Grundlegendes zum Debug-Client auf Wireless LAN-Controllern (WLCs)

Inhalt

Einleitung

In diesem Dokument werden detaillierte Informationen zu **debug client** Befehlsausgabe auf Wireless LAN Controllern (WLC).

Voraussetzungen

Anforderungen

In diesem Dokument werden folgende Themen behandelt:

- Vorgehensweise bei einem Wireless-Client
- Beheben von grundlegenden Assoziations- und Authentifizierungsproblemen

Die zu analysierende Ausgabe deckt das Szenario für ein WPA Pre-Shared Key (WPA-PSK)-Netzwerk ab.

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- Konfigurieren des WLC und des Lightweight Access Point (LAP) für den Grundbetrieb
- LWAPP (Lightweight Access Point Protocol) und Wireless-Sicherheitsverfahren
- Funktionsweise der 802.11-Authentifizierungs- und -Zuordnungsprozesse

Verwendete Komponenten

Die Informationen in diesem Dokument basierend auf folgenden Software- und Hardware-Versionen:

- Cisco AireOS WLCs (8540, 5520, vWLC) mit der Firmware 8.5 oder 8.10
- CAPWAP-basierte Access Points

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die möglichen Auswirkungen aller Befehle kennen.

Konventionen

Weitere Informationen zu Dokumentkonventionen finden Sie unter Cisco Technical Tips Conventions (Technische Tipps von Cisco zu Konventionen).

Debugclient

Der Befehl **debug client**
ist ein Makro, das acht Debugbefehle sowie einen Filter für die bereitgestellte MAC-Adresse aktiviert, sodass nur Meldungen angezeigt werden, die die angegebene MAC-Adresse enthalten. Die acht Debug-Befehle zeigen die wichtigsten Details zur Client-Zuordnung und -Authentifizierung an. Der Filter unterstützt Situationen, in denen mehrere Wireless-Clients vorhanden sind. Situationen, in denen z. B. zu viel ausgegeben wird oder der Controller überlastet ist, wenn das Debuggen ohne den Filter aktiviert ist.

Die gesammelten Informationen umfassen wichtige Details zur Client-Zuordnung und -Authentifizierung (mit zwei Ausnahmen, die weiter unten in diesem Dokument erwähnt werden).

Die aktivierten Befehle werden in der folgenden Ausgabe angezeigt:

<#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.
```

Diese Befehle umfassen Adressverhandlung, 802.11-Clientstatuscomputer, 802.1x-Authentifizierung, Richtliniendurchsetzungsmodul (PEM) und Adressverhandlung (DHCP).

Clientvariationen debuggen

In den meisten Szenarien wird **debug client**

genügt, um die erforderlichen Informationen zu erhalten. Es gibt jedoch zwei wichtige Situationen, in denen zusätzliches Debuggen erforderlich ist:

- Mobilität (Client-Roaming zwischen Controllern)
- Fehlerbehebung: EAP-Authentifizierung

Mobilität

In diesem Fall müssen Mobilitätsdebugs nach dem **debug client**
wurde eingeführt, um zusätzliche Informationen über die Interaktion des Mobilitätsprotokolls zwischen Controllern zu erhalten.

Hinweis: Details zu dieser Ausgabe werden in anderen Dokumenten behandelt.

Um Mobilitätsdebugs zu aktivieren, verwenden Sie die **debug client** und verwenden Sie dann den **debug mobility handoff enable** command:

<#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.
```

Fehlerbehebung: EAP-Authentifizierung

Um eine Fehlerbehebung für die Interaktion zwischen dem WLC und dem Authentifizierungsserver (externer RADIUS- oder interner EAP-Server) durchzuführen, verwenden Sie **debug AAA all enable** -Befehl, der die erforderlichen Details anzeigt. Dieser Befehl wird nach dem **debug client** -Befehls und kann bei Bedarf mit anderen Debug-Befehlen kombiniert werden (z. B. **handoff** Befehl).

<#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.
```

Client-Verbindung

Für die Zwecke dieses Dokuments ist *Client-Verbindung* der Prozess, den ein Wireless-Client durch die folgenden Schritte durchläuft:

Abschnitt 802.11

1. Prüfen Sie, ob ein gültiger Access Point für die Zuordnung vorhanden ist.
2. Authentifizierung: Kann offen (null) oder freigegeben sein. Normalerweise ist "Öffnen" ausgewählt.
3. Zuordnung: Anfordern von Datendiensten an den Access Point

Abschnitt "L2-Richtlinien"

1. Keine; PSK- oder EAP-Authentifizierung erfolgt auf Basis der Konfiguration.
2. Schlüsselaushandlung, wenn eine Verschlüsselungsmethode ausgewählt ist.

Abschnitt "L3-Richtlinien"

1. Address Learning
2. Web-Authentifizierung, falls ausgewählt.

> **Hinweis**: Diese Schritte stellen eine Teilmenge oder Zusammenfassung des gesamten Prozesses dar. In diesem Dokument wird ein vereinfachtes Szenario beschrieben, das 802.11- und L2-Richtlinien abdeckt und WPA-PSK sowie Address Learning verwendet. Es werden keine externen AAA- oder L3-Richtlinien für die Authentifizierung verwendet.

Controller-Prozesse

In jedem Abschnitt verwendet der Controller getrennte Prozesse, um den Zustand des Clients zu jedem Zeitpunkt zu verfolgen. Die Prozesse interagieren untereinander, um sicherzustellen, dass der Client der Verbindungstabelle hinzugefügt wird (gemäß den konfigurierten Sicherheitsrichtlinien). Um die Schritte zum Verbinden des Clients mit dem Controller zu verstehen, gibt es eine kurze Zusammenfassung der wichtigsten Prozesse:

- **Policy Enforcement Module (PEM)** - Steuert den Client-Status und erzwingt diesen über jede der Sicherheitsrichtlinien in der WLAN-Konfiguration.
- **Access Point Functions (APF**) - Im Wesentlichen der 802.11-Zustandsrechner.
- **Dot1x** - Implementiert den Statuscomputer für 802.1x, PSK-Authentifizierung und Schlüsselhandle für die Wireless-Clients.
- **Mobilität** - Nachverfolgung der Interaktion mit anderen Controllern derselben Mobilitätsgruppe
- **DTL (Data Transformation Layer**): Diese Schicht wird zwischen den Softwarekomponenten und der NPU (Network Hardware Acceleration) angeordnet und steuert die ARP-Informationen.

Richtliniendurchsetzungs-Modul (PEM)

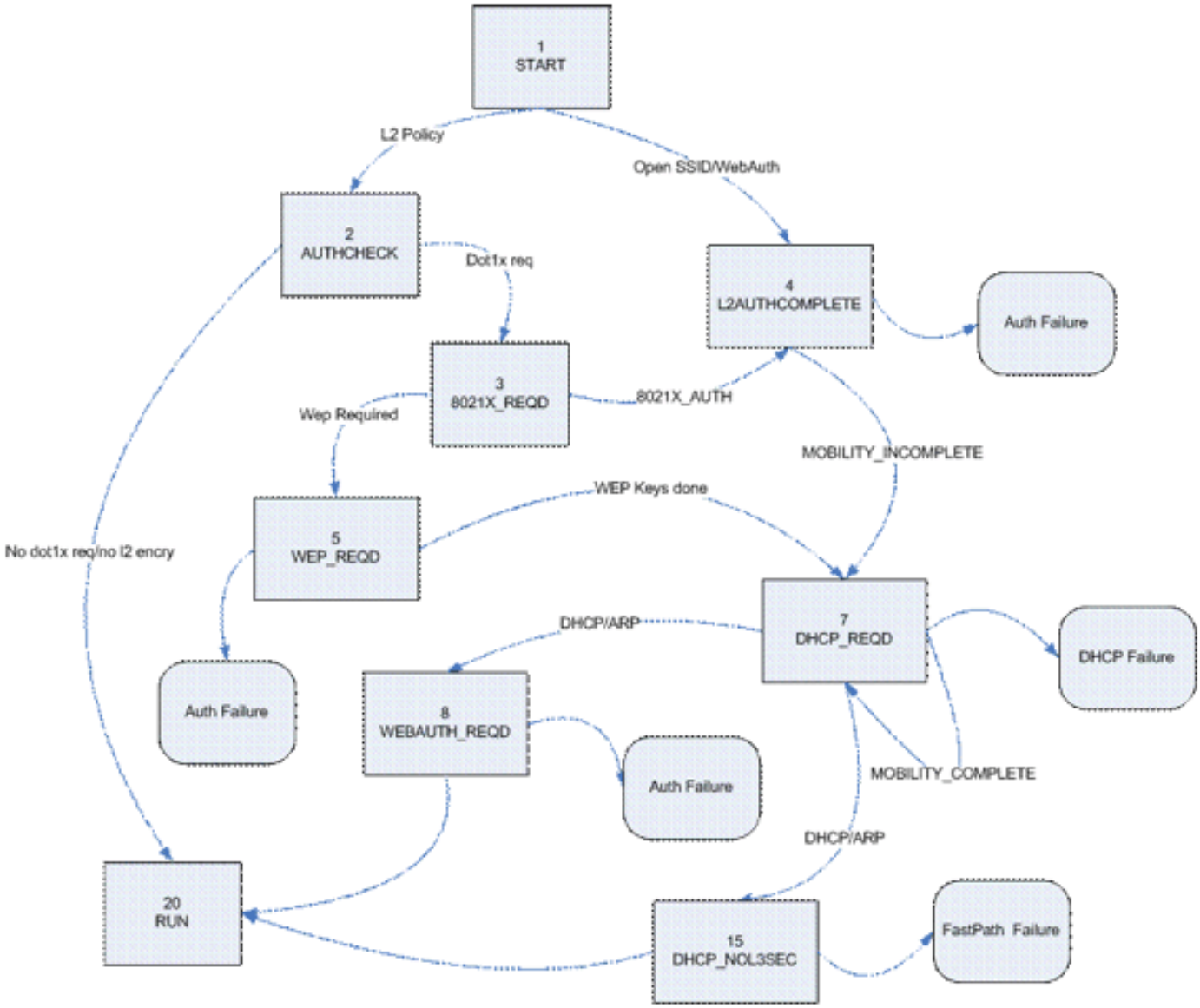
Je nach WLAN-Konfiguration durchläuft der Client eine Reihe von Schritten. PEM stellt sicher, dass dies geschieht, um die erforderlichen L2- und L3-Sicherheitsrichtlinien zu erfüllen.

Nachfolgend finden Sie eine Teilmenge der PEM-Zustände, die für die Analyse eines Client-Debugs relevant sind:

- **START** - Anfangsstatus für neuen Clienteintrag.
- **AUTHCHECK** - Das WLAN muss eine L2-Authentifizierungsrichtlinie durchsetzen.
- **8021X_REQD** - Der Client muss die 802.1x-Authentifizierung abschließen.
- **L2AUTHCOMPLETE** - Der Client hat die L2-Richtlinie erfolgreich abgeschlossen. Der Prozess kann jetzt mit L3-Richtlinien (Adresslernen, Webauthentifizierung usw.) fortfahren. Der Controller sendet hier die Mobilitätsankündigung, um L3-Informationen von anderen Controllern abzurufen, wenn es sich um einen Client-Roam in derselben Mobilitätsgruppe handelt.
- **WEP_REQD** - Der Client muss die WEP-Authentifizierung abschließen.
- **DHCP_REQD** - Der Controller muss die L3-Adresse vom Client abrufen. Dies geschieht entweder durch ARP-Anforderung, DHCP-Anforderung oder -Erneuerung oder durch Informationen, die er von einem anderen Controller in der Mobilitätsgruppe erhält. Wenn DHCP Required im WLAN markiert ist, werden nur DHCP- oder Mobilitätsinformationen verwendet.
- **WEBAUTH_REQD** - Der Client muss die Webauthentifizierung abschließen. (L3-Richtlinie)
- **RUN** - Der Client hat die erforderlichen L2- und L3-Richtlinien erfolgreich abgeschlossen und kann nun Datenverkehr an das Netzwerk übertragen.

Dieses Bild zeigt ein vereinfachtes PEM-Statussystem mit den Client-Übergängen bis zum Erreichen des

RUN-Status, bei dem der Client nun Datenverkehr an das Netzwerk senden kann:



Hinweis: Diese Zahl deckt nicht alle möglichen Übergänge und Zustände ab. Einige Zwischenschritte wurden der Übersichtlichkeit halber entfernt.

Weiterleitung von Client-Datenverkehr

Zwischen dem START-Status und dem letzten RUN-Status wird der Client-Datenverkehr nicht an das Netzwerk weitergeleitet, sondern zur Analyse an die Haupt-CPU auf dem Controller weitergeleitet. Die weitergeleiteten Informationen hängen vom Status und den geltenden Richtlinien ab. Wenn z. B. 802.1x aktiviert ist, wird der EAPOL-Datenverkehr an die CPU weitergeleitet. Ein weiteres Beispiel ist, dass bei Verwendung von Web Auth HTTP und DNS von der CPU zugelassen und abgefangen werden, um die Web-Umleitung durchzuführen und Client-Authentifizierungsdaten abzurufen.

Wenn der Client den RUN-Status erreicht, werden die Client-Informationen an die NPU gesendet, um FastPath-Switching zu aktivieren, das eine Weiterleitung des Benutzerdatenverkehrs mit Leitungsgeschwindigkeit an das Client-VLAN durchführt und die zentrale CPU von Benutzerdatenweiterleitungsaufgaben befreit.

Der weitergeleitete Datenverkehr hängt vom Client-Typ ab, der auf die NPU angewendet wird. In dieser

Tabelle werden die relevantesten Typen beschrieben:

Typ	Beschreibung
1	Normale Weiterleitung des Client-Datenverkehrs.
9	IP-Lernstatus. Ein Paket von diesem Client wird an die CPU gesendet, um die verwendete IP-Adresse zu ermitteln.
2	ACL-Passthrough. Wird verwendet, wenn es sich bei dem WLAN um eine ACL handelt, die zur Information der NPU konfiguriert wurde.

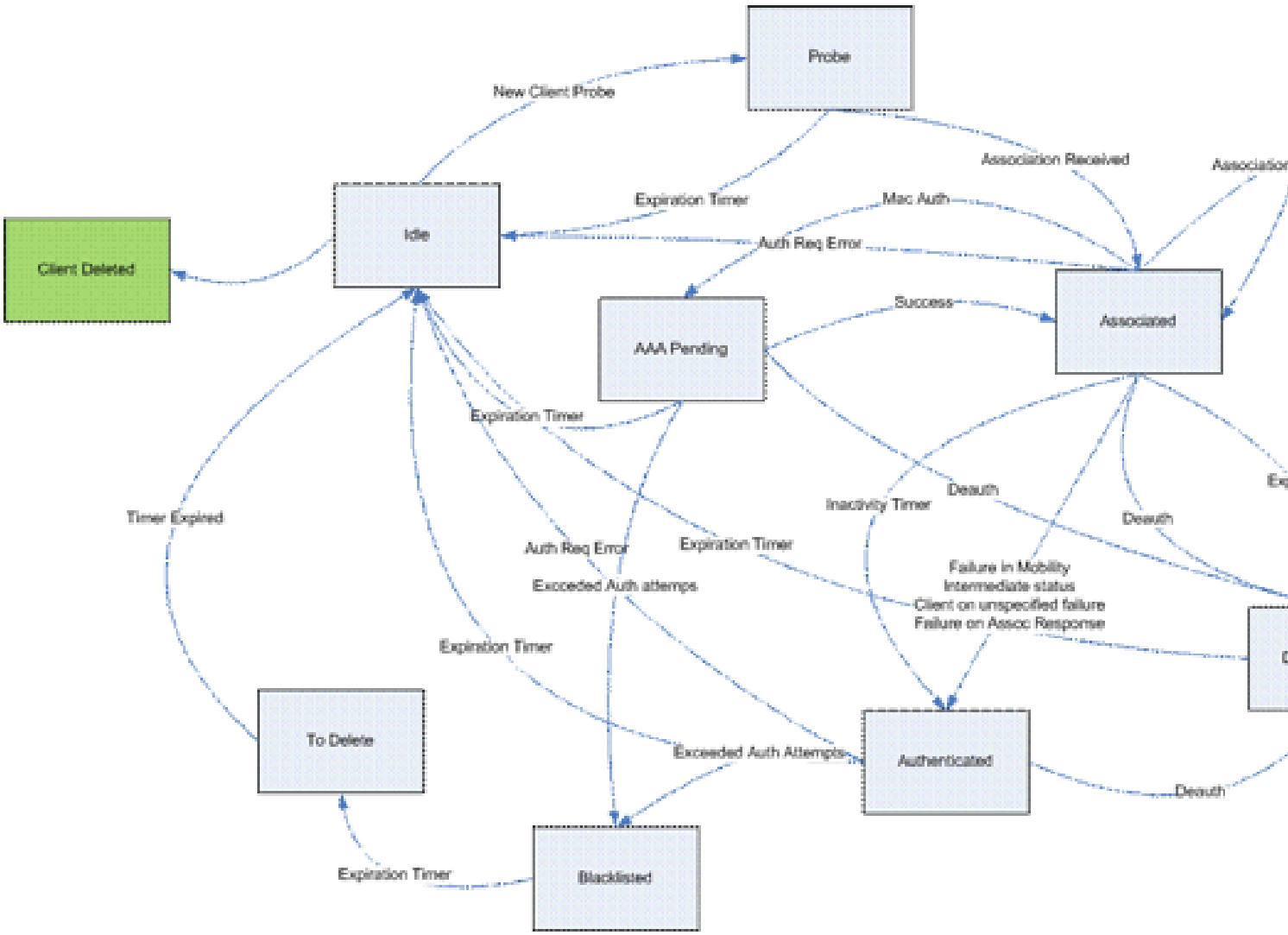
Access Point-Funktionen (APF)

Dieser Prozess behandelt den Client-Zustand über den 802.11-Computerzustand und interagiert mit Mobilitätscode, um die verschiedenen Roam-Szenarien zu validieren. In diesem Dokument werden weder die Mobilitätsdetails noch deren Status behandelt.

Diese Tabelle zeigt die relevanteren Client-Status, die auftreten können, wenn ein Client dem Controller zugeordnet wird:

Name	Beschreibung
Inaktivität	Neuer Client- oder temporärer Status in einigen Situationen.
AAA-Anhänger	Der Client wartet auf die MAC-Adressauthentifizierung.
Authentifiziert	In einigen Situationen ist die Authentifizierung erfolgreich oder im Zwischenzustand.
Zugeordnet	Der Client hat die MAC-Auth- und Open-Auth-Prozesse erfolgreich bestanden.
Getrennt	Der Client hat den Trennungs-/Deauthentifizierungs-Timer gesendet, oder der Zuordnungs-Timer ist abgelaufen.
Löschen	Client, der als gelöscht markiert wurde (normalerweise nach Ablauf des Ausschlusszeitgebers).
Sonde	Testanforderung für neuen Client empfangen.
Ausgeschlossen/Blockiert aufgeführt	Der Client wurde als ausgeschlossen markiert. Normalerweise bezogen auf WPS-Richtlinien.
Ungültig	Fehler beim Client-Status.

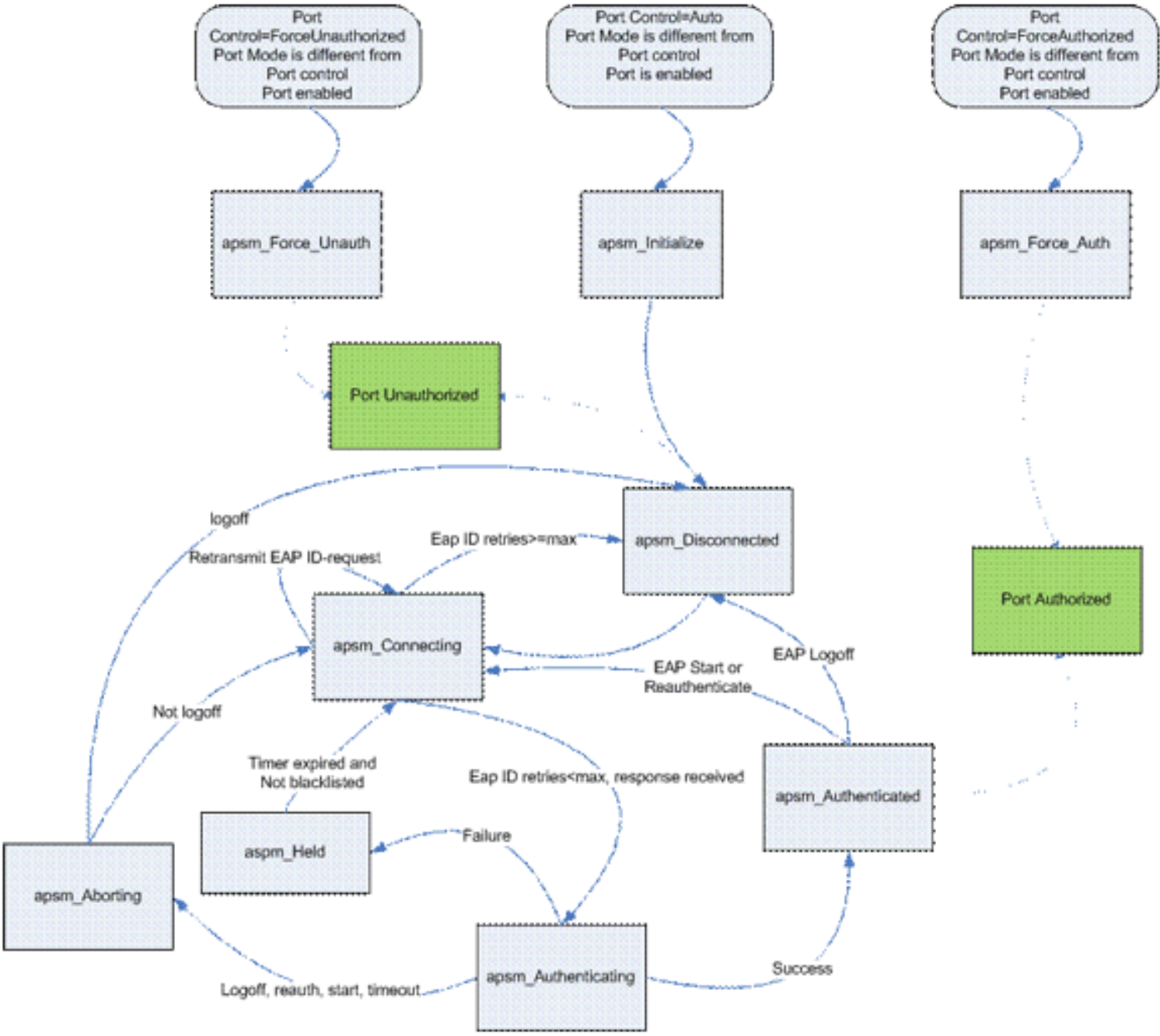
Dieses Bild stellt einen Zustandsmaschinenübergang dar und zeigt nur die wichtigsten Zustände und Übergänge:



802.1x-Authentifizierung (dot1x)

Der Dot1x-Prozess ist für die 802.1x-Authentifizierung und die Schlüsselverwaltung für den Client verantwortlich. Das bedeutet, dass selbst in WLANs, die keine EAP-Richtlinie haben, die 802.1x erfordert, dot1x für die Schlüsselerstellung und -verhandlung mit dem Client sowie für die Verarbeitung von zwischengespeicherten Schlüsseln (PMK oder CCKM) zuständig ist.

Dieser Statuscomputer zeigt die vollständigen 802.1x-Übergänge an:



Client-Analyse debuggen

In diesem Abschnitt wird der vollständige Prozess in den Protokollen angezeigt, wenn ein Client eine Verbindung mit einem WLAN herstellt.

<#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 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:

                    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 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 PLUMBFASTPATH: 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.
```

Beispiele zur Fehlerbehebung

Falsche Client-Verschlüsselungskonfiguration

Dieses Beispiel zeigt einen Client mit anderen Funktionen als dem Access Point. Der Client überprüft die SSID, aber da die Anfrage einige Parameter anzeigt, die nicht unterstützt werden, wird der Client niemals zu Authentifizierungs-/Zuordnungsphasen weitergeleitet.

Das Problem bestand insbesondere in einer Diskrepanz zwischen dem Client, der WPA verwendet, und dem AP, der nur WPA2 unterstützt:

```
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.
```

Falscher vorinstallierter Schlüssel

Dies zeigt, dass der Client versucht, sich über WPA-PSK in der Infrastruktur zu authentifizieren, jedoch aufgrund einer Diskrepanz zwischen dem vorinstallierten Schlüssel von Client und Controller nicht erfolgreich ist. Dies führt dazu, dass der Client der Ausschlussliste (Blockliste) schließlich hinzugefügt wird:

```
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 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
```

Zugehörige Informationen

- **Technischer Support und Downloads von Cisco**

Informationen zu dieser Übersetzung

Cisco hat dieses Dokument maschinell übersetzen und von einem menschlichen Übersetzer editieren und korrigieren lassen, um unseren Benutzern auf der ganzen Welt Support-Inhalte in ihrer eigenen Sprache zu bieten. Bitte beachten Sie, dass selbst die beste maschinelle Übersetzung nicht so genau ist wie eine von einem professionellen Übersetzer angefertigte. Cisco Systems, Inc. übernimmt keine Haftung für die Richtigkeit dieser Übersetzungen und empfiehlt, immer das englische Originaldokument (siehe bereitgestellter Link) heranzuziehen.