

Troubleshoot a Lightweight Access Point Not Joining a Wireless LAN Controller

Document ID: 99948

Introduction

Prerequisites

- Requirements

- Conventions

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

Debug from the Controller

- debug lwapp events enable

- debug pm pki enable

Debug from the LAP

Avoiding DHCP Related Issues

Using Syslog Servers to Troubleshoot the LAP Join Process

LAP Does Not Join the Controller, Why?

- Check the Basics First

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

- Problem 2: Mismatch in Regulatory domain

- Problem 3: Error Message AP cannot join because the maximum number of APs on interface 2 is reached

- Problem 4: With SSC APs, the SSC AP policy is disabled

- Problem 5: AP authorization list enabled on the WLC; LAP not in the authorization list

- Problem 6: The SSC public key-hash is wrong or missing

- Problem 7: There is a certificate or public key corruption on the AP

- Problem 8: The controller might be working in Layer 2 mode

- Problem 9: You receive this error message on the AP after conversion to LWAPP

- Problem 10: Controller receives AP discovery message on wrong VLAN (you see the discovery message debug, but not response)

- Problem 11: 1250 LAP Not Able to Join WLC

- Problem 12: AP Not Able to Join the WLC, Firewall Blocking Necessary Ports

- Problem 13: Duplicate IP address in the network

- Problem 14: LWAPP APs do not join WLC if network MTU is less than 1500 bytes

- Problem 15: 1142 series LAP not joining the WLC, Error message on WLC :

- lwapp_image_proc: unable to open tar file

- Problem 16: 1000 series LAPs not able to join the Wireless LAN controller, WLC runs version 5.0

- Problem 17: Error Message – Dropping primary discovery request from AP

- XX:AA:BB:XX:DD:DD – maximum APs joined 6/6

Related Information

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 (LWAPP)

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.

Originally, the controllers only operated in Layer 2 mode. In Layer 2 mode, the LAPs are required to be on the same subnet as the management interface and the Layer 3 mode AP-manager interface is not present on the controller. The LAPs communicate with the controller using Layer 2 encapsulation only (ethernet encapsulation) and do not Dynamic Host Configuration Protocol (DHCP) an IP address.

When Layer 3 mode on the controller was developed, a new Layer 3 interface called AP-manager was introduced. In Layer 3 mode, the LAPs would DHCP an IP address first and then send their discovery request to the management interface using IP addresses (Layer 3). This allowed the LAPs to be on a different subnet than the management interface of the controller. Layer 3 mode is the dominate mode today. Some controllers and LAPs can only perform Layer 3 mode.

However, this presented a new problem: how did the LAPs find the management IP address of the controller when it was on a different subnet?

In Layer 2 mode, they were required to be on the same subnet. In Layer 3 mode, the controller and LAP are essentially playing hide and seek in the network. If you do not tell the LAP where the controller is via DHCP option 43, DNS resolution of "Cisco-lwapp-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.

When the LAPs discover the controller, they do not know if the controller is in Layer 2 mode or Layer 3 mode. Therefore, the LAPs always connect to the management interface address of the controller first with a discovery request. The controller then tells the LAP which mode it is in the discovery reply. If the controller is in Layer 3 mode, the discovery reply contains the Layer 3 AP-manager IP address so the LAP can send a join request to the AP-manager interface next.

Note: By default both management and AP–manager interfaces are left untagged on their VLAN during configuration. In case these are tagged, make sure they are tagged to the same VLAN in order to properly receive discovery and join response from the WLC.

The LWAPP AP goes through this process on startup for Layer 3 mode:

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:
 - a. DHCP option 43 (good for global companies where offices and controllers are on different continents)
 - b. DNS entry for `cisco-lwapp-controller` (good for local businesses – can also be used to find where brand new APs join)
 - c. Management IP addresses of controllers the LAP remembers previously
 - d. A Layer 3 broadcast on the subnet
 - e. Over the air provisioning
 - f. Statically configured information

From 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 LAP s 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 AP s CLI. For information on how to statically configure controller information using the AP CLI, refer to [Manually Configuring Controller Information Using the Access Point CLI](#).

For a detailed explanation on the different discovery algorithms that LAPs use to find controllers, refer to [LAP Registration with WLC](#).

For information on configuring DHCP option 43 on a DHCP server, refer to [DHCP OPTION 43 for Lightweight Cisco Aironet Access Points 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):
 - a. Primary Controller system name (previously configured on LAP)
 - b. Secondary Controller system name (previously configured on LAP)

- c. Tertiary Controller system name (previously configured on LAP)
- d. 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)
- e. 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 lwapp events enable** Shows discovery packets and join packets.
- **debug lwapp packet enable** Shows packet level information of the discovery and join packets.
- **debug pm pki enable** Shows certificate validation process.
- **debug disable-all** 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-mac-address>
(in xx:xx:xx:xx:xx format)
debug client <ap-mac-address>

debug lwapp events enable
debug lwapp 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 lwapp events enable

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

This is the **debug lwapp 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 lwapp events enable

```
Wed Oct 24 16:59:35 2007: 00:0b:85:5b: fb:d0 Received LWAPP DISCOVERY REQUEST from AP 00:0b:85:5b: fb:d0 to 00:0b:85:33:52:80 on port '2'
```

!--- LWAPP discovery request sent to the WLC by the LAP.

```
Wed Oct 24 16:59:35 2007: 00:0b:85:5b:fb:d0 Successful transmission of LWAPP Discovery-Response to AP 00:0b:85:5b:fb:d0 on Port 2
```

!--- WLC responds to the discovery request from the LAP.

```
Wed Oct 24 16:59:46 2007: 00:0b:85:5b:fb:d0 Received LWAPP JOIN REQUEST from AP 00:0b:85:5b:fb:d0 to 00:0b:85:33:52:81 on port '2'
```

!--- LAP sends a join request to the WLC.

```
Wed Oct 24 16:59:46 2007: 00:0b:85:5b:fb:d0 AP ap:5b:fb:d0: txNonce 00:0B:85:33:52:80 rxNonce 00:0B:85:5B:FB:D0
```

```
Wed Oct 24 16:59:46 2007: 00:0b:85:5b:fb:d0 LWAPP Join-Request MTU path from AP 00:0b:85:5b:fb:d0 is 1500, remote debug mode is 0
```

```
Wed Oct 24 16:59:46 2007: 00:0b:85:5b:fb:d0 Successfully added NPU Entry for AP 00:0b:85:5b:fb:d0 (index 55) Switch IP: 10.77.244.211, Switch Port: 12223, intIfNum 2, vlanId 0 AP IP: 10.77.244.219, AP Port: 49085, next hop MAC: 00:0b:85:5b:fb:d0
```

```
Wed Oct 24 16:59:46 2007: 00:0b:85:5b:fb:d0 Successfully transmission of LWAPP Join-Reply to AP 00:0b:85:5b:fb:d0
```

!--- WLC responds with a join reply to the LAP.

```
Wed Oct 24 16:59:46 2007: 00:0b:85:5b:fb:d0 Register LWAPP event for AP 00:0b:85:5b:fb:d0 slot 0 -- LAP registers with the WLC
```

```
Wed Oct 24 16:59:48 2007: 00:0b:85:5b:fb:d0 Received LWAPP CONFIGURE REQUEST from AP 00:0b:85:5b:fb:d0 to 00:0b:85:33:52:81
```

!--- LAP requests for the configuration information from the WLC.

```
Wed Oct 24 16:59:48 2007: 00:0b:85:5b:fb:d0 Updating IP info for AP 00:0b:85:5b:fb:d0 -- static 1, 10.77.244.219/255.255.255.224, gtw 10.77.244.220
```

```
Wed Oct 24 16:59:48 2007: spamVerifyRegDomain RegDomain set for slot 0 code 0 regstring -A regDfromCb -AB
```

```
Wed Oct 24 16:59:48 2007: spamVerifyRegDomain RegDomain set for slot 1 code 0 regstring -A regDfromCb -AB
```

```
Wed Oct 24 16:59:48 2007: Send AP Timesync of 1193245188 source MANUAL
```

```
Wed Oct 24 16:59:48 2007: spamEncodeDomainSecretPayload:Send domain secret TSWEBRET<0d,59,aa,b3,7a,fb,dd,b4,e2,bd,b5,e7,d0,b2,52,4d,ad,21,1a,12> to AP 00:0b:85:5b:fb:d0
```

```
Wed Oct 24 16:59:48 2007: 00:0b:85:5b:fb:d0 Successfully transmission of LWAPP Config-Message to AP 00:0b:85:5b:fb:d0
```

!--- WLC responds by providing all the necessary configuration information to the LAP.

```
Wed Oct 24 16:59:48 2007: Running spamEncodeCreateVapPayload for SSID 'eap fast'
```

```
Wed Oct 24 16:59:48 2007: Running spamEncodeCreateVapPayload for SSID 'WPA'
```

```
Wed Oct 24 16:59:48 2007: Running spamEncodeCreateVapPayload for SSID 'webauth'
```

```

Wed Oct 24 16:59:48 2007: Running spamEncodeCreateVapPayload for SSID 'eap fast'
Wed Oct 24 16:59:48 2007: Running spamEncodeCreateVapPayload for SSID 'WPA'
Wed Oct 24 16:59:48 2007: Running spamEncodeCreateVapPayload for SSID 'webauth'
.
.
.
Wed Oct 24 16:59:48 2007: 00:0b:85:5b:fb:d0 Successfully transmission of
LWAPP Change-State-Event Response to AP 00:0b:85:5b:fb:d0
.
.
Wed Oct 24 16:59:48 2007: 00:0b:85:5b:fb:d0 Received LWAPP Up event for
AP 00:0b:85:5b:fb:d0 slot 0!

!--- LAP is up and ready to service wireless clients.

Wed Oct 24 16:59:48 2007: 00:0b:85:5b:fb:d0 Received LWAPP CONFIGURE COMMAND RES from
AP 00:0b:85:5b:fb:d0
.
.
.
Wed Oct 24 16:59:48 2007: 00:0b:85:5b:fb:d0 Received LWAPP RRM_CONTROL_RES from
AP 00:0b:85:5b:fb:d0

!--- 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 lwapp events enable** command output if the LAP downloads an image from the controller as a part of the registration process:

```

Wed Oct 24 17:49:40 2007: 00:0b:85:5b:fb:d0 Received LWAPP IMAGE_DATA_RES from
AP 00:0b:85:5b:fb:d0
Wed Oct 24 17:49:40 2007: 00:0b:85:5b:fb:d0 Received LWAPP IMAGE_DATA_RES from
AP 00:0b:85:5b:fb:d0
Wed Oct 24 17:49:40 2007: 00:0b:85:5b:fb:d0 Received LWAPP IMAGE_DATA_RES from
AP 00:0b:85:5b:fb:d0

```

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 LWAPP Join Request to the WLC, it embeds its X.509 certificate in the LWAPP message. The AP also generates a random session ID that is also included in the LWAPP Join Request. When the WLC receives the LWAPP 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 LWAPP Control Messages exchanged with the AP. Note that data packets are sent in the clear in the LWAPP 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.

```
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: locking ca cert table
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: calling x509_alloc() for user cert
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: calling x509_decode()
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: <subject> C=US, ST=California,
L=San Jose, O=airespace Inc, CN=000b8591c3c0, MAILTO=support@airespace.com
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: <issuer> C=US, ST=California,
L=San Jose, O=airespace Inc, OU=none, CN=ca, MAILTO=support@airespace.com
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: Mac Address in subject is
00:0b:85:91:c3:c0
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: Cert is issued by Airespace Inc.
Thu Oct 25 13:52:59 2007: sshpmGetCID: called to evaluate <bsnDefaultCaCert>
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 0, CA cert >bsnOldDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 1, CA cert >bsnDefaultRootCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 2, CA cert >bsnDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCertFromCID: called to get cert for CID 2d812f0c
Thu Oct 25 13:52:59 2007: sshpmGetCertFromCID: comparing to row 0, certname
>bsnOldDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCertFromCID: comparing to row 1, certname
>bsnDefaultRootCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCertFromCID: comparing to row 2, certname
>bsnDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmUserCertVerify: calling x509_decode()
Thu Oct 25 13:52:59 2007: sshpmGetCID: called to evaluate <bsnOldDefaultCaCert>
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 0, CA cert >bsnOldDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCertFromCID: called to get cert for CID 20f00bf3
Thu Oct 25 13:52:59 2007: sshpmGetCertFromCID: comparing to row 0, certname
>bsnOldDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmUserCertVerify: calling x509_decode()
Thu Oct 25 13:52:59 2007: sshpmUserCertVerify: user cert verified using
>bsnOldDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: ValidityString (current):
2007/10/25/13:52:59
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: AP version is 0x400d900,
sending Cisco ID cert...
Thu Oct 25 13:52:59 2007: sshpmGetCID: called to evaluate <cscDefaultIdCert>
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 0, CA cert >bsnOldDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 1, CA cert >bsnDefaultRootCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 2, CA cert >bsnDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 3, CA cert >bsnDefaultBuildCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 4, CA cert >cscDefaultNewRootCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 5, CA cert >cscDefaultMfgCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 0, ID cert >bsnOldDefaultIdCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 1, ID cert >bsnDefaultIdCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 2, ID cert >bsnSslWebadminCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 3, ID cert >bsnSslWebauthCert<
Thu Oct 25 13:52:59 2007: sshpmGetIssuerHandles: Airespace ID cert ok; sending it...
Thu Oct 25 13:52:59 2007: sshpmGetCID: called to evaluate <bsnOldDefaultIdCert>
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 0, CA cert >bsnOldDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 1, CA cert >bsnDefaultRootCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 2, CA cert >bsnDefaultCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 3, CA cert >bsnDefaultBuildCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 4, CA cert >cscDefaultNewRootCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 5, CA cert >cscDefaultMfgCaCert<
```

```

Thu Oct 25 13:53:03 2007: sshpmGetCID: comparing to row 0, ID cert >bsnOldDefaultIdCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromHandle: calling sshpmGetCertFromCID()
with CID 0x156af135
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: called to get cert for CID 156af135
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 0,
certname >bsnOldDefaultCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 1,
certname >bsnDefaultRootCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 2,
certname >bsnDefaultCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 3,
certname >bsnDefaultBuildCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 4,
certname >cscscoDefaultNewRootCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 5,
certname >cscscoDefaultMfgCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 0,
certname >bsnOldDefaultIdCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromHandle: calling sshpmGetCertFromCID()
with CID 0x156af135
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: called to get cert for CID 156af135
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 0,
certname >bsnOldDefaultCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 1,
certname >bsnDefaultRootCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 2,
certname >bsnDefaultCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 3,
certname >bsnDefaultBuildCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 4,
certname >cscscoDefaultNewRootCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 5,
certname >cscscoDefaultMfgCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 0,
certname >bsnOldDefaultIdCert<
Thu Oct 25 13:53:03 2007: sshpmPublicKeyEncrypt: called to encrypt 16 bytes
Thu Oct 25 13:53:03 2007: sshpmPublicKeyEncrypt: successfully encrypted, out is 192 bytes
Thu Oct 25 13:53:03 2007: sshpmPrivateKeyEncrypt: called to encrypt 196 bytes
Thu Oct 25 13:53:03 2007: sshpmGetOpensslPrivateKeyFromCID: called to get key for
CID 156af135
Thu Oct 25 13:53:03 2007: sshpmGetOpensslPrivateKeyFromCID: comparing to row 0,
certname >bsnOldDefaultIdCert<
Thu Oct 25 13:53:03 2007: sshpmGetOpensslPrivateKeyFromCID: match in row 0
Thu Oct 25 13:53:03 2007: sshpmPrivateKeyEncrypt: calling RSA_private_encrypt with
172 bytes
Thu Oct 25 13:53:03 2007: sshpmPrivateKeyEncrypt: RSA_private_encrypt returned 192
Thu Oct 25 13:53:03 2007: sshpmPrivateKeyEncrypt: calling RSA_private_encrypt with
24 bytes
Thu Oct 25 13:53:03 2007: sshpmPrivateKeyEncrypt: RSA_private_encrypt returned 192
Thu Oct 25 13:53:03 2007: sshpmPrivateKeyEncrypt: encrypted bytes: 384
Thu Oct 25 13:53:03 2007: sshpmFreePublicKeyHandle: called with 0xae0c358
Thu Oct 25 13:53:03 2007: sshpmFreePublicKeyHandle: freeing public key

```

For a LAP with a SSC, the **debug pm pki enable** command output will look like this. Notice that the SSC hash is also seen in this output.

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

```

(Cisco Controller) > debug pm pki enable

Mon May 22 06:34:10 2006: sshpmGetIssuerHandles: getting (old) aes ID cert handle...
Mon May 22 06:34:10 2006: sshpmGetCID: called to evaluate <bsnOldDefaultIdCert>
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 0, CA cert >bsnOldDefaultCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 1, CA cert bsnDefaultRootCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 2, CA cert >bsnDefaultCaCert<

```

```

Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 3, CA cert >bsnDefaultBuildCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 4, CA cert >ciscoDefaultNewRootCaCe
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 5, CA cert ciscoDefaultMfgCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 0, ID cert >bsnOldDefaultIdCert<
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles: Calculate SHA1 hash on Public Key Data
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 30820122 300d06092a864886 f70d0101
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 01050003 82010f003082010a 02820101
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 00c805cd 7d406ea0cad8df69 b366fd4c
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 82fc0df0 39f2bff7ad425fa7 face8f15
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data f356a6b3 9b87625143b95a34 49292e11
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 038181eb 058c782e56f0ad91 2d61a389
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data f81fa6ce cd1f400bb5cf7cef 06ba4375
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data dde0648e c4d63259774ce74e 9e2fde19
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 0f463f9e c77b79ea65d8639b d63aa0e3
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 7dd485db 251e2e079cd31041 b0734a55
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 463fbacc 1a61502dc54e75f2 6d28fc6b
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 82315490 881e3e3102d37140 7c9c865a
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 9ef3311b d514795f7a9bac00 d13ff85f
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 97e1a693 f9f6c5cb88053e8b 7fae6d67
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data ca364f6f 76cf78bcbc1acc13 0d334aa6
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 031fb2a3 b5e572df2c831e7e f765b7e5
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data fe64641f de2a6fe323311756 8302b8b8
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 1bfae1a8 eb076940280cbcd1 49b2d50f
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles: Key Data f7020301 0001
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles: SSC Key Hash is
9e4ddd8dfcdd8458ba7b273fc37284b31a384eb9

```

!--- This is the actual SSC key-hash value.

```

Mon May 22 06:34:14 2006: LWAPP Join-Request MTU path from AP 00:0e:84:32:04:f0 is 1500,
remote debug mode is 0

```

Debug from the LAP

If the controller debugs do not indicate a join request, you can debug the process from the LAP as long as the LAP has a console port. You can see the LAP 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 the join/discovery packets to the controller as well as DHCP and DNS queries (all of these are UDP packets. Port 12223 is the controller's source port).
- **debug lwapp client event** Shows LWAPP events for the AP.
- **undebug all** Disables debugs on the AP.

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

```
UDP: sent src=10.77.244.199(20679), dst=10.77.244.208(12223)

!--- LWAPP Discovery Request sent to a controller to which
!--- the AP was previously registered to.

UDP: sent src=10.77.244.199(20679), dst=172.16.1.50(12223)

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

UDP: sent src=10.77.244.199(20679), dst=255.255.255.255(12223)

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

UDP: sent src=10.77.244.199(20679), dst=172.16.1.51(12223)

!--- LWAPP Join Request sent to AP-Manager interface on statically configured controller.
```

Avoiding DHCP Related Issues

LAPs that use DHCP to find an IP address before they start the WLC discovery process might have trouble receiving a DHCP address due to the misconfiguration of DHCP related parameters. This section explains how DHCP works with WLCs and provides some of the best practices to avoid DHCP related issues.

For DHCP, the controller behaves like a router with an IP helper address. That is, it fills in the gateway IP address and forwards the request via a unicast packet directly to the DHCP server.

When the DHCP offer comes back to the controller, it changes the DHCP server IP address to its virtual IP address. The reason it does this is because when Windows roams between APs, the first thing it does is try to contact the DHCP server and renew the address.

With the DHCP server address being 1.1.1.1 (typical virtual IP address on a controller), the controller can intercept that packet and quickly respond to Windows.

This is also why the virtual IP address is the same on all controllers. If a Windows laptop roams to an AP on another controller, it will try to contact the virtual interface on the controller. Due to the mobility event and context transfer, the new controller that the Windows client roamed to already has all the information to respond to Windows again.

If you want to use the internal DHCP server on the controller, all you have to do is put the management IP address as the DHCP server on the dynamic interface you create for the subnet. Then assign that interface to the WLAN.

The reason the controller needs an IP address on each subnet is so it can fill in the DHCP gateway address in the DHCP request.

These are some of the points to remember when you configure DHCP servers for the WLAN:

1. The DHCP server IP address should not fall within any dynamic subnet that is on the controller. It will be blocked but can be overridden with this command:

`config network mgmt-via-dynamic-interface` on version 4.0 only
(command not available in version 3.2)

2. The controller will forward the DHCP via unicast from its dynamic interface (in later code) using its IP address on that interface. Make sure that any firewall allows this address to reach the DHCP server.
3. Make sure that the response from the DHCP server can reach the controller's dynamic address on that VLAN through any firewalls. Ping the dynamic interface address from the DHCP server. Ping the DHCP server with a source IP address of the dynamic interface's gateway address.
4. Make sure the AP's VLAN is allowed on the switches and routers, and that their ports are configured as trunks so the packets (includes DHCP) tagged with the VLAN are allowed through the wired network.
5. Ensure that the DHCP server is configured to assign an IP address on the VLAN of the AP. You can also configure the WLC as a DHCP server. For more information on how to configure the DHCP server on the WLC, refer to the Using the GUI to Configure DHCP section of Cisco Wireless LAN Controller Configuration Guide, Release 5.0.
6. Verify that the controller's IP address on its dynamic interface will fall within one of the DHCP scopes on the DHCP server.
7. Finally, verify that you are not using a DHCP server that does not respond to unicast DHCP requests such as PIX.

If you cannot resolve your DHCP issue, there are 2 solutions:

- Try an internal DHCP server. Configure the DHCP server address on the dynamic interface to be the management IP address and then the DHCP internal pool. If the DHCP scope is enabled, it should work.
- Verify that there is no response to the DHCP request by sending in the output on the CLI (console or SSH) from these debugs:

```
0. debug mac addr <mac address>
1. debug dhcp message enable
2. debug dhcp packet enable
```

This should indicate that the DHCP packet was forwarded but the controller did not receive a response.

Finally, because of security on the controller, it is not recommend putting a VLAN or subnet on the controller that also contains the LAPs, unless it is on the management interface subnet.

Note: The RADIUS server or DHCP server must not be on any of the controller's dynamic interface subnets. Security will block the return packets that try to communicate with the controller.

Using Syslog Servers to Troubleshoot the LAP Join Process

Controller software release 5.2 enables you to configure the APs to send all CAPWAP-related errors to a syslog server. You do not need to enable any debug commands on the controller because all of the CAPWAP error messages can be viewed from the syslog server itself. For more information on this feature and the commands used to enable it, read the section Troubleshooting the Access Point Join Process of the configuration guide Cisco Wireless LAN Controller Configuration Guide, Release 5.2.

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 discover at least a single WLC Console or telnet/ssh into the controller to run debugs.

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

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

Complete these steps in order to troubleshoot this problem:

1. Issue the **debug lwapp errors enable** and **debug pm pki enable** commands.

The **debug lwapp event enable** command output shows the debug of certificate messages that are passed between the AP and the WLC. The output clearly shows a message that the certificate is rejected.

Note: Make sure to account for the Coordinated Universal Time (UTC) offset.

This is the output of the **debug lwapp events enable** command on the controller:

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

```
Thu Jan 1 00:09:46 1970: 00:0b:85:5b:fb:d0 Received LWAPP DISCOVERY REQUEST
from AP 00:0b:85:5b:fb:d0 to ff:ff:ff:ff:ff:ff on port '2'
Thu Jan 1 00:09:46 1970: 00:0b:85:5b:fb:d0 Successful transmission of
LWAPP Discovery-Response to AP 00:0b:85:5b:fb:d0 on Port 2
Thu Jan 1 00:09:57 1970: 00:0b:85:5b:fb:d0 Received LWAPP JOIN REQUEST
from AP 00:0b:85:5b:fb:d0 to 00:0b:85:33:52:81 on port '2'
Thu Jan 1 00:09:57 1970: 00:0b:85:5b:fb:d0 LWAPP Join-Request does not
include valid certificate in CERTIFICATE_PAYLOAD from AP 00:0b:85:5b:fb:d0.
Thu Jan 1 00:09:57 1970: 00:0b:85:5b:fb:d0 Unable to free public key
for AP 00:0B:85:5B:FB:D0
Thu Jan 1 00:09:57 1970: spamProcessJoinRequest : spamDecodeJoinReq failed
```

This is the output from the **debug pm pki enable** command on the controller. This output follows the process for validation of the certificate.

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

```
Thu May 25 07:25:00 2006: sshpmGetIssuerHandles: locking ca cert table
Thu May 25 07:25:00 2006: sshpmGetIssuerHandles: calling x509_alloc() for user cert
Thu May 25 07:25:00 2006: sshpmGetIssuerHandles: calling x509_decode()
Thu May 25 07:25:00 2006: sshpmGetIssuerHandles: <subject> C=US, ST=California,
L=San Jose, O=Cisco Systems, CN=C1200-001563e50c7e, MAILTO=support@cisco.com
Thu May 25 07:25:00 2006: sshpmGetIssuerHandles: <issuer> O=Cisco Systems,
CN=Cisco Manufacturing CA
Thu May 25 07:25:00 2006: sshpmGetIssuerHandles: Mac Address in subject
is 00:15:63:e5:0c:7e
Thu May 25 07:25:00 2006: sshpmGetIssuerHandles: Cert is issued by Cisco Systems.
.....
.....
```

```

.....
.....
Fri Apr 15 07:55:03 2005: ssphmUserCertVerify: calling x509_decode()
Fri Apr 15 07:55:03 2005: ssphmUserCertVerify: user cert verified using
>ciscoDefaultMfgCaCert<
Fri Apr 15 07:55:03 2005: sshpmGetIssuerHandles: ValidityString (current):
2005/04/15/07:55:03
Fri Apr 15 07:55:03 2005: sshpmGetIssuerHandles: Current time outside
AP cert validity interval: make sure the controller time is set.
Fri Apr 15 07:55:03 2005: sshpmFreePublicKeyHandle: called with (nil)

```

This information clearly shows that the controller time is outside the certificate validity interval of the LAP. Therefore, the LAP cannot register with the controller. Certificates installed in the LAP 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 LAP'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 LAPs 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:

```

AP0015.63e5.0c7e#show crypto ca certificates
.....
.....
.....
.....
Certificate
Status: Available
Certificate Serial Number: 4BC6DAB80000000517AF
Certificate Usage: General Purpose
Issuer:
cn=Cisco Manufacturing CA
o=Cisco Systems
Subject:
Name: C1200-001563e50c7e
ea=support@cisco.com
cn=C1200-001563e50c7e
o=Cisco Systems
l=San Jose
st=California
c=US
CRL Distribution Point:
http://www.cisco.com/security/pki/crl/cmca.crl
Validity Date:
start date: 17:22:04 UTC Nov 30 2005
end date: 17:32:04 UTC Nov 30 2015
renew date: 00:00:00 UTC Jan 1 1970
Associated Trustpoints: Cisco_IOS_MIC_cert
.....
.....
.....

```

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 output, it is

Name: C1200-001563e50c7e. This is the actual certificate validity interval to be considered.

Problem 2: Mismatch in Regulatory domain

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

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

```
Wed Oct 24 17:13:20 2007: 00:0b:85:91:c3:c0 Received LWAPP DISCOVERY REQUEST
from AP 00:0b:85:91:c3:c0 to 00:0b:85:33:52:80 on port '2'
Wed Oct 24 17:13:20 2007: 00:0e:83:4e:67:00 Successful transmission of
LWAPP Discovery-Response to AP 00:0b:85:91:c3:c0 on Port 2
Wed Oct 24 17:13:46 2007: 00:0b:85:91:c3:c0 Received LWAPP JOIN REQUEST
from AP 00:0b:85:91:c3:c0 to 00:0b:85:33:52:81 on port '2'
Wed Oct 24 17:13:46 2007: 00:0b:85:91:c3:c0 AP ap:91:c3:c0:
txNonce 00:0B:85:33:52:80 rxNonce 00:0B:85:91:C3:C0
Wed Oct 24 17:13:46 2007: 00:0b:85:91:c3:c0 LWAPP Join-Request MTU path
from AP 00:0b:85:91:c3:c0 is 1500, remote debug mode is 0
Wed Oct 24 17:13:46 2007: 00:0b:85:91:c3:c0 Successfully added NPU Entry
for AP 00:0b:85:91:c3:c0 (index 48)
Switch IP: 10.77.244.211, Switch Port: 12223, intIfNum 2, vlanId 0
AP IP: 10.77.246.18, AP Port: 7228, next hop MAC: 00:17:94:06:62:88
Wed Oct 24 17:13:46 2007: 00:0b:85:91:c3:c0 Successfully transmission
of LWAPP Join-Reply to AP 00:0b:85:91:c3:c0
Wed Oct 24 17:13:46 2007: 00:0b:85:91:c3:c0 Register LWAPP event
for AP 00:0b:85:91:c3:c0 slot 0
Wed Oct 24 17:13:46 2007: 00:0b:85:91:c3:c0 Register LWAPP event
for AP 00:0b:85:91:c3:c0 slot 1
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 Received LWAPP CONFIGURE REQUEST
from AP 00:0b:85:91:c3:c0 to 00:0b:85:33:52:81
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 Updating IP info for AP 00:0b:85:91:c3:c0 --
static 0, 10.77.246.18/255.255.255.224, gw 10.77.246.1
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 Updating IP 10.77.246.18 ==> 10.77.246.18
for AP 00:0b:85:91:c3:c0
Wed Oct 24 17:13:47 2007: spamVerifyRegDomain RegDomain set for
slot 0 code 21 regstring -N regDfromCb -AB
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 AP 00:0b:85:91:c3:c0: 80211a Regulatory Domain
(-N) does not match with country (US ) reg. domain -AB for the slot 0
Wed Oct 24 17:13:47 2007: spamVerifyRegDomain RegDomain set for
slot 1 code 21 regstring -N regDfromCb -AB
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 AP 00:0b:85:91:c3:c0: 80211bg Regulatory
Domain (-N) does not match with country (US ) reg. domain -AB for the slot 1
Wed Oct 24 17:13:47 2007: spamVerifyRegDomain AP RegDomain check for the country US failed
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 AP 00:0b:85:91:c3:c0: Regulatory Domain
check Completely FAILED The AP will not be allowed to join
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 apfSpamProcessStateChangeInSpamContext:
Deregister LWAPP event for AP 00:0b:85:91:c3:c0 slot 0
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 apfSpamProcessStateChangeInSpamContext:
Deregister LWAPP event for AP 00:0b:85:91:c3:c0 slot 1
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 Deregister LWAPP event
for AP 00:0b:85:91:c3:c0 slot 0
Wed Oct 24 17:13:47 2007: 00:0b:85:91:c3:c0 Deregister LWAPP event
for AP 00:0b:85:91:c3:c0 slot 1
```

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 LAP 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 and WLCs, ensure that they share the same regulatory domain. Only then can the LAPs register with the WLC.

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

- b. Choose **SSC** as the certificate type.
- c. Add AP to the authorization list with MAC address and key–hash.

This key–hash can be obtained from the output of the **debug pm pki enable** command. See Problem 6 for information on getting the key–hash value.

Problem 5: 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 lwapp events enable** command:

```
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 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
LWAPP 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
LWAPP 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 LWAPP 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 LWAPP 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 lwapp 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 6: The SSC public key–hash is wrong or missing

Complete these steps in order to troubleshoot this problem:

1. Issue the **debug lwapp events enable** command.

This verifies that the AP tries to join.

2. Issue the **show auth-list** command.

This command shows the public key–hash that the controller has in storage.

3. Issue the **debug pm pki enable** command.

This command shows the actual public key–hash. The actual public key–hash must match the public key–hash that the controller has in storage. A discrepancy causes the problem. This is a sample output of this debug message:

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

```
(Cisco Controller) > debug pm pki enable
```

```
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles: getting (old) aes ID cert handle...
Mon May 22 06:34:10 2006: sshpmGetCID: called to evaluate <bsnOldDefaultIdCert>
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 0,
CA cert >bsnOldDefaultCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 1,
CA cert bsnDefaultRootCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 2,
CA cert >bsnDefaultCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 3,
CA cert >bsnDefaultBuildCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 4,
CA cert >cscsDefaultNewRootCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 5,
CA cert cscsDefaultMfgCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 0,
ID cert >bsnOldDefaultIdCert<
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles: Calculate SHA1 hash on
Public Key Data
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 30820122 300d06092a864886 f70d0101
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 01050003 82010f003082010a 02820101
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 00c805cd 7d406ea0cad8df69 b366fd4c
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 82fc0df0 39f2bfff7ad425fa7 face8f15
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data f356a6b3 9b87625143b95a34 49292e11
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 038181eb 058c782e56f0ad91 2d61a389
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data f81fa6ce cd1f400bb5cf7cef 06ba4375
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data dde0648e c4d63259774ce74e 9e2fde19
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 0f463f9e c77b79ea65d8639b d63aa0e3
Mon May 22 06:34:10 2006: sshpmGetIssuerHandles:
Key Data 7dd485db 251e2e079cd31041 b0734a55
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 463fbacc 1a61502dc54e75f2 6d28fc6b
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 82315490 881e3e3102d37140 7c9c865a
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 9ef3311b d514795f7a9bac00 d13ff85f
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 97e1a693 f9f6c5cb88053e8b 7fae6d67
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data ca364f6f 76cf78bcbc1acc13 0d334aa6
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data 031fb2a3 b5e572df2c831e7e f765b7e5
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
Key Data fe64641f de2a6fe323311756 8302b8b8
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
```

```
Key Data 1bfae1a8 eb076940280cbcd1 49b2d50f
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles: Key Data f7020301 0001
Mon May 22 06:34:14 2006: sshpmGetIssuerHandles:
SSC Key Hash is 9e4ddd8dfcdd8458ba7b273fc37284b31a384eb9
```

!--- This is the actual SSC key-hash value.

```
Mon May 22 06:34:14 2006: LWAPP Join-Request MTU path from
AP 00:0e:84:32:04:f0 is 1500, remote debug mode is 0
Mon May 22 06:34:14 2006: spamRadiusProcessResponse:
AP Authorization failure for 00:0e:84:32:04:f0
```

Complete these steps in order to resolve the problem:

1. Copy the public key-hash from the **debug pm pki enable** command output and use it to replace the public key-hash in the authentication list.
2. Issue the **config auth-list add ssc AP_MAC AP_key** command in order to add the AP MAC address and key-hash to the authorization list.

This is an example of this command:

Note: This command has been moved to the second line due to space constraints.

```
(Cisco Controller)>config auth-list add ssc 00:0e:84:32:04:f0
9e4ddd8dfcdd8458ba7b273fc37284b31a384eb9
```

Problem 7: 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 lwapp 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
LWAPP 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).
- SSC AP Downgrade to Cisco IOS® Software Release 12.3(7)JA.

If it is an AP with an SSC convert it back to IOS using the MODE button. Then use the lwapp upgrade tool again to convert back to LWAPP. This should create the certificate again.

Complete these steps in order to downgrade:

1. Use the reset button option.
2. Clear the controller settings.
3. Run the upgrade again.

For more information on downgrading an LAP, refer to [Upgrading Autonomous Cisco Aironet Access Points](#)

to Lightweight Mode.

If you have a WCS, you can push the SSCs to the new WLC. For more information on how to configure APs using the WCS, refer to the Configuring Access Points section of *Cisco Wireless Control System Configuration Guide, Release 5.1*.

Problem 8: The controller might be working in Layer 2 mode

Complete this step in order to troubleshoot this problem:

Check the mode of operation of the controller. Converted APs only support Layer 3 discovery. Converted APs do not support Layer 2 discovery.

Complete these steps in order to resolve the problem:

1. Set the WLC to be in Layer 3 mode.
2. Reboot and configure the AP-manager interface.

If you have a service port, such as the service port on a 4402 or 4404, you should have it in a different supernet than the AP-manager and management interfaces.

Problem 9: You receive this error message on the AP after conversion to LWAPP

You see this error message:

```
*Mar 1 00:00:23.535: %LWAPP-5-CHANGED: LWAPP changed state to DISCOVERY
*Mar 1 00:00:23.550: LWAPP_CLIENT_ERROR_DEBUG: lwapp_crypto_init_ssc_keys_and_certs
no certs in the SSC Private File
*Mar 1 00:00:23.550: LWAPP_CLIENT_ERROR_DEBUG:
*Mar 1 00:00:23.551: lwapp_crypto_init: PKI_StartSession failed
*Mar 1 00:00:23.720: %SYS-5-RELOAD: Reload requested by LWAPP CLIENT.
Reload Reason: FAILED CRYPTO INIT.
*Mar 1 00:00:23.721: %LWAPP-5-CHANGED: LWAPP changed state to DOWN
```

The AP reloads after 30 seconds and starts the process over again.

Complete these steps in order to resolve this problem:

1. You have an SSC AP. Convert back to an autonomous IOS image.
2. Clear the configuration by issuing the **write erase** command and reload. Do not save the configuration when reloading.

Problem 10: Controller receives AP discovery message on wrong VLAN (you see the discovery message debug, but not response)

You see this message in the **debug lwapp 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 11: 1250 LAP Not Able to Join WLC

The setup consists of a 2106 WLC that runs version 4.1.185.0. A Cisco 1250 AP is not able to join the controller.

The log on the WLC shows this:

```
Mon Jun 2 21:19:37 2008 AP with MAC f0:2x:cf:2x:1d:3x (APf02x.cf2x.1d3x) is unknown.
Mon Jun 2 21:19:37 2008 AP Associated. Base Radio MAC: f0:2x:cf:2x:1d:3x
Mon Jun 2 21:19:26 2008 AP Disassociated. Base Radio MAC:f0:2x:cf:2x:1d:3x
Mon Jun 2 21:19:20 2008 AP with MAC f0:2x:cf:2x:1d:3x (APf02x.cf2x.1d3x) is unknown.
Mon Jun 2 21:19:20 2008 AP Associated. Base Radio MAC: f0:2x:cf:2x:1d:3x
Mon Jun 2 21:19:09 2008 AP Disassociated. Base Radio MAC:f0:2x:cf:2x:1d:3x
Mon Jun 2 21:19:03 2008 AP with MAC f0:2x:cf:2x:1d:3x (APf02x.cf2x.1d3x) is unknown.
```

Solution: This is because the Cisco 1250 series LAP is not supported on version 4.1. The Cisco Aironet 1250 Series AP is supported from controller versions 4.2.61 and later. In order to fix this issue, upgrade the controller software to 4.2.61.0 or later.

Problem 12: 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 LWAPP traffic:
 - ◆ Data – 12222
 - ◆ Control – 12223
- Enable these UDP ports for mobility traffic:
 - ◆ 16666 – 16666
 - ◆ 16667 – 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 13: 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 14: LWAPP APs do not join WLC if network MTU is less than 1500 bytes

This is because of Cisco bug ID **CSCsd94967**. LWAPP APs might fail to join a WLC. If the LWAPP join request is larger than 1500 bytes, LWAPP must fragment the LWAPP join request. The logic for all LWAPP APs is that the size of the first fragment is 1500 bytes (including IP and UDP header) and the second fragment is 54 bytes (including IP and UDP header). If the network between the LWAPP APs and WLC has a MTU size less than 1500 (as might be encountered when using a tunneling protocol such as IPsec VPN, GRE, MPLS, etc.), WLC cannot handle the LWAPP join request.

You will encounter this problem under these conditions:

- WLC that runs version 3.2 software or earlier
- Network path MTU between the AP and WLC is less than 1500 bytes

In order to resolve this issue, use any one of these options:

- Upgrade to WLC software 4.0, if the platform supports it. In WLC version 4.0, this problem is fixed by allowing the LWAPP tunnel to reassemble up to 4 fragments.
- Increase the network path MTU to 1500 bytes.
- Use 1030 REAPs for the locations reachable via low MTU paths. REAP LWAPP connections to 1030 APs have been modified to handle this situation by reducing the MTU used for REAP mode.

Problem 15: 1142 series LAP not joining the WLC, Error message on WLC : lwapp_image_proc: unable to open tar file

The 1142 series LAPs are supported only with WLC release 5.2 and later. If you run WLC versions earlier than 5.2, you cannot register the LAP to the Controller and you will see an error message similar to this:

```
*Mar 27 15:04:38.596: %LWAPP-5-CHANGED: CAPWAP changed state to DISCOVERY
*Mar 27 15:04:38.597: %CAPWAP-5-CHANGED: CAPWAP changed state to DISCOVERY
*Mar 27 15:04:38.606: %LWAPP-3-CLIENTERRORLOG: not receive read response(3)
*Mar 27 15:04:38.609: lwapp_image_proc: unable to open tar file
```

```
Mar 12 15:47:27.237 spam_lrad.c:8317 LWAPP-3-IMAGE_DOWNLOAD_ERR3:
```

Refusing image download request from AP 0X:2X:D0:FG:a7:XX - unable to open image file /bsn/ap//c1140

In order to register the 1140 LAPs to the WLC, upgrade the firmware on the WLC to 5.2 or later versions.

Problem 16: 1000 series LAPs not able to join the Wireless LAN controller, WLC runs version 5.0

This is because WLC software release 5.0.148.0 or later is not compatible with Cisco Aironet 1000 series APs.

Problem 17: Error Message – Dropping primary discovery request from AP XX:AA:BB:XX:DD:DD – maximum APs joined 6/6

There is a limit to the number of LAPs that can be supported by a WLC. Each WLC supports a certain number of LAPs, which depends on the model and platform. This error message is seen on the WLC when it receives a discovery request after it has reached its maximum AP capacity.

Here is the number of LAPs supported on the different WLC platform and models:

- The 2100 series controller supports up to 6, 12, or 25 LAPs. This depends on the model of the WLC.
- The 4402 supports up to 50 LAPs, while the 4404 supports up to 100. This makes it ideal for large-sized enterprises and large-density applications.
- The Catalyst 6500 Series Wireless Services Module (WiSM) is an integrated Catalyst 6500 switch and two Cisco 4404 controllers that supports up to 300 LAPs.
- The Cisco 7600 Series Router WiSM is an integrated Cisco 7600 router and two Cisco 4404 controllers that supports up to 300 LAPs.
- The Cisco 28/37/38xx Series Integrated Services Router is an integrated 28/37/38xx router and Cisco controller network module that supports up to 6, 8, 12, or 25 LAPs, depending on the version of the network module. The versions that support 8, 12, or 25 APs and the NME-AIR-WLC6-K9 6-access-point version feature a high-speed processor and more on-board memory than the NM-AIR-WLC6-K9 6-access-point version.
- The Catalyst 3750G Integrated WLC Switch is an integrated Catalyst 3750 switch and Cisco 4400 series controller that supports up to 25 or 50 LAPs.

Related Information

- **Lightweight Access Point (LAP) Authorization in a Cisco Unified Wireless Network Configuration Example**
- **Lightweight AP (LAP) Registration to a Wireless LAN Controller (WLC)**
- **Cisco Wireless LAN Controller Configuration Guide, Release 4.1**
- **Technical Support & Documentation – Cisco Systems**

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2008 – 2009 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Aug 21, 2008

Document ID: 99948
