轻量级无线接入点不能加入到无线控制器的故障排除

症状 10：控制器不在当前的 VLAN 里收到 AP 的 discovery 信息（你可以看到 discovery 信息，但是看不到 response）

症状 11：1250 LAP 不能加入 WLC

症状 12：AP 不能加入 WLC，因为防火墙阻止了一些必要的端口

NetPro 论坛 - Featured Conversations

相关信息

导言

这篇文档简单介绍了无线控制器（以下简称 WLC 或者控制器）的发现和加入过程，也提供了有关于为什么轻量级无线接入点（以下简称 LAP）加入 WLC 失败以及如何解决这种问题的信息。

必备条件

知识需求

Cisco 建议您对以下知识点有所了解：

- 配置 LAP 和 WLC 的基本知识；
- 轻量级无线接入点协议（以下简称 LWAPP）的基本知识

参阅理解轻量级无线接入点协议(LWAPP)[Understanding the Lightweight Access Point Protocol (LWAPP)]以获得更多信息

通用定义

参阅思科技术通用提示[Cisco Technical Tips Conventions] 获得更多这方面的信息

无线控制器发现和加入过程概述

在思科统一无线网络里，LAP 在为无线客户端提供服务前，必须先发现并加入到 WLC。

早期的控制器只能在工作在 2 层模式。在 2 层模式下，LAP 必须和控制器的管理接口在同一个子网内，控制器并不提供 3 层模式的 AP 管理接口。LAP 和控制器之间只通过 2 层封装（以太网封装）方式进行通信，并不需要通过 DHCP 方

式获得一个 IP 地址。

当控制器发展到能够提供 3 层工作模式的时候，一个全新的被称之为 AP 管理接口的 3 层接口被引入。在 3 层模式下，LAP 可以先通过 DHCP 方式获得一个 IP 地址，然后通过 IP 地址的方式向管理接口发送发现请求(以下称为 discovery request)。这种方式可以允许 LAP 和控制器上的管理接口不在同一个子网内。目前，主要的工作模式是 3 层模式，有些控制器和 LAP 只能提供 3 层工作模式。

然而，这就带来了一个新的问题：当 LAP 和控制器位于不同的子网时，LAP 如何知道控制器管理接口的 IP 地址？

在 2 层工作模式下，LAP 和控制器必须在同一个子网内。在 3 层工作模式下，控制器和 LAP 之间就好比是在网络中玩捉迷藏。如果不通过 DHCP 选项 43 或者"Cisco-lwapp-controller@local_domain"的 DNS 方式或者静态配置的方式把控制器的管理接口 IP 地址通告给 LAP，LAP 就不知道如何在网络中找到控制器的管理接口。

在这些方式之外，LAP 能够通过 255.255.255.255 的本地广播方式自动来寻找同一子网内的控制器。同样的，LAP 在重启时能够储存任何一个曾经加入过的控制器的管理接口 IP 地址。因此，如果您将一个 LAP 第一次放入到本地子网内，它最终会找到控制器的管理接口并记住地址。这就是所谓的启动(priming)。如果您在 LAP 启动之后再放入，这并不能帮助 LAP 找到控制器。因此，思科建议使用 DHCP 选项 43 或者 DNS 方式。

当 LAP 在寻找控制器时，他们并不会知道控制器是工作在 2 层模式还是 3 层模式的。因此，LAP 在连接管理接口的地址之前，需要发送一个 discovery request，然后控制器通过发送发现回应（以下称为 discovery reply）来告诉 LAP 该控制器的工作模式。如果控制器是工作在 3 层模式的， discovery reply 会包含 3 层的 AP 管理地址，以便于 LAP 接下来可以向 AP 管理接口发送加入请求（以下称为 join request）。

LWAPP 模式的 AP 在启动时通过以下进程来启用 3 层模式：

1. 如果事先没有配置静态 IP 地址，LAP 在启动的时候需要通过 DHCP 方式来获得一个 IP 地址

2. LAP 通过各种发现算法向控制器发送 discovery request，并建立控制器清单。从本质上讲，为了要建立控制器清单，LAP 会学习到尽可能多的控制器的管理接口地址，通过：
    a. DHCP 选项 43（适合于全球化公司，因为这种公司，通常办公室和控制器在不同的地方）
    b. DNS 条目：“cisco-lwapp-controller”（适合于本地性公司，也可以被用来使全新的 AP 找到可供加入的 WLC）
    c. LAP 先前储存的控制器的管理口 IP 地址
    d. 子网内的 3 层广播
    e. 通过空中接口提供
    f. 静态的配置信息

从这些发现方式中，你会发现，最简单的方法是将 LAP 和控制器的管理接口部署在同一个子网内，并允许 LAP 通过 3 层广播方式去发现控制器。这种方法可以应用于网络规模比较小，并且没有本地 DNS 服务器的小型企业。

下一个简单的部署方法是使用 DNS 和 DHCP 方式。相同的 DNS 名字可以有多个条目，这可以让 LAP 发现多个控制器。这种方法可以应用于所有控制器在同一个地点，并拥有本地 DNS 服务器的企业。或者是拥有多个 DNS 后缀并通过后缀来区分控制器的企业。

大型企业通常会使用 DHCP 选项 43 方式，并通过 DHCP 方式使信息本地化。使用这种方式的大型企业通常都只会有一个 DNS 条目。例如，思科在欧洲，澳大利亚和美国都有办公场所，为了确保 LAP 只加入本地的控制器，就不能使用 DNS 方式，必须使用 DHCP 选项 43 告诉 LAP 他们本地控制器管理接口的 IP 地址。

最后，如果网络里面没有 DCHP 服务器的话，就需要使用静态配置了。您可以通过 console 口连接 AP，并通过 CLI 方式静态配置 LAP 所要加入到控制器的必要信息，关于如何使用的 AP CLI 方式静态配置控制器信息的方法，请参阅利用接入点 CLI 手动配置控制器[Manually Configuring Controller Information Using the Access Point CLI.].

如需了解 LAP 用来发现控制器的不同发现算法的详细解释，请参阅 LAP 注册到 WLC[LAP Registration with WLC.]

如需了解在 DHCP 服务器上配置 DHCP 选项 43 的更多信息，请参阅为思科 aironet 轻量级无线接入点提供 DHCP 选项 43 的配置实例[DHCP OPTION 43 for Lightweight Cisco Aironet Access Points Configuration Example.]

3. LAP 会向控制器列表上的所有的控制器发送 discovery request 并等待控制器回应 discovery reply，回应中会包括系统名字，AP 管理接口 IP 地址，每个 AP 管理接口上已经连接的 AP 数量，以及控制器剩余的可承载 AP 的数量。
4. 基于生成的控制器列表，LAP 通过以下顺序向控制器发送 join request（前提是 AP 收到该控制器发送的 discovery reply）
   a. 第一个（primary）控制器系统名（该系统名先前已经配置在 LAP 上）
   b. 第二个（secondary）控制器系统名（该系统名先前已经配置在 LAP 上）
   c. 第三个（tertiary）控制器系统名（该系统名先前已经配置在 LAP 上）
   d. 主控制器（如果 AP 没有配置 primary、secondary、tertiary 控制器名字，用来使新 LAP 知道所要加入的控制器）
   e. 如果以上几个环节没有的话，将启用控制器间的负载均衡。该负载均衡通过 discovery response 中所包含的控制器剩余承载能力这个值来实现。

      如果两个控制器剩余的承载能力相同的话，AP 会向第一个回应 discovery response 的控制器发送 join request。如果一个控制器上有多个 AP 管理接口，将向关联 AP 最少的 AP 管理接口发送 join request。

      控制器会回应所有的 discovery request，并不检查 AP 的证书或者 AP 是否符合信任策略。然而，join request 必须包含一个有效的认证信息，否则将无法从控制器处收到 join response。如果 LAP 无法从它选择的控制器处收到 join response，LAP 会尝试向列表中的下一个控制器发送 join request。如果控制器是配置好加入顺序的（primary/secondary/tertiary），就算 LAP 收不到 join response，也不会向加入顺序之外的其他控制器发送 join request。

5. 当 LAP 收到一个 join reply 之后，它会检查自己的软件镜像是否和控制器上的相同。如果不同，AP 需要从控制器处下载软件镜像，然后重启并加载新的软件镜像。重启之后，AP 需要重新从步骤 1 开始。
6. 如果 AP 和控制器上的软件镜像相同，AP 需要向控制器请求配置信息，然后 AP 在控制器上才会显示状态是已注册。

   在 AP 下载完配置信息后，AP 可能会再次重启来应用新的配置。因此，可能会有额外的重启发生，但这是一种正常的行为。

从控制器端 Debug

在控制器上，在 CLI 界面，可以使用一些 debug 命令来查看 LAP 加入无线控

制器的整个过程：

- debug lwapp events enable—显示 discovery 数据包和 join 数据包.
- debug lwapp packet enable—显示数据包级别的 discovery 和 join 信息
- debug pm pki enable—显示认证确认过程
- debug disable-all—关闭所有的 debug.

在终端上可以导出日志信息，通过 console 方式、SSH 或者 telnet 方式登陆到控制器，输入以下命令：

config session timeout 120
config serial timeout 120
show run-config　　(输入空格键来收集所有的信息)

debug mac addr *<ap-mac-address>*
(使用 xx:xx:xx:xx:xx 格式)
debug client *<ap-mac-address>*

debug lwapp events enable
debug lwapp errors enable
debug pm pki enable

在收集完 debug 信息之后，使用 debug disable-all 命令来关闭所有的 debug 命令。

下一个部分显示的是，当 LAP 注册到控制器的时候，这些 debug 命令所输出的信息。

debug lwapp events enable

这个命令提供了一些在 LWAPP discovery 和 join 过程中，发生的 LWAPP 的事件和错误信息。

如下所示，是 debug lwapp events enable 命令的输出信息，前提是 LAP 和它所要注册的 WLC 拥有相同的 image。

注意：因为空间的限制，有一些输出的信息被移到了下一行。

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'
```

!--- LAP 向 WLC 发送 LWAPP discovery request

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 向 LAP 回应 discovery request

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 向 WLC 发送 join request

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 向 LAP 回应 join reply

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 从 WLC 处请求配置信息

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 回应 LAP 并提供所有必需的配置信息。

```
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 已经启动并可以给无线客户端提供服务

```
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 将所有的 RRM（radio resource management）和其他的配置信息细节发送到 LAP 上。

正如上面章节所提到的，一旦 LAP 注册到 WLC 上，它会检测是否和控制器上的软件镜像一致。如果 LAP 上的软件镜像和 WLC 上的不同，LAP 会首先从 WLC 下载新的软件镜像。如果 LAP 上的软件镜像和 WLC 上的相同，LAP 会继续从 WLC 下

载配置和其他的一些参数。

如果 LAP 在注册过程中从控制器上下载软件镜像，因为这也是 LAP 注册的一部分，所以可以从 debug lwapp events enable 这条命令的输出中看到相应的信息：

```
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
```

一旦下载完毕，LAP 会重启，并再次运行 discovery 和 join 算法。

debug pm pki enable

在 LAP 的加入过程中，WLC 需要通过检测 LAP 的证书是否有效来对 LAP 进行验证。

在 AP 向 WLC 发送 LWAPP join request 的时候，它会把自己的 X.509 证书包含在 LWAPP 消息中。LWAPP 的 join request 消息里面还包括 AP 所生成的随机会话 ID。当 WLC 收到 LWAPP 的 join request 的时候，WLC 使用 AP 的公有密钥来验证 X.509 证书的有效性，同时也检测该证书是否由受信任的证书机构发布。

WLC 会通过验证证书起始日期和时间来检查 AP 证书的有效性，同时和自己的日期和时间进行对比。（因此控制器的时钟最好是和当前的日期和时间接近）。如果 X.509 的证书是有效的，WLC 会生成一个随机 AES 加密密钥。WLC 会将该加密密钥放到加密引擎中，使得后续 WLC 和 AP 之间交换的 LWAPP 控制信息能够被加密和解密。需要注意的是，在 LAP 和控制器之间的 LWAPP 隧道中的数据包是以明文形式进行传输的。

Debug pm pki enable 这条命令显示的是 LAP 在加入到控制器阶段时，证书确认过程的信息。如果 AP 有一个由 LWAPP 转换程式创建的自我签署证书（下面简称 SSC），Debug pm pki enable 这条命令也会显示加入过程中 AP 的哈希密钥。如果 AP 用的是由制造商安装的证书（下面简称 MIC），你将不能看到这个哈希密钥。

注意：所有在 2006 年 6 月后制造的 AP，都会有制造商安装的证书（MIC）。

当有 MIC 证书的 LAP 需要加入到控制器时，通过 debug pm pki enable 命令可以看到这种信息：

注意：因为空间的限制，有一些输出的信息被移到了下一行。

```
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: ssphmUserCertVerify: 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: ssphmUserCertVerify: calling x509_decode()
Thu Oct 25 13:52:59 2007: ssphmUserCertVerify: user cert verfied 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 <cscoDefaultIdCert>
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 >cscoDefaultNewRootCaCert<
```

```
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 5, CA cert >cscoDefaultMfgCaCert<
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 >cscoDefaultNewRootCaCert<
Thu Oct 25 13:52:59 2007: sshpmGetCID: comparing to row 5, CA cert >cscoDefaultMfgCaCert<
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 >cscoDefaultNewRootCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 5,
certname >cscoDefaultMfgCaCert<
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 >cscoDefaultNewRootCaCert<
```

```
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 5,
certname >cscoDefaultMfgCaCert<
Thu Oct 25 13:53:03 2007: sshpmGetCertFromCID: comparing to row 0,
certname >bsnOldDefaultIdCert<
Thu Oct 25 13:53:03 2007: ssphmPublicKeyEncrypt: called to encrypt 16 bytes
Thu Oct 25 13:53:03 2007: ssphmPublicKeyEncrypt: 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
```

如果是 SSC 证书的 LAP, **debug pm pki enable** 这条命令的输出信息是这样的。 在输出信息中还能看到 SSC 的哈希值。

注意：因为空间的限制，有一些输出的信息被移到了下一行。

```
(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 >cscoDefaultNewRootCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 5, CA cert cscoDefaultMfgCaCert<
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 eb076940280cbed1 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
```

!--- 这是目前所使用的 SSC 密钥的 hash 值

```
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
```

从 LAP 端 Debug

如果控制器上的 debug 命令不能显示 join request，连接 AP 的 console 口，从 LAP 端 debug 加入过程。使用下列命令你可以看到 LAP 的启动过程，但是首先

你要确保在 enable 的模式下（默认密码是 Cisco）：

- **debug dhcp detail**—显示有关 DHCP 选项 43 的信息
- **debug ip udp**—显示发向控制器的 join/discovery 数据包以及 DHCP 和 DNS 查询（所有的这些都是 UDP 数据包，12223 端口是控制器的源端口）.
- **debug lwapp client event**—显示 AP 的 LWAPP 事件.
- **undebug all**—关闭 AP 端的 debug 信息.

这里有一个 **debug ip udp** 命令所输出的信息示例。看完这部分的输出信息，你将对 LAP 在启动时，向控制器发送 discovery 和 join 的数据包有所了解。

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

!--- AP 向先前注册的控制器发送 LWAPP discovery request


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

!--- LWAPP Discovery Request 使用静态配置的控制器信息

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

!--- LWAPP Discovery Request 使用子网内广播形式发送.


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

!--- LWAPP Join Request 向静态设定的控制器 AP 管理接口发送
```

避免 DHCP 相关的一些问题

LAP 在开始发现或寻找 WLC 的过程前需要通过 DHCP 来获得一个 IP 地址，可能会因为 DHCP 的相关参数配置错误而导致获得 IP 地址有问题。这一章节将讲解 DHCP 怎么和 WLC 一起工作，并提供一些最佳的建议来规避 DHCP 相关的一些问题。

为了启用 DHCP 功能，控制器的行为就像一个拥有 IP 帮助地址的路由器。更加确切的说，在控制器上填入网关的 IP 地址，控制器会通过 IP 单播包的形式将所有的 DHCP 请求转发到 DHCP 服务器上。

当 DHCP 服务器提供的 DHCP 信息返回到控制器上时，控制器会将 DHCP 服务

器的 IP 地址变成自己的虚拟 IP 地址。这么做的原因是当 Windows 操作系统的客户端在不同的 AP 间做漫游的时候，它所做的第一件事是连接 DHCP 服务器并更新地址。

由于 DHCP 服务器的地址被设成了 1.1.1.1(这是典型的控制器的虚拟 IP 地址)，控制器可以中途截获数据包并向 Windows 客户端做出快速的响应。

这也是为什么所有的控制器都具有相同的虚拟 IP 地址的原因。如果一个安装 Windows 操作系统的笔记本向另外一个控制器下的 AP 漫游，它会尝试连接另外一台控制器的虚拟接口。由于移动性事件的发生和背景的转移， Windows 客户端所需要漫游到的控制器已经有了所有该客户端的信息，并将相关信息回复给这个客户端。

如果你需要使用控制器上的 DHCP 服务，你需要做的是将管理接口的 IP 地址作为 DHCP 服务器地址，并将这个 IP 地址赋给你为子网创建的动态接口上。然后指派这个动态接口给无线局域网（以下简称 WLAN）。

控制器需要在每个子网里有一个 IP 地址，这个地址作为网关地址回应 DHCP 请求。

以下是在为 WLAN 配置 DHCP 服务器时需要记住的一些要点：

1. 在控制器上 DHCP 服务器的 IP 地址不应该属于任何动态子网，否则 DHCP 服务会被阻止，但是可以用这条命令来解决这个问题：

   config network mgmt-via-dynamic-interface 这条命令只在 4.0 版本上有效

   (3.2 版本里不能使用这条命令)

2. 控制器从动态接口收到 DHCP 请求，用单播的方式向 DHCP 服务器转发，并使用动态接口的 IP 地址作为发起者。确保防火墙上的策略允许该地址到 DHCP 服务器。
3. 需要确保防火墙上的策略允许 DHCP 服务器的回应能够到达控制器动态接口的地址。你可以通过在 DHCP 服务器上 ping 控制器的动态地址，或者以动态接口的网关 IP 地址作为源地址，ping DHCP 服务器来验证。
4. 确保控制器上动态接口的 IP 地址落在 DHCP 服务器上的一个 DHCP 范围内。
5. 最后，确认你不是在使用那些不能回应单播 DHCP 请求的 DHCP 服务器，比如说 PIX。

如果您不能解决你的 DHCP 问题，有 2 个解决方案可以选择：

- 尝试使用内部的 DHCP 服务器。将动态接口上的管理 IP 地址作为 DHCP 服务器地址，然后配置内部的 DHCP 分配池。一旦启用了 DHCP 范围，就作为 DHCP 服务器开始提供 DHCP 服务了。
- 通过输入以下命令（console 或者 SSH 方式登陆到 CLI 界面）来确认发出 DHCP 请求后，是否没有收到 DHCP 回应：

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

  这可以表明 DHCP 数据包已经发送出去了，但是控制器没有收到回应。

最后，考虑到安全性，在控制器上的 VLAN 或者的子网不建议将 LAP 包括在内，除非是在管理接口的子网内。

注意：RADIUS 服务器或 DHCP 服务器绝不能在控制器的任何动态接口的子网内。安全特性会将所有发向控制器的回包阻止。

为什么 LAP 不能加入到控制器？

症状 1：控制器的时间在 LAP 证书有效期之外

通过完成以下步骤来确定是否为此问题：

1. 使用 **debug lwapp errors enable** 和 **debug pm pki enable** 命令

   Debug lwapp event enable 这条命令用来显示 AP 和 WLC 之间的证书信息，该信息能够清晰的显示出证书被拒绝。

   注意：确保解决和世界标准时间之间的偏移

   这是控制器上的 **debug lwapp events enable** 命令的输出例子：

   注意：因为空间的限制，有一些输出的信息被移到了下一行。

```
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
```

这是控制器上 **debug pm pki enable** 命令的输出例子。该输出的信息遵循证书的确认过程。

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

注意：因为空间的限制，有一些输出的信息被移到了下一行。

```
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 verfied using
>cscoDefaultMfgCaCert<
 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)
```

这个信息清晰的显示了控制器上的时间在 LAP 证书的有效时间之外。因此，LAP 不能注册到控制器上。LAP 上的证书安装时有一个预先定义的有效时间。控制器上的时间应该设置在 LAP 证书的有效时间内。

2. 在控制器的 CLI 界面里输入 **show time** 命令，确认控制器上设定的日期和时间是否在 LAP 证书的有效时间内。如果控制器时间比 LAP 证书有效时间早或者迟的话，就需要改变控制器的时间，使时间落在 LAP 证书的有效时间内。

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.

注意：如果控制器上时间设置不正确，在控制器 GUI 界面，commands 菜单下选择 set time，或者在控制器 CLI 界面使用 config time 命令来调整控制器时间。

3. 登陆到 LAP，在 CLI 界面下使用 **show crypto ca certificates** 命令来确认证书的状态。

   这条命令可以让你确认 AP 上设置的证书有效时间。这里有一个例子：

```
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
  ...............................
  ................................
  ..................................
```

以上并不是全部的输出，因为和这条命令相关的有效时间非常多。你需要考虑的仅仅是相关 trustpoint 的有效时间：Cisco_IOS_MIC_cert 和在 name 栏里面的相关 AP 名字。在这个例子里，AP 的名字是：C1200-001563e50c7e。这是需要考虑的证书有效时间。

症状 2：管理域不匹配

以下是键入 **debug lwapp events enable** 后输出的讯息：

注意：因为空间的限制，有一些输出的信息被移到了下一行。

```
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, gtw 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
```

这个消息很清晰的指出 LAP 和 WLC 之间的管理域不匹配。WLC 可以同时支持多个管理域，但是在 LAP 加入到 WLC 之前必须先指定 LAP 对应的管理域。举个例子，WLC 使用-A 的管理域就只能接入管理域也是-A 的 AP。当你在购买 AP 和 WLC 的时候，必须确保他们能够工作在相同的管理域下。只有这样，LAP 才能注册到 WLC 上。

注意：在一个 LAP 上，802.11b/g 和 802.11a 无线模块必须在同一个管理域内。

症状 3: Error Message AP cannot join because the maximum number of APs on interface 2 is reached （错误信息，因为接口 2 上的 AP 数量已经最大，所以 AP 不能加入）

当 AP 试图加入到控制器的时候，你可能会看到这个错误信息：

```
Fri May 19 16:18:06 2006  [ERROR] spam_lrad.c 1553: spamProcessJoinRequest :
spamDecodeJoinReq failed
Fri May 19 16:18:06 2006  [ERROR] spam_lrad.c 4498: AP cannot join
because the maximum number of APs on interface 2 is reached.
```

4400 系列控制器每个端口默认最多支持 40 个 AP，当你尝试在控制器上连接超过 48 个 AP 的时候，你会看到这个错误信息。然而，你可以使用下面方法中的任意一个，使得 4400 系列控制器一个端口下（每端口）支持更多的 AP：

- 链路汇聚(控制器需要工作在 3 层模式)
- 多个 AP 管理接口(控制器需要工作在 3 层模式)
- 连接额外的端口(控制器需要工作在 2 层模式)

欲了解更多有关这方面的信息，参阅配置 4400 系列控制器使其可以支持超过 48 个接入点[Configuring a 4400 Series Controller to Support More Than 48 Access Points.]

症状 4：SCC 证书的 AP，控制器 SCC AP 的策略是未启用的

如果控制器上的 SCC 策略是未启用的，在控制器上使用 **debug lwapp events enable** 和 **debug pm pki enable** 这两条命令可以看到下面这个错误信息：

```
Wed Aug 9 17:20:21 2006 [ERROR] spam_lrad.c 1553: spamProcessJoinRequest :
spamDecodeJoinReq failed
Wed Aug 9 17:20:21 2006 [ERROR] spam_crypto.c 1509: Unable to free public key for
AP 00:12:44:B3:E5:60
Wed Aug 9 17:20:21 2006 [ERROR] spam_lrad.c 4880: LWAPP Join-Request does not
include valid certificate in CERTIFICATE_PAYLOAD from AP 00:12:44:b3:e5:60.
Wed Aug 9 17:20:21 2006 [CRITICAL] sshpmPkiApi.c 1493: Not configured to
accept Self-signed AP cert
```

完成以下的步骤，来解决这个问题：

执行其中一个即可：

- 在控制器 CLI 界面，输入 **show auth-list** 命令来确认控制器是否接受 SSC 证书的 AP。

  这里有一个例子：

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

Mac Addr	Cert Type	Key Hash
00:09:12:2a:2b:2c	SSC	1234567890123456789012345678901234567890

- 在 GUI 界面，security 菜单下选取 AP policies
  a. 确认 **Accept Self Signed Certificate** 这个选择框是否被激活，如果没有，启用这个功能。
  b. 在证书类型里，选择 SSC
  c. 在授权列表里加入 AP 的 MAC 地址和哈希密钥

哈希密钥可以使用 **debug pm pki enable** 这条命令获取。关于如何取得哈希密钥的信息，请参考症状 6 。

症状 5：WLC 上启用了 AP 授权列表，但是 LAP 不在授权列表里面

在这种情况下，在控制器上通过 debug lwapp events enable 这条命令你可以看到这种输出信息：

```
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
```

如果是使用带有 console 口的 LAP，可以通过 **debug lwapp client error** 这条命令可以看到这种信息：

```
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.
```

这个信息再一次清楚的表明，这个 LAP 不在控制器的授权列表里面。

你可以通过这个命令来看到 AP 授权列表的状态信息：

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

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

使用 **config auth-list add mic** *<AP MAC Address>*将 LAP 加入到 AP 的授权列表里。欲了解更多有关如何配置 LAP 授权的信息，参阅思科统一无线网络 LAP

授权配置实例[Lightweight Access Point (LAP) Authorization in a Cisco Unified Wireless Network Configuration Example]。

症状 6：SSC 证书的公有 hash 密钥错误或丢失

完成以下步骤来确定是否为此问题：

1. 输入 **debug lwapp events enable** 命令

   这条命令用来验证 AP 是否尝试加入到控制器中

2. 输入 **show auth-list** 命令

   这条命令用来显示控制器上已经保存的公有 hash 密钥

3. 输入 **debug pm pki enable** 命令

   这条命令用来显示目前的公有 hash 密钥。目前的公有密钥必须和控制器上储存的公有 hash 密钥一致。如果两者不一致，将会出现问题。这里有一个 debug 命令所输出信息的例子：

   注意：因为空间的限制，有一些输出的信息被移到了下一行。

```
(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 >cscoDefaultNewRootCaCert<
Mon May 22 06:34:10 2006: sshpmGetCID: comparing to row 5,
CA cert cscoDefaultMfgCaCert<
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 eb076940280cbed1 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
```

!---这是目前的 hash 密钥值

```
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
```

完成以下步骤来解决这个问题：

1. 拷贝 **debug pm pki enable** 这条命令中显示的公有 hash 密钥，使用该密钥来替代授权列表中的公有 hash 密钥。
2. 使用 **config auth-list add ssc** *AP_MAC AP_key 命令，将 AP 的 MAC 地址和公有 hash 密钥加入到授权列表里。*

   这条命令的配置实例:

注意：因为空间的限制，有一些输出的信息被移到了下一行。

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

症状 7：AP 上的证书或者公有密钥损坏

如果证书有问题，LAP 肯定不能加入到控制器上。

通过使用 **debug lwapp errors enable** 和 **debug pm pki enable** 这两条命令，你可以看到证书或者是密钥被损坏的相关信息。

注意：因为空间的限制，有一些输出的信息被移到了下一行。

```
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
```

可以通过下面 2 个方法中的任意 1 个来解决这个问题：

- MIC AP—请求 return materials authorization 即(RMA).
- SSC AP—可以降级到 Cisco IOS® 软件版本 12.3(7)JA.

  如果是一个 SSC 证书的 AP，可以用 mode 键将 IOS 软件灌到 AP 中。然后再通过 LWAPP 升级工具将胖 AP 变回瘦 AP 模式。这样做可以重新创建证书。

完成以下步骤以使 AP 降级：

1. 使用复位按键选项
2. 清除控制器配置
3. 重新升级

欲了解更多有关于 LAP 降级的信息，请参阅升级胖 AP 到瘦 AP[Upgrading Autonomous Cisco Aironet Access Points to Lightweight Mode.]

WCS 可以将 SSC 证书推送给 WLC。与了解更多有关于如果用 WCS 配置 AP，参阅 cisco 无线控制系统配置向导，5.1 版本 Configuring Access Points 章节

症状 8：控制器可能工作在 2 层模式

完成以下步骤来定位这个问题：

检查控制器的工作模式，更改 AP 的配置，使 AP 只支持基于 3 层的控制器发现，更改 AP 的配置，使 AP 不支持基于 2 层的控制器发现。

完成以下步骤来解决这个问题：

1. 将 WLC 设成 3 层模式
2. 重启并配置 AP 管理接口

   如果有服务端口，比如说 4402 或者 4404，你应该让服务端口和 AP 管理接口以及管理接口在不同的网段里。

症状 9：转换成 LWAPP 后，在 AP 上看到这种错误信息

你会看到这种错误信息：

```
*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
```

AP 在 30 秒后重启，然后重新开始这一过程。

完成以下步骤来解决这个过程：

1. 如果是一个 SSC 证书的 AP。将这个 AP 重新转换成使用自治 IOS 的胖 AP。
2. 使用 **write erase** 命令来清除配置然后重启。在重启的时候不要保存配置。

症状 10：控制器不在当前的 VLAN 里收到 AP 的 discovery 信息（你可以看到 discovery 信息，但是看不到 response）

你可以通过 **debug lwapp events enable** 命令看到这种信息：

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

这个信息的意思是，控制器通过广播 IP 地址收到一个 discovery request，但是发送 discovery request 的源地址并不在控制器上的任何一个子网内。也就是说，控制器会丢掉这个 discovery request 数据包。

造成这个问题的原因是 AP 并没有将 discovery request 发向管理接口的 IP 地址。控制器报告广播的 discovery request 从一个控制器上没有的 VLAN 发出。这种情况较为典型的发生在，在 trunks 上对 vlan 做允许，而不是将他们限制为无线 vlan 的时候。

完成以下步骤来解决这个问题：

1. 如果控制器在另外一个子网内，AP 必须事先知道控制器的 IP 地址，或者 AP 必须通过其它任意一个发现方法来获得控制器的 IP 地址。
2. 通过配置交换机，允许一些不在控制器上的 vlan 得以通过。在 trunks 上严格限定 vlan。

症状 11：1250 LAP 不能加入 WLC

思科 1250 AP 不能加入到软件版本为 4.1.185.0 的 2106 WLC 中。

WLC 的日志信息显示：

```
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.
```

解决方式：这是因为 cisco 1250 系列 LAP 不能支持 4.1 版本的 WLC 软件。1250 系列 AP 能够支持 4.2.61.0 及以后的控制器软件版本。为了解决这个问题，将控制器的软件升级到 4.2.61.0 及以后的版本。

症状 12：AP 不能加入 WLC，因为防火墙阻止了一些必要的端口

如果在企业网络里有部署防火墙，为了使 LAP 能够加入到控制器，并保证 LAP 和控制器之间通信正常，需要确保防火墙对以下端口予以通行。

- LWAPP 流量所需的 UDP 端口：
  - 数据 - 12222
  - 控制 - 12223
- 移动性流量所需的 UDP 端口：
  - 16666 - 16666
  - 16667 - 16667
- SNMP 所需的 TCP 端口：161 and 162（无线控制系统[WCS]所用）

下面的端口是可选（根据你的需求）：

- TFTP：UDP 69
- GUI 界面登陆所用的端口，HTTP 是 TCP 80 端口，HTTPS 是 TCP 443 端口
- CLI 界面登陆所用的端口：使用 telnet 或者 SSH 方式登陆，TCP 23 端口和 TCP 22 端口

Netpro 论坛- Featured Conversations

Networking Professionals Connection 这个论坛提供网络专业人员分享有关于网络解决方案、产品和技术的问题、建议以及信息。下面的链接是一些这方面经常被提及的话题。

NetPro Discussion Forums - Featured Conversations for Wireless
Wireless - Mobility: WLAN 射频标准
Users are not able to connect to WAP- Dec 18, 2008 Upgrading antennas - Dec 18, 2008 New Antenna - Dec 18, 2008 AIR LAP1131AG-N-K9 wont join 4400 series WLAN controller - Dec 18, 2008 Converting 1242AG-S-K9 to ETSI - Dec 12, 2008
Wireless - Mobility: 安全和网络管理
WCS/WLC upgrade options - Dec 20, 2008 AP in WCS maps doesnt exist - Dec 20, 2008 Missing RX neighbours in WCS - Dec 19, 2008 WDS not authenticated with WLSE - Dec 19, 2008 MS PEAP, IAS and AP1200s - Dec 19, 2008

Wireless - Mobility: 无线语音和视频
Cisco Controllers & Ascom i75 VoIP Phones - Dec 19, 2008 7920 Phones - No AP Found - with LAP1131AG - Dec 18, 2008 Nokia E65 connected to a Aironet 1250 - Dec 15, 2008 Massive wifi phones deauthentication - Dec 11, 2008 7921 stuck at Configuring IP - Dec 10, 2008
Wireless - Mobility: 开始使用无线
PEAP and IAS - Dec 19, 2008 WLC Management Page not responding - Dec 18, 2008 Port of WLC are not coming up - Dec 18, 2008 Which antenna do I need? - Dec 18, 2008 Wlan - Dec 18, 2008
Wireless - Mobility: 常规性资料
WCS Now Statistics for WLAN-Clinet available - Dec 20, 2008 WCS: Client AP Association History empty - Dec 20, 2008 Multiple VLANs on c1200 - Dec 20, 2008 Cellular Over Cisco WLAN - Dec 20, 2008 AP Wont Come Up - Dec 19, 2008

相关信息

- 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

【译者注】

1、AP 和 WLC 之间通信用的 discovery request、join request 之类的保留了英文；
2、Netpro 论坛的链接保留了英文；
3、正文用的是小四字体，因为空间的问题，输出的 log 信息用小五的字体