In these conditions, if the AP is not on the same version, it needs to download the same version, therefore make sure you have a valid TFTP/SFTP server and location, with the AP images, configured in the **EWC** > **Administration** > **Software Management:**

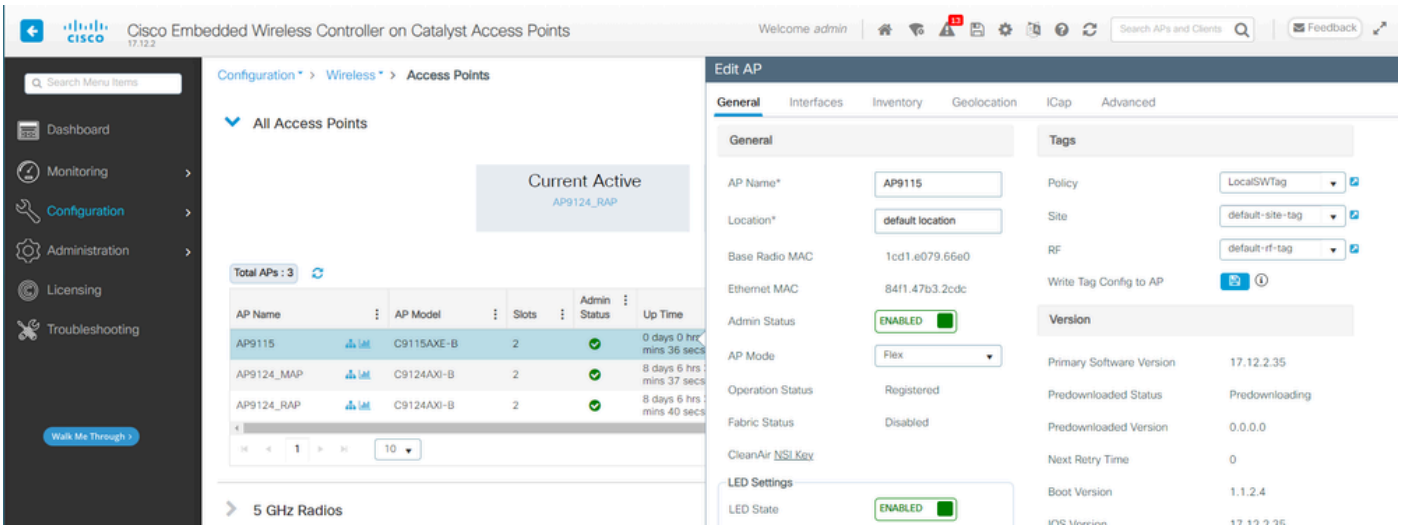


TFTP server with AP images folder

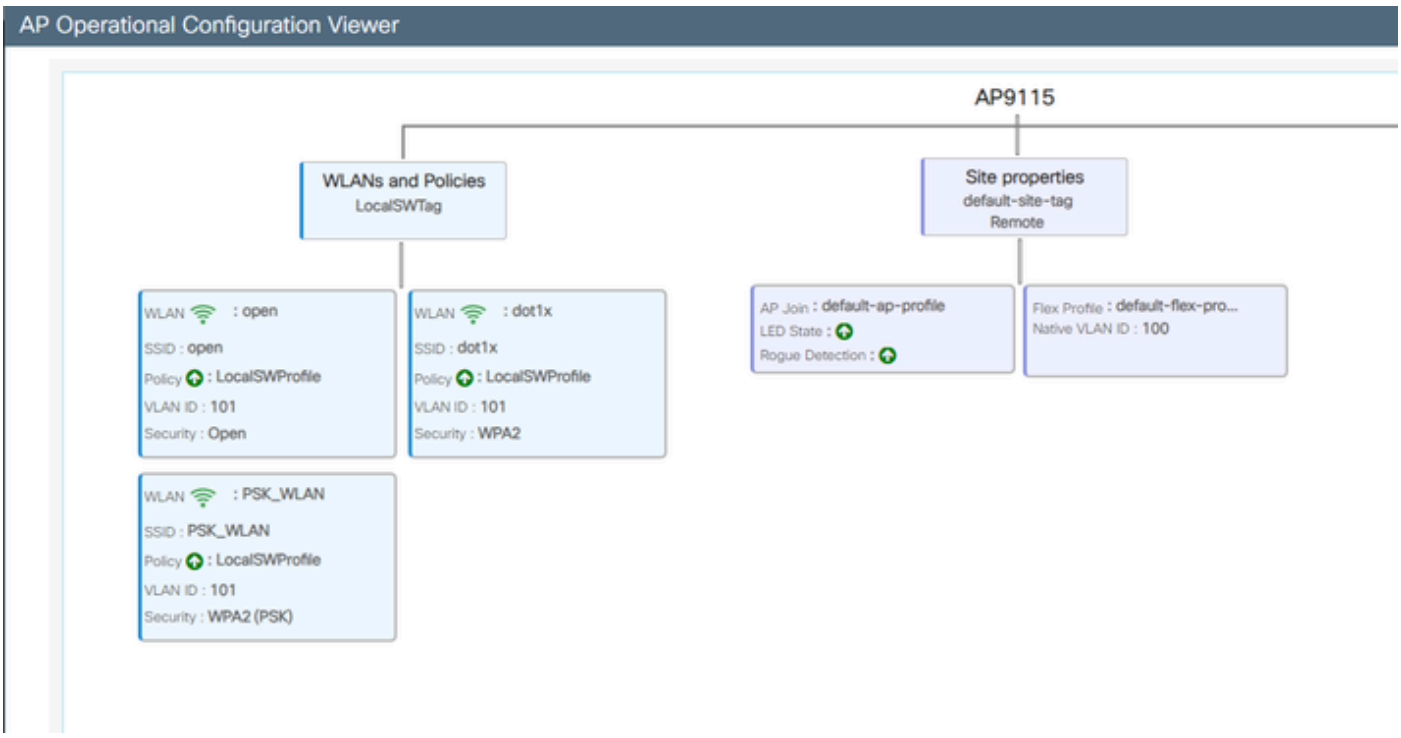


AP Images

The AP shows in the AP list and you can assign a PolicyTag:



AP List with 9115 details



AP Operational View

Verify

You can see the mesh tree via GUI which also gives the output from CLI if you use the command "*show wireless mesh ap tree*". On the GUI go to **Monitoring** > **Wireless** > **Mesh:**

Mesh AP tree

On the RAP and MAP you can verify the mesh backhaul using the command "*show mesh backhaul*":

```
AP9124_RAP#show mesh backhaul
Wired Backhaul: 0 [3C:57:31:C5:AC:2C]
idx Cost Uplink InterfaceType
0   16   TRUE   WIRED
Mesh Wired Adjacency Info
Flags: Parent(P), Child(C), Reachable(R), CapwapUp(W), BlockListed(B) Authenticated(A)
Address           Cost RawCost BlistCount Flags: P C R W B A  Reject reason
3C:57:31:C5:AC:2C 16   16      0            T/F: T F T T F T  Filtered


---------------------------------------------------------------
Wired Backhaul: 1 [3C:57:31:C5:AC:2C]
idx Cost    Uplink InterfaceType
1   Invalid FALSE  WIRED
Mesh Wired Adjacency Info
Flags: Parent(P), Child(C), Reachable(R), CapwapUp(W), BlockListed(B) Authenticated(A)
Address           Cost RawCost BlistCount Flags: P C R W B A  Reject reason
3C:57:31:C5:AC:2C 16   16      0            T/F: F F F F F F  Filtered


---------------------------------------------------------------
Radio Backhaul: 0 [4C:A6:4D:23:AE:F1]
idx State   Role   RadioState Cost    Uplink Downlink Access ShutDown ChildrenAllowed BlockChildState InterfaceType
2   INITIAL ACCESS UP         Invalid FALSE  FALSE    TRUE   FALSE    FALSE           ALLOWED         RADIO

No Radio Adjacency Exists


-------------------------------------------------------------------------------
Radio Backhaul: 1 [4C:A6:4D:23:AE:F1]
idx State Role     RadioState Cost    Uplink Downlink Access ShutDown ChildrenAllowed BlockChildState InterfaceType
3   MAINT DOWNLINK UP         Invalid FALSE  TRUE     FALSE  FALSE    TRUE            ALLOWED         RADIO
Mesh AWPP Radio adjacency info
Flags: Parent(P), Child(C), Neighbor(N), Reachable(R), CapwapUp(W),
       BlockListed(B), Authenticated(A), HTCapable(H), VHTCapable(V)
       OldParent(O), BGScan(S)
Address           Cost    RawCost LinkCost ReportedCost Snr BCount Ch Width  Bgn Flags: P O C N R W B A H V S Reject reason
4C:A6:4D:23:9D:51 Invalid Invalid 0        0            76  0      36 20 MHz -   (T/F): F F T F T F F T T T F -


-------------------------------------------------------------------------------
```

RAP show mesh backhaul

```
AP9124_MAP#show mesh backhaul
Wired Backhaul: 0 [3C:57:31:C5:A9:F8]
idx Cost    Uplink InterfaceType
0   Invalid FALSE  WIRED
Mesh Wired Adjacency Info
Flags: Parent(P), Child(C), Reachable(R), CapwapUp(W), BlockListed(B) Authenticated(A)
Address           Cost RawCost BlistCount Flags: P C R W B A  Reject reason
3C:57:31:C5:A9:F8 16   16      32           T/F: F F T F T T  Blocklisted: GW UNREACHABLE


---------------------------------------------------------------
Wired Backhaul: 1 [3C:57:31:C5:A9:F8]
idx Cost    Uplink InterfaceType
1   Invalid FALSE  WIRED
Mesh Wired Adjacency Info
Flags: Parent(P), Child(C), Reachable(R), CapwapUp(W), BlockListed(B) Authenticated(A)
Address           Cost RawCost BlistCount Flags: P C R W B A  Reject reason
3C:57:31:C5:A9:F8 16   16      0            T/F: F F F F F F  Filtered


---------------------------------------------------------------
Radio Backhaul: 0 [4C:A6:4D:23:9D:51]
idx State   Role   RadioState Cost    Uplink Downlink Access ShutDown ChildrenAllowed BlockChildState InterfaceType
2   INITIAL ACCESS UP         Invalid FALSE  FALSE    TRUE   FALSE    FALSE           ALLOWED         RADIO

No Radio Adjacency Exists


-------------------------------------------------------------------------------
Radio Backhaul: 1 [4C:A6:4D:23:9D:51]
Hops to Root: 1
idx State Role   RadioState Cost Uplink Downlink Access ShutDown ChildrenAllowed BlockChildState InterfaceType
3   MAINT UPLINK UP         217  TRUE   TRUE     FALSE  FALSE    TRUE            ALLOWED         RADIO
Mesh AWPP Radio adjacency info
Flags: Parent(P), Child(C), Neighbor(N), Reachable(R), CapwapUp(W),
       BlockListed(B), Authenticated(A), HTCapable(H), VHTCapable(V)
       OldParent(O), BGScan(S)
Address           Cost RawCost LinkCost ReportedCost Snr BCount Ch Width  Bgn Flags: P O C N R W B A H V S Reject reason
4C:A6:4D:23:AE:F1 217  272     256      16           70  0      36 20 MHz -   (T/F): T F F T T T F T T T F -


-------------------------------------------------------------------------------

AP9124_MAP#!
```

MAP show mesh backhaul

You can verify Mesh VLAN Trunking configuration on the AP side:

```
AP9124_RAP#show mesh ethernet vlan config static
Static (Stored) ethernet VLAN Configuration

Ethernet Interface: 0
Interface Mode: TRUNK
Native Vlan: 100
Allowed Vlan: 101,

Ethernet Interface: 1
Interface Mode: ACCESS
Native Vlan: 0
Allowed Vlan:
```

```
Ethernet Interface: 2
Interface Mode: ACCESS
Native Vlan: 0
Allowed Vlan:
```

Laptop2 connected on Switch2 received IP address from VLAN 101:

```
C:\Users\luke>ipconfig

Windows IP Configuration


Ethernet adapter usb_xhci:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 192.168.101.12
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.101.1
```

The Laptop1 placed on Switch1 received an IP from VLAN 101:

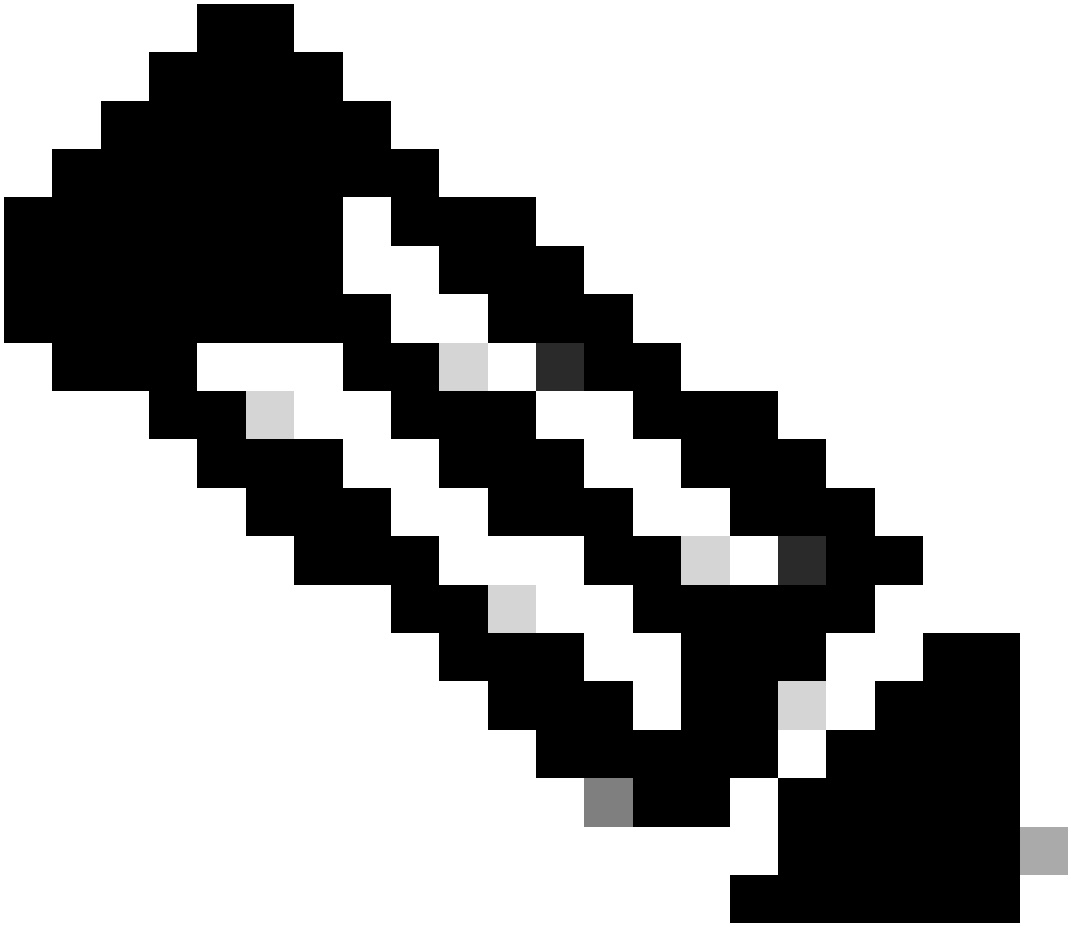
```
Ethernet adapter Ethernet 6_White:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::d1d6:f607:ff02:4217%18
IPv4 Address. . . . . . . . . . . : 192.168.101.13
Subnet Mask . . . . . . . . . . . : 255.255.255.0
Default Gateway . . . . . . . . . : 192.168.101.1
```

```
C:\Users\tantunes>ping 192.168.101.12 -i 192.168.101.13

Pinging 192.168.101.12 with 32 bytes of data:
Reply from 192.168.101.12: bytes=32 time=5ms TTL=128
Reply from 192.168.101.12: bytes=32 time=5ms TTL=128
Reply from 192.168.101.12: bytes=32 time=7ms TTL=128
Reply from 192.168.101.12: bytes=32 time=5ms TTL=128

Ping statistics for 192.168.101.12:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 5ms, Maximum = 7ms, Average = 5ms
```



Note: Please note that to test ICMP between Windows devices you need to allow ICMP on the system firewall. By default Windows devices block ICMP in the system firewall.

Another simple test to verify Ethernet bridging is having SVI for VLAN 101 on both switches and setting Switch2 SVI to DHCP. Switch2 SVI for VLAN 101 gets IP from VLAN 101 and you can ping Switch 1 VLAN 101 SVI for vlan 101 connectivity check:

<#root>

```
Switch2#show ip int br
Interface IP-Address OK? Method Status Protocol
Vlan1 unassigned YES NVRAM up down
Vlan100 192.168.100.61 YES DHCP up up
```

```
Vlan101 192.168.101.11 YES DHCP up up
```

```
GigabitEthernet0/1 unassigned YES unset up up
[...]
```

```
Switch2#
Switch2#ping 192.168.101.1 source 192.168.101.11
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.101.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.101.11
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 3/4/7 ms
Switch2#
```

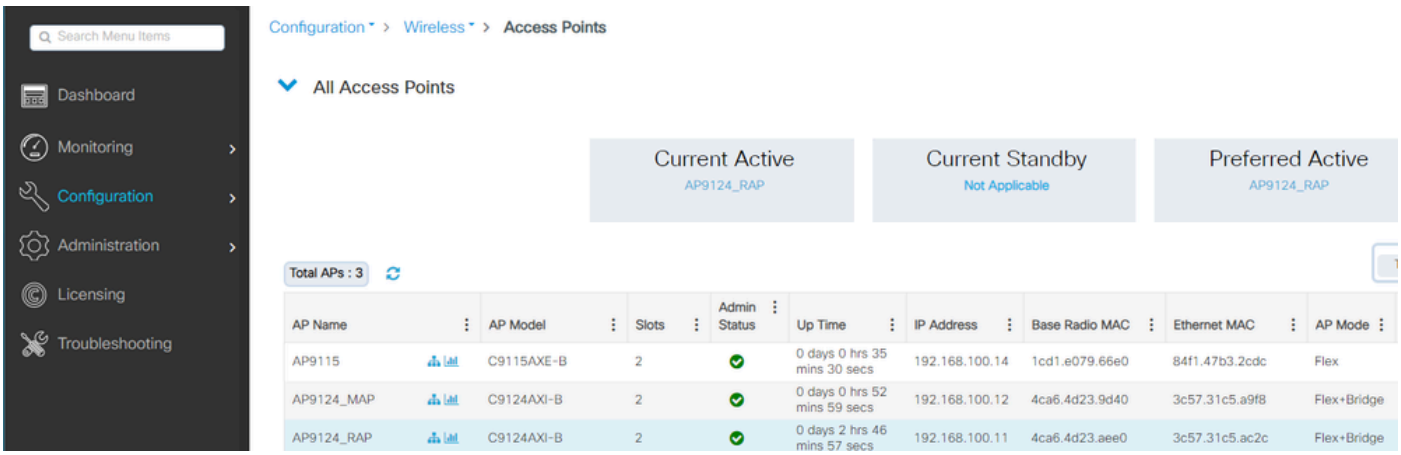
<#root>

```
Switch1#sh ip int br
Interface IP-Address OK? Method Status Protocol
Vlan1 192.168.1.11 YES NVRAM up up
Vlan100 192.168.100.1 YES NVRAM up up
```

```
Vlan101 192.168.101.1 YES NVRAM up up
```

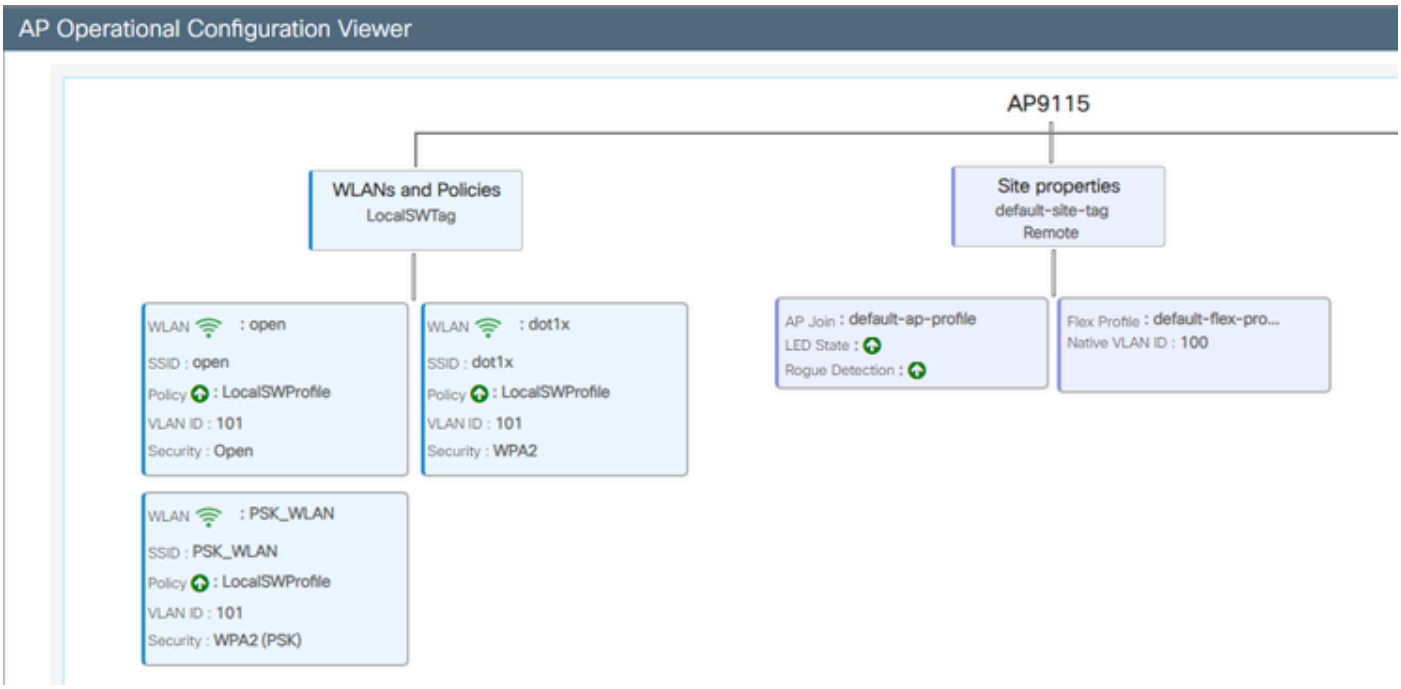
```
GigabitEthernet1/0/1 unassigned YES unset up up
[...]
Switch1#ping 192.168.101.11 source 192.168.101.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.101.11, timeout is 2 seconds:
Packet sent with a source address of 192.168.101.1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/6/8 ms
Switch1#
```

Local mode AP C9115 also joined the EWC:



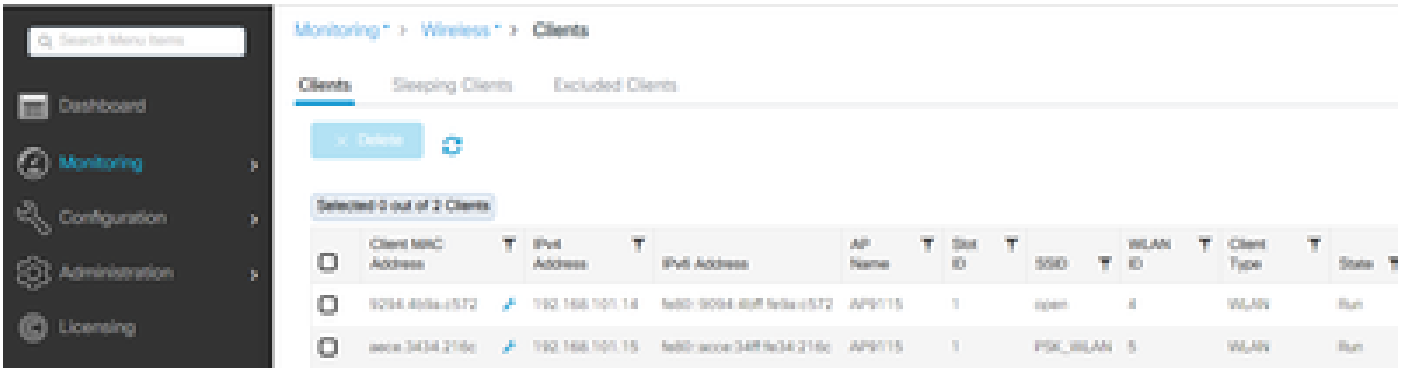
AP 9115 Joined to the EWC

Created 3 WLANs, open, PSK and dot1x mapped to a Policy Profile with VLAN 101 defined in the Access Policies:



AP9115 Operational Configuration

Wireless clients are able to connect to the WLANs:



Troubleshoot

In this section, useful commands and some tips, tricks and recommendations are presented.

Useful commands

On RAP/MAP:

```
AP9124_RAP#show mesh
  adjacency              MESH Adjacency
  backhaul               MESH backhaul
  bgscan                 MESH Background Scanning
  channel                MESH channels
  client-debug-filter    MESH client debugging filter set
  config                 MESH config paramenter
  convergence            MESH convergence info
  dfs                    MESH dfs information
  dhcp                   Flex-mesh Internal DHCP Server
  ethernet               show mesh ethernet bridging
  forwarding             MESH Forwarding
  history                MESH history of events
  least-congested-scan   Mesh least congested channel scan
  linktest               MESH linktest stats
  nat                    Flex-mesh NAT/PAT
  res                    MESH RES info
  security               MESH Security Show
  stats                  MESH stats
  status                 MESH status
  stp                    MESH daisychain STP info
  timers                 MESH Adjacency timers
```

show mesh

```
AP9124_RAP#debug mesh
  adjacency        MESH adjacency debugs
  ap-link          MESH link debugs
  bg-scan          Mesh background scanning debugs
  channel          MESH channel debugs
  clear            RESET all MESH debugs
  client           Debug mesh clients
  convergence      MESH convergence debugs
  dhcp             MESH Internal DHCP debugs
  dump-pkts        Dump mesh packets
  events           MESH events
  filter           MESH debug filter
  forward-mcast    Mesh forwarding mcast debugs
  forward-table    Mesh forwarding table debugs
  history          MESH history of events
  level            Enable different mesh debug levels
  linktest         Mesh linktest debugs
  nat              Mesh NAT debugs
  path-control     MESH path-control debugs
  port-control     MESH port-control debugs
  security         MESH security debugs
  stp              MESH daisychain STP debugs
  wpa_supplicant   Mesh WPA_SUPPLICANT debugs
  wstp             MESH WSTP debugs
```

RAP/MAP debug mesh options

On WLC:

```
9124EWC#show wireless mesh ?
  airtime-fairness           Shows Mesh AP Airtime Fairness information
  ap                         Shows mesh AP related information
  cac                        Shows Mesh AP cac related information
  config                     Show mesh configurations
  convergence                Show mesh convergence details.
  ethernet                   Show wireless mesh ethernet
  neighbor                   Show neighbors of all connected mesh Aps
  persistent-ssid-broadcast  Shows Mesh AP persistent ssid broadcast
                             information
  rrm                        Show wireless mesh rrm information
```

show wireless mesh

To debug on the WLC the best start point is to use RadioActive trace with the MAC address of the MAP/RAP.

Example 1: RAP receives adjacency from MAP and succeeds authentication

<#root>

```
AP9124_RAP#show debug
mesh:
adjacent packet debugging is enabled
event debugging is enabled
mesh linktest debug debugging is enabled
Jan 16 14:47:01 AP9124_RAP kernel: [*01/16/2024 14:47:01.9559] EVENT-MeshRadio
Jan 16 14:47:01 AP9124_RAP kernel: [*01/16/2024 14:47:01.9559] EVENT-MeshAwppA
Jan 16 14:47:01 AP9124_RAP kernel: [*01/16/2024 14:47:01.9560] EVENT-MeshAwppA
Jan 16 14:47:01 AP9124_RAP kernel: [*01/16/2024 14:47:01.9570] CLSM[4C:A6:4D:2
Jan 16 14:47:04 AP9124_RAP kernel: [*01/16/2024 14:47:04.9588] EVENT-MeshRadio
Jan 16 14:47:04 AP9124_RAP kernel: [*01/16/2024 14:47:04.9592] EVENT-MeshLink:
Jan 16 14:47:04 AP9124_RAP kernel: [*01/16/2024 14:47:04.9600] EVENT-MeshSecur
Jan 16 14:47:05 AP9124_RAP kernel: [*01/16/2024 14:47:05.1008] EVENT-MeshSecur
Jan 16 14:47:05 AP9124_RAP kernel: [*01/16/2024 14:47:05.1011] EVENT-MeshSecur
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.1172] EVENT-MeshSecur
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.1173] EVENT-MeshSecur
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.1173] EVENT-MeshSecur
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2033] EVENT-MeshSecur
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2139] EVENT-MeshSecur
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2139] EVENT-MeshSecur
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2143] EVENT-MeshSecur
```

Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2143] EVENT-MeshSecur

Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2143] EVENT-MeshLink:

Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2143] EVENT-MeshLink:

```
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2144] EVENT-MeshLink:
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2146] EVENT-MeshAwppA

Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2147] EVENT-MeshAwppA
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2151] EVENT-MeshAwppA
Jan 16 14:47:06 AP9124_RAP kernel: [*01/16/2024 14:47:06.2151] EVENT-MeshAwppA
Jan 16 14:47:19 AP9124_RAP kernel: [*01/16/2024 14:47:19.3576] EVENT-MeshRadio
Jan 16 14:47:19 AP9124_RAP kernel: [*01/16/2024 14:47:19.3577] EVENT-MeshRadio
Jan 16 14:47:19 AP9124_RAP kernel: [*01/16/2024 14:47:19.3577] EVENT-MeshRadio
```

Example 2: MAP Mac address not added to WLC or incorrectly added

<#root>

```
Jan 16 14:52:13 AP9124_RAP kernel: [*01/16/2024 14:52:13.6402] INFO-MeshRadioE
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7407] INFO-MeshRadioE
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7408] EVENT-MeshRadio
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7409] INFO-MeshRadioE
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7411] EVENT-MeshLink:
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7419] EVENT-MeshSecur
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7583] EVENT-MeshSecur
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7586] EVENT-MeshSecur
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7586] EVENT-MeshSecur

Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7620] INFO-MeshRadioE
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7620] INFO-MeshRadioE
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7621] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7621] 0x3c 0x57 0x31
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7621] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7621] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7621] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7622] 0xff 0xff 0xff
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7622] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7622] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7622] 0xaa 0xff 0x00
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7622] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7623] INFO-MeshAwppAc
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7623] 0xaa 0xff 0xaa
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7623] INFO-MeshRadioE
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7636] EVENT-MeshRadio
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7637] INFO-MeshRadioE
Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7642] EVENT-MeshLink:

Jan 16 14:52:15 AP9124_RAP kernel: [*01/16/2024 14:52:15.7642] EVENT-MeshSecur
```

Example 3: RAP looses MAP

<#root>

```
Jan 16 14:48:58 AP9124_RAP kernel: [*01/16/2024 14:48:58.9929] INFO-MeshRadioE
Jan 16 14:48:59 AP9124_RAP kernel: [*01/16/2024 14:48:59.2889] INFO-MeshAwppAc
Jan 16 14:48:59 AP9124_RAP kernel: [*01/16/2024 14:48:59.7894] INFO-MeshAwppAc
Jan 16 14:48:59 AP9124_RAP kernel: [*01/16/2024 14:48:59.9931] INFO-MeshRadioE
Jan 16 14:48:59 AP9124_RAP kernel: [*01/16/2024 14:48:59.9932] INFO-MeshRadioE
Jan 16 14:49:00 AP9124_RAP kernel: [*01/16/2024 14:49:00.2891] INFO-MeshAwppAc
Jan 16 14:49:00 AP9124_RAP kernel: [*01/16/2024 14:49:00.7891] INFO-MeshAwppAc
Jan 16 14:49:00 AP9124_RAP kernel: [*01/16/2024 14:49:00.9937] INFO-MeshRadioE
Jan 16 14:49:00 AP9124_RAP kernel: [*01/16/2024 14:49:00.9938] INFO-MeshRadioE
Jan 16 14:49:01 AP9124_RAP kernel: [*01/16/2024 14:49:01.2891] INFO-MeshAwppAc

Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5480] EVENT-MeshAwppA


Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5481] EVENT-MeshRadic

Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5481] EVENT-MeshRadic



Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5488] EVENT-MeshRadic


Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5489] INFO-MeshRadioE

Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5501] EVENT-MeshRadic



Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5501] EVENT-MeshAdj[1


Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5502] EVENT-MeshRadic
Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5511] EVENT-MeshLink
Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5512] EVENT-MeshSecur
Jan 16 14:49:25 AP9124_RAP kernel: [*01/16/2024 14:49:25.5513] EVENT-MeshLink
```

Tips, Tricks and Recommendations

- By upgrading the MAP and RAP to the same image version over the wire, we are avoiding image download going over the air (which can be problematic in "dirty" RF environments).
- It is highly recommended to test out the setup in a controlled environment before deploying it on site.
- If testing Ethernet bridging with windows laptops on each side, please note that to test ICMP between Windows devices you need to allow ICMP on the system firewall. By default Windows devices block ICMP in the system firewall.
- If APs with external antennas are being used, make sure to consult the deployment guide to check which antennas are compatible and which port they are supposed to be plugged in.
- In order to bridge the traffic from different VLANs over the mesh link, VLAN Transparent feature needs to be disabled.
- Consider having a syslog server local to the APs, as it can provide debug information otherwise only available with a console connection.

References

Cisco Embedded Wireless Controller on Catalyst Access Points Data Sheet

Cisco Embedded Wireless Controller on Catalyst Access Points (EWC) White Paper

Configuring Point-to-Point Mesh Link with Ethernet Bridging on Mobility Express APs