CUWNでの802.11 WLANおよび高速セキュアローミングの方法の確認

内容

概要

このドキュメントでは、Unified Wireless Network(CUWN)上のIEEE 802.11ワイヤレスLAN(WLAN)で使用可能なワイヤレスおよび高速セキュアローミングのタイプについて説明します

。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- IEEE 802.11 WLAN の基本
- IEEE 802.11 WLAN のセキュリティ
- IEEE 802.1X/EAP の基本

使用するコンポーネント

このドキュメントの情報は、Cisco WLANコントローラソフトウェアバージョン7.4に基づくものです。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

このドキュメントの情報は、Cisco WLANコントローラソフトウェアバージョン7.4に基づいていますが、説明されているデバッグ出力と動作のほとんどは、説明されている方法をサポートするソフトウェアバージョンに適用できます。ここで説明するすべての方式の詳細は、以降のCisco WLAN Controllerコード（この記事が更新されるまではバージョン8.3まで）でも同じです。

このドキュメントでは、Cisco Unified Wireless Network（CUWN）でサポートされている IEEE 802.11 ワイヤレス LAN（WLAN）で使用可能な、さまざまなタイプのワイヤレス ローミングおよび高速セキュア ローミングの方式について説明します。

このドキュメントでは、各方式の動作方法や設定方法の詳細については説明していません。このドキュメントの主な目的は、使用可能なさまざまな手法の違い、それぞれの利点と制約、および各方式でのフレーム交換について説明することです。WLANコントローラ(WLC)のデバッグの例を示し、ワイヤレスパケットイメージを使用して、説明されている各ローミング方式で発生するイベントを分析および説明します。

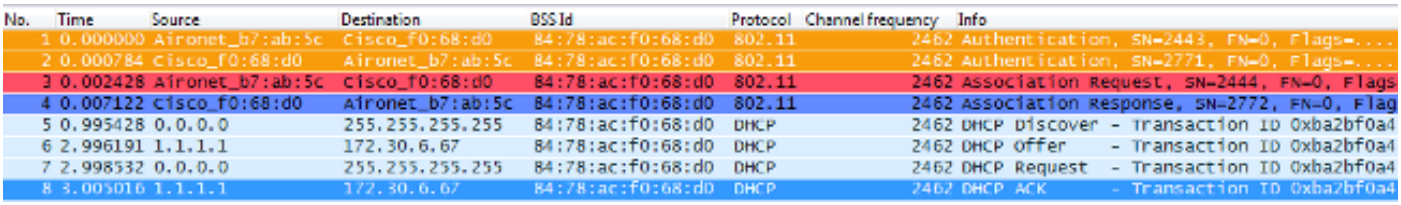
WLAN で使用できるさまざまな高速セキュア ローミング方式の説明の前に、WLAN 関連付けプロセスはどのように動作するか、サービス セット識別子（SSID）でセキュリティが設定されていない場合は通常どのようなローミング イベントが発生するか、理解することが重要です。

802.11 ワイヤレス クライアントは、アクセス ポイント（AP）に接続する際、トラフィック（ワイヤレス データ フレーム）を渡し始める前に、基本的な 802.11 オープン システム認証プロセスを通過する必要があります。次に、関連付けプロセスを完了する必要があります。オープンシステム認証プロセスは、クライアントが選択するAP上のケーブル接続に似ています。これは非常に重要なポイントです。なぜなら、どのAPを優先するかを選択するのは常にワイヤレスクライアントであり、決定はベンダーによって異なる複数の要因に基づいて行われるからです。そのため、このプロセスは、このドキュメントで後述するように、クライアントが選択した AP に認証フレ

ームを送信することで開始されます。接続の確立を AP の側から要求することはできません。

オープン システム認証プロセスが AP からの応答（「ケーブルが接続されたこと」）で正常に完了すると、関連付けプロセスにより、クライアントと AP 間のリンクを確立する 802.11 レイヤ 2（L2）のネゴシエーションが実質的に終了します。接続が正常に完了すると、AP は、クライアントに関連付け ID を割り当て、トラフィックを渡すことや、SSID に高レベル セキュリティ方式が設定されていればそれを実行することができるよう準備します。オープン システム認証プロセスは、2 つの管理フレームと関連付けプロセスで構成されています。認証フレームと関連付けフレームは、データ フレームではなくワイヤレス管理フレームであり、基本的に AP での接続プロセスに使用されるものです。

このプロセスの無線フレームの画像を次に示します。



No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:68:d0	84:78:ac:f0:68:d0	802.11	2462	Authentication, SN=2443, FN=0, Flags=....
2	0.000784	Cisco_f0:68:d0	Aironet_b7:ab:5c	84:78:ac:f0:68:d0	802.11	2462	Authentication, SN=2771, FN=0, Flags=....
3	0.002428	Aironet_b7:ab:5c	Cisco_f0:68:d0	84:78:ac:f0:68:d0	802.11	2462	Association Request, SN=2444, FN=0, Flags
4	0.007122	Cisco_f0:68:d0	Aironet_b7:ab:5c	84:78:ac:f0:68:d0	802.11	2462	Association Response, SN=2772, FN=0, Flag
5	0.995428	0.0.0.0	255.255.255.255	84:78:ac:f0:68:d0	DHCP	2462	DHCP Discover - Transaction ID 0xba2bf0a4
6	2.996191	1.1.1.1	172.30.6.67	84:78:ac:f0:68:d0	DHCP	2462	DHCP Offer - Transaction ID 0xba2bf0a4
7	2.998532	0.0.0.0	255.255.255.255	84:78:ac:f0:68:d0	DHCP	2462	DHCP Request - Transaction ID 0xba2bf0a4
8	3.005016	1.1.1.1	172.30.6.67	84:78:ac:f0:68:d0	DHCP	2462	DHCP ACK - Transaction ID 0xba2bf0a4

> **注**:802.11ワイヤレススニッフィングと、このドキュメントに記載されているイメージに対してWiresharkで使用されているフィルタや色について知りたい場合は、シスコサポートコミュニティの投稿『802.11スニファイメージの分析』を参照してください。

ワイヤレス クライアントが認証フレームで開始し、AP が別の認証フレームで応答します。すると、クライアントが関連付け要求フレームを送信し、AP が関連付け要求フレームで応答して終了します。DHCP パケットが示すとおり、802.11 オープン システム認証プロセスと関連付けプロセスを通過すると、クライアントはデータ フレームを渡し始めます。このケースでは、SSID にはセキュリティ方式が設定されていないため、クライアントは、暗号化されないデータ フレーム（この例では、DHCP）の送信をただちに開始します。

このドキュメントで後述するように、SSIDでセキュリティが有効になっている場合、関連付け応答の直後およびクライアントトラフィックのデータフレームが送信される前に、DHCP、Address Resolution Protocol(ARP)、アプリケーションパケットなどの特定のセキュリティ方式の高レベル認証および暗号化ハンドシェイクフレームが存在します。これらのフレームは暗号化されます。データ フレームは、設定されているセキュリティ方式に基づいてクライアントが完全に認証され、暗号キーがネゴシエートされるまで、送信できません。

前のイメージに基づいて、ワイヤレスクライアントがWLANへの新しいアソシエーションを開始したときにWLCの**debug client**コマンドの出力に表示されるメッセージを次に示します。

```
*apfMsConnTask_0: Jun 21 18:55:14.221: 00:40:96:b7:ab:5c
  Association received from mobile on BSSID 84:78:ac:f0:68:d0
!--- This is the Association Request from the wireless client
     to the selected AP.

*apfMsConnTask_0: Jun 21 18:55:14.222: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:68:d0
  (status 0) ApVapId 1 Slot 0
```

```
!--- This is the Association Response from the AP to the client.
```

> **注**：このドキュメントで示されている出力に使用されるWLCデバッグは**debug client**コマンドです。例では、出力全体ではなく、関連するメッセージの一部のみを示しています。このdebugコマンドの詳細については、『ワイヤレスLANコントローラ(WLC)でのデバッグクライアントについて』というドキュメントを参照してください。

これらのメッセージは、関連付け要求フレームと応答フレームを示します。このハンドシェイクはCUWN上のAPレベルで迅速に発生するため、初期認証フレームはWLCに記録されません。

クライアントがローミングする際には、どのような情報が表示されるでしょうか。クライアントは、AP への接続が確立されると常に、クライアントの関連付けの確立またはローミング イベントに起因する 4 つの管理フレームを交換します。クライアントが 1 度に確立する AP との接続は、1 つの AP との 1 つの接続のみです。WLAN インフラストラクチャへの新しい接続の場合とローミング イベントの場合で、フレーム交換に関する相違は 1 点のみです。それは、ローミング イベントの関連付けフレームが**再関連付けフレームと呼ばれる点です。これは、クライアントが別の AP からローミングするときには、実際には WLAN への新しい関連付けの確立を試行していないことを意味しています。**これらのフレームには、ローミングイベントをネゴシエートするために使用されるさまざまな要素を含めることができます。これは設定によって異なりますが、これらの詳細については、このドキュメントでは取り上げていません。

フレーム交換の例を次に示します。

No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:2a:90	84:78:ac:f0:2a:90	802.11	2437	Authentication, SN=2611, FN=0, Flags=.....
2	0.001608	Cisco_f0:2a:90	Aironet_b7:ab:5c	84:78:ac:f0:2a:90	802.11	2437	Authentication, SN=3010, FN=0, Flags=.....
3	0.003248	Aironet_b7:ab:5c	Cisco_f0:2a:90	84:78:ac:f0:2a:90	802.11	2437	Reassociation Request, SN=2612, FN=0, Flags
4	0.008122	Cisco_f0:2a:90	Aironet_b7:ab:5c	84:78:ac:f0:2a:90	802.11	2437	Reassociation Response, SN=3011, FN=0, Flag
5	4.291764	Aironet_b7:ab:5c	Broadcast	84:78:ac:f0:2a:90	ARP	2437	Who has 172.30.6.254? Tell 172.30.6.67
6	4.293918	Cisco_f5:4a:40	Aironet_b7:ab:5c	84:78:ac:f0:2a:90	ARP	2437	172.30.6.254 is at 00:1e:f7:f5:4a:40

デバッグ出力には次のメッセージが表示されます。

```
*apfMsConnTask_2: Jun 21 19:02:19.709: 00:40:96:b7:ab:5c
  Reassociation received from mobile on BSSID 84:78:ac:f0:2a:90
!--- This is the Reassociation Request from the wireless client
     to the selected AP.

*apfMsConnTask_2: Jun 21 19:02:19.710: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:2a:90
  (status 0) ApVapId 1 Slot 0
!--- This is the Reassociation Response from the AP to the client.
```

上に示すように、クライアントが新しい AP に再関連付け要求を送信し、その AP から再関連付け応答を受信すると、ローミング イベントが正常に実行されます。クライアントにはすでに IP アドレスがあるため、最初のデータ フレームは ARP パケット用です。

ローミングイベントが発生することが予想されるものの、クライアントが再関連付け要求（このドキュメントで前述したイメージやデバッグから確認できる）ではなく関連付け要求(ARQ)を送信した場合、クライアントは実際にはローミングしていません。クライアントは、切断が行われたかのように WLAN への新しい関連付けを開始し、初めからの再接続を試行します。このことは、クライアントがカバレッジ エリアから退出した後に関連付けを開始するのに十分な信号品質を持つ AP を見つけた場合など、複数の理由で発生します。しかし通常は、クライアント側の問題（ドライバ、ファームウェア、またはソフトウェアの問題のためにクライアントがローミング イベントを開始しない）が考えられます。

> **注**：問題の原因を特定するには、ワイヤレスクライアントベンダーに問い合わせてください

。

高レベル セキュリティでのローミング

SSID に基本的な 802.11 オープン システム認証に加えて L2 のより高レベルのセキュリティが設定されている場合、初回関連付けやローミングのためには、さらに多くのフレームが必要になります。このドキュメントでは、802.11 WLAN 向けに標準化および導入されている最も一般的な 2 つのセキュリティ方式について説明します。

- **WPA/WPA2-PSK（事前共有キー）：事前共有キーによるクライアントの認証。**
- **WPA/WPA2-EAP（Extensible Authentication Protocol）：802.1X/EAP 方式によるクライアントの認証。認証サーバの使用により、証明書、ユーザ名とパスワード、トークンなどの追加の安全なクレデンシャルを確認する。**

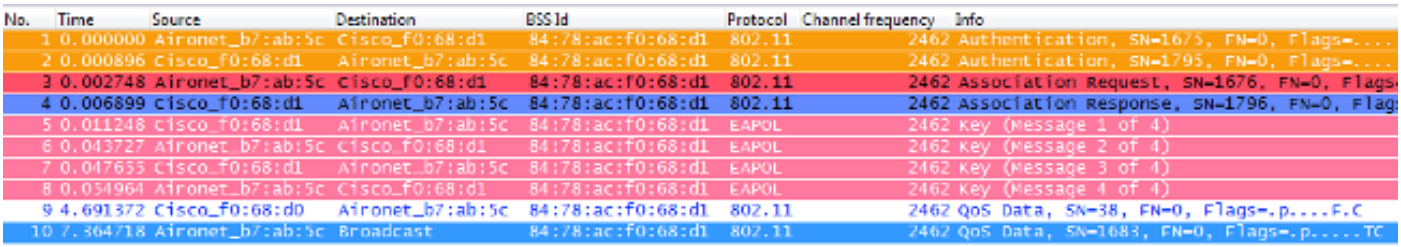
これらの 2 つの方式（PSK および EAP）は、クライアントを認証および確認する方法は異なりますが、キー管理プロセスには基本的に同じ WPA/WPA2 ルールが使用されていることを認識することが重要です。セキュリティが WPA/WPA2-PSK であっても WPA/WPA2-EAP であっても、使用される特定の認証方式でクライアントが確認されると、WPA/WPA2 の 4 方向ハンドシェイクと呼ばれるプロセスにより、WLC/AP とクライアント間のキー ネゴシエーションが、マスター セッション キー（MSK）を元のキー マテリアルとして開始されます。

プロセスの概要を次に示します。

1. MSK は、セキュリティ方式として 802.1X/EAP が使用される場合は EAP 認証のフェーズから、WPA/WPA2-PSK が使用される場合は PSK から、それぞれ導出されます。
2. この MSK から、クライアントと WLC/AP は ペアワイズ マスター キー（PMK）を導出し、WLC/AP はグループ マスター キー（GMK）を生成します。
3. これらの2つのマスターキーの準備が整うと、クライアントとWLC/APは、実際の暗号キーのネゴシエーションのシードとしてマスターキーを使用して、WPA/WPA2の4方向ハンドシェイク（このドキュメントの後半で説明するスクリーンイメージとデバッグを使用して）を開始します。
4. これらの最終的な暗号キーは、ペアワイズ一時キー（PTK）およびグループ一時キー（GTK）と呼ばれています。PTK は、PMK から導出され、クライアントでユニキャスト フレームを暗号化するために使用されます。グループ一時キー（GTK）は、GMK から導出され、当該の SSID/AP でマルチキャスト/ブロードキャストを暗号化するために使用されます。

WPA/WPA2-PSK

暗号化のために Temporal Key Integrity Protocol（TKIP）または Advanced Encryption Standard（AES）を使用して WPA-PSK または WPA2-PSK を実行する際、クライアントは、初回関連付けとローミング両方の際に、WPA 4 方向ハンドシェイクと呼ばれるプロセスを実行する必要があります。すでに説明したように、これは、基本的には WPA/WPA2 が暗号キーを導出できるようにするため使用されるキー管理プロセスです。ただし、PSK が実行される際には、クライアントが WLAN に接続するために有効な事前共有キーを持っていることを確認するためにも使用されます。次の図は、PSKを使用してWPAまたはWPA2を実行する場合の初期関連付けプロセスを示しています。



No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:68:d1	84:78:ac:f0:68:d1	802.11	2462	Authentication, SN=1675, FN=0, Flags=....
2	0.000896	Cisco_f0:68:d1	Aironet_b7:ab:5c	84:78:ac:f0:68:d1	802.11	2462	Authentication, SN=1795, FN=0, Flags=....
3	0.002748	Aironet_b7:ab:5c	Cisco_f0:68:d1	84:78:ac:f0:68:d1	802.11	2462	Association Request, SN=1676, FN=0, Flags
4	0.006899	Cisco_f0:68:d1	Aironet_b7:ab:5c	84:78:ac:f0:68:d1	802.11	2462	Association Response, SN=1796, FN=0, Flag
5	0.011248	Cisco_f0:68:d1	Aironet_b7:ab:5c	84:78:ac:f0:68:d1	EAPOL	2462	Key (Message 1 of 4)
6	0.043727	Aironet_b7:ab:5c	Cisco_f0:68:d1	84:78:ac:f0:68:d1	EAPOL	2462	Key (Message 2 of 4)
7	0.047655	Cisco_f0:68:d1	Aironet_b7:ab:5c	84:78:ac:f0:68:d1	EAPOL	2462	Key (Message 3 of 4)
8	0.054964	Aironet_b7:ab:5c	Cisco_f0:68:d1	84:78:ac:f0:68:d1	EAPOL	2462	Key (Message 4 of 4)
9	4.691372	Cisco_f0:68:d0	Aironet_b7:ab:5c	84:78:ac:f0:68:d1	802.11	2462	QoS Data, SN=38, FN=0, Flags=.p....F.C
10	7.364718	Aironet_b7:ab:5c	Broadcast	84:78:ac:f0:68:d1	802.11	2462	QoS Data, SN=1683, FN=0, Flags=.p.....TC

上に示すように、802.11 オープン システム認証および関連付けプロセスの後に、WPA 4 方向ハンドシェイクからの 4 つの EAPOL フレーム（AP による **message-1** で開始し、クライアントによる **message-4** で終了）がありますハンドシェイクが成功すると、クライアントはデータフレーム（DHCPなど）の送信を開始します。この場合、データフレームは4方向ハンドシェイクから取得されたキーで暗号化されます（ワイヤレスイメージからトラフィックの実際の内容とタイプを確認できない理由です）。

> **注**:EAPOLフレームは、APとクライアントの間ですべてのキー管理フレームと802.1X/EAP認証フレームを無線で転送するために使用され、無線データフレームとして送信されます。

デバッグ出力には次のメッセージが表示されます。

```
*apfMsConnTask_0: Jun 21 19:30:05.172: 00:40:96:b7:ab:5c
  Association received from mobile on BSSID 84:78:ac:f0:68:d1
*apfMsConnTask_0: Jun 21 19:30:05.173: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:68:d1
  (status 0) ApVapId 2 Slot 0
!--- The Association handshake is finished.

*dot1xMsgTask: Jun 21 19:30:05.178: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c
  state INITPMK (message 1), replay counter
  00.00.00.00.00.00.00.00
!--- Message-1 of the WPA/WPA2 4-Way handshake is sent
     from the WLC/AP to the client.

*Dot1x_NW_MsgTask_4: Jun 21 19:30:05.289: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 21 19:30:05.289: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTK_START state (message 2)
  from mobile 00:40:96:b7:ab:5c
!--- Message-2 of the WPA/WPA2 4-Way handshake is successfully
     received from the client.

*Dot1x_NW_MsgTask_4: Jun 21 19:30:05.290: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c
  state PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01
!--- Message-3 of the WPA/WPA2 4-Way handshake is sent
     from the WLC/AP to the client.

*Dot1x_NW_MsgTask_4: Jun 21 19:30:05.309: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 21 19:30:05.310: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile 00:40:96:b7:ab:5c
!--- Message-4 (final message) of the WPA/WPA2 4-Way handshake
     is successfully received from the client, which confirms
     the installation of the derived keys. They can now be used in
```

```
order to encrypt data frames with current AP.
```

ローミング時には、クライアントは基本的に同じフレーム交換を追跡します。新しいAPで新しい暗号キーを取得するには、WPAの4方向ハンドシェイクが必要です。その理由は、標準規格によって確立されているセキュリティ上の理由と、新しい AP が元のキーを認識しないことです。唯一の違いは、次の図に示すように、アソシエーションフレームではなく再関連付けフレームがあることです。

No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:2a:91	84:78:ac:f0:2a:91	802.11	2437	Authentication, SN=2356, FN=0, Flags=.....
2	0.000846	Cisco_f0:2a:91	Aironet_b7:ab:5c	84:78:ac:f0:2a:91	802.11	2437	Authentication, SN=3694, FN=0, Flags=.....
3	0.004296	Aironet_b7:ab:5c	Cisco_f0:2a:91	84:78:ac:f0:2a:91	802.11	2437	Reassociation Request, SN=2357, FN=0, Flags
4	0.010867	Cisco_f0:2a:91	Aironet_b7:ab:5c	84:78:ac:f0:2a:91	802.11	2437	Reassociation Response, SN=3695, FN=0, Flag
5	0.013109	Cisco_f0:2a:91	Aironet_b7:ab:5c	84:78:ac:f0:2a:91	EAPOL	2437	Key (Message 1 of 4)
6	0.034339	Aironet_b7:ab:5c	Cisco_f0:2a:91	84:78:ac:f0:2a:91	EAPOL	2437	Key (Message 2 of 4)
7	0.041124	Cisco_f0:2a:91	Aironet_b7:ab:5c	84:78:ac:f0:2a:91	EAPOL	2437	Key (Message 3 of 4)
8	0.056241	Aironet_b7:ab:5c	Cisco_f0:2a:91	84:78:ac:f0:2a:91	EAPOL	2437	Key (Message 4 of 4)
9	0.695758	Aironet_b7:ab:5c	Broadcast	84:78:ac:f0:2a:91	802.11	2437	QoS Data, SN=2360, FN=0, Flags=.p..R..TC
10	0.698357	Cisco_f5:4a:40	Aironet_b7:ab:5c	84:78:ac:f0:2a:91	802.11	2437	QoS Data, SN=42, FN=0, Flags=.p....F.C

デバッグ出力に表示されるメッセージは同じですが、クライアントからの最初のパケットは、すでに説明したように、関連付けではなく再関連付けです。

WPA/WPA2-EAP

802.1X/EAP 方式を使用してクライアントの認証を安全な SSID で行う場合、クライアントがトラフィックを渡し始める前に、さらに多くのフレームが必要になります。これらの追加フレームは、クライアントクレデンシャルを認証するために使用され、EAP方式に応じて、4 ~ 20フレームの間で使用できます。これらは、関連付けまたは再関連付けの後、かつ、WPA/WPA2 の 4 方向ハンドシェイクの前に来ます。その理由は、キー管理プロセス (4 方向ハンドシェイク) において最終的な暗号キーを生成するためのシードとして使用される MSK は、認証フェーズから導出されるためです。

次の図は、PEAPv0/EAP-MSCHAPv2を使用したWPAが実行された場合の初期関連付けでAPとワイヤレスクライアントの間で無線で交換されるフレームの例を示しています。

No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	802.11	2462	Authentication, SN=2465, FN=0, Fla
2	0.000783	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	802.11	2462	Authentication, SN=275, FN=0, Flag
3	0.002579	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	802.11	2462	Association Request, SN=2466, FN=0
4	0.007765	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	802.11	2462	Association Response, SN=276, FN=0
5	0.012140	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Identity
6	0.052606	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	EAPOL	2462	Start
7	0.055257	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Identity
8	0.061197	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	EAP	2462	Response, Identity
9	0.081402	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
10	0.117423	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	TLSv1	2462	Client Hello
11	0.145293	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
12	0.167145	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	EAP	2462	Response, Protected EAP (EAP-PEAP)
13	0.183267	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
14	0.196221	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	EAP	2462	Response, Protected EAP (EAP-PEAP)
15	0.201527	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
16	0.210076	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	TLSv1	2462	Certificate, Client Key Exchange,
17	0.220032	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
18	0.222784	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	EAP	2462	Response, Protected EAP (EAP-PEAP)
19	0.227233	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
20	0.291267	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	TLSv1	2462	Application Data, Application Data
21	0.291862	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	TLSv1	2462	Application Data, Application Data
22	0.295816	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
23	0.297766	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	TLSv1	2462	Application Data, Application Data
24	0.304666	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
25	0.313817	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
26	0.315942	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	TLSv1	2462	Application Data, Application Data
27	0.321376	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Request, Protected EAP (EAP-PEAP)
28	0.323863	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	TLSv1	2462	Application Data, Application Data
29	0.328766	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAP	2462	Success
30	0.330360	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAPOL	2462	Key (Message 1 of 4)
31	0.334225	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	EAPOL	2462	Key (Message 2 of 4)
32	0.338645	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	EAPOL	2462	Key (Message 3 of 4)
33	0.341932	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	EAPOL	2462	Key (Message 4 of 4)
34	1.366605	Cisco_f0:68:d8	Aironet_b7:ab:5c	84:78:ac:f0:68:d8	802.11	2462	QoS Data, SN=448, FN=0, Flags=.p..
35	1.383200	Aironet_b7:ab:5c	Cisco_f0:68:d8	84:78:ac:f0:68:d8	802.11	2462	QoS Data, SN=2482, FN=0, Flags=.p.

この交換では、フレームの数が異なることがあります。これには、EAP 方式の種類、問題発生に

よる再送信、クライアントの動作 (この例では、AP が最初の ID 要求を送信した後にクライアントが **EAPOL START を送信したことによる 2 件の ID 要求) 、またはクライアントがすでにサーバと証明書を交換しているかどうかなど、複数の要因があります。**802.1X/EAP 方式のために SSID を設定すると、必ず、フレーム (認証用) が多くなるため、クライアントがデータ フレームの送信を開始するまでにより多くの時間がかかります。

デバッグ メッセージの要約を次に示します。

```
*apfMsConnTask_0: Jun 21 23:41:19.092: 00:40:96:b7:ab:5c
  Association received from mobile on BSSID 84:78:ac:f0:68:d8
*apfMsConnTask_0: Jun 21 23:41:19.094: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:68:d8
  (status 0) ApVapId 9 Slot 0
!--- The Association handshake is finished.

*dot1xMsgTask: Jun 21 23:41:19.098: 00:40:96:b7:ab:5c
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
  (EAP Id 1)
!--- The EAP Identity Request is sent to the client once it is
     associated in order to begin the higher-level authentication
     process. This informs the client that an identity to start
     this type of 802.1X/EAP authentication must be provided.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.226: 00:40:96:b7:ab:5c
  Received EAPOL START from mobile 00:40:96:b7:ab:5c
!--- The wireless client decides to start the EAP authentication
     process, and informs the AP with an EAPOL START data frame.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.227: 00:40:96:b7:ab:5c
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
  (EAP Id 2)
!--- WLC/AP sends another EAP Identity Request to the client.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.235: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.235: 00:40:96:b7:ab:5c
  Received Identity Response (count=2) from mobile 00:40:96:b7:ab:5c
!--- The client responds with an EAP Identity Response on an EAPOL
     frame.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.301: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.301: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 3)
!--- Once the WLC/AP sends the client response to the Authentication
     Server on a RADIUS Access-Request packet, the server responds
     with a RADIUS Access-Challenge in order to officially start the
     EAP negotiation, handshake, and authentication with the client
     (sometimes with mutual authentication, dependent upon the EAP
     method). This response received by the WLC/AP is sent to the client.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.344: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.344: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 3, EAP Type 25)
!--- The client responds with an EAP Response on an EAPOL frame, which
     is sent to the Authentication Server on a RADIUS Access-Request
     packet. The server responds with another RADIUS Access-Challenge.
     This process continues, dependent upon the EAP method (the exchange
```

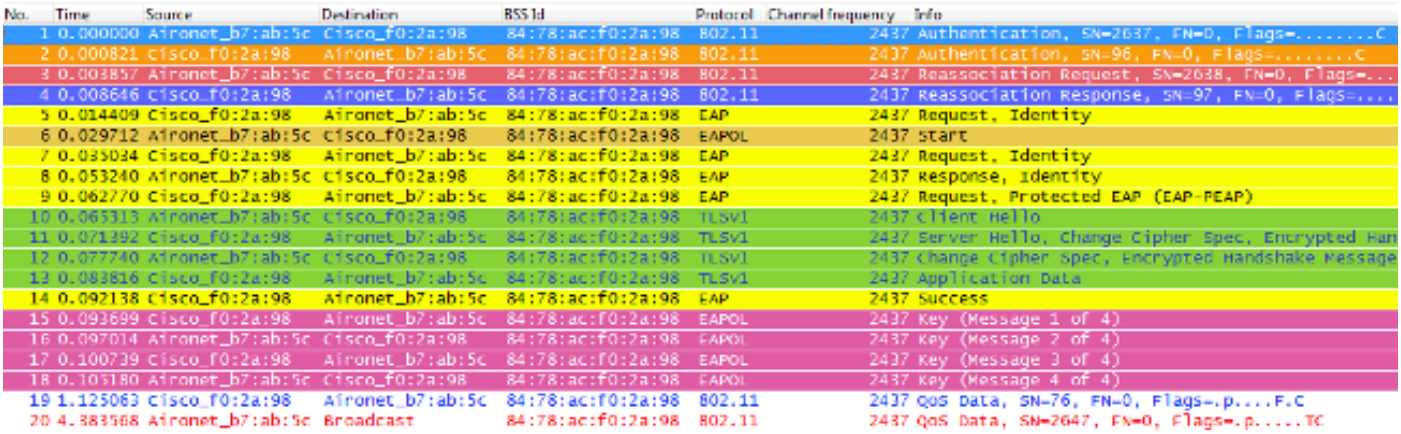
of certificates when used, the building of TLS tunnels, validation of client credentials, client validation of server identity when applicable). Hence, the next few messages are basically the same on the WLC/AP side, as this acts as a "proxy" between the client and the Authentication Server exchanges.

```
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.347: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.347: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 4)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.375: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.375: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 4, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.377: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.377: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 5)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.403: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.403: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 5, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.404: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.404: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 6)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.414: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.414: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 6, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.421: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.421: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 7)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.425: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.425: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 7, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.427: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c
```

```
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.427: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 8)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.434: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.434: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 8, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.436: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.436: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 9)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.440: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.440: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 9, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.442: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.442: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 10)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.449: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.449: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 10, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.452: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.452: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 11)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.457: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.457: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 11, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.459: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.459: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 13)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.469: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c
```

```
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.469: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 13, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.472: 00:40:96:b7:ab:5c
  Processing Access-Accept for mobile 00:40:96:b7:ab:5c
!--- The authentication finishes and is successful for this client,
     so the RADIUS Server sends a RADIUS Access-Accept to the WLC/AP.
     This RADIUS Access-Accept comes with the special attributes
     that are assigned to this client (if any are configured on the
     Authentication Server for this client). This Access-Accept also
     comes with the MSK derived with the client in the EAP
     authentication process, so the WLC/AP installs it in order to
     initiate the WPA/WPA2 4-Way handshake with the wireless client.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.473: 00:40:96:b7:ab:5c
  Sending EAP-Success to mobile 00:40:96:b7:ab:5c
  (EAP Id 13)
!--- The accept/pass of the authentication is sent to the client as
     an EAP-Success message.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.473: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c
  state INITPMK (message 1), replay counter
  00.00.00.00.00.00.00.00
!--- Message-1 of the WPA/WPA2 4-Way handshake is sent from the
     WLC/AP to the client.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.481: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.481: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTK_START state (message 2)
  from mobile 00:40:96:b7:ab:5c
!--- Message-2 of the WPA/WPA-2 4-Way handshake is successfully
     received from the client.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.481: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c
  state PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01
!--- Message-3 of the WPA/WPA2 4-Way handshake is sent from the
     WLC/AP to the client.

*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.487: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 21 23:41:19.487: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile 00:40:96:b7:ab:5c
!--- Message-4 (final message) of the WPA/WPA2 4-Way handshake
     is successfully received from the client, which confirms the
     installation of the derived keys. They can now be used in
     order to encrypt data frames with the current AP.
```

無線クライアントがここで通常のローミングを実行する場合（高速セキュアローミング方式を実装しない通常の動作）、クライアントはまったく同じプロセスを実行し、図に示すように認証サーバに対して完全な認証を実行する必要があります。唯一の違いは、クライアントは、実際には別の AP からローミングしていることを再関連付け要求を使用して新しい AP に通知しますが、それでも、全体的な確認と新しいキーの生成を実行する必要があることです。



No.	Time	Source	Destination	BSSId	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	802.11	2437	Authentication, SN=2637, FN=0, Flags=.....C
2	0.000821	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	802.11	2437	Authentication, SN=96, FN=0, Flags=.....C
3	0.003857	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	802.11	2437	Reassociation Request, SN=2638, FN=0, Flags=...
4	0.008646	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	802.11	2437	Reassociation Response, SN=97, FN=0, Flags=....
5	0.014409	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	EAP	2437	Request, Identity
6	0.029712	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	EAPOL	2437	Start
7	0.035034	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	EAP	2437	Request, Identity
8	0.053240	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	EAP	2437	Response, Identity
9	0.062770	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	EAP	2437	Request, Protected EAP (EAP-PEAP)
10	0.065313	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	TLSv1	2437	Client Hello
11	0.071292	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	TLSv1	2437	Server Hello, Change Cipher Spec, Encrypted Han
12	0.077740	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	TLSv1	2437	Change Cipher Spec, Encrypted Handshake Message
13	0.083816	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	TLSv1	2437	Application Data
14	0.092138	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	EAP	2437	Success
15	0.093699	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	EAPOL	2437	Key (Message 1 of 4)
16	0.097014	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	EAPOL	2437	Key (Message 2 of 4)
17	0.100739	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	EAPOL	2437	Key (Message 3 of 4)
18	0.105180	Aironet_b7:ab:5c	Cisco_f0:2a:98	84:78:ac:f0:2a:98	EAPOL	2437	Key (Message 4 of 4)
19	1.125063	Cisco_f0:2a:98	Aironet_b7:ab:5c	84:78:ac:f0:2a:98	802.11	2437	QoS Data, SN=76, FN=0, Flags=.p....F.C
20	4.383568	Aironet_b7:ab:5c	Broadcast	84:78:ac:f0:2a:98	802.11	2437	QoS Data, SN=2647, FN=0, Flags=.p.....TC

上に示すように、クライアントが新しい AP にローミングしたときにフレーム数が（前述した複数の要因によって）初期認証よりも少ない場合であっても、データ フレームを渡し続けるためには、（ローミング前にトラフィックがアクティブに送信されていた場合であっても）EAP 認証と WPA キー管理プロセスを完了する必要があります。そのため、クライアントで遅延の影響を受けやすいアプリケーション（音声トラフィック アプリケーション、タイムアウトの影響を受けやすいアプリケーションなど）がアクティブになっている場合、音声の途切れやアプリケーションの切断などの問題がローミング時にユーザに感知されることがあります。これは、クライアントがデータ フレームの送受信を続行するためのプロセスにかかる時間によって異なります。この遅延は、RF環境、クライアントの数、WLCとLAP間および認証サーバとの間のラウンドトリップ時間、およびその他の理由に応じて、より長くなる可能性があります。

このローミング イベントのデバッグ メッセージの要約を次に示します（前に示したデバッグと基本的に同じであるため、これらのメッセージの詳細は説明しません）。

```
*apfMsConnTask_2: Jun 21 23:47:54.872: 00:40:96:b7:ab:5c
  Reassociation received from mobile on BSSID 84:78:ac:f0:2a:98

*apfMsConnTask_2: Jun 21 23:47:54.874: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:2a:98
  (status 0) ApVapId 9 Slot 0

*dot1xMsgTask: Jun 21 23:47:54.879: 00:40:96:b7:ab:5c
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
  (EAP Id 1)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.895: 00:40:96:b7:ab:5c
  Received EAPOL START from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.895: 00:40:96:b7:ab:5c
  dot1x - moving mobile 00:40:96:b7:ab:5c into Connecting state

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.895: 00:40:96:b7:ab:5c
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
  (EAP Id 2)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.922: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.922: 00:40:96:b7:ab:5c
  Received Identity Response (count=2) from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.929: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.929: 00:40:96:b7:ab:5c
```

```
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 3)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.941: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.941: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 3, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.943: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.943: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 4)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.954: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.954: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 4, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.956: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.957: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 7)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.976: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.976: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 7, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.978: 00:40:96:b7:ab:5c
  Processing Access-Accept for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.978: 00:40:96:b7:ab:5c
  Sending EAP-Success to mobile 00:40:96:b7:ab:5c
  (EAP Id 7)

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.978: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c
  state INITPMK (message 1), replay counter
  00.00.00.00.00.00.00.00

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.995: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.995: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTK_START state (message 2)
  from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 23:47:54.995: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c
  state PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01

*Dot1x_NW_MsgTask_4: Jun 21 23:47:55.005: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c
```

```
*Dot1x_NW_MsgTask_4: Jun 21 23:47:55.005: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile 00:40:96:b7:ab:5c
```

802.1X/EAP および WPA/WPA2 のセキュリティ フレームワークは、このように動作します。アプリケーションやサービスが通常のローミング イベントによる遅延の影響を受けないようにするため、WiFi 業界では、WLAN/SSID でのセキュリティが使用されているときのローミング プロセスを高速化することを目指し、さまざまな高速セキュア ローミング方式が開発および導入されています。WLAN で高レベル セキュリティを導入すると、AP 間でローミングしながらトラフィックを渡し続ける際に、クライアントには多少の遅延が発生します。この原因は、すでに説明したように、セキュリティのセットアップにより、EAP 認証およびキー管理フレーム交換が必要とされることです。

高速セキュア ローミングとは、単に「WLAN でセキュリティが設定されているときのローミング プロセスを高速化するための方式やスキームの導入」を指して業界で使用されている用語であることを理解することが重要です。 次の項では、WLAN で使用でき、CUWN でサポートされているさまざまな高速セキュア ローミングの方式およびスキームについて説明します。

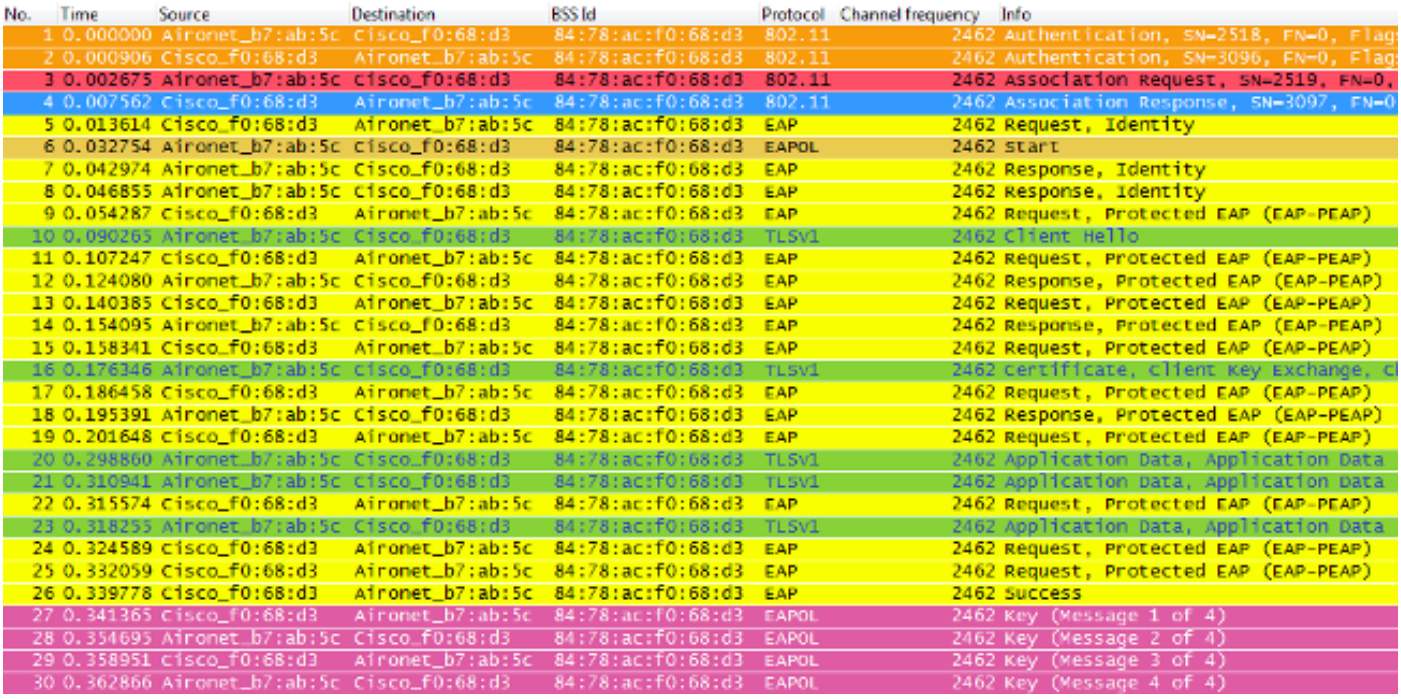
CCKM を使用しての高速セキュア ローミング

Cisco Centralized Key Management（CCKM）は、WLAN で 802.1X/EAP セキュリティを使用する場合に、これまでに説明した遅延を緩和するソリューションとしてシスコが作成した、エンタープライズ WLAN で開発および導入された最初の高速セキュア ローミング方式です。これは、シスコの独自プロトコルであるため、CCKM に関して Cisco Compatible Extension（CCX）と互換可能な Cisco WLAN インフラストラクチャのデバイスおよびワイヤレス クライアント（複数のベンダー）でのみサポートされています。

CCKMは、WEP、TKIP、AESなど、WLANで使用可能なさまざまな暗号化方式を使用して実装できます。また、デバイスでサポートされている CCX のバージョンによっては、WLAN で使用される 802.1X/EAP 認証方式のほとんどをサポートしています。

> **注**:CCX仕様の異なるバージョンでサポートされている機能の内容（サポートされているEAP方式を含む）の概要については、『CCXのバージョンと機能』ドキュメントを参照し、ワイヤレスクライアントでサポートされている正確なCCXバージョン（CCX互換の場合）を確認して、CCKMで使用するセキュリティ方式を実装できるかどうかを確認してください。

このワイヤレスイメージは、暗号化としてTKIPを使用してCCKMを実行し、802.1X/EAP方式としてPEAPv0/EAP-MSCHAPv2を実行する場合の初期関連付け時に交換されるフレームの例を示しています。これは、基本的に PEAPv0/EAP-MSCHAPv2 とともに WPA/TKIP を実行した場合と同じ交換ですが、ここでは、クライアントがローミングする必要があるときには高速セキュア ローミングを実行するため、別のキー階層とキャッシュ方式を使用するように、クライアントとインフラストラクチャの間の CCKM がネゴシエートされています。



No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	802.11	2462	Authentication, SN=2518, FN=0, Flag
2	0.000906	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	802.11	2462	Authentication, SN=3096, FN=0, Flag
3	0.002675	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	802.11	2462	Association Request, SN=2519, FN=0,
4	0.007562	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	802.11	2462	Association Response, SN=3097, FN=0
5	0.013614	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Identity
6	0.032754	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAPOL	2462	Start
7	0.042974	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAP	2462	Response, Identity
8	0.046855	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAP	2462	Response, Identity
9	0.054287	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
10	0.090265	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	TLSv1	2462	Client Hello
11	0.107247	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
12	0.124080	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAP	2462	Response, Protected EAP (EAP-PEAP)
13	0.140385	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
14	0.154095	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAP	2462	Response, Protected EAP (EAP-PEAP)
15	0.158341	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
16	0.176346	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	TLSv1	2462	Certificate, Client Key Exchange, C
17	0.186458	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
18	0.195391	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAP	2462	Response, Protected EAP (EAP-PEAP)
19	0.201648	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
20	0.298860	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	TLSv1	2462	Application Data, Application Data
21	0.310941	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	TLSv1	2462	Application Data, Application Data
22	0.315574	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
23	0.318255	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	TLSv1	2462	Application Data, Application Data
24	0.324589	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
25	0.332059	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Request, Protected EAP (EAP-PEAP)
26	0.339778	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAP	2462	Success
27	0.341365	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAPOL	2462	Key (Message 1 of 4)
28	0.354695	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAPOL	2462	Key (Message 2 of 4)
29	0.358951	Cisco_f0:68:d3	Aironet_b7:ab:5c	84:78:ac:f0:68:d3	EAPOL	2462	Key (Message 3 of 4)
30	0.362866	Aironet_b7:ab:5c	Cisco_f0:68:d3	84:78:ac:f0:68:d3	EAPOL	2462	Key (Message 4 of 4)

デバッグ メッセージの要約を次に示します (出力を減らすため、一部の EAP 交換を削除しています) 。

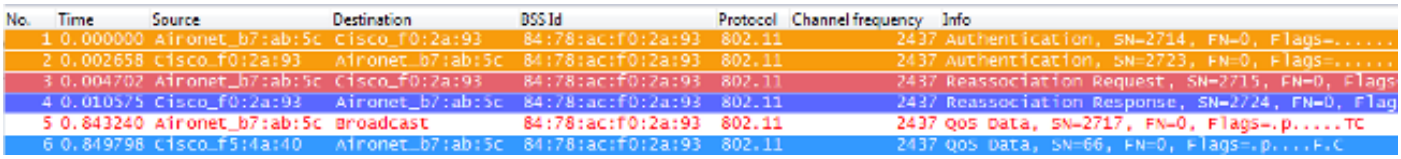
```
*apfMsConnTask_0: Jun 25 15:41:41.507: 00:40:96:b7:ab:5c
  Association received from mobile on BSSID 84:78:ac:f0:68:d3
!--- This is the Association Request from the client.

*apfMsConnTask_0: Jun 25 15:41:41.507: 00:40:96:b7:ab:5c
  Processing WPA IE type 221, length 22 for mobile
  00:40:96:b7:ab:5c
*apfMsConnTask_0: Jun 25 15:41:41.507: 00:40:96:b7:ab:5c
  CCKM: Mobile is using CCKM
!--- The WLC/AP finds an Information Element that claims CCKM
    support on the Association request that is sent from the client.

*apfMsConnTask_0: Jun 25 15:41:41.507: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 8
!--- This is the key cache index for this client, which is set temporally.

*apfMsConnTask_0: Jun 25 15:41:41.508: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:68:d3
  (status 0) ApVapId 4 Slot 0
!--- The Association Response is sent to the client.

*dot1xMsgTask: Jun 25 15:41:41.513: 00:40:96:b7:ab:5c
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
  (EAP Id 1)
!--- An EAP Identity Request is sent to the client once it is
     associated in order to begin the higher-level authentication
     process. This informs the client that an identity to start
     this type of 802.1X/EAP authentication must be provided.
     Further EAP messages are not described, as they are basically
     the same as the ones previously-explained.

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.536: 00:40:96:b7:ab:5c
  Received EAPOL START from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.536: 00:40:96:b7:ab:5c
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
```

```
  (EAP Id 2)

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.546: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.546: 00:40:96:b7:ab:5c
  Received EAP Response packet with mismatching id
  (currentid=2, eapid=1) from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.550: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.550: 00:40:96:b7:ab:5c
  Received Identity Response (count=2) from mobile
  00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.555: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.555: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 3)

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.594: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.594: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 3, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.840: 00:40:96:b7:ab:5c
  Processing Access-Accept for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.841: 00:40:96:b7:ab:5c
  Creating a PKC PMKID Cache entry for station 00:40:96:b7:ab:5c
  (RSN 0)<br/ >
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.841: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 8
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.841: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 0
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.841: 00:40:96:b7:ab:5c
  CCKM: Create a global PMK cache entry
!--- WLC creates a global PMK cache entry for this client,
     which is for CCKM in this case.

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.841: 00:40:96:b7:ab:5c
  Sending EAP-Success to mobile 00:40:96:b7:ab:5c
  (EAP Id 13)
!--- The client is informed of the successful EAP authentication.

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.841: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c state
  INITPMK(message 1),replay counter 00.00.00.00.00.00.00.00
!--- Message-1 of the initial 4-Way handshake is sent from the
     WLC/AP to the client.

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.858: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.858: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTK_START state (message 2) from mobile
  00:40:96:b7:ab:5c
!--- Message-2 of the initial 4-Way handshake is received
     successfully from the client.
```

```
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.858: 00:40:96:b7:ab:5c
  CCKM: Sending cache add
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.858: CCKM: Sending CCKM PMK
  (Version_1) information to mobility group
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.858: CCKM: Sending CCKM PMK
  (Version_2) information to mobility group
!--- The CCKM PMK cache entry for this client is shared with
     the WLCs on the mobility group.

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.858: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c
  state PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01
!--- Message-3 of the initial 4-Way handshake is sent from the
     WLC/AP to the client.

*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.866: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c
*Dot1x_NW_MsgTask_4: Jun 25 15:41:41.866: 00:40:96:b7:ab:5c Received
  EAPOL-key in PTKINITNEGOTIATING state (message 4) from mobile
  00:40:96:b7:ab:5c
!--- Message-4 (final message) of this initial 4-Way handshake
     is received successfully from the client, which confirms the
     installation of the derived keys. They can now be used in order
     to encrypt data frames with the current AP.
```

CCKMでは、WLANへの最初の関連付けは通常のWPA/WPA2と似ており、MSK(ここではネットワークセッションキー(NSK)とも呼ばれます)がクライアントとRADIUSサーバの間で相互に生成されます。このプライマリキーは、認証が成功した後にサーバからWLCに送信され、このWLANとのクライアント関連付けのライフタイムにおける以降のすべてのキーの導出の基礎としてキャッシュされます。ここから、WLCとクライアントは、CCKMに基づく高速セキュアローミングに使用されるシード情報を取得します。最初のAPでユニキャスト(PTK)とマルチキャスト/ブロードキャスト(GTK)の暗号化キーを取得するために、WPA/WPA2と同様の4方向のハンドシェイクを実行します。

大きな違いは、ローミング時に明らかになります。この場合、CCKMクライアントはAP/WLCに単一の再関連付け要求フレーム（MICと順次増加するランダム番号を含む）を送信し、新しいPTKを取得するために十分な情報(新しいAP MACアドレス**-BSSID-**を含む)を提供します。WLC と新しい AP は、新しい PTK を導出するための 十分な情報もこの再関連付け要求から得られるため、再関連付け応答で応答するのみです。これで、次の図に示すように、クライアントは引き続きトラフィックを渡すことができます。



このローミング イベントの WLC デバッグの要約を次に示します。

```
*apfMsConnTask_2: Jun 25 15:43:33.749: 00:40:96:b7:ab:5c
  CCKM: Received REASSOC REQ IE
*apfMsConnTask_2: Jun 25 15:43:33.749: 00:40:96:b7:ab:5c
  Reassociation received from mobile on BSSID
  84:78:ac:f0:2a:93
*apfMsConnTask_2: Jun 25 15:43:33.750: 00:40:96:b7:ab:5c
  Processing WPA IE type 221, length 22 for mobile
  00:40:96:b7:ab:5c
*apfMsConnTask_2: Jun 25 15:43:33.750: 00:40:96:b7:ab:5c
```

```
  CCKM: Mobile is using CCKM
!--- The Reassociation Request is received from the client,
      which provides the CCKM information needed in order to
      derive the new keys with a fast-secure roam.

*apfMsConnTask_2: Jun 25 15:43:33.750: 00:40:96:b7:ab:5c
  Setting active key cache index 0 ---> 8

*apfMsConnTask_2: Jun 25 15:43:33.750: 00:40:96:b7:ab:5c
  CCKM: Processing REASSOC REQ IE

*apfMsConnTask_2: Jun 25 15:43:33.750: 00:40:96:b7:ab:5c
  CCKM: using HMAC MD5 to compute MIC
!--- WLC computes the MIC used for this CCKM fast-roaming
      exchange.

*apfMsConnTask_2: Jun 25 15:43:33.750: 00:40:96:b7:ab:5c
  CCKM: Received a valid REASSOC REQ IE

*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  CCKM: Initializing PMK cache entry with a new PTK
!--- The new PTK is derived.

*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 8

*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 8

*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 0

*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  Creating a PKC PMKID Cache entry for station
  00:40:96:b7:ab:5c (RSN 0) on BSSID 84:78:ac:f0:2a:93
!--- The new PMKID cache entry is created for this new
      AP-to-client association.

*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  CCKM: using HMAC MD5 to compute MIC
*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  Including CCKM Response IE (length 62) in Assoc Resp to mobile
*apfMsConnTask_2: Jun 25 15:43:33.751: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:2a:93
  (status 0) ApVapId 4 Slot 0
!--- The Reassociation Response is sent from the WLC/AP to
      the client, which includes the CCKM information required
      in order to confirm the new fast-roam and key derivation.

*dot1xMsgTask: Jun 25 15:43:33.757: 00:40:96:b7:ab:5c
  Skipping EAP-Success to mobile 00:40:96:b7:ab:5c
!--- EAP is skipped due to the fast roaming, and CCKM does not
      require further key handshakes. The client is now ready to
      pass encrypted data frames on the new AP.
```

次に示すように、新しい暗号キーが引き続き生成されますが、CCKMネゴシエーションスキームに基づくため、EAP認証フレームが回避され、さらに4方向ハンドシェイクが実行される間に、高速セキュアローミングが実行されます。高速セキュア ローミングは、ローミングの再アソシエーション フレームと、クライアントおよび WLC によって以前にキャッシュされた情報によって完了します。

FlexConnect での CCKM の使用

- 中央認証がサポートされています。これには、ローカルおよび中央のデータスイッチングが含まれます。APは同じFlexConnectグループに属している必要があります。
- Flex Local Authenticationがサポートされています。接続モードでは、キャッシュはAPからコントローラに配布され、次にFlexConnectグループ内の残りのAPに配布されます。
- スタンドアロンモードがサポートされます。キャッシュがすでにAP上に存在する場合（以前の配布による）、高速ローミングが動作します。スタンドアロンモードの新しい認証では、高速セキュアローミングはサポートされません。

CCKM の長所

- CCKM は、エンタープライズ WLAN で最も多く導入されている、最速の高速セキュア ローミング方式です。 クライアントは、AP 間の移動の際に新しい鍵を導出するためにキー管理のハンドシェイクを行う必要がなく、当該の WLAN でのクライアントのライフタイム中は、新しい AP で 再度 802.1X/EAP 認証を実行する必要がありません。
- CCKM では、レガシー クライアントで引き続き使用されているシスコ独自のレガシー方式に加え、802.11 標準規格の範囲で使用できるすべての暗号化方式（WEP、TKIP、および AES）をサポートしています。

CCKM の短所

- CCKM はシスコ独自の方式であるため、導入およびサポートは、Cisco WLAN インフラストラクチャおよび CCX ワイヤレス クライアントに限られます。
- CCXバージョン5は広く採用されていないため、WPA2/AESを使用したCCKMは多くのCCXワイヤレスクライアントでサポートされていません（主な理由は、これらのクライアントのほとんどがWPA/TKIPを使用したCCKMをすでにサポートしているため、これはまだ非常に安全です）。

PMKID キャッシング/Sticky Key Caching を使用しての高速セキュア ローミング

Pairwise Think Key ID(PMKID)キャッシング、つまりSticky Key Caching(SKC)は、802.11iセキュリティ修正案のIEEE 802.11標準によって提案された最初の高速セキュアローミング方式であり、主な目的はWLANの高レベルのセキュリティを標準化することです。この高速セキュア ローミング手法は、WPA2 デバイスに向けた任意の方法として、このセキュリティが実行されているときのローミングを改善するために追加されました。

これが可能なのは、クライアントと認証サーバは、クライアントが完全に EAP 認証されるたびに、PMK の導出に使用される MSK を導出するためです。これは、（クライアントが別のAPにローミングするか、セッションの期限が切れるまで）セッションに使用される最終的なユニキャスト暗号化キー(PTK)を導出するために、WPA2の4方向ハンドシェイクのシードとして使用されます。したがって、この方法は、クライアントとAPによってキャッシュされた元のPMKを再利用するため、ローミング時のEAP認証フェーズを防止します。新しい暗号キーを導出するためにクライアントで必要となるのは、WPA2 の 4 方向ハンドシェイクを実行することだけです。

この方式は、802.11 標準規格で推奨される高速セキュア ローミング方式として広く導入されているわけではありません。その主な理由は次のとおりです。

- 802.11i修正の目的は高速セキュアローミングに関係せず、IEEEはすでにWLANの高速セキュ

アローミングを標準化する別の修正（802.11r、このドキュメントで後述）に取り組んでいるため、この方法はオプションであり、すべてのWPA2デバイスでサポートされているわけではありません。

- この方法には実装に大きな制限があります。無線クライアントは、以前に認証および接続したAPに戻ってローミングする場合にのみ、高速セキュアローミングを実行できます。

この方式では、APへの最初の関連付けは、WLANへの通常の初回認証と同様です。この場合、次の画面イメージに示すように、クライアントがデータフレームを送信する前に、認証サーバに対する802.1X/EAP認証全体とキー生成のための4方向ハンドシェイクが実行される必要があります。

No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	802.11	2462	Authentication, SN=2, FN=0, Flags=.....
2	0.000814	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	802.11	2462	Authentication, SN=4052, FN=0, Flags=...
3	0.002747	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	802.11	2462	Association Request, SN=3, FN=0, Flags=.
4	0.007357	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	802.11	2462	Association Response, SN=4053, FN=0, Fla
5	0.011957	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Identity
6	0.022896	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAP	2462	Response, Identity
7	0.044470	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
8	0.069885	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Client Hello
9	0.093349	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
10	0.095916	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAP	2462	Response, Protected EAP (EAP-PEAP)
11	0.112358	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
12	0.116114	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAP	2462	Response, Protected EAP (EAP-PEAP)
13	0.120221	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
14	0.129519	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Certificate, Client Key Exchange, Change
15	0.139156	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
16	0.162262	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAP	2462	Response, Protected EAP (EAP-PEAP)
17	0.166459	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
18	0.171454	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data
19	0.175710	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
20	0.178181	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data
21	0.182858	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
22	0.187006	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data
23	0.192835	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
24	0.197049	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data
25	0.202860	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
26	0.205372	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data
27	0.210763	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAP	2462	Success
28	0.212505	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 1 of 4)
29	0.215434	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 2 of 4)
30	0.219023	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 3 of 4)
31	0.221930	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 4 of 4)
32	0.224559	Apple_15:39:32	Cisco_f5:4a:40	84:78:ac:f0:68:d2	802.11	2462	QoS Data, SN=0, FN=0, Flags=.p.....TC

デバッグでは、ここで使用するキー キャッシング手法に関する出力が追加されており、他の方式での WLAN への初期認証時に同じ EAP 認証フレーム交換が行われていることがわかります。これらのデバッグ出力は、EAP フレーム交換の全体を示すものではなく、新しい情報を中心に示すために抜粋したものです。このようにした理由は、認証サーバに対するクライアントの認証が行われるたびに交換される情報は、基本的に同じであるためです。これは、これまでに示されており、パケットイメージに示されているEAP認証フレームと関連付けられているため、わかりやすくするために、デバッグ出力からほとんどのEAPメッセージが削除されています。

```
*apfMsConnTask_0: Jun 22 00:23:15.097: ec:85:2f:15:39:32
  Association received from mobile on BSSID 84:78:ac:f0:68:d2
!--- This is the Association Request from the client.

*apfMsConnTask_0: Jun 22 00:23:15.098: ec:85:2f:15:39:32
  Processing RSN IE type 48, length 20 for mobile ec:85:2f:15:39:32
!--- The WLC/AP finds an Information Element that claims PMKID
     Caching support on the Association request that is sent
     from the client.

*apfMsConnTask_0: Jun 22 00:23:15.098: ec:85:2f:15:39:32
  Received RSN IE with 0 PMKIDs from mobile ec:85:2f:15:39:32
!--- Since this is an initial association, the Association
     Request comes without any PMKID.
```

```
*apfMsConnTask_0: Jun 22 00:23:15.098: ec:85:2f:15:39:32
  Setting active key cache index 8 ---> 8

*apfMsConnTask_0: Jun 22 00:23:15.099: ec:85:2f:15:39:32
  Sending Assoc Response to station on BSSID 84:78:ac:f0:68:d2
  (status 0) ApVapId 3 Slot 0
!--- The Association Response is sent to the client.

*dot1xMsgTask: Jun 22 00:23:15.103: ec:85:2f:15:39:32
  Sending EAP-Request/Identity to mobile ec:85:2f:15:39:32
  (EAP Id 1)

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.118: ec:85:2f:15:39:32
  Received EAPOL EAPPKT from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.118: ec:85:2f:15:39:32
  Received Identity Response (count=1) from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.126: ec:85:2f:15:39:32
  Processing Access-Challenge for mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.126: ec:85:2f:15:39:32
  Sending EAP Request from AAA to mobile ec:85:2f:15:39:32
  (EAP Id 2)

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.146: ec:85:2f:15:39:32
  Received EAPOL EAPPKT from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.146: ec:85:2f:15:39:32
  Received EAP Response from mobile ec:85:2f:15:39:32
  (EAP Id 2, EAP Type 25)

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274: ec:85:2f:15:39:32
  Processing Access-Accept for mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274: ec:85:2f:15:39:32
  Creating a PKC PMKID Cache entry for station ec:85:2f:15:39:32
  (RSN 2)
*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274: ec:85:2f:15:39:32
  Setting active key cache index 8 ---> 8
*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274: ec:85:2f:15:39:32
  Setting active key cache index 8 ---> 0
*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274: ec:85:2f:15:39:32
  Adding BSSID 84:78:ac:f0:68:d2 to PMKID cache at index 0
  for station ec:85:2f:15:39:32
*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274:
  New PMKID: (16)
*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274:
  [0000] c9 4d 0d 97 03 aa a9 0f 1b c8 33 73 01 f1 18 f5
!--- WLC creates a PMK cache entry for this client, which is
     used for SKC in this case, so the PMKID is computed with
     the AP MAC address (BSSID 84:78:ac:f0:68:d2).

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.274: ec:85:2f:15:39:32
  Sending EAP-Success to mobile ec:85:2f:15:39:32
  (EAP Id 12)

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.275:
  Including PMKID in M1  (16)
!--- The hashed PMKID is included on the Message-1 of the
     WPA/WPA2 4-Way handshake.

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.275:
```

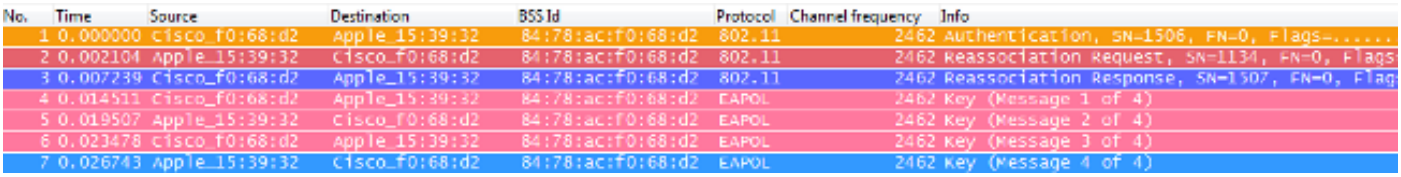
```
  [0000] c9 4d 0d 97 03 aa a9 0f 1b c8 33 73 01 f1 18 f5
!--- This is the hashed PMKID.

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.275: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32
  state INITPMK (message 1), replay counter
  00.00.00.00.00.00.00.00
!--- Message-1 of the WPA/WPA2 4-Way handshake is sent from
     the WLC/AP to the client.

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.284: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32
*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.284: ec:85:2f:15:39:32
  Received EAPOL-key in PTK_START state (message 2) from mobile
  ec:85:2f:15:39:32
!--- Message-2 of the WPA/WPA-2 4-Way handshake is successfully
     received from the client.

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.284: ec:85:2f:15:39:32
  PMK: Sending cache add

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.285: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32
  state PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01
!--- Message-3 of the WPA/WPA2 4-Way handshake is sent from
     the WLC/AP to the client.

*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.291: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32
*Dot1x_NW_MsgTask_2: Jun 22 00:23:15.291: ec:85:2f:15:39:32
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile ec:85:2f:15:39:32
!--- Message-4 (final message) of this initial WPA/WPA2 4-Way
     handshake is successfully received from the client, which
     confirms the installation of the derived keys. They can
     now be used in order to encrypt data frames with the current AP.
```

この方式では、AP とワイヤレス クライアントは、確立済みのセキュアな関連付けの PMK をキャッシュします。したがって、無線クライアントが関連付けされていない新しいAPにローミングする場合、クライアントは新しいAPにローミングする次の図に示すように、完全なEAP認証を再度実行する必要があります。

No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Apple_15:39:32	Cisco_f0:2a:92	84:78:ac:f0:2a:92	802.11	2437	Authentication, SN=462, FN=0, Flags=.
2	0.000819	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	802.11	2437	Authentication, SN=3633, FN=0, Flags=.
3	0.002754	Apple_15:39:32	Cisco_f0:2a:92	84:78:ac:f0:2a:92	802.11	2437	Reassociation Request, SN=463, FN=0,
4	0.007638	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	802.11	2437	Reassociation Response, SN=3634, FN=0
5	0.013519	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	EAP	2437	Request, Identity
6	0.043063	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	EAP	2437	Request, Protected EAP (EAP-PEAP)
7	0.054400	Apple_15:39:32	Cisco_f0:2a:92	84:78:ac:f0:2a:92	TLSv1	2437	Client Hello
8	0.060031	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	TLSv1	2437	Server Hello, Change Cipher Spec, Enc
9	0.093278	Apple_15:39:32	Cisco_f0:2a:92	84:78:ac:f0:2a:92	TLSv1	2437	Change Cipher Spec, Encrypted Handsha
10	0.099981	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	TLSv1	2437	Application Data
11	0.105545	Apple_15:39:32	Cisco_f0:2a:92	84:78:ac:f0:2a:92	TLSv1	2437	Application Data
12	0.110891	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	EAP	2437	Success
13	0.112656	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	EAPOL	2437	Key (Message 1 of 4)
14	0.115722	Apple_15:39:32	Cisco_f0:2a:92	84:78:ac:f0:2a:92	EAPOL	2437	Key (Message 2 of 4)
15	0.119364	Cisco_f0:2a:92	Apple_15:39:32	84:78:ac:f0:2a:92	EAPOL	2437	Key (Message 3 of 4)
16	0.123520	Apple_15:39:32	Cisco_f0:2a:92	84:78:ac:f0:2a:92	EAPOL	2437	Key (Message 4 of 4)
17	2.374472	Apple_15:39:32	IPv6mcast_00:00:0(	84:78:ac:f0:2a:92	802.11	2437	QoS Data, SN=6, FN=0, Flags=.p.....TC

ただし、ワイヤレス クライアントが以前に関連付け/認証が実行された AP にローミングする場合は、クライアントは、複数の PMKID をリストした再関連付け要求フレームを送信し、クライアントが以前認証を行ったすべての AP からキャッシュされた PMK を AP に通知します。したがって、クライアントは、元の AP と同様に当該クライアントのためにキャッシュされた PMK を持っている AP に再度ローミングするため、新しい PMK を導出するために EAP で再認証を行う必要はありません。新しい一時的な暗号キーを導出するためにクライアントで必要となるのは、

WPA2 の 4 方向ハンドシェイクを実行することだけです。



No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	802.11	2462	Authentication, SN=1506, FN=0, Flags=.....
2	0.002104	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	802.11	2462	Reassociation Request, SN=1134, FN=0, Flags
3	0.007239	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	802.11	2462	Reassociation Response, SN=1507, FN=0, Flag
4	0.014511	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 1 of 4)
5	0.019507	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 2 of 4)
6	0.023478	Cisco_f0:68:d2	Apple_15:39:32	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 3 of 4)
7	0.026743	Apple_15:39:32	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 4 of 4)

注：この図には、クライアントからの最初の802.11オープンシステム認証フレームは示されていませんが、このフレームは常に必要であるため、実装されている方式によるものではありません。理由は、この例ではOver-the-Air（OTA；無線）フレームをスニファするために使用されるアダプタまたはワイヤレスパケットイメージソフトウェアによって、この特定のフレームがイメージされていないためです。ただし、この例では教育目的のためにこのフレームを残しています。Over-the-Air（OTA；無線）パケットイメージを実行する際に、これが発生する可能性があることに注意してください。一部のフレームはイメージで欠落する可能性がありますが、実際にはクライアントとAPの間で交換されます。そうでないと、この例ではローミングが開始しません。

この高速セキュア ローミング方式の WLC デバッグの要約を次に示します。

```
*apfMsConnTask_0: Jun 22 00:26:40.787: ec:85:2f:15:39:32
  Reassociation received from mobile on BSSID
  84:78:ac:f0:68:d2
!--- This is the Reassociation Request from the client.

*apfMsConnTask_0: Jun 22 00:26:40.787: ec:85:2f:15:39:32
  Processing RSN IE type 48, length 38 for mobile
  ec:85:2f:15:39:32
!--- The WLC/AP finds an Information Element that claims PMKID
     Caching support on the Association request that is sent
     from the client.

*apfMsConnTask_0: Jun 22 00:26:40.787: ec:85:2f:15:39:32
  Received RSN IE with 1 PMKIDs from mobile
  ec:85:2f:15:39:32
!--- The Reassociation Request from the client comes with
     one PMKID.

*apfMsConnTask_0: Jun 22 00:26:40.787:
  Received PMKID:  (16)
*apfMsConnTask_0: Jun 22 00:26:40.788:
  [0000] c9 4d 0d 97 03 aa a9 0f 1b c8 33 73 01 f1 18 f5
!--- This is the PMKID that is received.

*apfMsConnTask_0: Jun 22 00:26:40.788: ec:85:2f:15:39:32
  Searching for PMKID in MSCB PMKID cache for mobile
  ec:85:2f:15:39:32
!--- WLC searches for a matching PMKID on the database.

*apfMsConnTask_0: Jun 22 00:26:40.788: ec:85:2f:15:39:32
  Found an cache entry for BSSID 84:78:ac:f0:68:d2 in
  PMKID cache at index 0 of station ec:85:2f:15:39:32
```

```
*apfMsConnTask_0: Jun 22 00:26:40.788: ec:85:2f:15:39:32
  Found a valid PMKID in the MSCB PMKID cache for mobile
  ec:85:2f:15:39:32
!--- The WLC validates the PMKID provided by the client,
      and confirms that it has a valid PMK cache for this
      client-and-AP pair.

*apfMsConnTask_0: Jun 22 00:26:40.788: ec:85:2f:15:39:32
  Setting active key cache index 1 ---> 0

*apfMsConnTask_0: Jun 22 00:26:40.788: ec:85:2f:15:39:32
  Sending Assoc Response to station on BSSID
  84:78:ac:f0:68:d2(status 0) ApVapId 3 Slot 0
!--- The Reassociation Response is sent to the client, which
      validates the fast-roam with SKC.

*dot1xMsgTask: Jun 22 00:26:40.795: ec:85:2f:15:39:32
  Initiating RSN with existing PMK to mobile
  ec:85:2f:15:39:32
!--- WLC initiates a Robust Secure Network association with
      this client-and-AP pair based on the cached PMK found.
Hence, EAP is avoided as per the next message.

*dot1xMsgTask: Jun 22 00:26:40.795: ec:85:2f:15:39:32
  Skipping EAP-Success to mobile ec:85:2f:15:39:32

*dot1xMsgTask: Jun 22 00:26:40.795: ec:85:2f:15:39:32
  Found an cache entry for BSSID 84:78:ac:f0:68:d2 in
  PMKID cache at index 0 of station ec:85:2f:15:39:32

*dot1xMsgTask: Jun 22 00:26:40.795: Including PMKID in M1(16)
!--- The hashed PMKID is included on the Message-1 of the
      WPA/WPA2 4-Way handshake.

*dot1xMsgTask: Jun 22 00:26:40.795:
  [0000] c9 4d 0d 97 03 aa a9 0f 1b c8 33 73 01 f1 18 f5
!--- The PMKID is hashed. The next messages are the same
      WPA/WPA2 4-Way handshake messages described thus far
      that are used in order to finish the encryption keys
      generation/installation.

*dot1xMsgTask: Jun 22 00:26:40.795: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32 state
  INITPMK (message 1), replay counter 00.00.00.00.00.00.00.00

*Dot1x_NW_MsgTask_2: Jun 22 00:26:40.811: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:26:40.812: ec:85:2f:15:39:32
  Received EAPOL-key in PTK_START state (message 2) from mobile
  ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:26:40.812: ec:85:2f:15:39:32
  PMK: Sending cache add

*Dot1x_NW_MsgTask_2: Jun 22 00:26:40.812: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32 state
  PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01

*Dot1x_NW_MsgTask_2: Jun 22 00:26:40.820: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 22 00:26:40.820: ec:85:2f:15:39:32
```

```
Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
from mobile ec:85:2f:15:39:32
```

FlexConnect での PMKID キャッシング/Sticky Key Caching の使用

- FlexConnectのセットアップでこの方式を使用すると、動作が正常に行われ、WLCに中央認証（中央スイッチングまたはローカルスイッチングを使用）を使用して戻ると、動作は前述の説明と同様に見えますが、このSKC方式はFlexConnectではサポートされていません。
- この方式は、ローカル モード AP を有する CUWN でのみ正式にサポートされており、FlexConnect やその他のモードではサポートされていません。

PMKID キャッシング/Sticky Key Caching の長所

この方式は、キャッシュされたキーを管理する集中型デバイスを使用せず、自律型の独立した AP を使用してローカルに導入することができます。

PMKID キャッシング/Sticky Key Caching の短所

- このドキュメントで前に説明したように、この方式の主な制約は、クライアントが高速セキュア ローミングを実行できるのは、以前に関連付けおよび認証が実行された AP にローミングする場合のみであることです。新しい AP にローミングするときは、クライアントは、EAP 認証の全体を再度完了する必要があります。
- ワイヤレス クライアントと AP は、新しい認証のたびに導出されるすべての PMK を記憶する必要があるため、この機能は通常、キャッシュされる PMK の数に制限されます。この制限は標準規格によって明確に定義されていないため、ベンダーは、それぞれの SKC 導入でさまざまな制限を定義することができます。たとえば、Cisco WLAN コントローラでは現在、PMK のキャッシュができる AP の数は、1 クライアントにつき 8 つまでとなっています。クライアントが 1 セッションにおいてローミングした AP の数が 8 を超えると、新たにキャッシュされたエントリを保存するため、最も古い AP がキャッシュ リストから削除されます。
- この方式はオプションであり、現在でも多くのWPA2デバイスでサポートされていないため、広く採用および導入されているわけではありません。
- SKC は、AP が同じモビリティ グループに属する場合であっても、コントローラ間のローミングが実行されるときはサポートされません。コントローラの間ローミングとは、異なる WLC で管理されている AP の間を移動する場合に発生するものです。

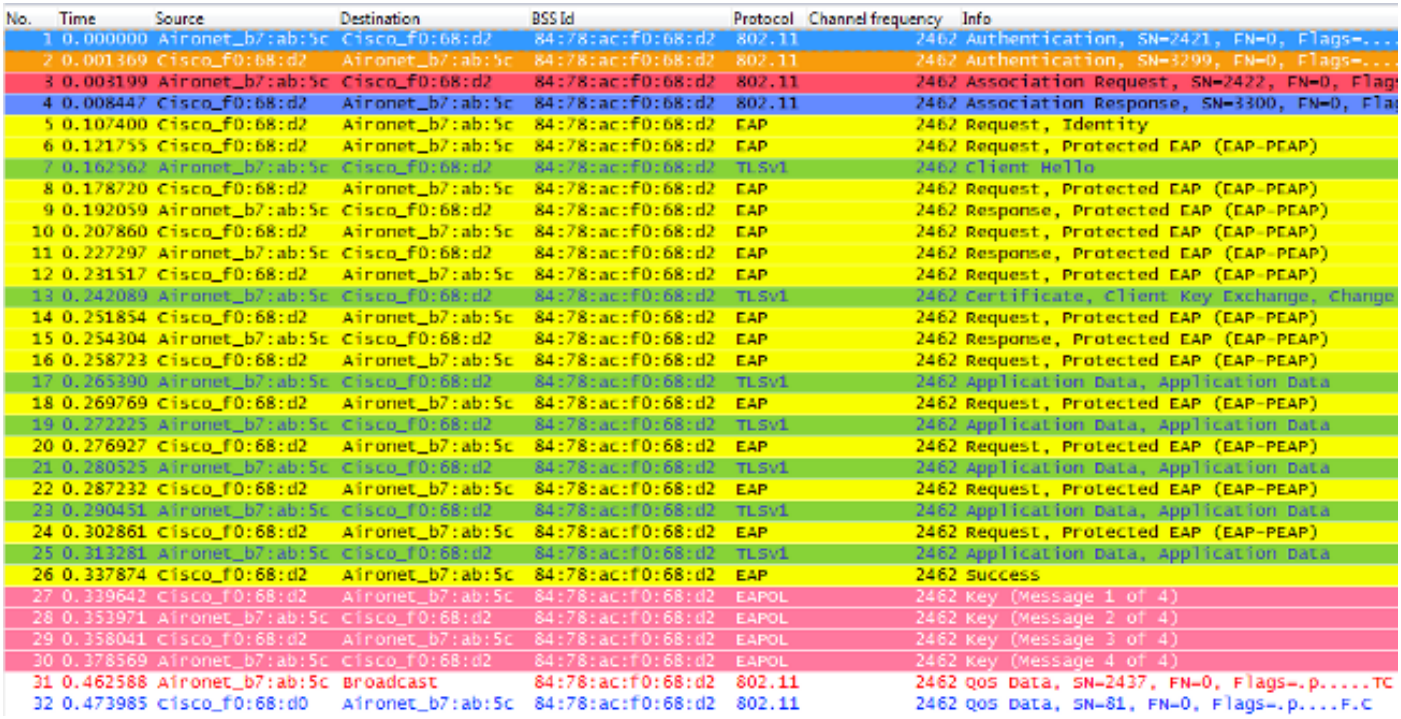
Opportunistic Key Caching を使用しての高速セキュア ローミング

Opportunistic Key Caching(OKC)は、Proactive Key Caching(PKC)（この用語については、次に説明する注で詳しく説明します）とも呼ばれ、基本的には前述したWPA2 PMKIDキャッシング方式の拡張であるため、Proactive/Opportunistic PMKID Cachingとも呼ばれます。したがって、標準規格 802.11 によって定義される高速セキュア ローミングの方式ではなく、多くのデバイスでサポートされていないことには注意が必要です。ただし、PMKID キャッシングと同様に WPA2-EAP と一緒に使用することができます。

この手法では、複数の AP 間でローミングする場合であっても、すべての AP が元の PMK（すべての WPA2 の 4 方向ハンドシェイクでシードとして使用される）を共有するため、ワイヤレス クライアントと WLAN インフラストラクチャは、クライアントと WLAN との関連付けのライフ

タイムにわたり、1 つの PMK（認証サーバでの初回の 802.1X/EAP 認証の後に MSK から導出されるもの）のみをキャッシュするだけで済みます。WPA2 の 4 方向ハンドシェイクは、SKC の場合と同様、クライアントが AP と再関連付けするたびに新しい暗号キーを生成するため、依然として必要です。複数の AP がクライアント セッションからの元の 1 つの PMK を共有するためには、元の PMK をキャッシュしてすべての AP に配布する集中型デバイスを使用し、それらの AP を何らかの管理制御下に置く必要があります。これはCUWNに似ています。CUWNでは、WLCが制御下のすべてのLAPに対してこのジョブを実行し、複数のWLC間でこのPMKを処理するためにモビリティグループを使用します。したがって、これはAutonomous AP環境に対する制限です。

この方式では、PMKID キャッシング（SKC）の場合と同様、WLAN への通常の初回認証が、AP への初回関連付けとなります。そこでは、データ フレームを送信する前に、認証サーバに対する 802.1X/EAP 認証全体と、キーを生成するための 4 方向ハンドシェイクを完了する必要があります。これを示す画面イメージを次に示します。



No.	Time	Source	Destination	BSS Id	Protocol	Channel frequency	Info
1	0.000000	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	802.11	2462	Authentication, SN=2421, FN=0, Flags=...
2	0.001369	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	802.11	2462	Authentication, SN=3299, FN=0, Flags=...
3	0.003199	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	802.11	2462	Association Request, SN=2422, FN=0, Flag
4	0.008447	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	802.11	2462	Association Response, SN=3300, FN=0, Flag
5	0.107400	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Identity
6	0.121755	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
7	0.162562	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Client Hello
8	0.178720	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
9	0.192059	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAP	2462	Response, Protected EAP (EAP-PEAP)
10	0.207860	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
11	0.227297	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAP	2462	Response, Protected EAP (EAP-PEAP)
12	0.231517	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
13	0.242089	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Certificate, Client Key Exchange, Change
14	0.251854	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
15	0.254304	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAP	2462	Response, Protected EAP (EAP-PEAP)
16	0.258723	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
17	0.265390	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data, Application Data
18	0.269769	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
19	0.272225	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data, Application Data
20	0.276927	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
21	0.280525	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data, Application Data
22	0.287232	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
23	0.290451	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data, Application Data
24	0.302861	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Request, Protected EAP (EAP-PEAP)
25	0.313281	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	TLSv1	2462	Application Data, Application Data
26	0.337874	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAP	2462	Success
27	0.339642	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 1 of 4)
28	0.353971	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 2 of 4)
29	0.358041	Cisco_f0:68:d2	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 3 of 4)
30	0.378569	Aironet_b7:ab:5c	Cisco_f0:68:d2	84:78:ac:f0:68:d2	EAPOL	2462	Key (Message 4 of 4)
31	0.462588	Aironet_b7:ab:5c	Broadcast	84:78:ac:f0:68:d2	802.11	2462	QoS Data, SN=2437, FN=0, Flags=.p.....TC
32	0.473985	Cisco_f0:68:d0	Aironet_b7:ab:5c	84:78:ac:f0:68:d2	802.11	2462	QoS Data, SN=81, FN=0, Flags=.p....F.C

デバッグ出力には、WLANへの初期認証時に（図に示すように）、このドキュメントで説明されているその他の方式と基本的に同じEAP認証フレーム交換が示され、WLCで使用されるキーキャッシング技術に関する出力が追加されています。次のデバッグ出力も抜粋であり、関連する情報だけを示しています。

```
*apfMsConnTask_0: Jun 21 21:46:06.515: 00:40:96:b7:ab:5c
  Association received from mobile on BSSID
  84:78:ac:f0:68:d2
!--- This is the Association Request from the client.

*apfMsConnTask_0: Jun 21 21:46:06.516: 00:40:96:b7:ab:5c
  Processing RSN IE type 48, length 20 for mobile
  00:40:96:b7:ab:5c
!--- The WLC/AP finds an Information Element that claims
     PMKID Caching support on the Association request that
     is sent from the client.

*apfMsConnTask_0: Jun 21 21:46:06.516: 00:40:96:b7:ab:5c
  Received RSN IE with 0 PMKIDs from mobile
  00:40:96:b7:ab:5c
!--- Since this is an initial association, the Association
```

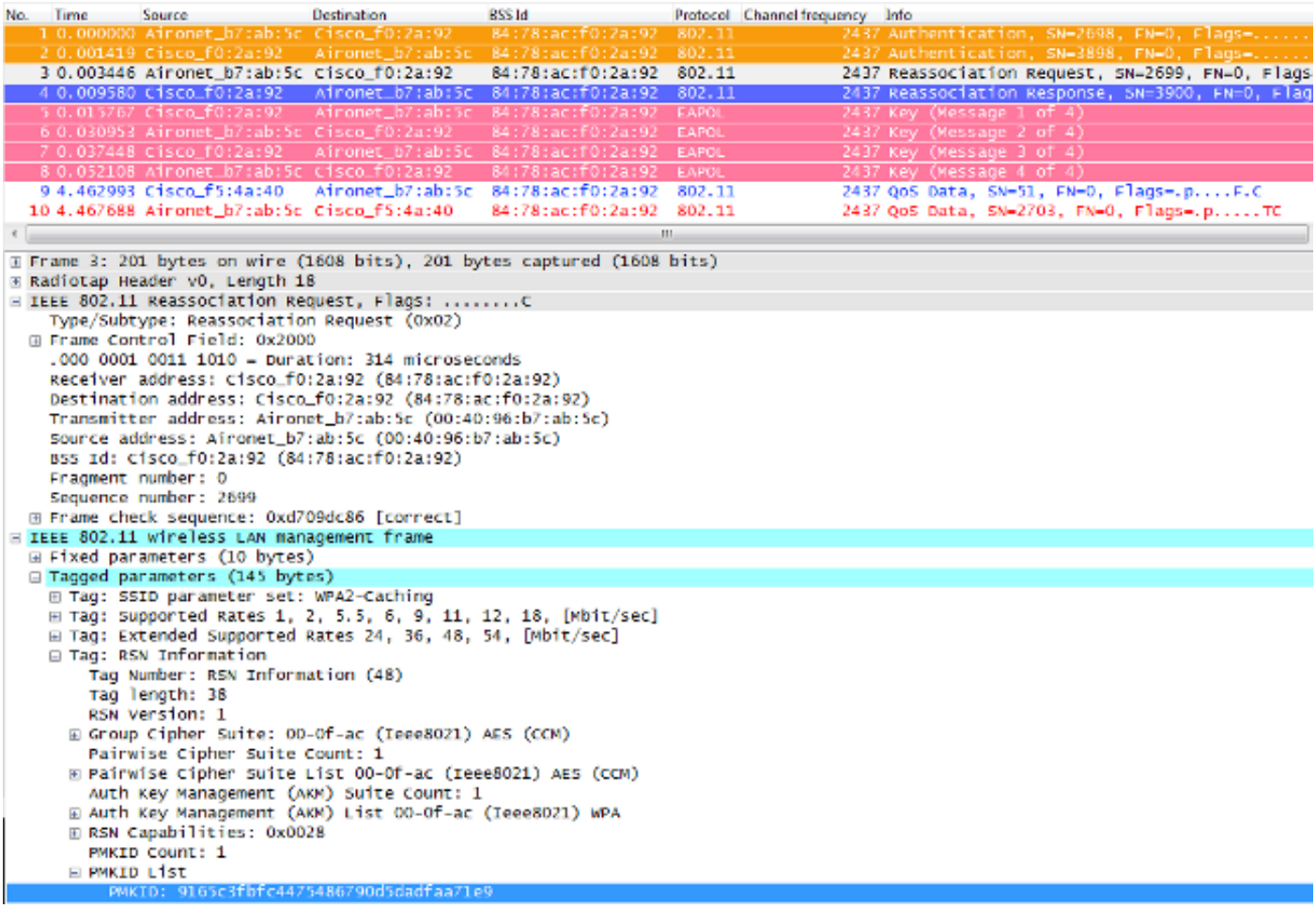
```
     Request comes without any PMKID.

*apfMsConnTask_0: Jun 21 21:46:06.516: 00:40:96:b7:ab:5c
  Setting active key cache index 0 ---> 8

*apfMsConnTask_0: Jun 21 21:46:06.516: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID
  84:78:ac:f0:68:d2 (status 0) ApVapId 3 Slot
!--- The Association Response is sent to the client.

*dot1xMsgTask: Jun 21 21:46:06.522: 00:40:96:b7:ab:5
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
  (EAP Id 1)

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.614: 00:40:96:b7:ab:5c
  Received EAPOL START from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.614: 00:40:96:b7:ab:5c
  Sending EAP-Request/Identity to mobile 00:40:96:b7:ab:5c
  (EAP Id 2)

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.623: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.623: 00:40:96:b7:ab:5c
  Received Identity Response (count=2) from mobile
  00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.630: 00:40:96:b7:ab:5c
  Processing Access-Challenge for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.630: 00:40:96:b7:ab:5c
  Sending EAP Request from AAA to mobile 00:40:96:b7:ab:5c
  (EAP Id 3)

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.673: 00:40:96:b7:ab:5c
  Received EAPOL EAPPKT from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.673: 00:40:96:b7:ab:5c
  Received EAP Response from mobile 00:40:96:b7:ab:5c
  (EAP Id 3, EAP Type 25)

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.843: 00:40:96:b7:ab:5c
  Processing Access-Accept for mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  Creating a PKC PMKID Cache entry for station
  00:40:96:b7:ab:5c (RSN 2)
*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 8
*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  Setting active key cache index 8 ---> 0
*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  Adding BSSID 84:78:ac:f0:68:d2 to PMKID cache at index 0
  for station 00:40:96:b7:ab:5
*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: New PMKID: (16)
*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844:
  [0000] 4e a1 7f 5a 75 48 9c f9 96 e3 a8 71 25 6f 11 d0
!--- WLC creates a PMK cache entry for this client, which is
     used for OKC in this case, so the PMKID is computed
     with the AP MAC address (BSSID 84:78:ac:f0:68:d2).

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  PMK sent to mobility group
```

```
!--- The PMK cache entry for this client is shared with the
     WLCs on the mobility group.

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  Sending EAP-Success to mobile 00:40:96:b7:ab:5c (EAP Id 13)

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  Found an cache entry for BSSID 84:78:ac:f0:68:d2 in PMKID
  cache at index 0 of station 00:40:96:b7:ab:5

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: Including PMKID
  in M1  (16)
!--- The hashed PMKID is included on the Message-1 of the
     WPA/WPA2 4-Way handshake.

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844:
  [0000] 4e a1 7f 5a 75 48 9c f9 96 e3 a8 71 25 6f 11 d0
!--- This is the hashed PMKID. The next messages are the same
     WPA/WPA2 4-Way handshake messages described thus far that
     are used in order to finish the encryption keys
     generation/installation.

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.844: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c state
  INITPMK (message 1), replay counter 00.00.00.00.00.00.00.00

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.865: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.865: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTK_START state (message 2)
  from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.865: 00:40:96:b7:ab:5c
  PMK: Sending cache add

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.865: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c state
  PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.889: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:46:06.890: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile 00:40:96:b7:ab:5c
```

この方式では、ワイヤレス クライアントと (すべての管理対象 AP の) WLC は、最初に確立されたセキュアな関連付けから、元の 1 つの PMK をキャッシュします。基本的に、ワイヤレスクライアントが特定のAPに接続するたびに、クライアントのMACアドレス、APのMACアドレス（WLANのBSSID）、およびそのAPで取得されるPMKに基づいてPMKIDがハッシュされます。したがって、OKC は すべての AP と特定のクライアントのために同じ元 PMK をキャッシュするため、このクライアントが別の AP に (再) 関連付けされる際に新しい PMKID をハッシュするために変更される値は、新しい AP MAC アドレスのみです。

クライアントが新しいAPへのローミングを開始し、再関連付け要求フレームを送信すると、キャッシュされたPMKが高速セキュアローミングに使用されることをAPに通知する必要がある場合、クライアントはWPA2 RSN情報要素にPMKIDを追加します。ローミング先のBSSID(AP)のMACアドレスはすでに認識されており、クライアントはこの再関連付け要求で使用される新しいPMKIDをハッシュするだけです。クライアントからこの要求を受信すると、AP も、すでに保持している値 (キャッシュされた PMK、クライアントの MAC アドレス、および当該 AP の MAC

アドレス）を使用して PMKID をハッシュし、PMKID の一致を確認する成功の再関連付け応答で応答します。キャッシュされたPMKは、新しい暗号キーを取得するために（およびEAPをスキップするために）、WPA2の4方向ハンドシェイクを開始するシードとして使用できます。



この図では、クライアントからの再関連付け要求フレームが選択され、展開されて、フレームの詳細を確認できます。MACアドレス情報と、802.11i - WPA2に準拠したRobust Security Network（RSN；堅牢なセキュリティネットワーク）の情報要素が含まれており、この関連付けに使用されるWPA2設定に関する情報が示されています（強調表示されているのは、ハッシュされた数式から取得されたPMKIDです）。

OKC を使用した高速セキュア ローミング方式の WLC デバッグの要約を次に示します。

```
*apfMsConnTask_2: Jun 21 21:48:50.562: 00:40:96:b7:ab:5c
  Reassociation received from mobile on BSSID
  84:78:ac:f0:2a:92
!--- This is the Reassociation Request from the client.

*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  Processing RSN IE type 48, length 38 for mobile
  00:40:96:b7:ab:5c
!--- The WLC/AP finds and Information Element that claims
      PMKID Caching support on the Association request that
      is sent from the client.

*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  Received RSN IE with 1 PMKIDs from mobile
  00:40:96:b7:ab:5c
!--- The Reassociation Request from the client comes with
      one PMKID.
```

```
*apfMsConnTask_2: Jun 21 21:48:50.563:
  Received PMKID:  (16)

*apfMsConnTask_2: Jun 21 21:48:50.563:
  [0000] 91 65 c3 fb fc 44 75 48 67 90 d5 da df aa 71 e9

*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  Searching for PMKID in MSCB PMKID cache for mobile
  00:40:96:b7:ab:5c
*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  No valid PMKID found in the MSCB PMKID cache for mobile
  00:40:96:b7:ab:5
!--- As the client has never authenticated with this new AP,
      the WLC cannot find a valid PMKID to match the one provided
      by the client. However, since the client performs OKC
      and not SKC (as per the following messages), the WLC computes
      a new PMKID based on the information gathered (the cached PMK,
      the client MAC address, and the new AP MAC address).

*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  Trying to compute a PMKID from MSCB PMK cache for mobile
  00:40:96:b7:ab:5c

*apfMsConnTask_2: Jun 21 21:48:50.563:
  CCKM: Find PMK in cache: BSSID =  (6)
*apfMsConnTask_2: Jun 21 21:48:50.563:
  [0000] 84 78 ac f0 2a 90
*apfMsConnTask_2: Jun 21 21:48:50.563:
  CCKM: Find PMK in cache: realAA =  (6)
*apfMsConnTask_2: Jun 21 21:48:50.563:
  [0000] 84 78 ac f0 2a 92
*apfMsConnTask_2: Jun 21 21:48:50.563:
  CCKM: Find PMK in cache: PMKID =  (16)
*apfMsConnTask_2: Jun 21 21:48:50.563:
  [0000] 91 65 c3 fb fc 44 75 48 67 90 d5 da df aa 71 e9
*apfMsConnTask_2: Jun 21 21:48:50.563:
  CCKM: AA (6)
*apfMsConnTask_2: Jun 21 21:48:50.563:
  [0000] 84 78 ac f0 2a 92
*apfMsConnTask_2: Jun 21 21:48:50.563:
  CCKM: SPA (6)
*apfMsConnTask_2: Jun 21 21:48:50.563:
  [0000] 00 40 96 b7 ab 5c
*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  Adding BSSID 84:78:ac:f0:2a:92 to PMKID cache at
  index 0 for station 00:40:96:b7:ab:5c
*apfMsConnTask_2: Jun 21 21:48:50.563:
  New PMKID: (16)
*apfMsConnTask_2: Jun 21 21:48:50.563:
  [0000] 91 65 c3 fb fc 44 75 48 67 90 d5 da df aa 71 e9
*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  Computed a valid PMKID from MSCB PMK cache for mobile
  00:40:96:b7:ab:5c
!--- The new PMKID is computed and validated to match the
      one provided by the client, which is also computed with
      the same information. Hence, the fast-secure roam is
      possible.

*apfMsConnTask_2: Jun 21 21:48:50.563: 00:40:96:b7:ab:5c
  Setting active key cache index 0 ---> 0

*apfMsConnTask_2: Jun 21 21:48:50.564: 00:40:96:b7:ab:5c
  Sending Assoc Response to station on BSSID 84:78:ac:f0:2a:92
```

```
  (status 0) ApVapId 3 Slot
!--- The Reassociation response is sent to the client, which
     validates the fast-roam with OKC.

*dot1xMsgTask: Jun 21 21:48:50.570: 00:40:96:b7:ab:5c
  Initiating RSN with existing PMK to mobile
  00:40:96:b7:ab:5c
!--- WLC initiates a Robust Secure Network association with
     this client-and AP pair with the cached PMK found.
Hence, EAP is avoided, as per the the next message.

*dot1xMsgTask: Jun 21 21:48:50.570: 00:40:96:b7:ab:5c
  Skipping EAP-Success to mobile 00:40:96:b7:ab:5c

*dot1xMsgTask: Jun 21 21:48:50.570: 00:40:96:b7:ab:5c
  Found an cache entry for BSSID 84:78:ac:f0:2a:92 in
  PMKID cache at index 0 of station 00:40:96:b7:ab:5c

*dot1xMsgTask: Jun 21 21:48:50.570:
  Including PMKID in M1  (16)
!--- The hashed PMKID is included on the Message-1 of the
     WPA/WPA2 4-Way handshake.

*dot1xMsgTask: Jun 21 21:48:50.570:
  [0000] 91 65 c3 fb fc 44 75 48 67 90 d5 da df aa 71 e9
!--- The PMKID is hashed. The next messages are the same
     WPA/WPA2 4-Way handshake messages described thus far,
     which are used in order to finish the encryption keys
     generation/installation.

*dot1xMsgTask: Jun 21 21:48:50.570: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c state
  INITPMK (message 1), replay counter 00.00.00.00.00.00.00.00

*Dot1x_NW_MsgTask_4: Jun 21 21:48:50.589: 00:40:96:b7:ab:5
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:48:50.589: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTK_START state (message 2) from mobile
  00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:48:50.589: 00:40:96:b7:ab:5c
  PMK: Sending cache add

*Dot1x_NW_MsgTask_4: Jun 21 21:48:50.590: 00:40:96:b7:ab:5c
  Sending EAPOL-Key Message to mobile 00:40:96:b7:ab:5c state
  PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01

*Dot1x_NW_MsgTask_4: Jun 21 21:48:50.610: 00:40:96:b7:ab:5c
  Received EAPOL-Key from mobile 00:40:96:b7:ab:5c

*Dot1x_NW_MsgTask_4: Jun 21 21:48:50.610: 00:40:96:b7:ab:5c
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile 00:40:96:b7:ab:5c
```

デバッグ出力の最初に示すように、クライアントからの再アソシエーション要求が受信された後で、PMKID が計算される必要があります。これは、PMKID を検証することで、キャッシュされた PMK が WPA2 4 ウェイ ハンドシェイクに使用され、暗号化キーが生成され、高速セキュア ローミングが完了したことを確認するために必要となります。デバッグのCCKM エントリを混同しないでください。これは、前述したように、CCKMではなくOKCを実行するために使用されます。ここでの CCKM は、出力用に WLC によって使用される名前にすぎず、PMKID を計算するために値を処理する機能の名前のようなものです。

FlexConnect での Opportunistic Key Caching の使用

- 中央認証がサポートされています。これには、ローカルおよび中央のデータスイッチングが含まれます。APが同じFlexConnectグループの一部である場合、高速セキュアローミングはAPによって行われ、それ以外の場合、高速セキュアローミングはコントローラによって行われます。
  注：この設定は、APが同じFlexConnectグループ上にない場合に機能しますが、推奨またはサポートされている設定ではありません。
- Flex Local Authenticationがサポートされています。接続モードでは、キャッシュはAPからコントローラに配布され、次にFlexConnectグループ内の残りのAPに配布されます。
- スタンドアロンモードがサポートされます。 キャッシュがすでにAP上に存在する場合（以前の配布による）、高速セキュアローミングが動作します。スタンドアロンモードの新しい認証では、高速セキュアローミングはサポートされません。

Opportunistic Key Caching の長所

- ワイヤレス クライアントと WLAN インフラストラクチャは、複数の PMKID を記憶する必要がなく、WLAN への初期認証から 1 つの 元の PMK をキャッシュするだけで済みます。その後、高速セキュア ローミングの検証を行うため、各 AP のセキュアな関連付けに必要な、適切な PMKID（再関連付け要求に使用される）を再ハッシュする必要があります。
- ここで、ワイヤレス クライアントは、新しい AP への高速セキュア ローミングを、（SKC とは違って）その AP に関連付けされたことがない場合でも、同じ WLAN/SSID で実行します。クライアントのローミング先となるすべての AP の PMK キャッシュが処理される集中型の導入によって管理されている AP のいずれか 1 つに対して初回の 802.1X/EAP 認証を行っていれば、クライアントは、この WLAN でのライフタイムが終わるまで、全体的な認証を再度行う必要はありません。

Opportunistic Key Caching の短所

- この方式は、集中型の環境、つまり、クライアント セッションから 1 つの元の PMK をキャッシュして共有する役割を担う何らかの管理制御（WLAN コントローラなど）の下にすべての AP が置かれる環境にのみ導入されます。したがって、この方式は、自律型 AP の環境に限定されます。
- この方式に適用される手法は、802.11 規格では提案も記述もされていません。したがって、サポートはデバイスごとに大きく異なります。にもかかわらず、この方式は、802.11r の登場までに比較的多く採用されています。

「Proactive Key Caching」という用語について

Proactive Key Caching(PKC)はOKC(Opportunistic Key Caching)と呼ばれており、この2つの用語は、ここで説明する同じ方法を説明する際に同じ意味で使用されます。しかし、この用語は、2001 年の通信業界では単に古いキャッシング方式の意味で使用されていましたが、802.11i 標準規格により、「事前認証」（高速セキュア ローミングの一種。後に概略を説明します）の基盤という意味で使用されるようになりました。PKC は OKC（Opportunistic Key Caching）とも事前認証とも異なりますが、文献などで言及される PKC は、基本的には事前認証ではなく OKC を指しています。

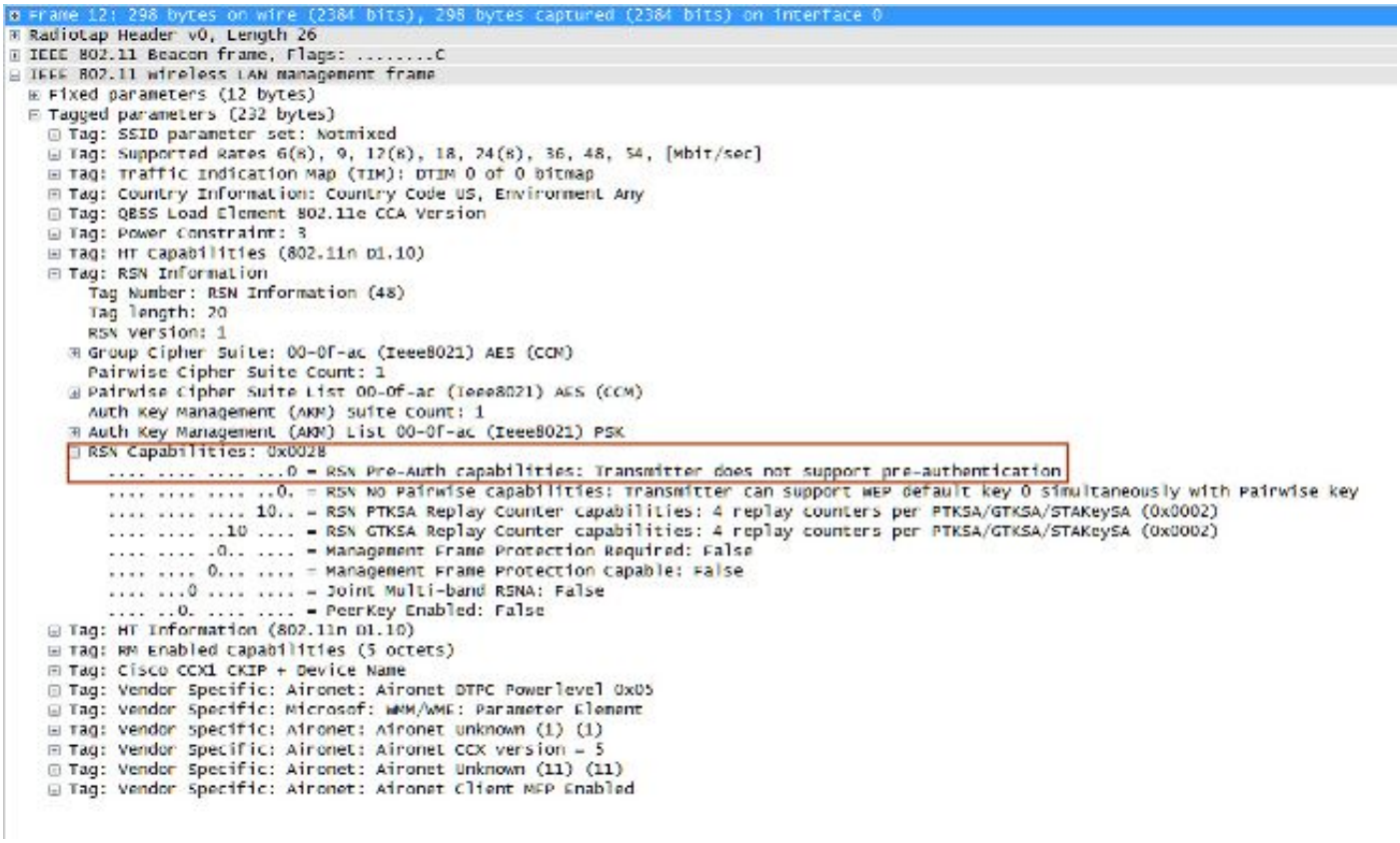
事前認証を使用しての高速セキュア ローミング

この方式も、IEEE 802.11 標準規格の 802.11i セキュリティ修正版で提案されたものであるため、WPA2 でも動作します。しかし、この方式は、Cisco WLAN インフラストラクチャがサポートしていない唯一の高速セキュア ローミング方式です。したがって、ここでは、出力を使用せず簡単に説明するのみとします。

事前認証を使うと、ワイヤレス クライアントは、現在の AP に関連付けられた状態で、1 度に複数の AP への認証ができます。これが発生するとき、クライアントは、現在接続中の AP に EAP 認証フレームを送信しますが、この EAP 認証フレームの宛先は、クライアントが事前認証を行おうとする他の（単数または複数の）AP（ローミング先の候補となりうる近隣の AP）です。現在の AP は、これらのフレームを、分散システムを介してターゲット AP に送信します。新しい AP は、RADIUS サーバに対してこのクライアントの認証の全体を行います。それによって全体的な新しい EAP 認証 ハンドシェイクが完了し、この新しい AP は、オーセンティケータの役割を果たします。

考え方は、クライアントが近隣の AP にローミングを行う前に、それらの AP での認証と PMK 導出を実行するというものです。したがって、ローミングを実行するときには、クライアントはすでに認証されており、PMK も、AP とクライアントの間のこの新しい安全な関連付けのため、すでにキャッシュされています。したがって、クライアントが初回の再関連付け要求を送信した後は、AP と クライアントが 4 方法ハンドシェイクを行うだけで、高速ローミングができます。

事前認証のサポートをアドバタイズするRSN IEフィールドを示すAPビーコンの画像を次に示します（これは、事前認証がサポートされていないことを確認したCisco APのものです）。



事前認証の長所

PMK は、AP と クライアントとの安全な関連付け 1 つにつき 1 つ存在します。このことは、AP が攻撃を受けてキーが盗まれた場合、セキュリティ上の利点とみなすことができます（キーは他

の AP には使用できません）。ただし、このセキュリティ上の利点は、WLAN インフラストラクチャにより、他の方式においてさまざまな方法で対応されています。

事前認証の短所

- PMK は AP ごとに 1 つずつであるため、クライアントが保持できる事前認証可能な AP の数が制限されます。
- クライアントが新しい AP との事前認証を行うたびに EAP 認証全体の交換が行われるため、ネットワークと認証サーバへの負荷が大きくなります。
- この方式はあまり採用されていないため（OKC のほうが多く採用されています）、ほとんどのワイヤレス クライアントはこの方式をサポートしていません。

802.11r を使用しての高速セキュア ローミング

802.11r 修正版に基づく高速セキュア ローミング手法（802.11 標準規格での正式名称は**高速 BSS 移行、別名 FT**）は、IEEE が AP（Basic Service Set（BSS））間の高速移行を実行するためのソリューションとして 802.11 規格で初めて（2008 年）正式に承認した方式であり、WLAN でキーを処理およびキャッシュする際に使用されるキー階層を明確に定義しています。ただし、その採用はあまり進んでいません。主な理由は、高速移行が実際に必要なときに使用可能なソリューションは、すでにほかにもあることです（このドキュメントですでに説明した方式のいずれかの VoWLAN 導入での使用など）。現在（2013 年まで）、何らかの FT オプションをサポートしているデバイスはごく少数です。

この手法は、新しい概念や、さまざまなデバイス（役割が異なる各デバイス）でキャッシュされる PMK の複数のレイヤが導入されていることから、他の方式より説明が複雑になります。また、高速セキュア ローミングのさらなるオプションを提供します。したがって、この方式については、方式の説明と使用可能な各オプションを伴う導入方法を概略的にまとめます。

802.11r は、SKC および OKC とは次の点が異なります。

- ハンドシェイク メッセージング（PMKID、ANonce、および SNonce の交換など）は、再関連付けフレームではなく、802.11 認証フレームまたはアクション フレームで行われます。PMKID キャッシング方式とは異なり、（再）関連付けメッセージの交換後に実行される別途の 4 方向ハンドシェイク フェーズは回避されます。新しい AP によるキー ハンドシェイクは、クライアントによる新しい AP へのローミングまたは関連付けが完全に行われる前に開始されます。
- 高速ローミングハンドシェイクには、AIR経由と分散システム(DS)経由の2つの方法が用意されています。
- 802.11r は、キー階層のレイヤが他の方式より多くなっています。
- このプロトコルは、クライアントがローミングする際にキー管理のための 4 方向ハンドシェイクを回避する（このハンドシェイクを必要とせずに新しい暗号キー（PTK および GTK）を生成する）ため、認証に 802.1X/EAP を使用する場合だけでなく、PSK を使用する WPA2 セットアップにも適用することができます。これにより、これらの EAP または 4 方向ハンドシェイクの交換が行われないセットアップでは、ローミングがさらに高速化されます。

この方式では、ワイヤレス クライアントは、WLAN インフラストラクチャに対する初期認証を、最初の AP への接続が確立されたときに 1 回だけ行って、同じ FT モビリティ ドメインの AP 間でローミングを行う際に高速セキュア ローミングを実行します。

FT モビリティ ドメインとは、新しい概念の 1 つで、基本的には、同じ SSID を使用する複数の

AP（Extended Service Set または ESS と呼ばれます）で、同じ FT キーを使用するものを指します。これは、これまでに説明した他の方式と似ています。APがFTモビリティドメインキーを処理する方法は、通常、WLCやモビリティグループなどの中央集中型セットアップに基づいています。ただし、この方法はAutonomous AP環境にも実装できます。

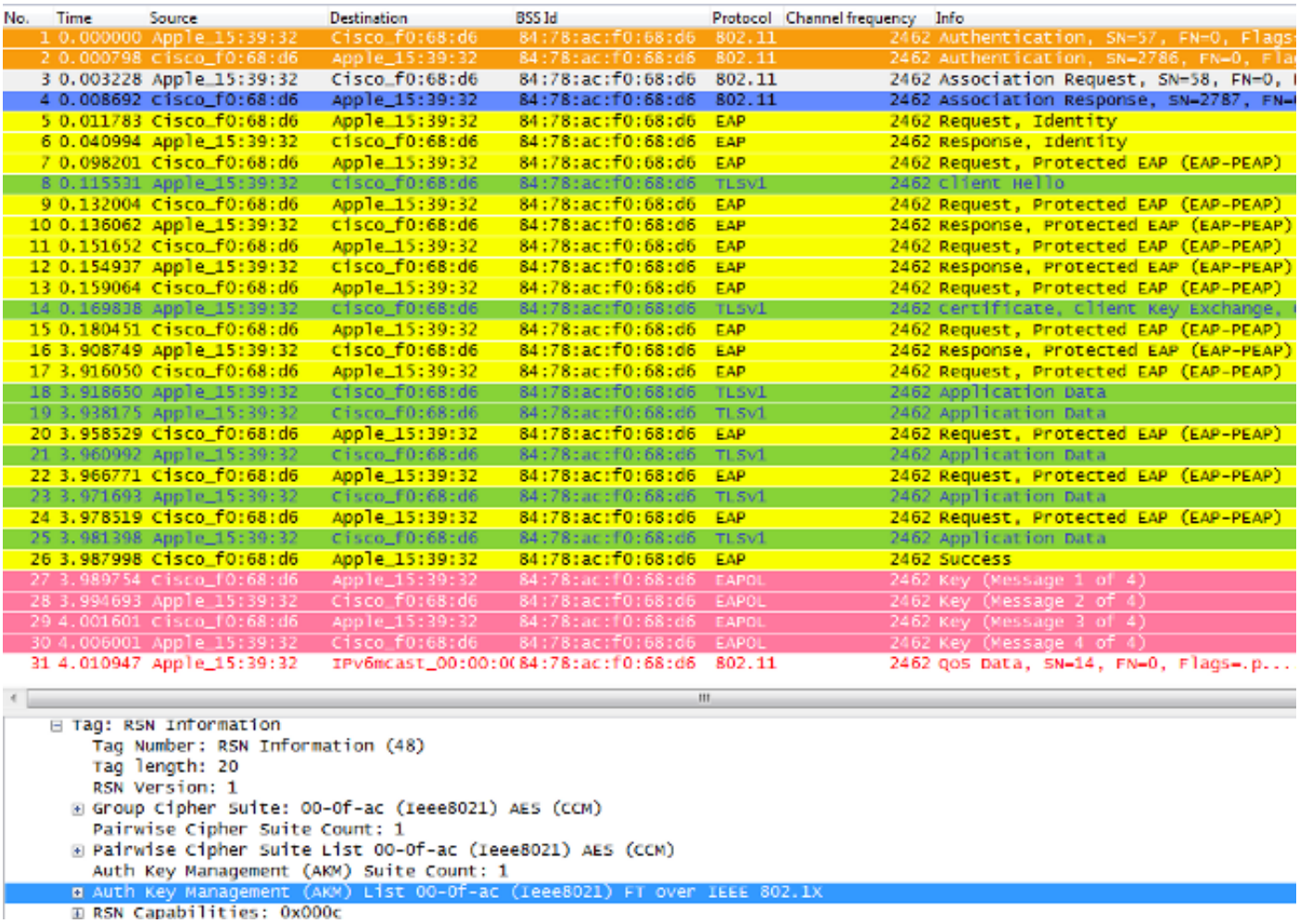
キー階層の概要を次に示します。

- MSK は、ここでも、クライアント サプリカントと認証サーバについて、初回の 802.1X/EAP 認証フェーズから導出されます（認証が成功すると、認証サーバからオーセンティケータ（WLC）に転送されます）。この MSK は、他の方式の場合と同様、FT キー階層のシードとして使用されます。EAP 認証方式ではなく WPA2-PSK を使用する場合、基本的に PSK はこの MSK です。
- ペアワイズ マスター キー R0（PMK-R0）は、FT キー階層の最初のレベルのキーである MSK から導出されます。この PMK-R0 のキー ホルダーは、WLC とクライアントです。
- 2 番目のレベルのキーは、ペアワイズ マスター キー R1（PMK-R1）と呼ばれ、PMK-R0 から導出されます。このキー のホルダーは、クライアントと、PMK-R0 を持つ WLC によって管理される AP です。
- FT キー階層の 3 番目かつ最終レベルのキーは、PTK です。これは、802.11 ユニキャストのデータ フレームを暗号化するために使用される最終的なキーです（WPA/TKIP または WPA2/AES を使用する他の方式と同様です）。この PTK は、FT では、PMK-R1 から導出されます。キー ホルダーは、クライアントと、WLC によって管理される AP です。

注:WLANベンダーと実装の設定（Autonomous AP、FlexConnect、メッシュなど）に応じて、WLANインフラストラクチャは異なる方法でキーを転送および処理できます。キーホルダーの役割を変更することさえできますが、これはこのドキュメントの範囲外であるため、前述のキー階層の要約に基づく例が次の焦点となります。違いは、ソフトウェアの問題を検出するためにインフラストラクチャ デバイス（およびコード）を詳しく分析することが実際に必要でない限り、実際には、プロセスの理解にはそれほど関連しません。

Over-the-Air での高速 BSS 移行

この方式では、APへの最初の関連付けはWLANへの通常の初回認証です。この場合、次の画面イメージに示すように、データフレームが送信される前に、認証サーバに対する802.1X/EAP認証全体とキー生成のための4方向ハンドシェイクが発生する必要があります。



主な違いは次のとおりです。

- 認証キー管理ネゴシエーションが通常の WPA/WPA2 とは若干異なるため、FT をサポートする WLAN インフラストラクチャへの関連付けが行われるときに、このネゴシエーションを行うための追加情報が使用されます。図に示すように、クライアントからの関連付け要求フレームが選択され、RNS情報要素のAKMフィールドが強調表示されて、このクライアントが802.1X/EAP経由でFTを実行する必要があることを示します。
- また、モビリティドメイン情報要素（FTの一部）も示されています。この場合、**FT Capability and Policy**フィールドは、高速ローミング時に高速BSS移行がOver-the-AirまたはOver-the-DSで完了したかどうかを示します（この図ではOver-the-Airを示しています）。
- FT ローミング時に FT 認証シーケンスを実行するのに必要な情報を持つ別の情報要素（このドキュメントで後述する 高速 BSS 移行または FT IE）も追加されています。
- キー階層のため、キー生成が異なります。そのため、FT の 4 方向ハンドシェイクは WPA/WPA2 の 4 方向ハンドシェイクと似ていますが、実際には内容が若干異なります。

デバッグでは、（図から確認したとおり）WLANへの初期認証時に、基本的に他の方式と同じEAP認証フレーム交換が示されますが、WLCで使用されるキーキャッシング方式に関する出力が追加されています。したがって、次のデバッグ出力は、関連情報のみを表示するためにカットされています。

```
*apfMsConnTask_0: Jun 27 19:25:23.426: ec:85:2f:15:39:32
  Association received from mobile on BSSID
  84:78:ac:f0:68:d6
!--- This is the Association request from the client.

*apfMsConnTask_0: Jun 27 19:25:23.427: ec:85:2f:15:39:32
```

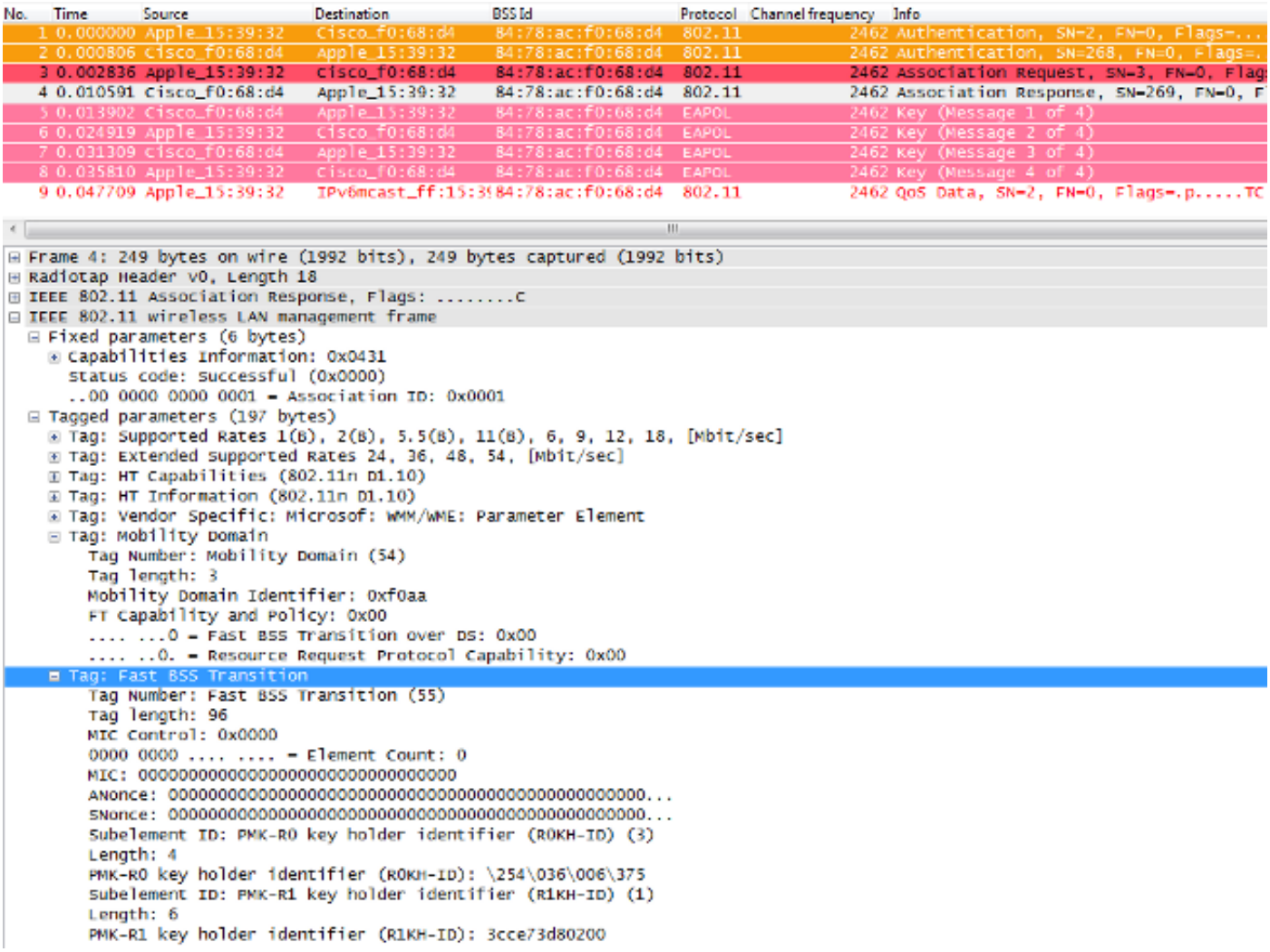
```
  Marking this mobile as TGr capable.
!--- WLC recognizes that the client is 802.11r-capable.

*apfMsConnTask_0: Jun 27 19:25:23.427: ec:85:2f:15:39:32
  Processing RSN IE type 48, length 20 for mobile
  ec:85:2f:15:39:32
!--- The WLC/AP finds an Information Element that claims FT
     support on the Association request that is sent from the client.

*apfMsConnTask_0: Jun 27 19:25:23.427:
  Sending assoc-resp station:ec:85:2f:15:39:32
  AP:84:78:ac:f0:68:d0-00 thread:144be808
*apfMsConnTask_0: Jun 27 19:25:23.427:
  Adding MDIE, ID is:0xaaf0
*apfMsConnTask_0: Jun 27 19:25:23.427: ec:85:2f:15:39:32
  Including FT Mobility Domain IE (length 5) in Initial
  assoc Resp to mobile
*apfMsConnTask_0: Jun 27 19:25:23.427: ec:85:2f:15:39:32
  Sending R0KH-ID as:-84.30.6.-3
*apfMsConnTask_0: Jun 27 19:25:23.427: ec:85:2f:15:39:32
  Sending R1KH-ID as 3c:ce:73:d8:02:00
*apfMsConnTask_0: Jun 27 19:25:23.427: ec:85:2f:15:39:32
  Including FT IE (length 98) in Initial Assoc Resp to mobile
*apfMsConnTask_0: Jun 27 19:25:23.427: ec:85:2f:15:39:32
  Sending Assoc Response to station on BSSID 84:78:ac:f0:68:d6
  (status 0) ApVapId 7 Slot 0
!--- The Association Response is sent to the client once the
     FT information is computed (as per the previous messages),
     so this is included in the response.

*dot1xMsgTask: Jun 27 19:25:23.432: ec:85:2f:15:39:32
  Sending EAP-Request/Identity to mobile ec:85:2f:15:39:32
  (EAP Id 1)
!--- EAP begins, and follows the same exchange explained so far.

*apfMsConnTask_0: Jun 27 19:25:23.436: ec:85:2f:15:39:32
  Got action frame from this client.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.449: ec:85:2f:15:39:32
  Received EAPOL EAPPKT from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.449: ec:85:2f:15:39:32
  Received Identity Response (count=1) from mobile
  ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.456: ec:85:2f:15:39:32
  Processing Access-Challenge for mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.456: ec:85:2f:15:39:32
  Sending EAP Request from AAA to mobile ec:85:2f:15:39:32
  (EAP Id 2)

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.479: ec:85:2f:15:39:32
  Received EAPOL EAPPKT from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.479: ec:85:2f:15:39:32
  Received EAP Response from mobile ec:85:2f:15:39:32
  (EAP Id 2, EAP Type 25)

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.627: ec:85:2f:15:39:32
  Processing Access-Accept for mobile ec:85:2f:15:39:32
!--- The client is validated/authenticated by the RADIUS Server.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.627: ec:85:2f:15:39:32
```

```
  Creating a PKC PMKID Cache entry for station
  ec:85:2f:15:39:32 (RSN 2)
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.627: ec:85:2f:15:39:32
  Resetting MSCB PMK Cache Entry 0 for station
  ec:85:2f:15:39:32
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.627: ec:85:2f:15:39:32
  Setting active key cache index 8 ---> 8
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.628: ec:85:2f:15:39:32
  Setting active key cache index 8 ---> 0
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.628: ec:85:2f:15:39:32
  Adding BSSID 84:78:ac:f0:68:d6 to PMKID cache at index 0
  for station ec:85:2f:15:39:32
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.628: New PMKID: (16)
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.628:
  [0000] 52 b8 8f cf 50 a7 90 98 2b ba d6 20 79 e4 cd f9
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.629: ec:85:2f:15:39:32
  Created PMK Cache Entry for TGr AKM:802.1x ec:85:2f:15:39:32
!--- WLC creates a PMK cache entry for this client, which is
     used for FT with 802.1X in this case, so the PMKID is
     computed with the AP MAC address (BSSID 84:78:ac:f0:68:d6).

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.629:
  ec:85:2f:15:39:32   R0KH-ID:172.30.6.253
  R1KH-ID:3c:ce:73:d8:02:00  MSK Len:48 pmkValidTime:1807
!--- The R0KH-ID and R1KH-ID are defined, as well as the PMK
     cache validity period.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.630: ec:85:2f:15:39:32
  PMK sent to mobility group
!--- The FT PMK cache entry for this client is shared with the
     WLCs on the mobility group.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.630: ec:85:2f:15:39:32
  Sending EAP-Success to mobile ec:85:2f:15:39:32 (EAP Id 12)

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.630: ec:85:2f:15:39:32
  Found an cache entry for BSSID 84:78:ac:f0:68:d6 in PMKID
  cache at index 0 of station ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.630: Including PMKID in
  M1  (16)
!--- The hashed PMKID is included on the Message-1 of the
     initial FT 4-Way handshake.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.630:
  [0000] 52 b8 8f cf 50 a7 90 98 2b ba d6 20 79 e4 cd f9

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.630: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32 state
  INITPMK (message 1), replay counter 00.00.00.00.00.00.00.0
!--- Message-1 of the FT 4-Way handshake is sent from the
     WLC/AP to the client.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.639: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.639: ec:85:2f:15:39:32
  Received EAPOL-key in PTK_START state (message 2) from
  mobile ec:85:2f:15:39:32
!--- Message-2 of the FT 4-Way handshake is received
     successfully from the client.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.639: ec:85:2f:15:39:32
  Calculating PMKR0Name
!--- The PMKR0Name is calculated.
```

```
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.639: ec:85:2f:15:39:32
  DOT11R: Sending cache add

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.639: Adding MDIE,
  ID is:0xaaf0
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.639: ec:85:2f:15:39:32
  Adding TIE for reassociation deadtime:20000 milliseconds
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.639: ec:85:2f:15:39:32
  Adding TIE for R0Key-Data valid time :1807
*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.640: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32 state
  PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01
!--- After the MDIE, TIE for reassociation deadtime, and TIE
     for R0Key-Data valid time are calculated, the Message-3
     of this FT 4-Way handshake is sent from the WLC/AP to the
     client with this information.

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.651: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:25:23.651: ec:85:2f:15:39:32
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile ec:85:2f:15:39:32
!--- Message-4 (final message) of this initial FT 4-Way handshake
     is received successfully from the client, which confirms the
     installation of the derived keys. They can now be used in order
     to encrypt data frames with the current AP.
```

注：この方式をデバッグして、ここに示す追加の802.11r/FT出力に到達するには、**debug client**とともに追加のデバッグを有効にします。これは**debug ft events enable**です。

802.1X/EAP方式ではなく、WPA2-PSKを使用してFTを実行した場合のWLANへの初期関連付けのイメージとデバッグを次に示します。ここでは、Fast BSS Transition Information Element（強調表示）を表示するために、APからのAssociation Response(CRR)フレームが選択されています。FT の 4 方向ハンドシェイクを実行するために必要となるキー 情報の一部も示されています。



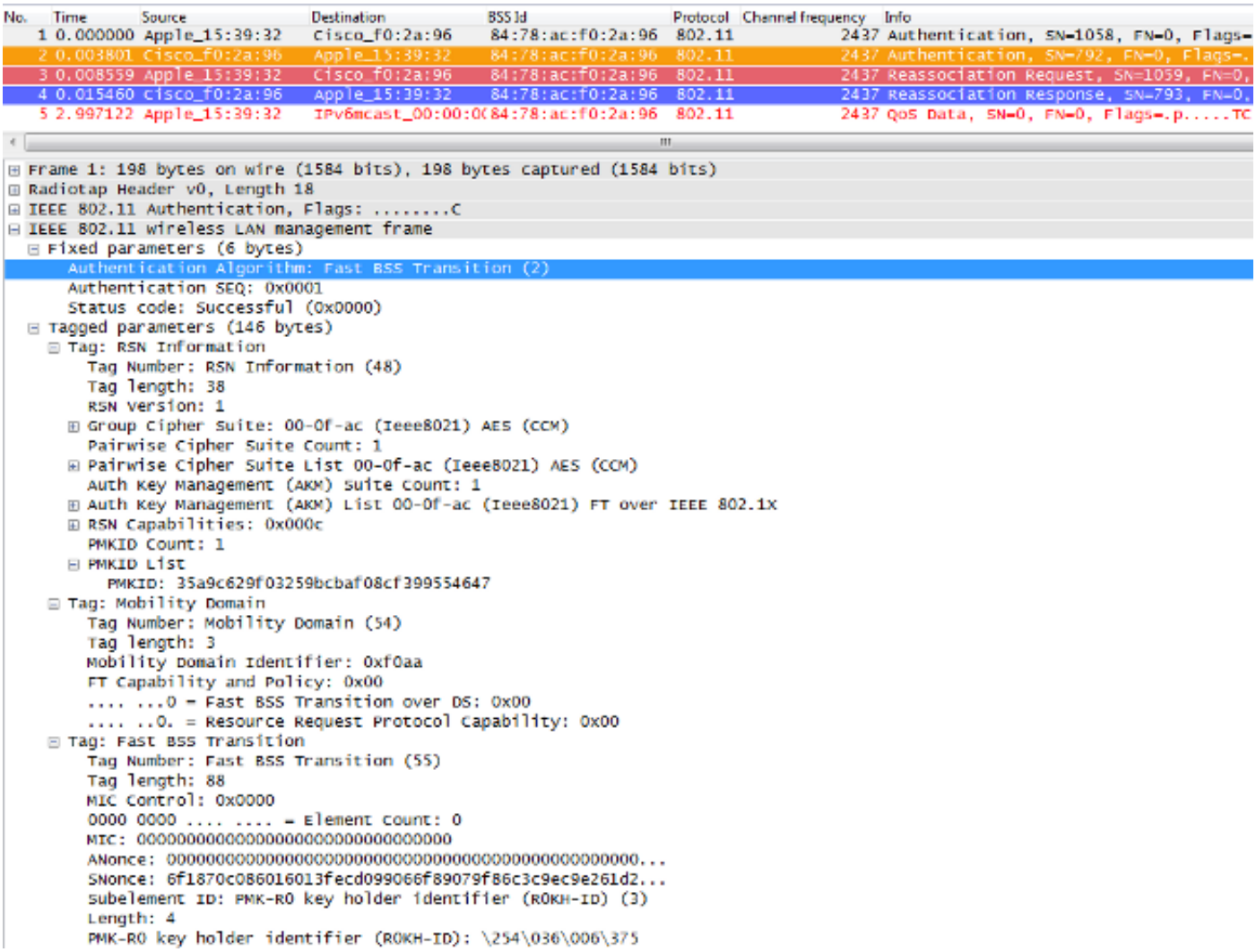

```
*apfMsConnTask_0: Jun 27 19:29:09.136: ec:85:2f:15:39:32
  Association received from mobile on BSSID
  84:78:ac:f0:68:d4

*apfMsConnTask_0: Jun 27 19:29:09.137: ec:85:2f:15:39:32
  Marking this mobile as TGr capable.

*apfMsConnTask_0: Jun 27 19:29:09.137: ec:85:2f:15:39:32
  Processing RSN IE type 48, length 20 for mobile
  ec:85:2f:15:39:32

*apfMsConnTask_0: Jun 27 19:29:09.137: Sending assoc-resp
  station:ec:85:2f:15:39:32 AP:84:78:ac:f0:68:d0-00
  thread:144be808

*apfMsConnTask_0: Jun 27 19:29:09.137: Adding MDIE,
  ID is:0xaaf0

*apfMsConnTask_0: Jun 27 19:29:09.137: ec:85:2f:15:39:32
  Including FT Mobility Domain IE (length 5) in Initial
  assoc Resp to mobile

*apfMsConnTask_0: Jun 27 19:29:09.137: ec:85:2f:15:39:32
  Sending R0KH-ID as:-84.30.6.-3

*apfMsConnTask_0: Jun 27 19:29:09.137: ec:85:2f:15:39:32
  Sending R1KH-ID as 3c:ce:73:d8:02:00

*apfMsConnTask_0: Jun 27 19:29:09.137: ec:85:2f:15:39:32
```

```
  Including FT IE (length 98) in Initial Assoc Resp to mobile

*apfMsConnTask_0: Jun 27 19:29:09.138: ec:85:2f:15:39:32
  Sending Assoc Response to station on BSSID 84:78:ac:f0:68:d4
  (status 0) ApVapId 5 Slot 0

*dot1xMsgTask: Jun 27 19:29:09.141: ec:85:2f:15:39:32
  Creating a PKC PMKID Cache entry for station
  ec:85:2f:15:39:32 (RSN 2)

*dot1xMsgTask: Jun 27 19:29:09.141: ec:85:2f:15:39:32
  Resetting MSCB PMK Cache Entry 0 for station
  ec:85:2f:15:39:32

*dot1xMsgTask: Jun 27 19:29:09.141: ec:85:2f:15:39:32
  Setting active key cache index 8 ---> 8

*dot1xMsgTask: Jun 27 19:29:09.141: ec:85:2f:15:39:32
  Setting active key cache index 8 ---> 0

*dot1xMsgTask: Jun 27 19:29:09.141: ec:85:2f:15:39:32
  Adding BSSID 84:78:ac:f0:68:d4 to PMKID cache at
  index 0 for station ec:85:2f:15:39:32

*dot1xMsgTask: Jun 27 19:29:09.142: New PMKID: (16)

*dot1xMsgTask: Jun 27 19:29:09.142:
  [0000] 17 4b 17 5c ed 5f c7 1d 66 39 e9 5d 3a 63 69 e7

*dot1xMsgTask: Jun 27 19:29:09.142: ec:85:2f:15:39:32
  Creating global PMK cache for this TGr client

*dot1xMsgTask: Jun 27 19:29:09.142: ec:85:2f:15:39:32
  Created PMK Cache Entry for TGr AKM:PSK
  ec:85:2f:15:39:32

*dot1xMsgTask: Jun 27 19:29:09.142: ec:85:2f:15:39:32
  R0KH-ID:172.30.6.253   R1KH-ID:3c:ce:73:d8:02:00
  MSK Len:48 pmkValidTime:1813

*dot1xMsgTask: Jun 27 19:29:09.142: ec:85:2f:15:39:32
  Initiating RSN PSK to mobile ec:85:2f:15:39:32

*dot1xMsgTask: Jun 27 19:29:09.142: ec:85:2f:15:39:32
  Found an cache entry for BSSID 84:78:ac:f0:68:d4 in
  PMKID cache at index 0 of station ec:85:2f:15:39:32

*dot1xMsgTask: Jun 27 19:29:09.142: Including PMKID
  in M1  (16)

*dot1xMsgTask: Jun 27 19:29:09.142:
  [0000] 17 4b 17 5c ed 5f c7 1d 66 39 e9 5d 3a 63 69 e7

*dot1xMsgTask: Jun 27 19:29:09.143: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32
  state INITPMK (message 1), replay counter
  00.00.00.00.00.00.00.00

*apfMsConnTask_0: Jun 27 19:29:09.144: ec:85:2f:15:39:32
  Got action frame from this client.

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.152: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32
```

```
*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.153: ec:85:2f:15:39:32
  Received EAPOL-key in PTK_START state (message 2) from
  mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.153: ec:85:2f:15:39:32
  Calculating PMKR0Name

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.153: Adding MDIE,
  ID is:0xaaf0

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.153: ec:85:2f:15:39:32
  Adding TIE for reassociation deadtime:20000 milliseconds

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.153: ec:85:2f:15:39:32
  Adding TIE for R0Key-Data valid time :1813

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.154: ec:85:2f:15:39:32
  Sending EAPOL-Key Message to mobile ec:85:2f:15:39:32 state
  PTKINITNEGOTIATING (message 3), replay counter
  00.00.00.00.00.00.00.01

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.163: ec:85:2f:15:39:32
  Received EAPOL-Key from mobile ec:85:2f:15:39:32

*Dot1x_NW_MsgTask_2: Jun 27 19:29:09.163: ec:85:2f:15:39:32
  Received EAPOL-key in PTKINITNEGOTIATING state (message 4)
  from mobile ec:85:2f:15:39:32
```

802.11r では、他の高速セキュア ローミング方式の場合と同様に、WLAN への初回関連付けが、この手法で使用されるベース キーの導出に使用される基盤になります。主な違いは、クライアントがローミングを開始する際に生じます。FTは、802.1X/EAPを使用しないだけでなく、初期の802.11オープンシステム認証と再関連付けフレーム（AP間のローミング時に常に使用および必要）を組み合わせて、FT情報を交換し、4方向ハンドシェイクの代わりに新しい動的暗号キーを取得する、より効率的なローミング方法を実行します。

次の図は、802.1X/EAPセキュリティを使用したFast BSS Transition Over-the-Airの実行時に交換されるフレームを示しています。FT のキー ネゴシエーションを開始するために必要な FT プロトコル情報要素を表示するため、クライアントから AP へのオープン システム認証フレームを選択しています。これは、新しい AP での新しい PTK を（PMK-R1 に基づいて）導出するために使用します。このクライアントが単なるオープン システム認証ではなく高速 BSS 移行を実行していることを示すため、認証アルゴリズムを示すフィールドを強調表示しています。

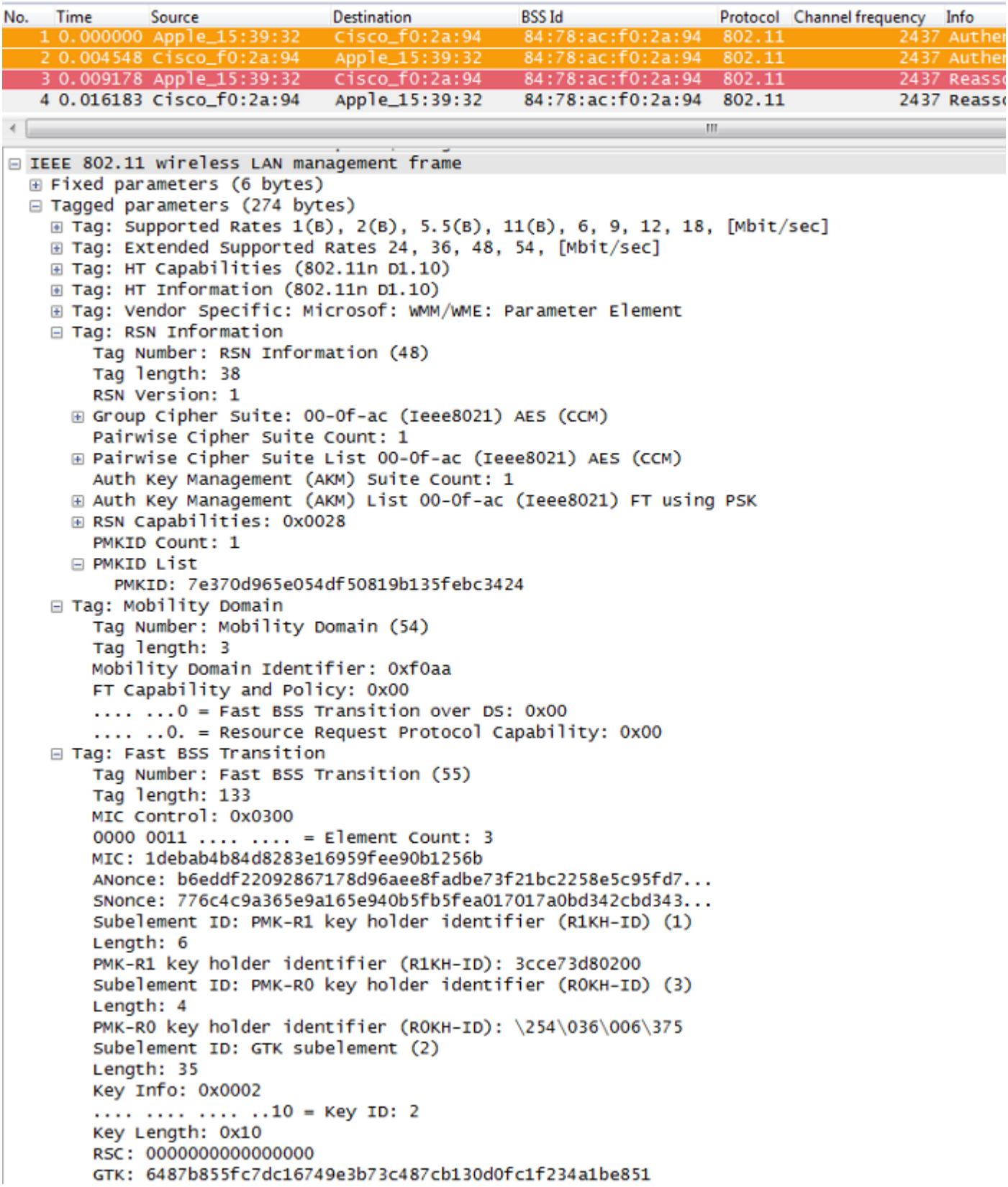


この FT ローミング イベントが 802.1X/EAP の使用時に発生したときの WLC からのデバッグ出力を次に示します。

```
*apfMsConnTask_2: Jun 27 19:25:48.751: ec:85:2f:15:39:32
  Doing preauth for this client over the Air
!--- WLC begins FT fast-secure roaming over-the-Air with
      this client and performs a type of preauthentication,
      because the client asks for this with FT on the Authentication
      frame that is sent to the new AP over-the-Air
      (before the Reassociation Request).

*apfMsConnTask_2: Jun 27 19:25:48.751: ec:85:2f:15:39:32
  Doing local roaming for destination address
  84:78:ac:f0:2a:96
!--- WLC performs the local roaming event with the new AP to
      which the client roams.

*apfMsConnTask_2: Jun 27 19:25:48.751: ec:85:2f:15:39:32
  Got 1 AKMs in RSNIE
*apfMsConnTask_2: Jun 27 19:25:48.751: ec:85:2f:15:39:32
  RSNIE AKM matches with PMK cache entry :0x3
!--- WLC receives one PMK from this client (known as AKM here),
      which matches the PMK cache entry hold for this client.

*apfMsConnTask_2: Jun 27 19:25:48.751: ec:85:2f:15:39:32
  Created a new preauth entry for AP:84:78:ac:f0:2a:96
*apfMsConnTask_2: Jun 27 19:25:48.751: Adding MDIE,
  ID is:0xaaf0
```

```
!--- WLC creates a new preauth entry for this AP-and-Client pair,
      and adds the MDIE information.

*apfMsConnTask_2: Jun 27 19:25:48.763: Processing assoc-req
  station:ec:85:2f:15:39:32 AP:84:78:ac:f0:2a:90-00
  thread:144bef38
*apfMsConnTask_2: Jun 27 19:25:48.763: ec:85:2f:15:39:32
  Reassociation received from mobile on BSSID
  84:78:ac:f0:2a:96
!--- Once the client receives the Authentication frame reply from the
      WLC/AP, the Reassociation request is sent, which is received at
      the new AP to which the client roams.

*apfMsConnTask_2: Jun 27 19:25:48.764: ec:85:2f:15:39:32
  Marking this mobile as TGr capable.

*apfMsConnTask_2: Jun 27 19:25:48.764: ec:85:2f:15:39:32
  Processing RSN IE type 48, length 38 for mobile
  ec:85:2f:15:39:32

*apfMsConnTask_2: Jun 27 19:25:48.765: ec:85:2f:15:39:32
  Roaming succeed for this client.
!--- WLC confirms that the FT fast-secure roaming is successful
      for this client.

*apfMsConnTask_2: Jun 27 19:25:48.765: Sending assoc-resp
  station:ec:85:2f:15:39:32 AP:84:78:ac:f0:2a:90-00
  thread:144bef38
*apfMsConnTask_2: Jun 27 19:25:48.766: Adding MDIE,
  ID is:0xaaf0
*apfMsConnTask_2: Jun 27 19:25:48.766: ec:85:2f:15:39:32
  Including FT Mobility Domain IE (length 5) in
  reassociation assoc Resp to mobile
*apfMsConnTask_2: Jun 27 19:25:48.766: ec:85:2f:15:39:32
  Sending Assoc Response to station on BSSID 84:78:ac:f0:2a:96
  (status 0) ApVapId 7 Slot 0
!--- The Reassociation response is sent to the client, which
      includes the FT Mobility Domain IE.

*dot1xMsgTask: Jun 27 19:25:48.769: ec:85:2f:15:39:32
  Finishing FT roaming for mobile ec:85:2f:15:39:32
!--- FT roaming finishes and EAP is skipped (as well as any
      other key management handshake), so the client is ready
      to pass encrypted data frames with the current AP.

*dot1xMsgTask: Jun 27 19:25:48.769: ec:85:2f:15:39:32
  Skipping EAP-Success to mobile ec:85:2f:15:39:32
```

次の図は、WPA2-PSKセキュリティを使用したFast BSS Transition Over-the-Airを示しています。このFT交換の詳細を表示するために、APからクライアントへの最終的な再関連付け応答(RR)フレームが選択されています。



この FT のローミング イベントが PSK の使用時に発生したときのデバッグ出力を次に示します。これは、802.1X/EAP を使用した場合のデバッグ出力と同様です。

```
*apfMsConnTask_2: Jun 27 19:29:29.854: ec:85:2f:15:39:32
  Doing preauth for this client over the Air

*apfMsConnTask_2: Jun 27 19:29:29.854: ec:85:2f:15:39:32
  Doing local roaming for destination address
```

```
  84:78:ac:f0:2a:94

*apfMsConnTask_2: Jun 27 19:29:29.854: ec:85:2f:15:39:32
  Got 1 AKMs in RSNIE

*apfMsConnTask_2: Jun 27 19:29:29.854: ec:85:2f:15:39:32
  RSNIE AKM matches with PMK cache entry :0x4

*apfMsConnTask_2: Jun 27 19:29:29.854: ec:85:2f:15:39:32
  Created a new preauth entry for AP:84:78:ac:f0:2a:94

*apfMsConnTask_2: Jun 27 19:29:29.854: Adding MDIE,
  ID is:0xaaf0

*apfMsConnTask_2: Jun 27 19:29:29.867: Processing assoc-req
  station:ec:85:2f:15:39:32 AP:84:78:ac:f0:2a:90-00
  thread:144bef38

*apfMsConnTask_2: Jun 27 19:29:29.867: ec:85:2f:15:39:32
  Reassociation received from mobile on BSSID
  84:78:ac:f0:2a:94

*apfMsConnTask_2: Jun 27 19:29:29.868: ec:85:2f:15:39:32
  Marking this mobile as TGr capable.

*apfMsConnTask_2: Jun 27 19:29:29.868: ec:85:2f:15:39:32
  Processing RSN IE type 48, length 38 for mobile
  ec:85:2f:15:39:32

*apfMsConnTask_2: Jun 27 19:29:29.869: ec:85:2f:15:39:32
  Roaming succeed for this client.

*apfMsConnTask_2: Jun 27 19:29:29.869: Sending assoc-resp
  station:ec:85:2f:15:39:32 AP:84:78:ac:f0:2a:90-00
  thread:144bef38

*apfMsConnTask_2: Jun 27 19:29:29.869: Adding MDIE,
  ID is:0xaaf0

*apfMsConnTask_2: Jun 27 19:29:29.869: ec:85:2f:15:39:32
  Including FT Mobility Domain IE (length 5) in
  reassociation assoc Resp to mobile

*apfMsConnTask_2: Jun 27 19:29:29.870: ec:85:2f:15:39:32
  Sending Assoc Response to station on BSSID
  84:78:ac:f0:2a:94 (status 0) ApVapId 5 Slot 0

*dot1xMsgTask: Jun 27 19:29:29.874: ec:85:2f:15:39:32
  Finishing FT roaming for mobile ec:85:2f:15:39:32
```

図に示すように、WLANへの初期関連付け時にFast BSS移行がネゴシエートされると、ローミングに使用および必要な4つのフレーム（クライアントからのオープンシステム認証、APからのオープンシステム認証、再関連付け要求、および再関連付け応答）が、新しいPTK（ユニキャスト暗号化キー）およびGTK（マルチキャスト/ブロードキャスト暗号化キー）を取得するためのFT 4方向ハンドシェイクとして基本的に使用されます。

これは、通常はこれらのフレームの交換後に発生する 4 方向ハンドシェイクの代わりになります。これらのフレームでの FT の内容およびキー ネゴシエーションは、セキュリティ方式として 802.1X/EAP または PSK のいずれを使用する場合でも、基本的に同じです。図に示すように、AKMフィールドが主な違いで、クライアントがPSKまたは802.1XでFTを実行するかどうかを確認します。したがって、キー ネゴシエーションのためのこのタイプのセキュリティ情報は、通常これら 4 つのフレームに含まれていないことに注意する必要があります。この情報は、802.11r が

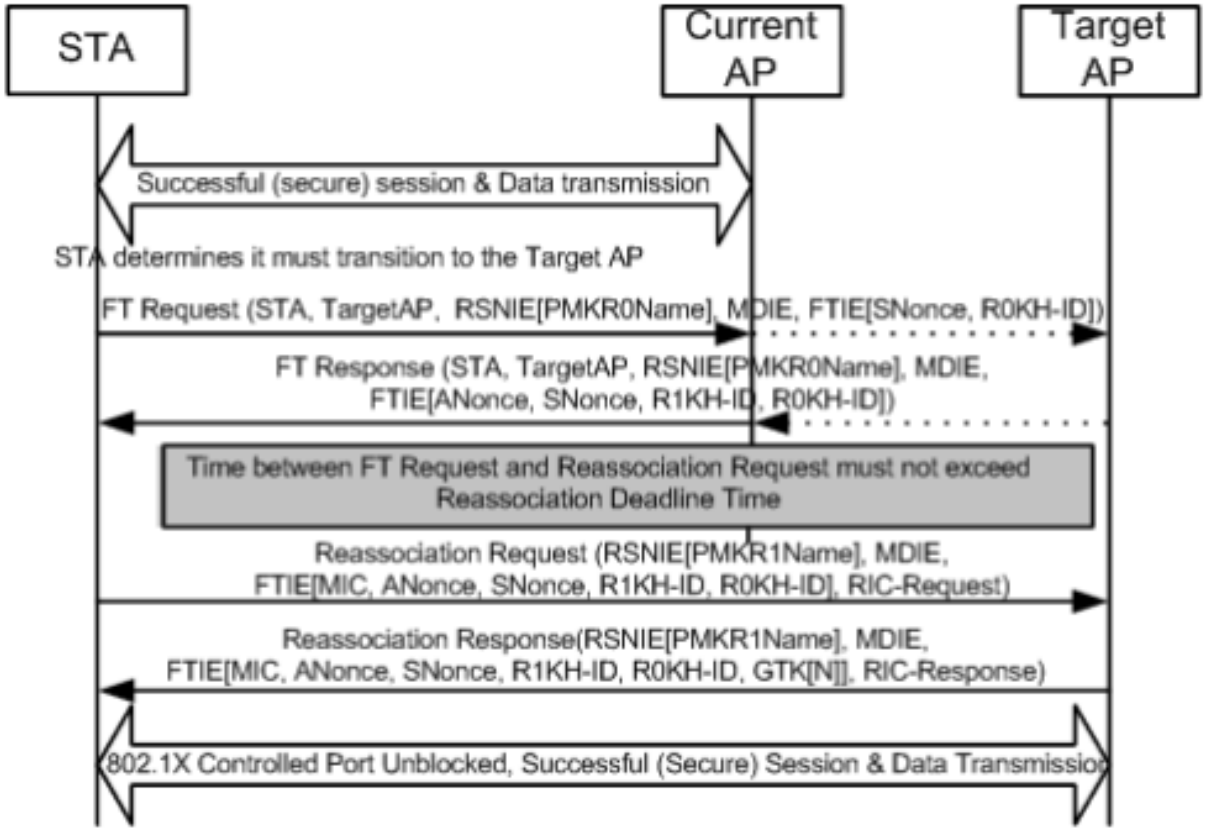
実行されている場合にクライアントが FT ローミングを行うときのみ含まれ、初回関連付けの後にクライアントと WLAN インフラストラクチャの間でネゴシエートされます。

Over-the-DS での高速 BSS 移行

802.11r では、高速 BSS 移行の別の実装も可能です。クライアントによる 新しい AP への FT ローミングが、Over-the-Air ではなく Over-the-DS（分散システム）で行われる実装です。この場合、キー ネゴシエーションの開始には、オープン システム認証フレームではなく FT アクション フレームが使用されます。

基本的に、クライアントは、より適切なAPにローミングできると判断すると、ローミング前に現在接続している元のAPにFTアクション要求フレームを送信します。クライアントは、FT ローミング先となるターゲット AP の BSSID（MAC アドレス）を示します。元の AP は、この FT アクション要求フレームを分散システム（通常は有線インフラストラクチャ）を介してターゲット AP に転送し、ターゲット AP は、FT アクション応答フレームでクライアントに応答します（これも DS を介するため、最終的には Over-the-Air でクライアントに送信することができます）。このFTアクションフレーム交換が成功すると、クライアントはFTローミングを終了します。クライアントは再関連付け要求をターゲットAPに送信し（今回は地上波）、ローミングと最終的なキーの導出を確認するために新しいAPから再関連付け応答を受信します。

つまり、高速 BSS 移行のネゴシエートおよび新しい暗号キーの導出を行うためには 4 つのフレームがありますが、ここでは、オープン システム認証フレームが FT アクション要求/応答フレームに置き換えられており、これらが、現在の AP とターゲット AP の間で分散システムを介して交換されます。この方式は、802.1X/EAPとPSKの両方のセキュリティ方式にも有効で、すべてCisco Wireless LAN Controllerでサポートされています。ただし、このOver-the-DS移行はWiFi業界のほとんどのワイヤレスクライアントでサポートおよび実装されていないため（フレーム交換とデバッグ出力は基本的に同じであるため）、このドキュメントでは例を示しません。その代わりに、Over-the-DS での 高速 BSS 移行を視覚化する次の図を示します。



FlexConnect での 802.11r の使用

- 中央認証がサポートされています。これには、ローカルおよび中央のデータスイッチングが含まれます。APは同じFlexConnectグループに属している必要があります。
- ローカル認証はサポートされていません。
- スタンドアロンモードはサポートされていません。

802.11r の長所

- この方式は、IEEE が 802.11 標準規格修正版（802.11r）で明確に定義するキー階層を使用する最初の方式です。そのため、これらの FT 手法の実装は、ベンダー間の互換性が高く、解釈の齟齬もありません。
- 802.11r により、必要に応じて有用な複数の手法が可能になります（Over-the-Air と Over-the-DS、802.1x/EAP セキュリティと PSK セキュリティ）。
- ワイヤレス クライアントは、新しい AP への高速セキュア ローミングを、その AP との関連付けを行ったことがなくても、同じ WLAN/SSID で、複数の PMKID を保存する必要なく実行します。
- この方式は、PSK セキュリティが使用される場合であっても高速なローミングが可能な最初の高速セキュア ローミング方式であり、WPA/WPA2 PSK の使用時に AP 間でローミングするときに必要な 4 方向ハンドシェイクが不要です。高速セキュアローミング方式の主な目的は、このセキュリティ方式を実装する際に802.1X/EAPハンドシェイクを回避することですが、PSKセキュリティでは、4ウェイハンドシェイクを回避する802.11rを使用すると、ローミングイベントがさらに高速化されます。

802.11r の短所

- 高速 BSS 移行を実際にサポートするワイヤレス クライアント デバイスは少数であり、ほとんどの場合、これらのデバイスは、802.11r で使用可能な手法をすべてサポートしているわけではありません。
- これらの実装は非常に若いため、発生する可能性のある警告に対処するには、実稼働環境からのテスト結果や十分なデバッグ結果が得られません。
- WLAN/SSID をいずれかの FT 方式を使用するために設定する場合、その WLAN/SSID に接続できるのは、802.11r をサポートするワイヤレス クライアントのみです。FT の設定はクライアントが選択するものではないため、802.11r をサポートしないワイヤレス クライアントは、FT がまったく設定されていない別の WLAN/SSID と接続する必要があります。

適応型802.11r

- 一部のレガシークライアントは、「混合モード」でも802.11rが有効になっているWLAN/SSIDと関連付けることができません（802.11rをサポートし、サポートしていない同じSSIDクライアントに関連付けることができることを希望します）。これは、Robust Security Network Information Element(RSN IE)の解析を担当するクライアントサプリカントのドライバが古く、IE内の追加のAKMスイートを認識していない場合です。この制限により、クライアントは802.11rサポートをアドバタイズするWLANにアソシエーション要求を送信できません。そのため、802.11rクライアント用に1つのWLAN/SSIDを設定し、802.11rをサポートしないクライアント用に別のWLAN/SSIDを設定する必要があります。
- この問題を解決するために、CiscoワイヤレスLANインフラストラクチャではAdaptive

802.11r機能が導入されました。WLANレベルでFTモードがAdaptiveに設定されている場合、WLANは802.11i対応WLANで802.11rモビリティドメインIDをアドバタイズします。一部のApple iOS10クライアントデバイスは、80211i/WPA2 WLAN上のMDIEの存在を識別し、802.11rアソシエーションを確立するために独自のハンドシェイクを実行します。クライアントが正常な802.11rアソシエーションを完了すると、通常の802.11r対応WLANと同様にFTローミングを実行できます。FTアダプティブは、選択したApple iOS10（およびそれ以降）のデバイスにのみ適用できます。他のすべてのクライアントはWLAN上で引き続き802.11i/WPA2アソシエーションを持つことができ、サポートされている適切なFSR方式を実行します。

- 802.11rが実際には有効になっていない（他の非802.11rクライアントが正常に接続できるように）WLAN/SSIDで802.11rを実行するためにiOS10デバイスに導入されたこの新機能についての詳細なドキュメントは、『CiscoワイヤレスLAN上のCisco IOSデバイスに関するエンタープライズベストプラクティス』を参照してください。

まとめ

- 特定の AP にローミングすることを決定するのは常にクライアントであり、WLC/AP がクライアントに代わって決定することはできないことに注意してください。ローミングイベントは、ローミングが必要であると判断された時点で、ワイヤレスクライアントによって開始されます。
- WLC は、同じ WLAN/SSID で、ほとんどまたはすべての FSR（高速セキュア ローミング）方式の組み合わせを一括してサポートしています。ただし、これは通常機能しないことに注意してください。サポートの可否や、サポート対象であると WLC がアドバタイズしようとしているものを理解することは、クライアントの動作（モバイル デバイス間で大きく異なる）に多大に依存するためです。1 つだけの SSID で相互運用性を達成することは、通常、それで解決される以上の問題があるため、推奨されません。これが本当に必要な場合は、このWLANで使用可能なすべてのクライアントを使用した詳細なテストを完了する必要があります。
- 高速セキュア ローミング方式を開発する目的は、WLAN/SSID にセキュリティを有効にした状態で AP 間を移動する際に WLAN ローミング プロセスを高速化することであることを理解することが非常に重要です。セキュリティが実行されていない場合、クライアントと AP がデータ フレームの送信前に行うことは、AP 間のローミング時に常に必要であるワイヤレス管理フレーム（クライアントからのオープン システム認証、AP からのオープン システム認証、再関連付け要求、および再関連付け応答）の交換だけなので、高速化の対象がありません。したがって、これ以上高速化することはできません。セキュリティがない状態でローミングの問題が発生した場合、高速ローミング方式でローミングを改善することはできません。ワイヤレス クライアント ステーションが AP カバレッジ セルの間で適宜ローミングできるように WLAN/SSID が適切にセットアップおよび設計されているか確認することしか方法はありません。
- 802.11r/FTは、802.11rセクションで説明されているように、このセキュリティでローミングイベントを高速化し、4方向ハンドシェイクを回避するために、WPA2-PSKで実装されます。
- すべての方式にはそれぞれ長所と短所がありますが、最終的には、導入する方式がワイヤレス クライアント ステーションでサポートされているかどうか、使用可能なすべての方式が Cisco WLAN インフラストラクチャでサポートされているかどうかを必ず確認する必要があります。そのため、特定の WLAN/SSID に接続するワイヤレス クライアントによって実際にサポートされている最適な方式を選択する必要があります。たとえば、一部の導入では、

CiscoワイヤレスIP Phone用にCCKMを使用してWLAN/SSIDを作成し（CCKMを使用してWPA2/AESをサポートするが、802.11rはサポートしない）、この高速セキュアローミング方式をサポートするワイヤレスクライアント用にWPA2/AESを使用して別のWLAN/SSIDを作成できます（サポートされている場合ははOKCを使用）。

- ワイヤレスクライアントが利用可能な高速セキュアローミング方式をサポートしていない場合は、802.1X/EAPセキュリティを使用したWLAN/SSID上のAP間のローミング（クライアントアプリケーションやサービスの中断を引き起こす可能性がある）時に、これらのクライアントがこのドキュメントで説明されている遅延を常に試すことができるという事実を受け入れる必要があります。
- SKC（WPA2 PMKID キャッシング）を除くすべての方式は、AP が同じモビリティ グループにある限り、さまざまな WLC によって管理される AP 間の高速セキュア ローミング（コントローラ間ローミング）でサポートされています。
- CUWNは、802.1X/EAP認証をWPA/WPA2に使用する場合、この記事で説明する各種の高速セキュアローミング方式をすべて完全にサポートします。CUWNは、高速ローミング方式がほとんど必要ないPSK(WPA2-Personal)が使用されている場合に、WPA2-RSNで動作する方式(CCKM、PMKID Caching/SKC、OKC/PKC)で高速セキュアローミングをサポートしません。ただし、CUWNでは、この記事でも説明されているように、PSKを使用するWPA2-FT(802.11r)の場合に高速セキュアローミング(FSR)をサポートしています。

関連情報

- 802.11r BSS高速移行導入ガイド
- シスコテクニカルサポートおよびダウンロード

翻訳について
シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。