Troubleshoot a Lightweight Access Point Not Joining a Wireless LAN Controller

Contents

Introduction

This document gives an overview of the Wireless LAN Controller (WLC) Discovery and Join Process. This document also provides information on some of the issues why a Lightweight Access Point (LAP) fails to join a WLC and how to troubleshoot the issues.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Basic knowledge of the configuration of LAPs and Cisco WLCs
- Basic knowledge of Lightweight Access Point Protocol (CAPWAP)

Conventions

Refer to Cisco Technical Tips Conventions for more information on document conventions.

Overview of the Wireless LAN Controller (WLC) Discovery and Join Process

In a Cisco Unified Wireless network, the LAPs must first discover and join a WLC before they can service wireless clients.

However, this presents a question: how did the LAPs find the management IP address of the controller when it is on a different subnet?

If you do not tell the LAP where the controller is via DHCP option 43, DNS resolution of "Cisco-capwap-controller.local_domain", or statically configure it, the LAP does not know where in the network to find the management interface of the controller.

In addition to these methods, the LAP does automatically look on the local subnet for controllers with a 255.255.255.255 local broadcast. Also, the LAP remembers the management IP address of any controller it joins across reboots. Therefore, if you put the LAP first on the local subnet of the management interface, it will find the controller's management interface and remember the address. This is called priming. This does not help find the controller if you replace a LAP later on. Therefore, Cisco recommends using the DHCP option 43 or DNS methods.

The LAPs always connect to the management interface address of the controller first with a discovery request. The controller then tells the LAP the Layer 3 AP-manager interface (which can also be the management by default)IP address so the LAP can send a join request to the AP-manager interface next.

The AP goes through this process on startup:

1. The LAP boots and DHCPs an IP address if it was not previously assigned a static IP address.
2. The LAP sends discovery requests to controllers through the various discovery algorithms and builds a controller list. Essentially, the LAP learns as many management interface addresses for the controller list as possible via:DHCP option 43 (good for global companies where offices and controllers are on different continents)DNS entry for `cisco-capwap-controller` (good for local businesses - can also be used to find where brand new APs join)**Note:** If you use CAPWAP, make sure that there is a DNS entry for `cisco-capwap-controller`.Management IP addresses of controllers the LAP remembers previouslyA Layer 3 broadcast on the subnetStatically configured informationControllers present in the mobility group of the WLC the AP last joinedFrom this list, the easiest method to use for deployment is to have the LAPs on the same subnet as the management interface of the controller and allow the LAPs Layer 3 broadcast to find the controller. This method should be used for companies that have a small network and do not own a local DNS server.The next easiest method of deployment is to use a DNS entry with DHCP. You can have multiple entries of the same DNS name. This allows the LAP to discover multiple controllers. This method should be used by companies that have all of their controllers in a single location and own a local DNS server. Or, if the company has multiple DNS suffixes and the controllers are segregated by suffix.DHCP option 43 is used by large companies to localize the information via the DHCP. This method is used by large enterprises that have a single DNS suffix. For

example, Cisco owns buildings in Europe, Australia, and the United States. In order to ensure that the LAPs only join controllers locally, Cisco cannot use a DNS entry and must use DHCP option 43 information to tell the LAPs what the management IP address of their local controller is.Finally, static configuration is used for a network that does not have a DHCP server.You can statically configure the information necessary to join a controller via the console port and the APs CLI. For information on how to statically configure controller information using the AP CLI, use the following command:

```
AP#capwap ap primary-base <WLCName> <WLCIP>
```

For information on configuring DHCP option 43 on a DHCP server, refer to the DHCP option 43 configuration example

3. Send a discovery request to every controller on the list and wait for the controller's discovery reply which contains the system name, AP-manager IP addresses, the number of APs already attached to each AP-manager interface, and overall excess capacity for the controller.
4. Look at the controller list and send a join request to a controller in this order (only if the AP received a discovery reply from it):Primary Controller system name (previously configured on LAP)Secondary Controller system name (previously configured on LAP)Tertiary Controller system name (previously configured on LAP)Master controller (if the LAP has not been previously configured with any Primary, Secondary, or Tertiary controller names. Used to always know which controller brand new LAPs join)If none of the above are seen, load balance across controllers using the excess capacity value in the discovery response.If two controllers have the same excess capacity, then send the join request to the first controller that responded to the discovery request with a discovery response. If a single controller has multiple AP-managers on multiple interfaces, choose the AP-manager interface with the least number of APs.The controller will respond to all discovery requests without checking certificates or AP credentials. However, join requests must have a valid certificate in order to get a join response from the controller. If the LAP does not receive a join response from its choice, the LAP will try the next controller in the list unless the controller is a configured controller (Primary/Secondary/Tertiary).
5. When it receives the join reply, the AP checks to make sure it has the same image as that of the controller. If not, the AP downloads the image from the controller and reboots to load the new image and starts the process all over again from step 1.
6. If it has the same software image, it asks for the configuration from the controller and moves into the registered state on the controller.After you download the configuration, the AP might reload again to apply the new configuration. Therefore, an extra reload can occur and is a normal behavior.

Debug from the Controller

There are a few **debug** commands on the controller you can use in order to see this entire process on the CLI .

- **debug capwap events enable**â€”Shows discovery packets and join packets.
- **debug capwap packet enable**â€” Shows packet level information of the discovery and join packets.
- **debug pm pki enable**â€”Shows certificate validation process.
- **debug disable-all**â€”Turns off debugs.

With a terminal application that can capture output to a log file, console in or secure shell

(SSH)/Telnet to your controller, and enter these commands:

```
config session timeout 120
config serial timeout 120
show run-config     (and spacebar thru to collect all)

debug mac addr <ap-radio-mac-address>     (in xx:xx:xx:xx:xx format)
debug client <ap-mac-address>
debug capwap events enable
debug capwap errors enable
debug pm pki enable
```

After capturing the debugs, use the **debug disable-all** command to turn off all debugs.

The next sections show the output of these **debug** commands when the LAP registers with the controller.

debug capwap events enable

This command provides information on the CAPWAP events and errors that occur during the CAPWAP discovery and join process.

This is the **debug capwap events enable** command output for a LAP which has the same image as that of the WLC:

Note: Some lines of the output has been moved to the second line due to space constraints.

```
debug capwap events enable

*spamApTask7: Jun 16 12:37:36.038: 00:62:ec:60:ea:20 Discovery Request from
172.16.17.99:46317
!--- CAPWAP discovery request sent to the WLC by the LAP. *spamApTask7: Jun 16
12:37:36.039: 00:62:ec:60:ea:20 Discovery Response sent to 172.16.17.99 port 46317 !-
-- WLC responds to the discovery request from the LAP. *spamApTask7: Jun 16
12:38:43.469: 00:62:ec:60:ea:20 Join Request from 172.16.17.99:46317 !--- LAP sends a
join request to the WLC.
 *spamApTask7: Jun 16 12:38:33.039: 00:62:ec:60:ea:20 Join Priority Processing status
= 0, Incoming Ap's Priority 1, MaxLrads = 75, joined Aps =0
*spamApTask7: Jun 16 12:38:43.469: 00:62:ec:60:ea:20 Join Request from
172.16.17.99:46317

*spamApTask7: Jun 16 12:38:43.472: 00:62:ec:60:ea:20 Join Version: = 134256640

*spamApTask7: Jun 16 12:38:43.473: 00:62:ec:60:ea:20 apType = 46 apModel: AIR-
CAP2702I-E-K9

*spamApTask7: Jun 16 12:38:43.473: 00:62:ec:60:ea:20 Join resp: CAPWAP Maximum Msg
element len = 90

*spamApTask7: Jun 16 12:38:43.473: 00:62:ec:60:ea:20 Join Response sent to
172.16.17.99:46317
*spamApTask7: Jun 16 12:38:43.473: 00:62:ec:60:ea:20 CAPWAP State: Join
!--- WLC responds with a join reply to the LAP.
*spamApTask7: Jun 16 12:38:43.964: 00:62:ec:60:ea:20 Configuration Status from
172.16.17.99:46317
*spamApTask7: Jun 16 12:38:43.964: 00:62:ec:60:ea:20 CAPWAP State: Configure !--- LAP
```

```
requests for the configuration information from the WLC.

*spamApTask7: Jun 16 12:38:43.964: 00:62:ec:60:ea:20 Updating IP info for AP
00:62:ec:60:ea:20 -- static 0, 172.16.17.99/255.255.254.0, gtw 172.16.16.1
*spamApTask7: Jun 16 12:38:43.964: 00:62:ec:60:ea:20 Updating IP 172.16.17.99 ===>
172.16.17.99 for AP 00:62:ec:60:ea:20
*spamApTask7: Jun 16 12:38:43.964: 00:62:ec:60:ea:20 Running
spamDecodeVlanProfMapPayload for00:62:ec:60:ea:20
*spamApTask7: Jun 16 12:38:43.964: 00:62:ec:60:ea:20 Setting MTU to 1485
*spamApTask7: Jun 16 12:38:44.019: 00:62:ec:60:ea:20 Configuration Status Response
sent to 172:16:17:99

 !--- WLC responds by providing all the necessary configuration information to the
LAP. *spamApTask7: Jun 16 12:38:46.882: 00:62:ec:60:ea:20 Change State Event Request
from 172.16.17.99:46317

*spamApTask7: Jun 16 12:38:46.883: 00:62:ec:60:ea:20 Radio state change for slot: 0
state: 2 cause: 0 detail cause: 69
*spamApTask7: Jun 16 12:38:46.883: 00:62:ec:60:ea:20 Change State Event Response sent
to 172.16.17.99:46317
.
.
.
.

*spamApTask7: Jun 16 12:38:46.883: 00:62:ec:60:ea:20 CAPWAP State: Run

*spamApTask7: Jun 16 12:38:46.883: 00:62:ec:60:ea:20 Sending the remaining config to
AP 172.16.17.99:46317!
.
.
.
. !--- LAP is up and ready to service wireless clients. *spamReceiveTask: Jun 16
12:38:46.897: 00:62:ec:60:ea:20 Configuration update request for RrmInterferenceCtrl
payload sent to 172:16:17:99

*spamReceiveTask: Jun 16 12:38:46.897: 00:62:ec:60:ea:20 Configuration update request
for RrmNeighbourCtrl payload sent to 172.16.17.99

*spamReceiveTask: Jun 16 12:38:46.897: 00:62:ec:60:ea:20 Configuration update request
for RrmReceiveCtrl payload sent to 172:16:17:99

*spamReceiveTask: Jun 16 12:38:46.897: 00:62:ec:60:ea:20 Configuration update request
for CcxRmMeas payload sent to 172.16.17.99
!--- WLC sends all the RRM and other configuration parameters to the LAP.
```

As mentioned in the previous section, once a LAP registers with the WLC, it checks to see if it has the same image as the controller. If the images on the LAP and the WLC are different, the LAPs download the new image from the WLC first. If the LAP has the same image, it continues to download the configuration and other parameters from the WLC.

You will see these messages in the **debug capwap events enable** command output if the LAP downloads an image from the controller as a part of the registration process:

```
*spamApTask6: Jun 17 14:23:28.677: 00:62:ec:60:ea:20 Sending image data block of
length 1324 and msgLength = 1327

*spamApTask6: Jun 17 14:23:28.677: 00:62:ec:60:ea:20 Image Data Request sent to
172.16.17.201:46318
```

```
*spamApTask6: Jun 17 14:23:28.693: 00:62:ec:60:ea:20 Image data Response from
172.16.17.201:46318
```

Once the image download is complete, the LAP will reboot and run the discovery and join the algorithm again.

debug pm pki enable

As a part of the join process, the WLC authenticates each LAP by verifying that its certificate is valid.

When the AP sends the CAPWAP Join Request to the WLC, it embeds its X.509 certificate in the CAPWAP message. The AP also generates a random session ID that is also included in the CAPWAP Join Request. When the WLC receives the CAPWAP Join Request, it validates the signature of the X.509 certificate using the AP's public key and checks that the certificate was issued by a trusted certificate authority.

It also looks at the starting date and time for the AP certificate's validity interval and compares that date and time to its own date and time (hence the controller's clock needs to be set to close to the current date and time). If the X.509 certificate is validated, the WLC generates a random AES encryption key. The WLC plumbs the AES key into its crypto engine so that it can encrypt and decrypt future CAPWAP Control Messages exchanged with the AP. Note that data packets are sent in the clear in the CAPWAP tunnel between the LAP and the controller.

The **debug pm pki enable** command shows the certification validation process that occurs during the join phase on the controller. The **debug pm pki enable** command will also display the AP hash key during the join process if the AP has a self-signed certificate (SSC) created by the LWAPP conversion program. If the AP has a Manufactured Installed Certificate (MIC), you will not see a hash key.

Note: All APs manufactured after June 2006 have a MIC.

Here is the output of the **debug pm pki enable** command when the LAP with a MIC joins the controller:

Note: Some lines of the output has been moved to the second line due to space constraints.

```
*spamApTask4: Mar 20 11:05:15.687: [SA] OpenSSL Get Issuer Handles: locking ca cert
table

*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: x509 subject_name
/C=US/ST=California/L=San Jose/O=Cisco Systems/
CN=AP3G2-1005cae83a42/emailAddress=support@cisco.com
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: issuer_name
/O=Cisco Systems/CN=Cisco Manufacturing CA
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: CN AP3G2-
1005cae83a42
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: issuerCertCN
Cisco Manufacturing CA
*spamApTask4: Mar 20 11:05:15.688: [SA] GetMac: MAC: 1005.cae8.3a42
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: openssl Mac
Address in subject is 10:05:ca:e8:3a:42
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: CN AP3G2-
```

1005cae83a42
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: issuerCertCN Cisco Manufacturing CA
*spamApTask4: Mar 20 11:05:15.688: [SA] GetMac: MAC: 1005.cae8.3a42
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: openssl Mac Address in subject is 10:05:ca:e8:3a:42
*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: Cert Name in subject is AP3G2-1005cae83a42

*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: Extracted cert issuer from subject name.

*spamApTask4: Mar 20 11:05:15.688: [SA] OpenSSL Get Issuer Handles: **Cert is issued by Cisco Systems.**

*spamApTask4: Mar 20 11:05:15.688: [SA] Retrieving x509 cert for CertName cscoDefaultMfgCaCert
*spamApTask4: Mar 20 11:05:15.688: [SA] sshpmGetCID: called to evaluate <cscoDefaultMfgCaCert>
*spamApTask4: Mar 20 11:05:15.688: [SA] sshpmGetCID: Found matching CA cert cscoDefaultMfgCaCert in row 5
*spamApTask4: Mar 20 11:05:15.688: [SA] Found CID 260e5e69 for certname cscoDefaultMfgCaCert

*spamApTask4: Mar 20 11:05:15.688: [SA] CACertTable: Found matching CID cscoDefaultMfgCaCert in row 5 x509 0x2cc7c274

*spamApTask4: Mar 20 11:05:15.688: [SA] Retrieving x509 cert for CertName cscoDefaultNewRootCaCert
*spamApTask4: Mar 20 11:05:15.688: [SA] sshpmGetCID: called to evaluate <cscoDefaultNewRootCaCert>
*spamApTask4: Mar 20 11:05:15.688: [SA] sshpmGetCID: Found matching CA cert cscoDefaultNewRootCaCert in row 4

*spamApTask4: Mar 20 11:05:15.688: [SA] Found CID 28d7044e for certname cscoDefaultNewRootCaCert
*spamApTask4: Mar 20 11:05:15.688: [SA] CACertTable: Found matching CID cscoDefaultNewRootCaCert in row 4 x509 0x2cc7c490
*spamApTask4: Mar 20 11:05:15.691: [SA] Verify User Certificate: X509 Cert Verification return code: 1
*spamApTask4: Mar 20 11:05:15.691: [SA] Verify User Certificate: X509 Cert Verification result text: ok
*spamApTask4: Mar 20 11:05:15.691: [SA] sshpmGetCID: called to evaluate <cscoDefaultMfgCaCert>
*spamApTask4: Mar 20 11:05:15.691: [SA] sshpmGetCID: Found matching CA cert cscoDefaultMfgCaCert in row 5

*spamApTask4: Mar 20 11:05:15.691: [SA] **Verify User Certificate: OPENSSL X509_Verify: AP Cert Verfied Using >cscoDefaultMfgCaCert<**

*spamApTask4: Mar 20 11:05:15.691: [SA] OpenSSL Get Issuer Handles: **Check cert validity times (allow expired NO)**
*spamApTask4: Mar 20 11:05:15.691: [SA] sshpmGetCID: called to evaluate <cscoDefaultIdCert>
*spamApTask4: Mar 20 11:05:15.691: [SA] sshpmGetCID: Found matching ID cert cscoDefaultIdCert in row 2
*spamApTask4: Mar 20 11:05:15.691: [SA] sshpmFreePublicKeyHandle: called with 0x1b0b9380

*spamApTask4: Mar 20 11:05:15.691: [SA] sshpmFreePublicKeyHandle: **freeing public key**

Debug from the AP

If the controller debugs do not indicate a join request, you can debug the process from the AP as long as the AP has a console port. You can see the AP boot up process with these commands, but you must first get into enable mode (default password is Cisco):

- **debug dhcp detail :** Shows DHCP option 43 information.
- **debug ip udp** :Shows all UDP packets recieved and transmitted by the AP
- **debug capwap client event :** Shows capwap events for the AP.
- **debug capwap client error:** Shows capwap errors for AP.
- **debug dtls client event:** Shows DTLS events for the AP.
- **debug dtls error enable:** Shows DTLS errors for the AP.
- **undebug all**: Disables debugs on the AP.

Here is an example of the output from the **debug capwap** commands. This partial output gives an idea of the packets that are sent by the AP during the boot process to discover and join a controller.

```
AP can discover the WLC via one of the following options :

!--- AP discovers the WLC via option 43

*Jun 28 08:43:05.839: %CAPWAP-5-DHCP_OPTION_43: Controller address 10.63.84.78
obtained through DHCP
*Jun 28 08:43:15.963: %CAPWAP-3-EVENTLOG: Discovery Request sent to 10.63.84.78 with
discovery type set to 2

!--- capwap Discovery Request using the statically configured controller information.

*Jun 28 08:43:15.963: %CAPWAP-3-EVENTLOG: Discovery Request sent to 10.63.84.32 with
discovery type set to 1

!--- Capwap Discovery Request sent using subnet broadcast.

*Jun 28 08:43:15.963: %CAPWAP-3-EVENTLOG: Discovery Request sent to 255.255.255.255
with discovery type set to 0

!--- capwap Join Request sent to AP-Manager interface on DHCP discovered controller.

*Jun 28 08:40:29.031: %CAPWAP-5-SENDJOIN: sending Join Request to 10.63.84.78
```

LAP Does Not Join the Controller, Why?

Check the Basics First

- Can the AP and the WLC communicate?
- Make sure the AP is getting an address from DHCP (check the DHCP server leases for the AP's MAC address).
- Try pinging the AP from the controller.
- Check if the STP configuration on the switch is done right so that packets to the VLANs are

not blocked.
- If pings are successful, ensure the AP has at least one method by which to discovery at least a single WLC Console or telnet/ssh into the controller to run debugs.
- Each time the AP reboots, it initiates the WLC discovery sequence and tries to locate the AP. Reboot the AP and check if it joins the WLC.

Here are some of the commonly seen issues due to which the LAPs do not join the WLC.

Field Notice: FN - 63942

https://www.cisco.com/c/en/us/support/docs/field-notices/639/fn63942.html

Problem 1: The controller time is outside the certificate validity interval

Complete these steps in order to troubleshoot this problem:

1. Issue **debug dtls client error + debug dtls client event** commands on the AP :

```
*Jun 28 09:21:25.011: DTLS_CLIENT_EVENT: dtls_process_Certificate: Processing...Peer
certificate verification failed 001A
*Jun 28 09:21:25.031: DTLS_CLIENT_ERROR:
../capwap/base_capwap/capwap/base_capwap_wtp_dtls.c:509 Certificate verified failed!
*Jun 28 09:21:25.031: DTLS_CLIENT_EVENT: dtls_send_Alert: Sending FATAL : Bad certificate
Alert
*Jun 28 09:21:25.031: DTLS_CLIENT_EVENT: dtls_client_process_record: Error processing
Certificate.
*Jun 28 09:21:25.031: DTLS_CLIENT_EVENT: dtls_disconnect: Disconnecting DTLS connection
0x8AE7FD0
*Jun 28 09:21:25.031: DTLS_CLIENT_EVENT: dtls_free_connection: Free Called... for
Connection 0x8AE7FD0
*Jun 28 09:21:25.031: DTLS_CLIENT_EVENT: dtls_send_Alert: Sending FATAL : Close notify
Alert
```

   This information clearly shows that the controller time is outside the certificate validity interval of the AP. Therefore, the AP cannot register with the controller. Certificates installed in the AP have a predefined validity interval. The controller time should be set in such a way that it is within the certificate validity interval of the AP's certificate.
2. Issue the **show time** command from the controller CLI in order to verify that the date and time set on your controller falls within this validity interval. If the controller time is higher or lower than this certificate validity interval, then change the controller time to fall within this interval.**Note:** If the time is not set correctly on the controller, choose **Commands > Set Time** in the controller GUI mode, or issue the **config time** command in the controller CLI in order to set the controller time.
3. On APs with CLI access, verify the certificates with the **show crypto ca certificates** command from the AP CLI.This command allows you to verify the certificate validity interval set in the AP. This is an example:

```
AP00c1.649a.be5c#show crypto ca cert
.....................................
.....................................
```

```
.....................................
.....................................
Certificate
Status: Available
Certificate Serial Number (hex): 7D1125A900000002A61A
Certificate Usage: General Purpose
Issuer:
cn=Cisco Manufacturing CA SHA2
o=Cisco
Subject:
Name: AP1G2-00c1649abe5c
e=support@cisco.com
cn=AP1G2-00c1649abe5c
o=Cisco Systems
l=San Jose
st=California
c=US
CRL Distribution Points:
http://www.cisco.com/security/pki/crl/cmca2.crl
Validity Date:
start date: 01:05:37 UTC Mar 24 2016
end date: 01:15:37 UTC Mar 24 2026
Associated Trustpoints: Cisco_IOS_M2_MIC_cert
Storage:
..............................
..............................
..................................
```

The entire output is not listed as there can be many validity intervals associated with the output of this command. You need to consider only the validity interval specified by the Associated Trustpoint: Cisco_IOS_MIC_cert with the relevant AP name in the name field. In this example ouput, it is `Name: C1200-001563e50c7e`. This is the actual certificate validity interval to be considered.

4. please refer to CSCuq19142 LAP/WLC MIC or SSC lifetime expiration causes DTLS failure : https://bst.cloudapps.cisco.com/bugsearch/bug/CSCuq19142

Problem 2: Mismatch in Regulatory domain

You see this message in the **debug capwap events enable** command output:

*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 AP 00:cc:fc:13:e5:e0: Country code is not configured(BE).
*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 AP 00:cc:fc:13:e5:e0: Country code is not configured(BE).
*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 Setting MTU to1485
*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 AP 00:cc:fc:13:e5:e0: **Country code is not configured(BE).**
*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 Regulatory Domain Mismatch: AP 00:cc:fc:13:e5:e0 not allowed to join. Allowed domains: 802.11bg:-A
*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 Finding DTLS connection to delete for AP (192:168:47:29/60390)
*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 Disconnecting DTLS Capwap-Ctrl session 0x1d4df620 for AP (192:168:47:29/60390). Notify(true)
*spamApTask7: Jun 28 11:56:49.177: 00:cc:fc:13:e5:e0 acDtlsPlumbControlPlaneKeys: lrad:192.168.47.29(60390) mwar:10.63.84.78(5246)

WLC msglog will show the following :

```
*spamApTask5: Jun 28 11:52:06.536: %CAPWAP-3-DTLS_CLOSED_ERR: capwap_ac_sm.c:7095
00:cc:fc:13:e5:e0: DTLS connection
closed forAP 192:168:47:28 (60389), Controller: 10:63:84:78 (5246) Regulatory Domain
Mismatch
```

The message clearly indicates that there is a mismatch in the regulatory domain of the LAP and the WLC. The WLC supports multiple regulatory domains but each regulatory domain must be selected before an AP can join from that domain. For example, the WLC that uses regulatory domain -A can only be used with APs that use regulatory domain -A (and so on). When you purchase APs , ensure that they share the same regulatory domain. Only then can the APs register with the WLC.

Note: Both 802.1b/g and 802.11a radios must be in the same regulatory domain for a single AP.

Problem 3: AP authorization list enabled on the WLC; LAP not in the authorization list

In such cases, you will see this message on the controller in the output of the **debug capwap events enable** command:

```
Wed Sep 12 17:42:39 2007: 00:0b:85:51:5a:e0 Received CAPWAP DISCOVERY REQUEST
from AP 00:0b:85:51:5a:e0 to 00:0b:85:33:52:80 on port '1'
Wed Sep 12 17:42:39 2007: 00:0b:85:51:5a:e0 Successful transmission of
CAPWAP Discovery-Response to AP 00:0b:85:51:5a:e0 on Port 1
Wed Sep 12 17:42:39 2007: 00:0b:85:51:5a:e0 Received LWAPP DISCOVERY REQUEST
from AP 00:0b:85:51:5a:e0 to ff:ff:ff:ff:ff:ff on port '1'
Wed Sep 12 17:42:39 2007: 00:0b:85:51:5a:e0 Successful transmission of
CAPWAP Discovery-Response to AP 00:0b:85:51:5a:e0 on Port 1
Wed Sep 12 17:42:50 2007: 00:0b:85:51:5a:e0 Received CAPWAP JOIN REQUEST
from AP 00:0b:85:51:5a:e0 to 00:0b:85:33:52:80 on port '1'
Wed Sep 12 17:42:50 2007: 00:0b:85:51:5a:e0 AP ap:51:5a:e0: txNonce
00:0B:85:33:52:80
rxNonce  00:0B:85:51:5A:E0
Wed Sep 12 17:42:50 2007: 00:0b:85:51:5a:e0 CAPWAP Join-Request MTU path from
AP 00:0b:85:51:5a:e0 is 1500, remote debug mode is 0
Wed Sep 12 17:42:50 2007: spamRadiusProcessResponse: AP Authorization failure
for 00:0b:85:51:5a:e0
```

If you are using a LAP that has a console port, you will see this message when you issue the **debug capwap client error** command:

```
AP001d.a245.a2fb#
*Mar  1 00:00:52.267: LWAPP_CLIENT_ERROR_DEBUG: spamHandleJoinTimer: Did not receive
the
Join response
*Mar  1 00:00:52.267: LWAPP_CLIENT_ERROR_DEBUG: No more AP manager IP addresses
remain.
```

This again is a clear indication that the LAP is not part of the AP authorization list on the controller.

You can view the status of the AP authorization list using this command:

```
(Cisco Controller) >show auth-list

Authorize APs against AAA ..... enabled
Allow APs with Self-signed Certificate (SSC) .... disabled
```

In order to add an LAP to the AP authorization list, use the **config auth-list add mic *<AP MAC Address>*** command. For more information on how to configure LAP authorization, refer to Lightweight Access Point (LAP) Authorization in a Cisco Unified Wireless Network Configuration Example.

Problem 4: There is a certificate or public key corruption on the AP

The LAP does not join a controller because of a certificate issue.

Issue the **debug capwap errors enable** and **debug pm pki enable** commands. You see messages that indicate the certificates or keys that are corrupted.

Note: Some lines of the output have been moved to second lines due to space constraints.

```
Tue Aug 12 17:36:09 2008: 00:0f:24:a9:52:e0
 CAPWAP Join Request does not include valid certificate in CERTIFICATE_PAYLOAD
 from AP 00:0f:24:a9:52:e0.
Tue Aug 12 17:36:09 2008: 00:0f:24:a9:52:e0
 Deleting and removing AP 00:0f:24:a9:52:e0 from fast path
Tue Aug 12 17:36:09 2008: 00:0f:24:a9:52:e0 Unable to free public key for AP
```

Use one of these two options in order to resolve the problem:

- MIC AP -Request a return materials authorization (RMA).
- LSC AP- Re provision your LSC certificate

Problem 5: Controller receives AP discovery message on wrong VLAN (you see the discovery message debug, but not response)

You see this message in the **debug capwap events enable** command output:

```
Received a Discovery Request with subnet broadcast with wrong AP IP address
(A.B.C.D)!
```

This message means that the controller received a discovery request via a broadcast IP address that has a source IP address which is not in any configured subnets on the controller. This also means the controller is dropping the packet.

The problem is that the AP is not sending the discovery request to the management IP address. The controller is reporting a broadcast discovery request from a VLAN that is not configured on the controller. This typically occurs when the customer trunks allowed VLANs instead of restricting them to wireless VLANs.

Complete these steps in order to resolve this problem:

1. If the controller is on another subnet, the APs must be **primed** for the controller IP address, or the APs must receive the controllers IP address using any one of the discovery methods.
2. The switch is configured to allow some VLANs that are not on the controller. Restrict the allowed VLANs on the trunks.

Problem 6: AP Not Able to Join the WLC, Firewall Blocking Necessary Ports

If a firewall is used in the enterprise network, ensure that the following ports are enabled on the firewall for the LAP to be able to join and communicate with the controller.

You must enable these ports:

- Enable these UDP ports for CAPWAP traffic:Data - 5247Control - 5246
- Enable these UDP ports for mobility traffic:16666 - 1666616667 - 16667
- Enable UDP ports 5246 and 5247 for CAPWAP traffic.
- TCP 161 and 162 for SNMP (for the Wireless Control System [WCS])

These ports are optional (depending on your requirements):

- UDP 69 for TFTP
- TCP 80 and/or 443 for HTTP or HTTPS for GUI access
- TCP 23 and/or 22 for Telnet or SSH for CLI access

Problem 7: Duplicate IP address in the network

This is another common issue that is seen when the AP tries to join the WLC. You might see this error message when the AP tries to join the controller.

```
No more AP manager IP addresses remain
```

One of the reasons for this error message is when there is a duplicate IP address on the network that matches the AP manager IP address. In such a case, the LAP keeps power cycling and cannot join the controller.

The debugs will show that the WLC receives LWAPP discovery requests from the APs and transmits a LWAPP discovery response to the APs. However, WLCs do not receive LWAPP join requests from the APs.

In order to troubleshoot this issue, ping the AP manager from a wired host on the same IP subnet as the AP manager. Then, check the ARP cache. If a duplicate IP address is found, remove the device with the duplicate IP address or change the IP address on the device so that it has a unique IP address on the network.

The AP can then join the WLC.

Problem 8: LAPs with Mesh image not able to join WLC

The Lightweight Access Point does not register with the WLC. The log displays this the error message

```
AAA Authentication Failure for UserName:5475xxx8bf9c User
 Type: WLAN USER
```

This can happen if the Lightweight Access Point was shipped with a mesh image and is in Bridge mode. If the LAP was ordered with mesh software on it, you need to add the LAP to the AP authorization list. Choose **Security > AP Policies** and add **AP** to the Authorization List. The AP should then join, download the image from the controller, then register with the WLC in bridge mode. Then you need to change the AP to local mode. The LAP downloads the image, reboots and registers back to the controller in local mode.

Problem 9: Bad address "Microsoft DHCP".

Access points can renew their IP addresses pretty quickly when trying to join a WLC, which can cause Windows DHCP servers to mark these IPs as "BAD_ADDRESS" which could quickly deplete the DHCP pool.

Check for more information : https://www.cisco.com/c/en/us/td/docs/wireless/controller/8-2/config-guide/b_cg82/b_cg82_chapter_0101000.html#dhcp-release-override-on-aps