

Comprendre le client de débogage sur les contrôleurs LAN sans fil (WLC)

Table des matières

- [Introduction](#)
- [Conditions préalables](#)
- [Exigences](#)
- [Composants utilisés](#)
- [Conventions](#)
- [Client de débogage](#)
- [Déboguer les variantes du client](#)
- [Mobilité](#)
- [Dépannage de l'authentification EAP](#)
- [Connexion client](#)
- [Processus du contrôleur](#)
- [Module PEM \(Policy Enforcement Module\)](#)
- [Transfert du trafic client](#)
- [Fonctions des points d'accès \(APF\)](#)
- [Authentification 802.1x \(Dot1x\)](#)
- [Analyse du client de débogage](#)
- [DépannageExemples](#)
- [Configuration de chiffrement client incorrecte](#)
- [Clé pré-partagée incorrecte](#)
- [Informations connexes](#)

Introduction

Ce document décrit en détail les **debug client** sur les contrôleurs LAN sans fil (WLC).

Conditions préalables

Exigences

Ce document couvre les sujets suivants :

- Comment un client sans fil est géré
- Résolution des problèmes d'association et d'authentification de base

Le résultat à analyser couvre le scénario d'un réseau WPA à clé prépartagée (WPA-PSK).

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Comment configurer le WLC et le point d'accès léger (LAP) pour le fonctionnement de base
- Protocole LWAPP (Lightweight Access Point Protocol) et méthodes de sécurité sans fil
- Fonctionnement des processus d'authentification et d'association 802.11

Composants utilisés

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- WLC Cisco AireOS (8540, 5520, vWLC) qui exécute le microprogramme 8.5 ou 8.10.
- Points d'accès CAPWAP.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Conventions

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à Conventions relatives aux conseils techniques Cisco.

Client de débogage

La commande `debug client`

est une macro qui active huit commandes de débogage, plus un filtre sur l'adresse MAC fournie, de sorte que seuls les messages qui contiennent l'adresse MAC spécifiée sont affichés. Les huit commandes `debug` affichent les détails les plus importants sur l'association et l'authentification des clients. Le filtre aide dans les situations où il y a plusieurs clients sans fil. Des situations telles que lorsque trop de sortie est générée ou que le contrôleur est surchargé lorsque le débogage est activé sans le filtre.

Les informations collectées couvrent des détails importants sur l'association et l'authentification du client (avec deux exceptions mentionnées plus loin dans ce document).

Les commandes qui sont activées sont affichées dans ce résultat :

```
<#root>
```

```
(Cisco Controller) >
```

```
show debug
```

```
MAC address ..... 00:00:00:00:00:00
```

```
Debug Flags Enabled:
```

```
  dhcp packet enabled.  
  dot11 mobile enabled.  
  dot11 state enabled.  
  dot1x events enabled.  
  dot1x states enabled.  
  pem events enabled.  
  pem state enabled.
```

Ces commandes couvrent la négociation d'adresse, la machine d'état client 802.11, l'authentification 802.1x, le module PEM (Policy Enforcement Module) et la négociation d'adresse (DHCP).

Déboguer les variantes du client

Pour la plupart des scénarios, la `debug client`

est suffisante pour obtenir les informations nécessaires. Cependant, il y a deux situations importantes où un débogage supplémentaire est nécessaire :

- Mobilité (itinérance client entre les contrôleurs)
- Dépannage de l'authentification EAP

Mobilité

Dans ce cas, les débogages de mobilité doivent être activés après le `debug client` a été introduite afin d'obtenir des informations supplémentaires sur l'interaction du protocole de mobilité entre les contrôleurs.

Remarque : les détails de ce résultat sont traités dans d'autres documents.

Afin d'activer les débogages de mobilité, utilisez la `debug client` , puis utilisez la commande `debug mobility handoff enable` commande :

```
<#root>
```

```
(Cisco Controller) >
```

```
debug client 00:00:00:00:00:00
```

```
(Cisco Controller) >
```

```
debug mobility handoff enable
```

```
(Cisco Controller) >
```

```
show debug
```

```
MAC address ..... 00:00:00:00:00:00
```

```
Debug Flags Enabled:
```

```
  dhcp packet enabled.
```

```
  dot11 mobile enabled.
```

```
  dot11 state enabled
```

```
  dot1x events enabled.
```

```
  dot1x states enabled.
```

```
  mobility handoff enabled.
```

```
  pem events enabled.
```

```
  pem state enabled.
```

Dépannage de l'authentification EAP

Afin de dépanner l'interaction entre le WLC et le serveur d'authentification (RADIUS externe ou serveur EAP interne), utilisez la `debug AAA all enable` , qui affiche les détails requis. Cette commande est utilisée après le `debug client` et peut être combinée à d'autres commandes de débogage si nécessaire (par exemple, la commande `handoff`).

<#root>

(Cisco Controller) >

```
debug client 00:00:00:00:00:00
```

(Cisco Controller) >

```
debug aaa all enable
```

(Cisco Controller) >

```
show debug
```

```
MAC address ..... 00:00:00:00:00:00
```

```
Debug Flags Enabled:
```

```
aaa detail enabled.
```

```
aaa events enabled.
```

```
aaa packet enabled.
```

```
aaa packet enabled.
```

```
aaa ldap enabled.
```

```
aaa local-auth db enabled.
```

```
aaa local-auth eap framework errors enabled.
```

```
aaa local-auth eap framework events enabled.
```

```
aaa local-auth eap framework packets enabled.
```

```
aaa local-auth eap framework state machine enabled.
```

```
aaa local-auth eap method errors enabled.
```

```
aaa local-auth eap method events enabled.
```

```
aaa local-auth eap method packets enabled.
```

```
aaa local-auth eap method state machine enabled.
```

```
aaa local-auth shim enabled.
```

```
aaa tacacs enabled.
```

```
dhcp packet enabled.
```

```
dot11 mobile enabled.
```

```
dot11 state enabled
```

```
dot1x events enabled
```

```
dot1x states enabled.
```

```
mobility handoff enabled.
```

```
pem events enabled.
```

```
pem state enabled.
```

Connexion client

Aux fins de ce document, la *connexion client* est le processus pour qu'un client sans fil passe par ces étapes :

Section 802.11

1. Sondez, pour trouver un AP valide à associer.
2. Authentication : peut être ouvert (null) ou partagé. Normalement, l'option Ouvrir est sélectionnée.
3. Association : demandez des services de données au point d'accès.

Section Stratégies L2

1. Aucun ; l'authentification PSK ou EAP a lieu en fonction de la configuration.
2. Négociation de clé, si une méthode de cryptage est sélectionnée.

Section Stratégies C3

1. Adresse d'apprentissage.
2. Authentification Web, si cette option est sélectionnée.

Remarque : ces étapes représentent un sous-ensemble ou un résumé du processus complet. Ce document décrit un scénario simplifié qui couvre les politiques 802.11 et L2 et utilise WPA-PSK, ainsi que l'apprentissage des adresses. Aucune stratégie AAA ou L3 externe pour l'authentification n'est utilisée.

Processus du contrôleur

Dans chaque section, le contrôleur utilise des processus séparés afin de suivre l'état du client à chaque instant. Les processus interagissent entre eux pour garantir que le client est ajouté à la table de connexion (conformément aux stratégies de sécurité configurées). Afin de comprendre les étapes de connexion du client au contrôleur, voici un bref résumé des processus les plus pertinents :

- **Policy Enforcement Module (PEM)** - Contrôle l'état du client et le force à passer par chacune des stratégies de sécurité de la configuration WLAN.
- **Access Point Functions (APF)** - Fondamentalement, l'automate fini 802.11.
- **Dot1x** - Implémente la machine d'état pour 802.1x, l'authentification PSK et le handle de clé pour les clients sans fil.
- **Mobilité** - Suit l'interaction avec d'autres contrôleurs du même groupe de mobilité.
- **Data Transformation Layer (DTL)** - Se situe entre les composants logiciels et l'accélération matérielle du réseau (NPU) ; contrôle les informations ARP.

Module PEM (Policy Enforcement Module)

En fonction de la configuration WLAN, le client passe par une série d'étapes. Le PEM s'assure que cette opération est effectuée afin de se conformer aux politiques de sécurité C2 et C3 requises.

Voici un sous-ensemble des états PEM pertinents pour l'analyse d'un débogage client :

- **START** - État initial de la nouvelle entrée client.
- **AUTHCHECK** - Le WLAN a une stratégie d'authentification L2 à appliquer.
- **8021X_REQD** - Le client doit effectuer l'authentification 802.1x.
- **L2AUTHCOMPLETE** - Le client a terminé avec succès la stratégie L2. Le processus peut maintenant passer aux politiques de couche 3 (apprentissage d'adresses, authentification Web, etc.). Le contrôleur envoie ici l'annonce de mobilité pour apprendre les informations de couche 3 d'autres contrôleurs s'il s'agit d'une itinérance client dans le même groupe de mobilité.
- **WEP_REQD** - Le client doit terminer l'authentification WEP.
- **DHCP_REQD** - Le contrôleur doit apprendre l'adresse L3 du client, ce qui est fait soit par requête ARP, requête DHCP ou renouvellement, soit par des informations apprises d'un autre contrôleur dans le groupe de mobilité. Si DHCP Required est marqué sur le WLAN, seules les informations DHCP ou de mobilité sont utilisées.
- **WEBAUTH_REQD** - Le client doit terminer l'authentification Web. (politique de couche 3)
- **EXÉCUTER** - Le client a correctement appliqué les stratégies L2 et L3 requises et peut désormais transmettre le trafic au réseau.

Cette image montre une machine d'état PEM simplifiée avec les transitions du client jusqu'à ce qu'il atteigne l'état RUN, où le client peut maintenant envoyer du trafic au réseau :



Remarque : cette figure ne couvre pas toutes les transitions et tous les états possibles. Certaines étapes intermédiaires ont été supprimées pour plus de clarté.

Transfert du trafic client

Entre l'état START et avant l'état RUN final, le trafic client n'est pas transféré au réseau, mais est transmis au processeur principal sur le contrôleur pour analyse. Les informations qui sont transmises dépendent de l'état et des politiques en place ; par exemple, si 802.1x est activé, le trafic EAPOL est transféré au processeur. Par exemple, si l'authentification Web est utilisée, HTTP et DNS sont autorisés et interceptés par le processeur pour effectuer la redirection Web et obtenir les informations d'authentification du client.

Lorsque le client atteint l'état RUN, les informations du client sont envoyées à la NPU afin d'activer la commutation FastPath, qui effectue un transfert à débit de câble du trafic utilisateur vers le VLAN client et libère le CPU central des tâches de transfert de données utilisateur.

Le trafic qui est transféré dépend du type de client qui est appliqué à la NPU. Ce tableau décrit les types les plus pertinents :

Type	Description
1	Transfert de trafic client normal.

9	État d'â€™apprentissage IP. Un paquet de ce client est envoyé au processeur afin d'apprendre l'adresse IP utilisée.
2	Passthrough ACL. Utilisé lorsque le WLAN est une ACL configurée pour informer le NPU.

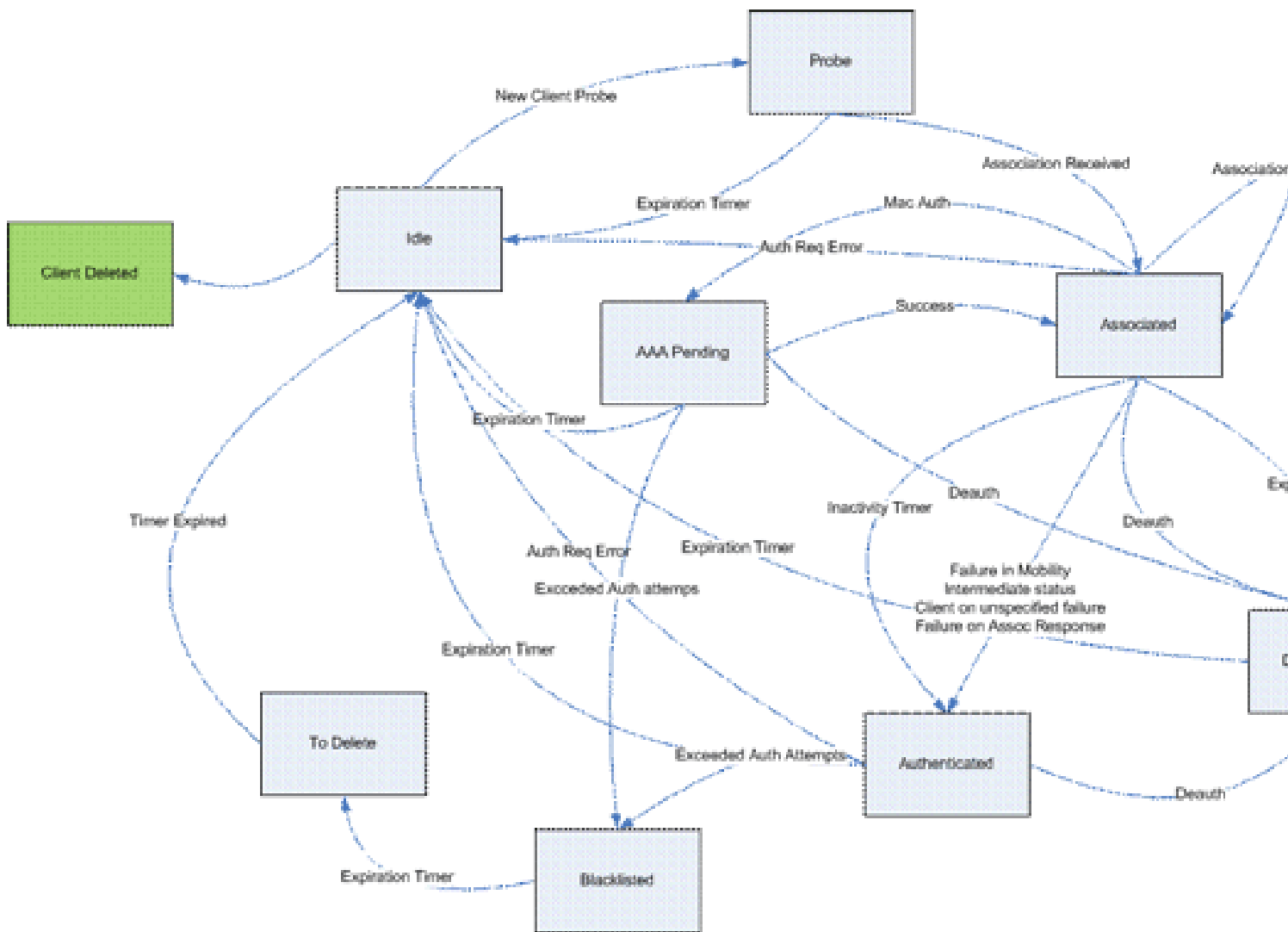
Fonctions des points d'accès (APF)

Ce processus gère l'état du client via l'état de la machine 802.11 et interagit avec le code de mobilité afin de valider les différents scénarios d'itinérance. Ce document ne couvre pas les détails de la mobilité ou ses états.

Ce tableau montre les états de client les plus pertinents qui peuvent se produire lorsqu'un client est associé au contrôleur :

Nom	Description
Inactif	Nouvel état client ou temporaire dans certaines situations.
Pendentif AAA	Le client attend l'authentification de l'adresse MAC.
Authentifié	Authentification ouverte réussie ou état intermédiaire dans certaines situations.
Associé	Le client a réussi les processus d'authentification MAC et d'authentification ouverte.
Dissocié	Le client a envoyé la désassociation/désauthentification ou le minuteur d'association a expiré.
Pour supprimer	Client marqué comme devant être supprimé (normalement après expiration du délai d'exclusion).
Sonde	Requête d'analyse reçue pour le nouveau client.
Exclus/Bloquer la liste	Le client a été marqué comme exclu. Normalement lié aux politiques WPS.
Non valide	Erreur sur l'état du client.

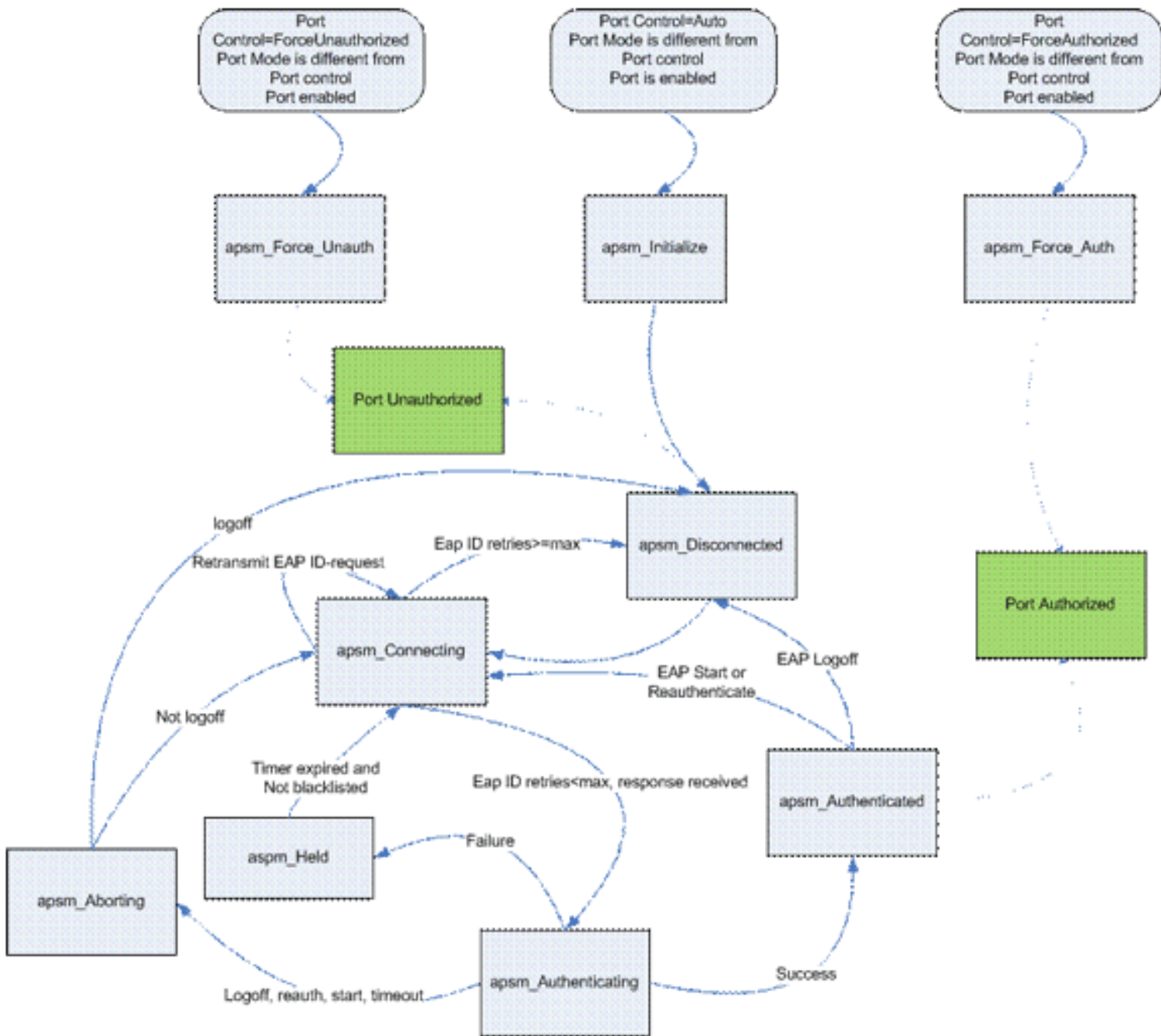
Cette image représente une transition de machine d'états et affiche uniquement les états et transitions les plus pertinents :



Authentication 802.1x (Dot1x)

Le processus Dot1x est responsable de l'authentification 802.1x et de la gestion des clés pour le client. Cela signifie que, même sur les WLAN qui n'ont pas de politique EAP qui nécessite 802.1x, dot1x participe pour gérer la création et la négociation de clé avec le client et aussi pour la gestion de clé mise en cache (PMK ou CCKM).

Cet ordinateur d'état affiche les transitions 802.1x complètes :



Analyse du client de débogage

Cette section montre le processus complet dans les journaux lorsqu'un client se connecte à un WLAN.

<#root>

APF Process

```
Wed Oct 31 10:46:13 2007: 00:1b:77:42:07:69 Adding mobile on LWAPP AP
00:1c:0j:ca:5f:c0(0)
```

```
!--- A new station is received. After validating type, it is added to the
!--- AP that received it. This can happen both on processing association
!--- request or probe requests
```

Wed Oct 31 10:46:13 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile Station: (callerId: 23) in 5 seconds

!--- Sets an expiration timer for this entry in case it does not progress beyond probe status. 5 Seconds corresponds to Probe Timeout. This message might appear with other time values since, during client processing, other functions might set different timeouts that depend on state.

Wed Oct 31 10:46:13 2007: 00:1b:77:42:07:69 apfProcessProbeReq (apf_80211.c:4057) Changing state for mobile 00:1b:77:42:07:69 on AP 00:1c:0j:ca:5f:c0 from Idle to Probe

!--- APF state machine is updated.

Wed Oct 31 10:46:13 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile Station: (callerId: 24) in 5 seconds

!--- New Probe request update sent AP about client. IMPORTANT: Access points do not forward all probe requests to the controller; they summarize per time interval (by default 500 msec). This information is used later by location and load balancing processes.

Wed Oct 31 10:46:14 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile Station: (callerId: 24) in 5 seconds

!--- New Probe request update sent AP about client.

Wed Oct 31 10:46:14 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile Station: (callerId: 24) in 5 seconds

!--- New Probe request update sent AP about client.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile Station: (callerId: 24) in 5 seconds

!--- New Probe request update sent AP about client.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Association received from mobile on AP 00:1c:0j:ca:5f:c0

!--- Access point reports an association request from the client. When the process reaches this point, the client is not excluded and not in mobility intermediate state

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 STA - rates (8): 140 18 152 36 176 72 96 108 0 0 0 0 0 0 0

!--- Controller saves the client supported rates into its connection table. Units are values of 500 kbps, basic (mandatory) rates have the Most Significant bit (MSb) set. The above would be 6mbps basic, 9, 12 basic, 18, 24 basic, 36, 48, 54

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Processing WPA IE type 221,
length 24 for mobile 00:1b:77:42:07:69

!--- Controller validates the 802.11i security information element.

PEM Process

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 START (0) Deleted mobile
LWAPP rule on AP [00:1c:0j:ca:5f:c0]

*!--- As the client requests new association, APF requests to PEM to delete the
!--- client state and remove any traffic forwarding rules that it could have.*

APF Process

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Updated location for station old
AP 00:00:00:00:00:00-0, new AP 00:1c:0j:ca:5f:c0-1

*!--- APF updates where this client is located. For example, this client is
!--- a new addition; therefore, no value exists for the old location.*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 START (0) Initializing
policy

*!--- PEM notifies that this is a new user. Security policies are checked
!--- for enforcement.*

PEM Process

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 START (0) Change state
to AUTHCHECK (2) last state AUTHCHECK (2)

!--- PEM marks as authentication check needed.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 AUTHCHECK (2) Change
state to 8021X_REQD (3) last state 8021X_REQD

*!--- After the WLAN configuration is checked, the client will need either
!--- 802.1x or PSK authentication*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 8021X_REQD (3) Plumbed
mobile LWAPP rule on AP 00:1c:0j:ca:5f:c0

*!--- PEM notifies the LWAPP component to add the new client on the AP with
!--- a list of negotiated capabilities, rates, Qos, etc.*

APF Process

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 apfPemAddUser2 (apf_policy.c:209)
Changing state for mobile 00:1b:77:42:07:69 on AP 00:1c:0j:ca:5f:c0 from
Probe to Associated

*!--- APF notifies that client has been moved successfully into associated
!--- state.*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Stopping deletion of Mobile
Station: (callerId: 48)

*!--- The expiration timer for client is removed, as now the session timeout
!--- is taking place. This is also part of the above notification
!--- (internal code callerId: 48).*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Sending Assoc Response to
station on BSSID 00:1c:0j:ca:5f:c0 (status 0)

!--- APF builds and sends the association response to client.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 apfProcessAssocReq
(apf_80211.c:3838) Changing state for mobile 00:1b:77:42:07:69 on AP
00:1c:0j:ca:5f:c0 from Associated to Associated

*!--- The association response was sent successfully; now APF keeps the
!--- client in associated state and sets the association timestamp on this point.*

Dot1x Process

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Creating a new PMK Cache Entry
for station 00:1b:77:42:07:69 (RSN 0)

*!--- APF calls Dot1x to allocate a new PMK cached entry for the client.
!--- RSN is disabled (zero value).*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Initiating WPA PSK to mobile
00:1b:77:42:07:69

!--- Dot1x signals a new WPA or WPA2 PSK exchange with mobile.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 dot1x - moving mobile
00:1b:77:42:07:69 into

Force Auth state

*!--- As no EAPOL authentication takes place, the client port is marked as
!--- forced Auth. Dot1x performs key negotiation with PSK clients only.*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Skipping EAP-Success to mobile
00:1b:77:42:07:69

*!--- For PSK, CCKM or RSN, the EAP success is not sent to client, as there
!--- was no EAPOL authentication taking place.*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Sending EAPOL-Key Message to
mobile
00:1b:77:42:07:69

state INITPMK (message 1), replay counter 00.00.00.00.00.00.00.00

*!--- Dot1x starts the exchange to arrive into PTK. PMK is known, as this
!--- is PSK auth. First message is ANonce.*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Received EAPOL-Key from mobile
00:1b:77:42:07:69

!--- Message received from client.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Received EAPOL-key in PKT_START
state (message 2) from mobile 00:1b:77:42:07:69

*!--- This signals the start of the validation of the second message
!--- from client (SNonce+MIC). No errors are shown, so process continues.
!--- Potential errors at this point could be: deflection attack (ACK bit
!--- not set on key), MIC errors, invalid key type, invalid key length, etc.*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Stopping retransmission timer
for mobile 00:1b:77:42:07:69

!--- Dot1x got an answer for message 1, so retransmission timeout is stopped.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Sending EAPOL-Key Message to
mobile 00:1b:77:42:07:69

state PTKINITNEGOTIATING (message 3), replay counter
00.00.00.00.00.00.00.01

!--- Derive PTK; send GTK + MIC.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Received EAPOL-Key from mobile
00:1b:77:42:07:69

!--- Message received from client.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 Received EAPOL-key in

PTKINITNEGOTIATING state (message 4) from mobile 00:1b:77:42:07:69

*!--- This signals the start of validation of message 4 (MIC), which
!--- means client installed the keys. Potential errors after this message
!--- are MIC validation errors, invalid key types, etc.*

PEM Process

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 8021X_REQD (3) Change state to L2AUTHCOMPLETE (4) last state L2AUTHCOMPLETE (4)

!--- PEM receives notification and signals the state machine to change to L2 authentication completed.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 L2AUTHCOMPLETE (4) Plumbed mobile LWAPP rule on AP 00:1c:0j:ca:5f:c0

!--- PEM pushes client status and keys to AP through LWAPP component.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 L2AUTHCOMPLETE (4) Change state to DHCP_REQD (7) last state DHCP_REQD (7)

>!--- PEM sets the client on address learning status.

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7) pemAdvanceState2 4238, Adding TMP rule

*!--- PEM signals NPU to allow DHCP/ARP traffic to be inspected by controller
!--- for the address learning.*

Wed Oct 31 10:46:15 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7) Adding Fast Path rule

type = Airespace AP - Learn IP address

on AP 00:1c:0j:ca:5f:c0, slot 1, interface = 1, QOS = 0

ACL Id = 255, Jumbo Frames = NO, 802.1P = 0, DSCP = 0, TokenID = 5006

*!--- Entry is built for client and prepared to be forwarded to NPU.
!--- Type is 9 (see the table in the Client Traffic Forwarding section of
!--- this document) to allow controller to learn the IP address.*

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7) Successfully plumbed mobile rule (ACL ID 255)

*!--- A new rule is successfully sent to internal queue to add the client
!--- to the NPU.*

Dot1x Process

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Stopping retransmission timer for mobile 00:1b:77:42:07:69

!--- Dot1x received message from client.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Sending EAPOL-Key Message to mobile 00:1b:77:42:07:69

state PTKINITDONE (message 5 - group), replay counter 00.00.00.00.00.00.00.02

!--- Group key update prepared for client.

PEM Process

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 Added NPU entry of type 9

!--- NPU reports that entry of type 9 is added (learning address state).

!--- See the table in the Client Traffic Forwarding section of this document.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Sent an XID frame

!--- No address known yet, so the controller sends only XID frame

!--- (destination broadcast, source client address, control 0xAF).

Dot1x Process

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Sent EAPOL-Key M5 for mobile 00:1b:77:42:07:69

!--- Key update sent.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Received EAPOL-Key from mobile 00:1b:77:42:07:69

!--- Key received.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Received EAPOL-key in REKEYNEGOTIATING state (message 6) from mobile 00:1b:77:42:07:69

!--- Successfully received group key update.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Stopping retransmission timer

for mobile 00:1b:77:42:07:69

!--- Group key timeout is removed.

DHCP Process

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 DHCP received op BOOTREQUEST
(1) (len 308, port 1, encap 0xec03)

!--- First DHCP message received from client.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 DHCP dropping packet due to
ongoing mobility handshake exchange, (siaddr 0.0.0.0, mobility
state = 'apfMsMmQueryRequested')

PEM Process

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7) mobility
role update request from Unassociated to Local

Peer = 0.0.0.0, Old Anchor = 0.0.0.0, New Anchor = 192.168.100.11

*!--- NPU is notified that this controller is the local anchor, so to
!--- terminate any previous mobility tunnel. As this is a new client,
!--- old address is empty.*

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7) State
Update from Mobility-Incomplete to Mobility-Complete, mobility
role=Local

!--- Role change was successful.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7)
pemAdvanceState2 3934, Adding TMP rule

*!--- Adding temporary rule to NPU for address learning now with new mobility
!--- role as local controller.*

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7)
Replacing Fast Path rule

type = Airespace AP - Learn IP address

on AP 00:1c:0j:ca:5f:c0, slot 1, interface = 1, QOS = 0

ACL Id = 255, Jumbo Frames = NO, 802.1P = 0, DSCP = 0, TokenID = 5006

!--- Entry is built.

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 DHCP_REQD (7)
Successfully plumbed mobile rule (ACL ID 255)

*!--- A new rule is successfully sent to internal queue to add the
!--- client to the NPU.*

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 0.0.0.0 Added NPU entry of type 9

*!--- Client is on address learning state; see the table in the
!--- Client Traffic Forwarding section of this document. Now mobility
!--- has finished.*

Wed Oct 31 10:46:19 2007: 00:1b:77:42:07:69 Sent an XID frame

*!--- No address known yet, so controller sends only XID frame (destination
!--- broadcast, source client address, control 0xAF).*

DHCP Process

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP received op BOOTREQUEST
(1) (len 308, port 1, encap 0xec03)

!--- DHCP request from client.

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP selecting relay 1 -
control block settings:

dhcpServer: 0.0.0.0, dhcpNetmask: 0.0.0.0,

dhcpGateway: 0.0.0.0, dhcpRelay: 0.0.0.0 VLAN: 0

*!--- Based on the WLAN configuration, the controller selects the identity to
!--- use to relay the DHCP messages.*

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP selected relay 1 -
192.168.100.254 (local address 192.168.100.11, gateway 192.168.100.254,
VLAN 100, port 1)

!--- Interface selected.

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
transmitting DHCP DISCOVER (1)

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
op: BOOTREQUEST, htype: Ethernet, hlen: 6, hops: 1

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
xid: 0xd3d3b6e9 (3553867497), secs: 1024, flags: 0

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP

chaddr: 00:1b:77:42:07:69

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
ciaddr: 0.0.0.0, yiaddr: 0.0.0.0

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
siaddr: 0.0.0.0, giaddr: 192.168.100.11

!--- Debug parsing of the frame sent. The most important fields are included.

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP sending REQUEST to
192.168.100.254 (len 350, port 1, vlan 100)

!--- DHCP request forwarded.

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP selecting relay 2 -
control block settings:

 dhcpServer: 0.0.0.0, dhcpNetmask: 0.0.0.0,
 dhcpGateway: 0.0.0.0, dhcpRelay: 192.168.100.11 VLAN: 100

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP selected relay 2 ? NONE

*!--- No secondary server configured, so no additional DHCP request are
!--- prepared (configuration dependant).*

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP received op BOOTREPLY (2)
(len 308, port 1, encap 0xec00)

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP setting server from OFFER
(server 192.168.100.254, yiaddr 192.168.100.105)

*!--- DHCP received for a known server. Controller discards any offer not on
!--- the DHCP server list for the WLAN/Interface.*

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP sending REPLY to STA
(len 416, port 1, vlan 100)

!--- After building the DHCP reply for client, it is sent to AP for forwarding.

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP transmitting DHCP OFFER (2)

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
op: BOOTREPLY, htype: Ethernet, hlen: 6, hops: 0

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
xid: 0xd3d3b6e9 (3553867497), secs: 0, flags: 0

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
chaddr: 00:1b:77:42:07:69

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
ciaddr: 0.0.0.0, yiaddr: 192.168.100.105

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
siaddr: 0.0.0.0, giaddr: 0.0.0.0

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP
server id: x.x.x.x rcvd server id: 192.168.100.254

!--- Debug parsing of the frame sent. The most important fields are included.

Wed Oct 31 10:46:21 2007: 00:1b:77:42:07:69 DHCP received op BOOTREQUEST (1)
(len 316, port 1, encap 0xec03)

!--- Client answers

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP selecting relay 1 -
control block settings:

dhcServer: 192.168.100.254, dhcpNetmask: 0.0.0.0,

dhcGateway: 0.0.0.0, dhcpRelay: 192.168.100.11 VLAN: 100

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP selected relay 1 -
192.168.100.254 (local address 192.168.100.11, gateway 192.168.100.254,
VLAN 100, port 1)

!--- DHCP relay selected per WLAN config

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP transmitting DHCP REQUEST (3)

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
op: BOOTREQUEST, htype: Ethernet, hlen: 6, hops: 1

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
xid: 0xd3d3b6e9 (3553867497), secs: 1024, flags: 0

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
chaddr: 00:1b:77:42:07:69

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
ciaddr: 0.0.0.0, yiaddr: 0.0.0.0

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
siaddr: 0.0.0.0, giaddr: 192.168.100.11

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
requested ip: 192.168.100.105

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
server id: 192.168.100.254 rcvd server id: x.x.x.x

!--- Debug parsing of the frame sent. The most important fields are included.

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP sending REQUEST to
192.168.100.254 (len 358, port 1, vlan 100)

!--- Request sent to server.

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP selecting relay 2 -
control block settings:

dhcpServer: 192.168.100.254, dhcpNetmask: 0.0.0.0,

dhcpGateway: 0.0.0.0, dhcpRelay: 192.168.100.11 VLAN: 100

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP selected relay 2 ? NONE

!--- No other DHCP server configured.

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP received op BOOTREPLY
(2) (len 308, port 1, encap 0xec00)

!--- Server sends a DHCP reply, most probably an ACK (see below).

PEM Process

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 192.168.100.105 DHCP_REQD
(7) Change state to RUN (20) last state RUN (20)

!--- DHCP negotiation successful, address is now known, and client

!--- is moved to RUN status.

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 192.168.100.105 RUN (20)
Reached PLUMBFASPATH: from line 4699

!--- No L3 security; client entry is sent to NPU.

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 192.168.100.105 RUN (20)
Replacing Fast Path rule

type = Airespace AP Client

on AP 00:1c:0j:ca:5f:c0, slot 1, interface = 1, QOS = 0

ACL Id = 255, Jumbo Frames = NO, 802.1P = 0, DSCP = 0, TokenID = 5006

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 192.168.100.105 RUN (20)
Successfully plumbed mobile rule (ACL ID 255)

DHCP Process

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 Assigning Address
192.168.100.105 to mobile

Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP sending REPLY to STA
(len 416, port 1, vlan 100)

```
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP transmitting DHCP ACK (5)
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
  op: BOOTREPLY, htype: Ethernet, hlen: 6, hops: 0
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
  xid: 0xd3d3b6e9 (3553867497), secs: 0, flags: 0
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
  chaddr: 00:1b:77:42:07:69
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
  ciaddr: 0.0.0.0, yiaddr: 192.168.100.105
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
  siaddr: 0.0.0.0, giaddr: 0.0.0.0
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 DHCP
  server id: x.x.x.x rcvd server id: 192.168.100.254
```

PEM Process

```
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 192.168.100.105 Added NPU
  entry of type 1
```

```
!--- Client is now successfully associated to controller.
!--- Type is 1; see the table in the Client Traffic Forwarding
!--- section of this document.
```

```
Wed Oct 31 10:46:25 2007: 00:1b:77:42:07:69 Sending a gratuitous ARP for
  192.168.100.105, VLAN Id 100
```

```
!--- As address is known, gratuitous ARP is sent to notify.
```

Exemples de dépannage

Configuration de chiffrement client incorrecte

Cet exemple montre un client avec des capacités différentes de celles du point d'accès. Le client sonde le SSID, mais comme la demande de sonde montre certains paramètres non pris en charge, le client ne passe jamais aux phases d'authentification/association.

En particulier, le problème introduit était une non-correspondance entre le client qui utilise WPA, et le point d'accès annonçant uniquement la prise en charge WPA2 :

```
Wed Oct 31 10:51:37 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
  Station: (callerId: 23) in 5 seconds
Wed Oct 31 10:51:37 2007: 00:1b:77:42:07:69 apfProcessProbeReq
  (apf_80211.c:4057) Changing state for mobile 00:1b:77:42:07:69 on AP
```

00:1c:b0:ea:5f:c0 from Idle to Probe

!--- Controller adds the new client, moving into probing status

```
Wed Oct 31 10:51:37 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:38 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:38 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
```

!--- AP is reporting probe activity every 500 ms as configured

```
Wed Oct 31 10:51:41 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:41 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:41 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:41 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:44 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:44 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:44 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:44 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:51:49 2007: 00:1b:77:42:07:69 apfMsExpireCallback (apf_ms.c:433)
Expiring Mobile!
Wed Oct 31 10:51:49 2007: 00:1b:77:42:07:69 0.0.0.0 START (0) Deleted mobile
LWAPP rule on AP [00:1c:b0:ea:5f:c0]
Wed Oct 31 10:51:49 2007: 00:1b:77:42:07:69 Deleting mobile on AP
00:1c:b0:ea:5f:c0(0)
```

*!--- After 5 seconds of inactivity, client is deleted, never moved into
!--- authentication or association phases.*

Clé pré-partagée incorrecte

Cela montre que le client tente de s'authentifier par WPA-PSK auprès de l'infrastructure, mais échoue en raison d'une non-correspondance de la clé pré-partagée entre le client et le contrôleur, ce qui entraîne l'ajout éventuel du client à la liste d'exclusion (de blocage) :

```
Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Adding mobile on LWAPP AP
00:1c:b0:ea:5f:c0(0)
Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 23) in 5 seconds
Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 apfProcessProbeReq (apf_80211.c:
4057) Changing state for mobile 00:1b:77:42:07:69 on AP 00:1c:b0:ea:5f:c0
from Idle to Probe
Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile
Station: (callerId: 24) in 5 seconds
Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Association received from mobile
on AP 00:1c:b0:ea:5f:c0
```

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 STA - rates (8): 130 132 139 150
12 18 24 36 0 0 0 0 0 0 0

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 STA - rates (12): 130 132 139 150
12 18 24 36 48 72 96 108 0 0 0 0

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Processing WPA IE type 221,
length 24 for mobile 00:1b:77:42:07:69

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 0.0.0.0 START (0)
Initializing policy

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 0.0.0.0 START (0) Change state to
AUTHCHECK (2) last state AUTHCHECK (2)

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 0.0.0.0 AUTHCHECK (2) Change
state to 8021X_REQD (3) last state 8021X_REQD (3)

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 0.0.0.0 8021X_REQD (3) Plumbed
mobile LWAPP rule on AP 00:1c:b0:ea:5f:c0

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 apfPemAddUser2 (apf_policy.c:209)
Changing state for mobile 00:1b:77:42:07:69 on AP 00:1c:b0:ea:5f:c0 from
Probe to Associated

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Stopping deletion of Mobile
Station: (callerId: 48)

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Sending Assoc Response to station
on BSSID 00:1c:b0:ea:5f:c0 (status 0)

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 apfProcessAssocReq (apf_80211.c:
3838) Changing state for mobile 00:1b:77:42:07:69 on AP 00:1c:b0:ea:5f:c0
from Associated to Associated

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Creating a new PMK Cache Entry
for station 00:1b:77:42:07:69 (RSN 0)

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Initiating WPA PSK to mobile
00:1b:77:42:07:69

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 dot1x - moving mobile
00:1b:77:42:07:69 into Force Auth state

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Skipping EAP-Success to mobile
00:1b:77:42:07:69

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Sending EAPOL-Key Message to
mobile 00:1b:77:42:07:69
state INITPMK (message 1), replay counter 00.00.00.00.00.00.00.00

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Received EAPOL-Key from mobile
00:1b:77:42:07:69

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Received EAPOL-key in PKT_START
state (message 2) from mobile 00:1b:77:42:07:69

Wed Oct 31 10:55:55 2007: 00:1b:77:42:07:69 Received EAPOL-key M2 with
invalid MIC from mobile 00:1b:77:42:07:69

Wed Oct 31 10:55:56 2007: 00:1b:77:42:07:69 802.1x 'timeoutEvt' Timer expired
for station 00:1b:77:42:07:69

Wed Oct 31 10:55:56 2007: 00:1b:77:42:07:69 Retransmit 1 of EAPOL-Key M1
(length 99) for mobile 00:1b:77:42:07:69

Wed Oct 31 10:55:56 2007: 00:1b:77:42:07:69 Received EAPOL-Key from mobile
00:1b:77:42:07:69

Wed Oct 31 10:55:56 2007: 00:1b:77:42:07:69 Received EAPOL-key in PKT_START
state (message 2) from mobile 00:1b:77:42:07:69

Wed Oct 31 10:55:56 2007: 00:1b:77:42:07:69 Received EAPOL-key M2 with invalid
MIC from mobile 00:1b:77:42:07:69

!--- MIC error due to wrong preshared key

Wed Oct 31 10:55:57 2007: 00:1b:77:42:07:69 802.1x 'timeoutEvt' Timer expired
for station 00:1b:77:42:07:69

Wed Oct 31 10:55:57 2007: 00:1b:77:42:07:69 Retransmit 2 of EAPOL-Key M1
(length 99) for mobile 00:1b:77:42:07:69

Wed Oct 31 10:55:57 2007: 00:1b:77:42:07:69 Received EAPOL-Key from mobile
00:1b:77:42:07:69

Wed Oct 31 10:55:57 2007: 00:1b:77:42:07:69 Received EAPOL-key in PKT_START
state (message 2) from mobile 00:1b:77:42:07:69

Wed Oct 31 10:55:57 2007: 00:1b:77:42:07:69 Received EAPOL-key M2 with invalid MIC from mobile 00:1b:77:42:07:69
Wed Oct 31 10:55:58 2007: 00:1b:77:42:07:69 802.1x 'timeoutEvt' Timer expired for station 00:1b:77:42:07:69
Wed Oct 31 10:55:58 2007: 00:1b:77:42:07:69 Retransmit failure for EAPOL-Key M1 to mobile 00:1b:77:42:07:69, retransmit count 3, mscb deauth count 0
Wed Oct 31 10:55:58 2007: 00:1b:77:42:07:69 Sent Deauthenticate to mobile on BSSID 00:1c:b0:ea:5f:c0 slot 0(caller 1x_ptsm.c:462)

!--- Client is deauthenticated, after three retries

!--- The process is repeated three times, until client is block listed

Wed Oct 31 10:56:10 2007: 00:1b:77:42:07:69 Block listing (if enabled) mobile 00:1b:77:42:07:69
Wed Oct 31 10:56:10 2007: 00:1b:77:42:07:69 apfBlacklistMobileStationEntry2 (apf_ms.c:3560) Changing state for mobile 00:1b:77:42:07:69 on AP 00:1c:b0:ea:5f:c0 from Associated to Exclusion-list (1)
Wed Oct 31 10:56:10 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile Station: (callerId: 44) in 10 seconds
Wed Oct 31 10:56:10 2007: 00:1b:77:42:07:69 0.0.0.0 8021X_REQD (3) Change state to START (0) last state 8021X_REQD (3)
Wed Oct 31 10:56:10 2007: 00:1b:77:42:07:69 0.0.0.0 START (0) Reached FAILURE: from line 3522
Wed Oct 31 10:56:10 2007: 00:1b:77:42:07:69 Scheduling deletion of Mobile Station: (callerId: 9) in 10 seconds

Informations connexes

- [Assistance technique et téléchargements Cisco](#)

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.