Understand Opportunistic Wireless Encryption Flow

Contents

Introduction

This document describes OWE transition flow and how it works on the Catalyst 9800 Wireless LAN Controller (WLC).

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- How to configure the 9800 WLC, the Access Point (AP) for basic operation
- How to configure WLAN and Policy Profiles.

Components Used

The information in this document is based on these software and hardware versions:

- C9800-80, Cisco IOS® XE 17.12.4 and also tested in Cisco IOS® XE 17.9.6
- AP model : C9136I, checked in both local and flex connect mode.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Description

- OWE (Opportunistic Wireless Encryption) is an extension to IEEE 802.11 which provides encryption for the wireless medium. The purpose of OWE based authentication is to avoid open unsecured wireless connectivity between the AP's and clients.
- The OWE uses the Diffie-Hellman algorithms-based Cryptography to setup the wireless encryption.
- With OWE, the client and AP perform a Diffie-Hellman key exchange during the access procedure and use the resulting pairwise secret with the 4-way handshake.
- The use of OWE enhances wireless network security for deployments where Open or shared PSK based networks are deployed.

Steps

1. Configure one OPEN WLAN without any encryption/security and enable broadcasting.
2. Configure another SSID with OWE security settings and map the OPEN WLAN ID number in transition-mode-wlan-id. Disable broadcast SSID option in this OWE transition SSID.
3. Map the OWE transition WLAN ID number in OPEN WLAN "transition-mode-wlan-id" field.

Details of Lab Repro

- Open SSID Name: OPEN-OWE
- OWE Transition SSID Name: OWE-Transition
- BSSID of OPEN-OWE: 40:ce:24:dd:2e:87
- BSSID of OWE-Transition: 40:ce:24:dd:2e:8f

OWE FLOW

1. Beacons can be broadcast for OPEN SSID. You can see it by its SSID name in AIR PCAP.
2. We can also see the hidden security enabled SSID with the name "Wildcard" instead of its own SSID name in AIR PCAP.
3. Once the clients receive the beacon frame for OPEN SSID, if it has or supports OWE, then it can start sending probe request to OWE transition SSID (which is that security enabled SSID instead of OPEN SSID).

4. OWE supported clients can get probe response from transition SSID.
5. OPEN authentication can happen between client and AP.
6. Client can send association request to the AP with DH key exchange details and use the resulting pairwise secret for 4-way handshake.
7. AP can send association response.
8. Four-Way handshake can happen between AP and client device.
9. After successful key management, L2 PSK can be successful.
10. Client can get IP from DHCP,ARP etc,.
11. Client can go to RUN state.
12. If client devices which are not supporting OWE, then it can send probe request to OPEN SSID itself and it can directly get IP than it can go to RUN state.

Original Beacon frame

- Here, AIR PCAP shows that, the SSID "OPEN-OWE" broadcasting (Beacon Frame). Which contains transition SSID details and it called "OWE-Transition".

Image-1 : Beacon Frame of OPEN SSID

Hidden SSID Beacons

- As per WLAN configuration, "broadcasting" is disabled for this "OWE-Transition" SSID, however, you can see hidden SSID beacons in AIR PCAP which contain the SSID Name "Wildcard". However, if you check that packet, it contains OWE-Transition details.
- Get the BSSID of hidden SSID by using the this packet, such as "40:ce:24:dd:2e:8f" and search it in packet capture.
- In this packet, it shows that, SSID "Missing" and it contains its transition SSID as "OPEN-OWE" and its BSSID "40:ce:24:dd:2e:87".

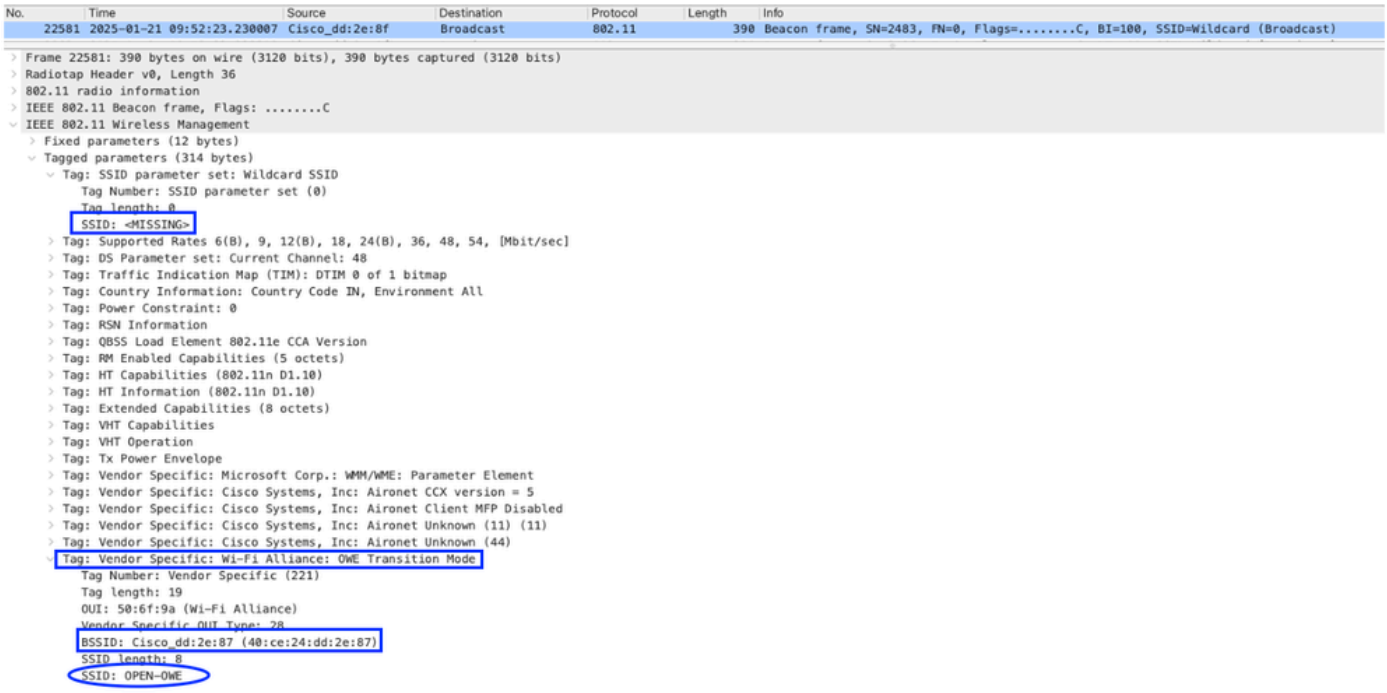

Image-2 : Hidden SSID - OWE Transition

Probe Request sent from client to OWE Transition SSID

- Based on the beacon frame "OPEN-OWE" SSID, the client comes to know the other SSID details which it needs to connect, in this scenario, it is "OWE-Transition". If client is able to support OWE encryption, then it can send the probe request to "OWE-Transition" SSID and get a response.
- Probe request sent to OWE-Transition BSSID "40:ce:24:dd:2e:8f" and got a response. Inside this probe response packet also, you can see OPEN-OWE SSID details.

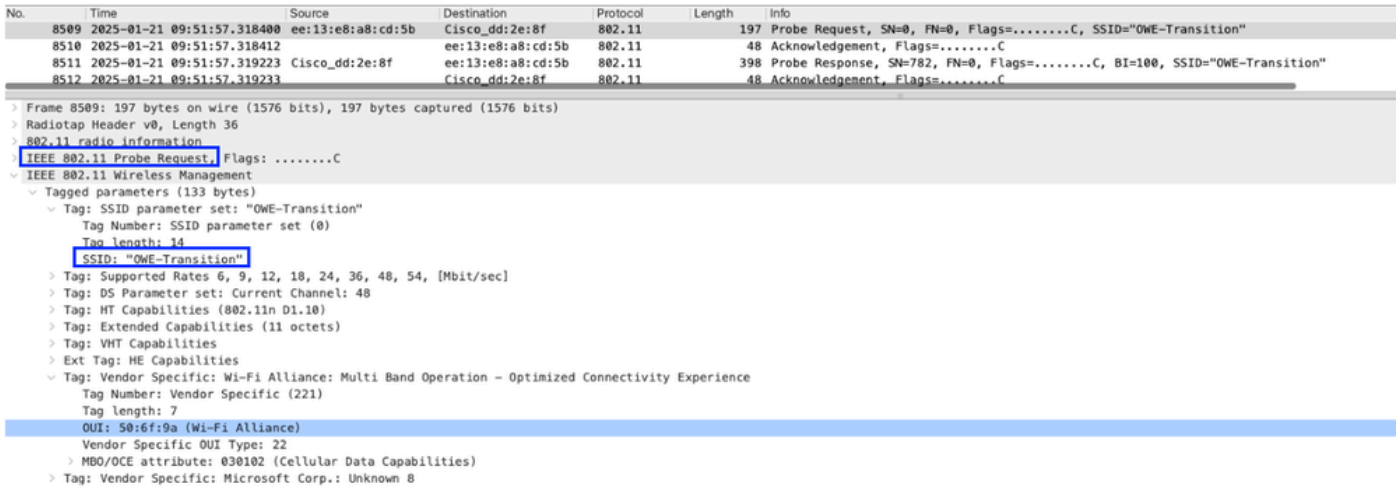

Image-3 : Probe Request

Probe Response sent from AP to Client

- Client received probe response for the SSID “OWE-Transition” however it has its original SSID details “OPEN-OWE” in WiFi Alliance.

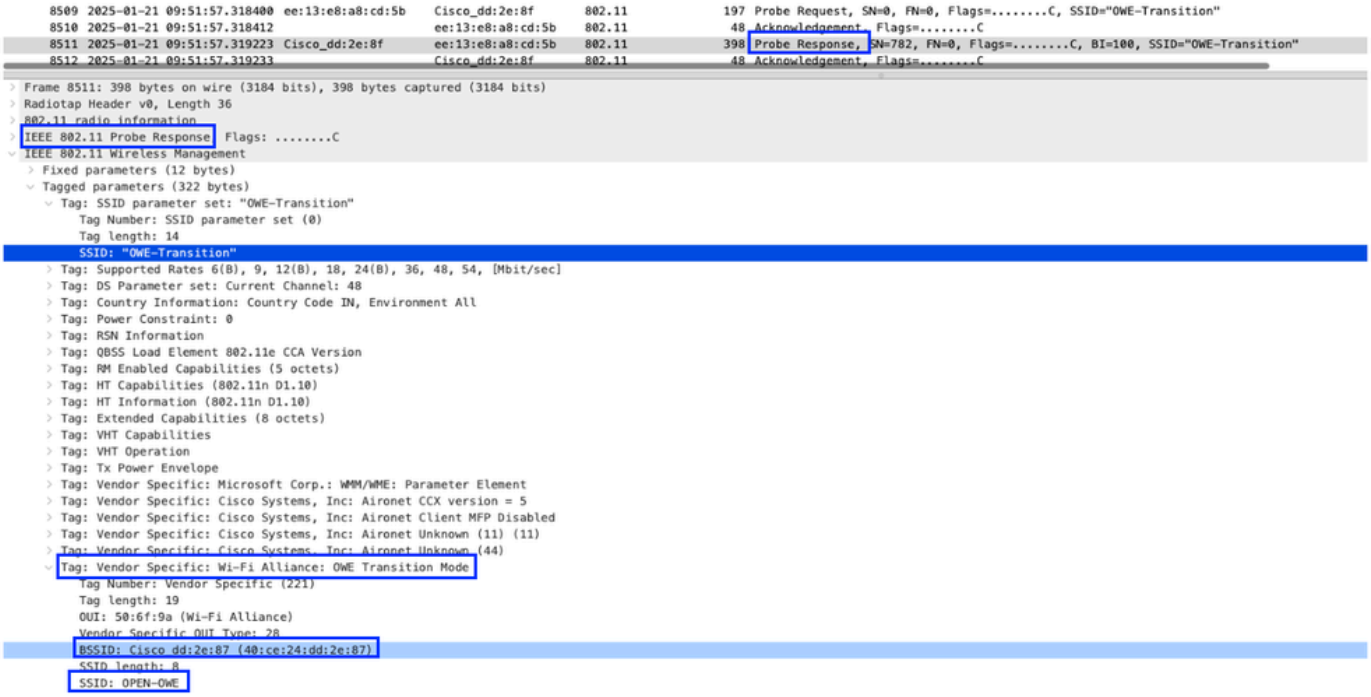

Image-4 : Probe Response

OPEN authentication

- After getting probe response, OPEN authentication can happen between Client and AP to check the clients wifi details/capabilities, before association.

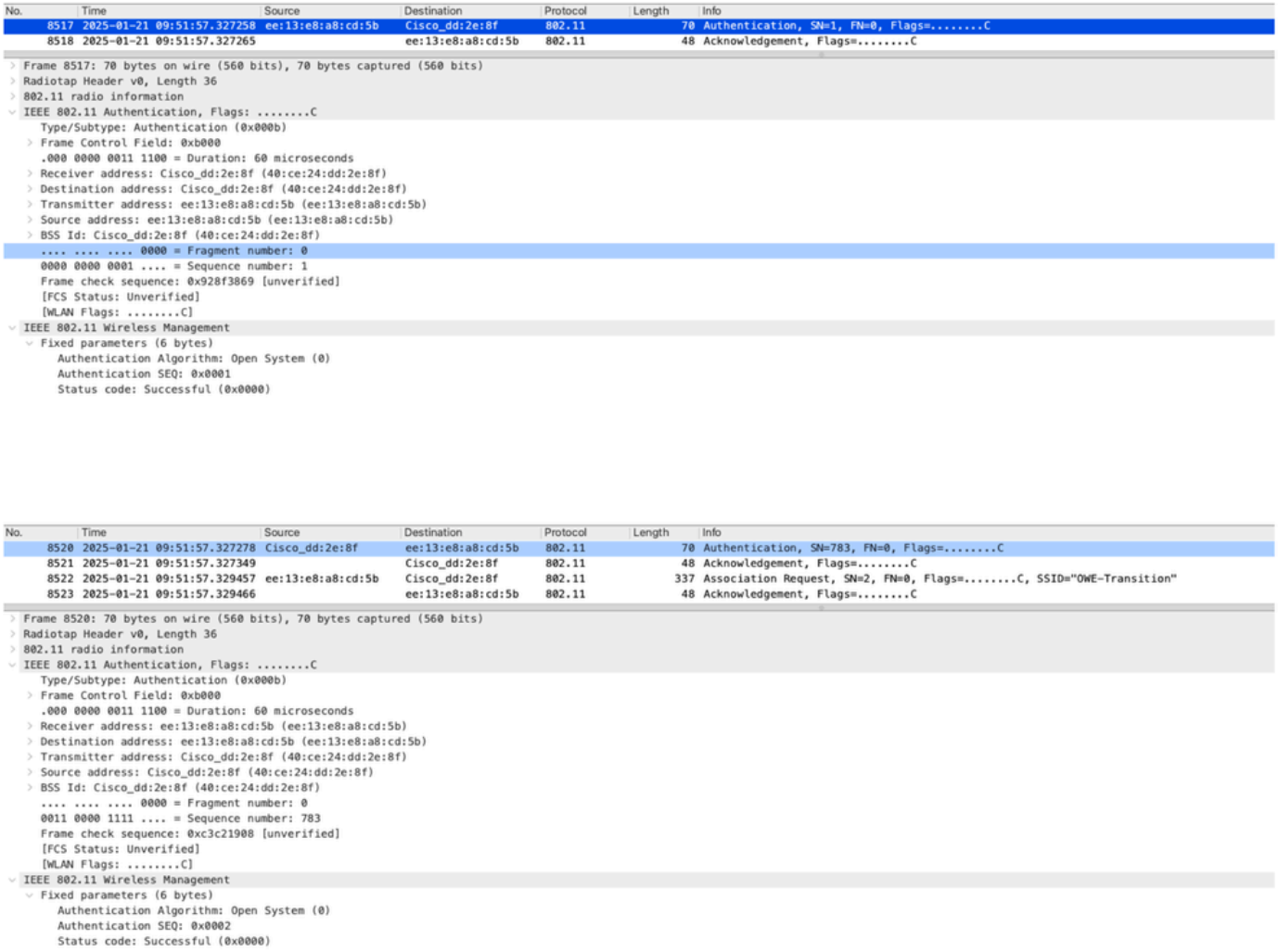

Image-5 : OPEN Authentication after successful Probe

Association Request from client to AP

- In this packet, notice the client can be attaching Diffie-Hellman parameter value for encryption.

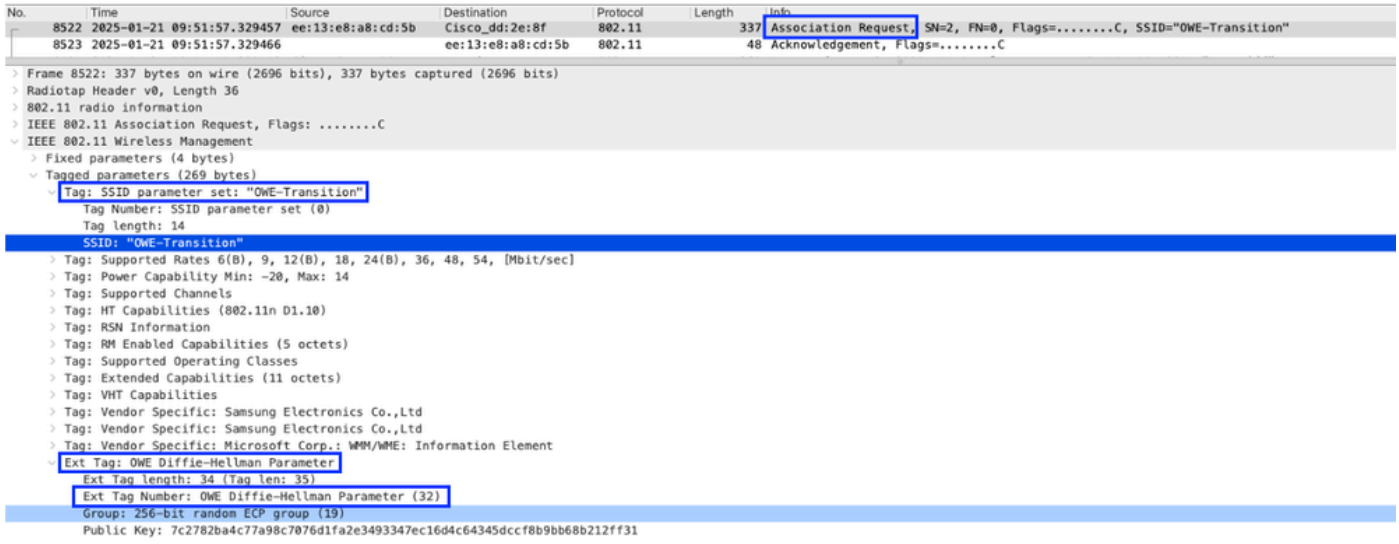

Image-6 : Association Request

- In RA trace, you can start seeing the client logs from Association phase,

```
2025/01/21 15:21:57.391071821 {wncd_x_R0-0}{1}: [client-orch-sm] [21675]: (note): MAC: ee13.e8a8.cd5b  A
2025/01/21 15:21:57.391117645 {wncd_x_R0-0}{1}: [client-orch-sm] [21675]: (debug): MAC: ee13.e8a8.cd5b
```

Association Response sent from AP to Client

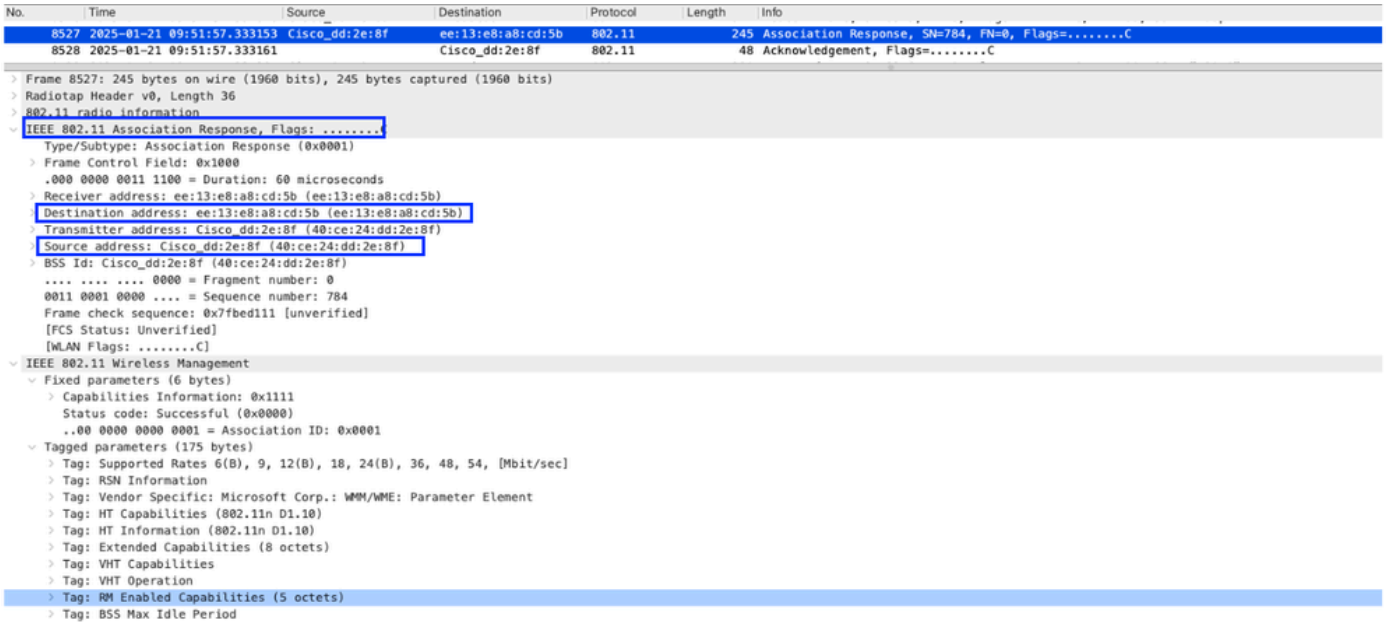

Image-7 : Association Response

```
2025/01/21 15:21:57.391334260 {wncd_x_R0-0}{1}: [client-orch-state] [21675]: (note): MAC: ee13.e8a8.cd5
2025/01/21 15:21:57.392296819 {wncd_x_R0-0}{1}: [dot11] [21675]: (note): MAC: ee13.e8a8.cd5b  Associati
```

Key Exchange

4-way handshake can happen between AP and client device.

Key-1 send by AP

Key-2 send by client

Key-3 send by AP

Key-4 send by client

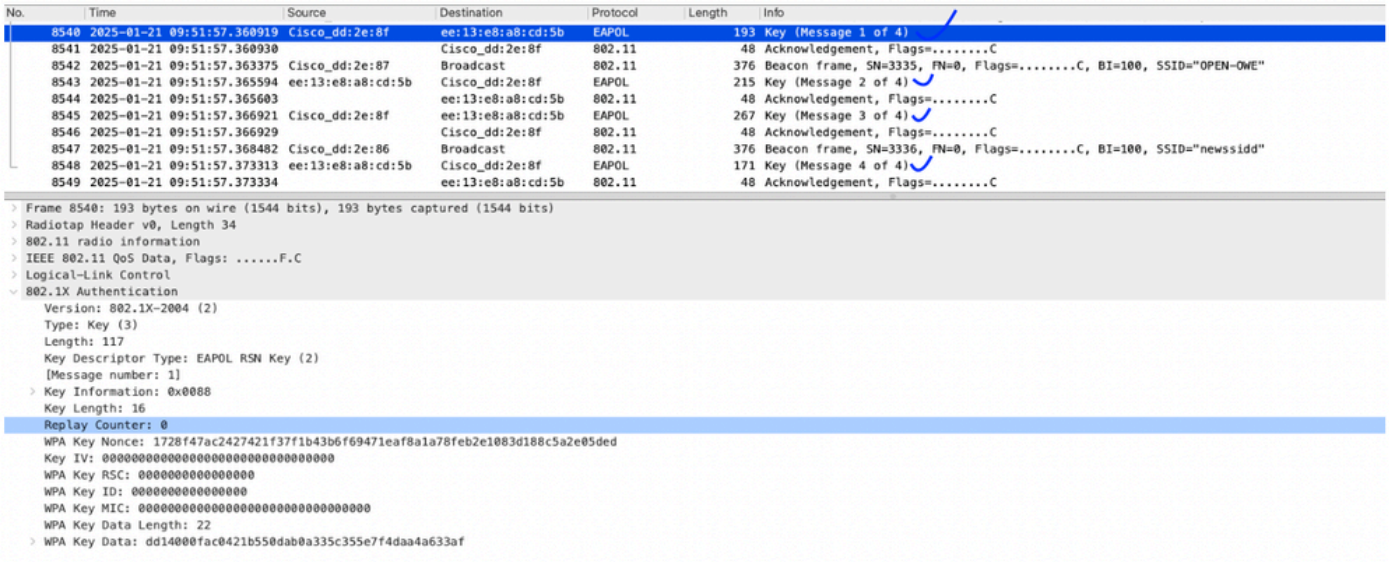

Image-8 : 4-Way Handshake

```
2025/01/21 15:21:57.392538716 {wncd_x_R0-0}{1}: [client-orch-sm] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.392557538 {wncd_x_R0-0}{1}: [client-orch-state] [21675]: (note): MAC: ee13.e8a8.cd5
2025/01/21 15:21:57.392640494 {wncd_x_R0-0}{1}: [client-auth] [21675]: (note): MAC: ee13.e8a8.cd5b  L2
2025/01/21 15:21:57.394830551 {wncd_x_R0-0}{1}: [client-auth] [21675]: (info): MAC: ee13.e8a8.cd5b  Cli
2025/01/21 15:21:57.395171903 {wncd_x_R0-0}{1}: [client-auth] [21675]: (info): MAC: ee13.e8a8.cd5b  Cli
2025/01/21 15:21:57.420590731 {wncd_x_R0-0}{1}: [client-auth] [21675]: (info): MAC: ee13.e8a8.cd5b  Cli

2025/01/21 15:21:57.420706435 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (info): MAC: ee13.e8a8.cd5b

2025/01/21 15:21:57.420775720 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (info): MAC: ee13.e8a8.cd5b

2025/01/21 15:21:57.426548998 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (info): MAC: ee13.e8a8.cd5b

2025/01/21 15:21:57.426725965 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (info): MAC: ee13.e8a8.cd5b

2025/01/21 15:21:57.426727805 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (info): MAC: ee13.e8a8.cd5b

2025/01/21 15:21:57.434078994 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (info): MAC: ee13.e8a8.cd5b

2025/01/21 15:21:57.434099154 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (note): MAC: ee13.e8a8.cd5b
```

L2 Authentication Successful

```
2025/01/21 15:21:57.434111288 {wncd_x_R0-0}{1}: [client-keymgmt] [21675]: (info): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.434250308 {wncd_x_R0-0}{1}: [client-auth] [21675]: (note): MAC: ee13.e8a8.cd5b  L2
2025/01/21 15:21:57.434286035 {wncd_x_R0-0}{1}: [client-auth] [21675]: (info): MAC: ee13.e8a8.cd5b  Cli
2025/01/21 15:21:57.434308953 {wncd_x_R0-0}{1}: [client-orch-sm] [21675]: (debug): MAC: ee13.e8a8.cd5b
```

IP Learning State

```
2025/01/21 15:21:57.434789679 {wncd_x_R0-0}{1}: [client-orch-sm] [21675]: (note): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.436611026 {wncd_x_R0-0}{1}: [client-orch-state] [21675]: (note): MAC: ee13.e8a8.cd5
2025/01/21 15:21:57.437239513 {wncd_x_R0-0}{1}: [client-orch-state] [21675]: (note): MAC: ee13.e8a8.cd5
2025/01/21 15:21:57.437508189 {wncd_x_R0-0}{1}: [client-iplearn] [21675]: (info): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.534166453 {wncd_x_R0-0}{1}: [sisf-packet] [21675]: (info): TX: DHCPv4 from interfac
2025/01/21 15:21:57.535325325 {wncd_x_R0-0}{1}: [client-iplearn] [21675]: (note): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.535874658 {wncd_x_R0-0}{1}: [sisf-packet] [21675]: (info): TX: DHCPv4 from interfac
2025/01/21 15:21:57.536500021 {wncd_x_R0-0}{1}: [client-orch-sm] [21675]: (debug): MAC: ee13.e8a8.cd5b
```

Client in RUN state

```
2025/01/21 15:21:57.537017277 {wncd_x_R0-0}{1}: [client-orch-state] [21675]: (note): MAC: ee13.e8a8.cd5
```

Clients which are not supported for OWE encryption

- By reviewing a beacon frame itself, the clients come to know, whether they are able to support this encryption method or not. If it is not supported, then it can just send a probe request to open SSID “OPEN-OWE” and can do a normal open authentication, get IP address, then it can go the RUN state.

```
2025/01/16 15:36:06.178370757 {wncd_x_R0-2}{1}: [client-orch-sm] [17332]: (note): MAC: d037.4587.8f35
2025/01/16 15:36:06.209288788 {wncd_x_R0-2}{1}: [dot11] [17332]: (note): MAC: d037.4587.8f35  Associati
2025/01/16 15:36:06.248651191 {wncd_x_R0-2}{1}: [client-auth] [17332]: (note): MAC: d037.4587.8f35  Ope
2025/01/16 15:36:06.248751507 {wncd_x_R0-2}{1}: [client-orch-state] [17332]: (note): MAC: d037.4587.8f3
2025/01/16 15:36:06.281808554 {wncd_x_R0-2}{1}: [client-orch-state] [17332]: (note): MAC: d037.4587.8f3
2025/01/16 15:36:06.303307756 {wncd_x_R0-2}{1}: [client-orch-state] [17332]: (note): MAC: d037.4587.8f3
2025/01/16 15:36:10.305041414 {wncd_x_R0-2}{1}: [client-iplearn] [17332]: (note): MAC: d037.4587.8f35
2025/01/16 15:36:10.305777492 {wncd_x_R0-2}{1}: [client-orch-state] [17332]: (note): MAC: d037.4587.8f3
```

Fast Transition Information

- We are able to configure OWE only in OPEN authentication or in Webauth (CWA/LWA/EWA).
- FT is not supported in OWE transition.
- If you enable FT, you get this error message,

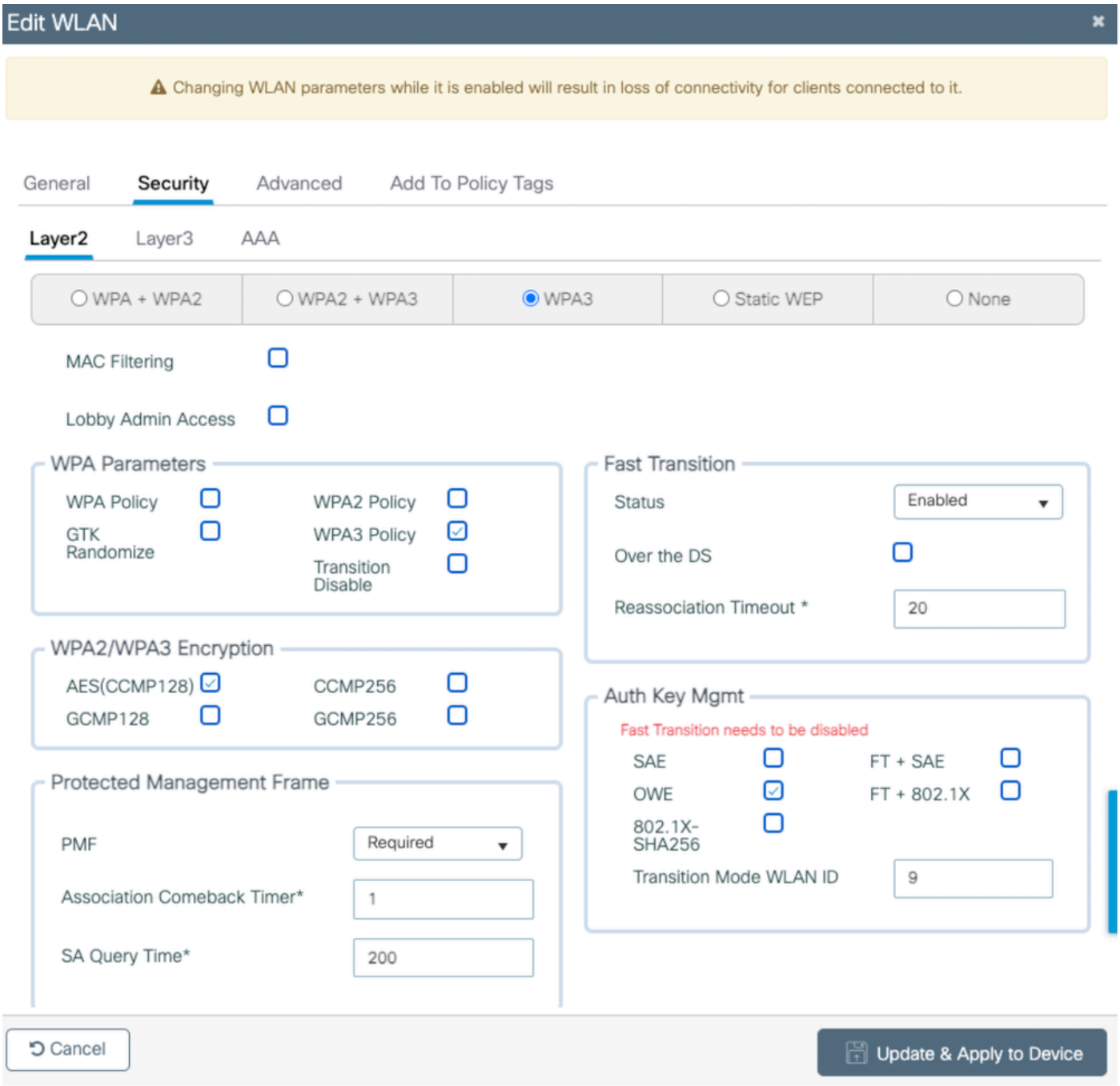

Image-9 : Error Message when we enable FT in OWE Transition SSID

OWE is not supported with PSK/dot1x

We are not able to enable OWE in these combinations,

1. 802.1x or FT+802.1x

2. PSK or FT+PSK or PSK-SHA256

3. SAE or FT+SAE

4. 802.1x-SHA256 or FT+802.1x-SHA256

If you try to enable any one of the these methods, you can get the error message,

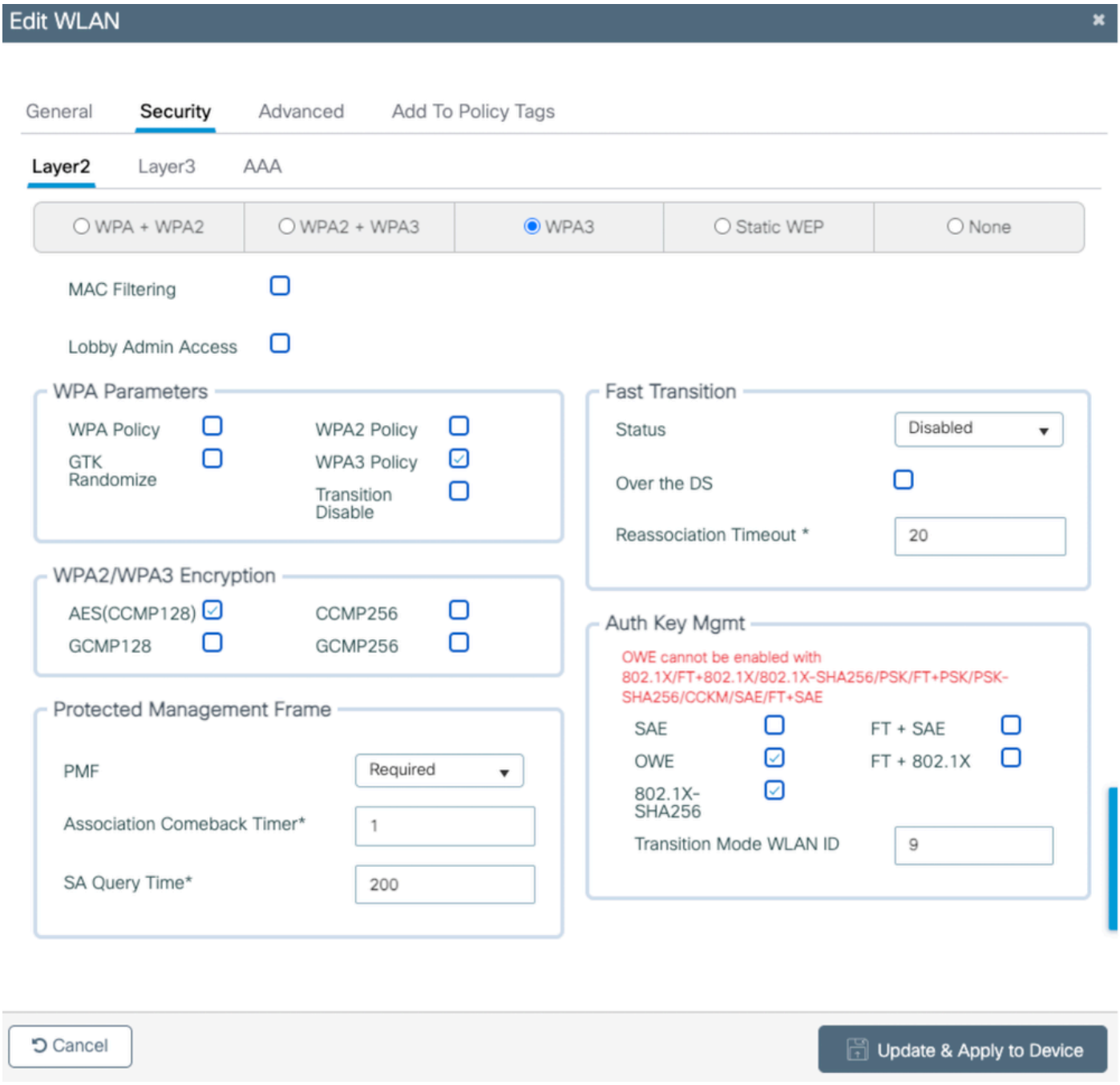

Image-10: Error Message getting while enabling other authentication methods in OWE SSID

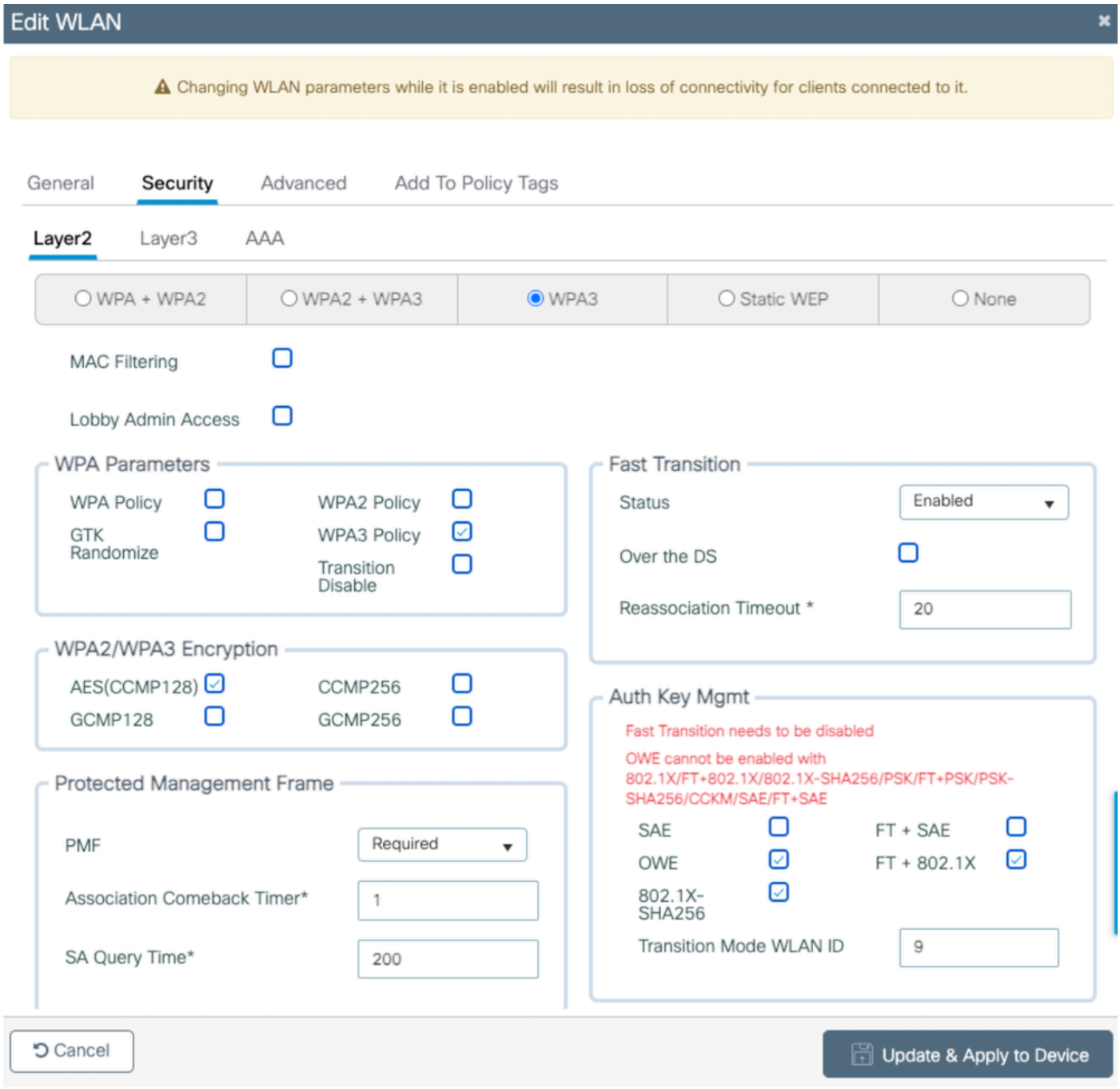

Image-11: Error Message while enabling AKM

- In Cisco IOS® XE 17.9.6 IOS version, you can see "OWE" option under AKM when you select "WPA2+WPA3" however you can get the error message, you are not able to use OWE with this combination.

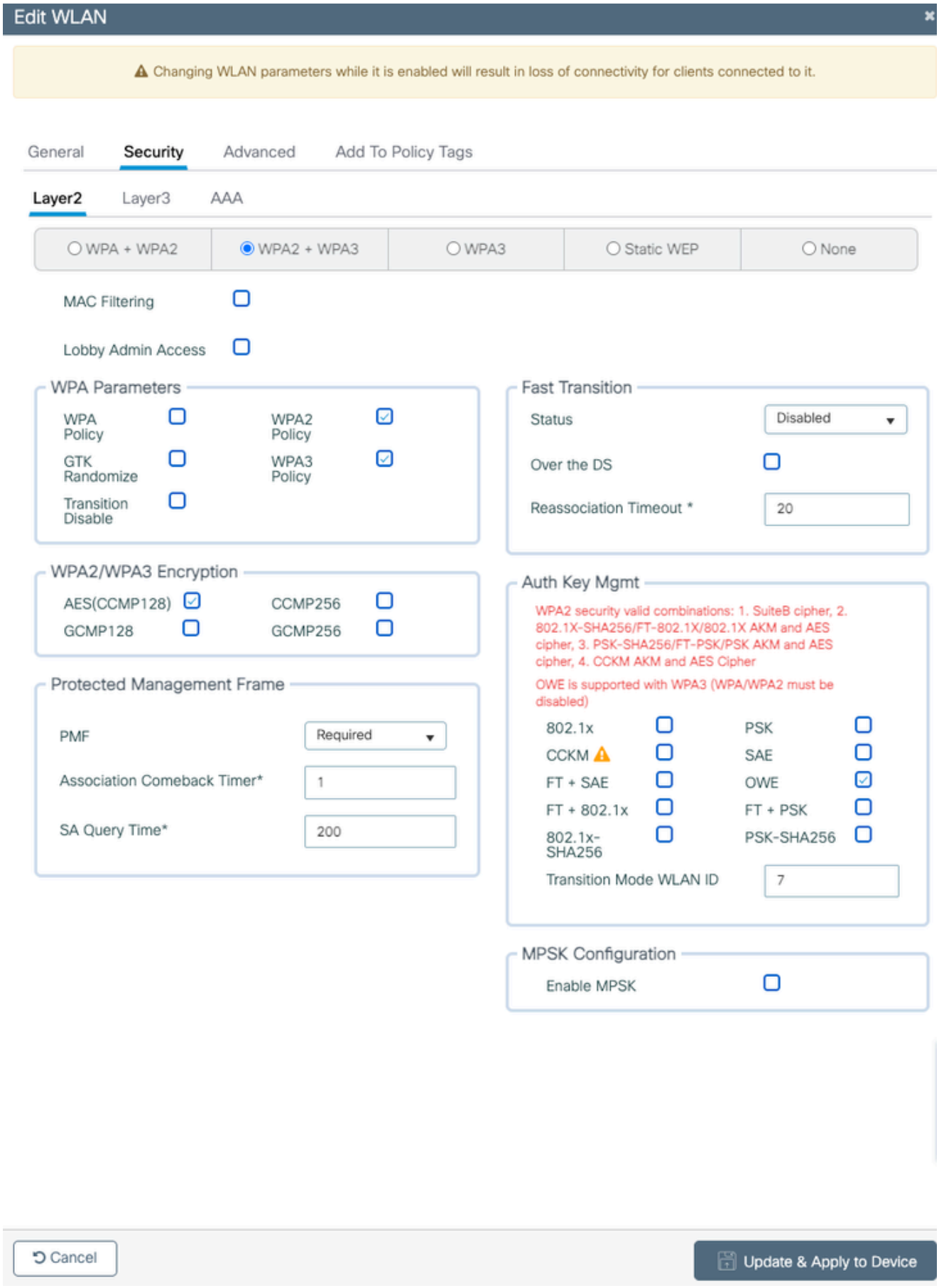

Image-12: Error message when we choose WPA2+WPA3

- In Cisco IOS® XE 17.12.4 version, when you choose "WPA2+WPA3", you cannot get the option "OWE" in AKM,

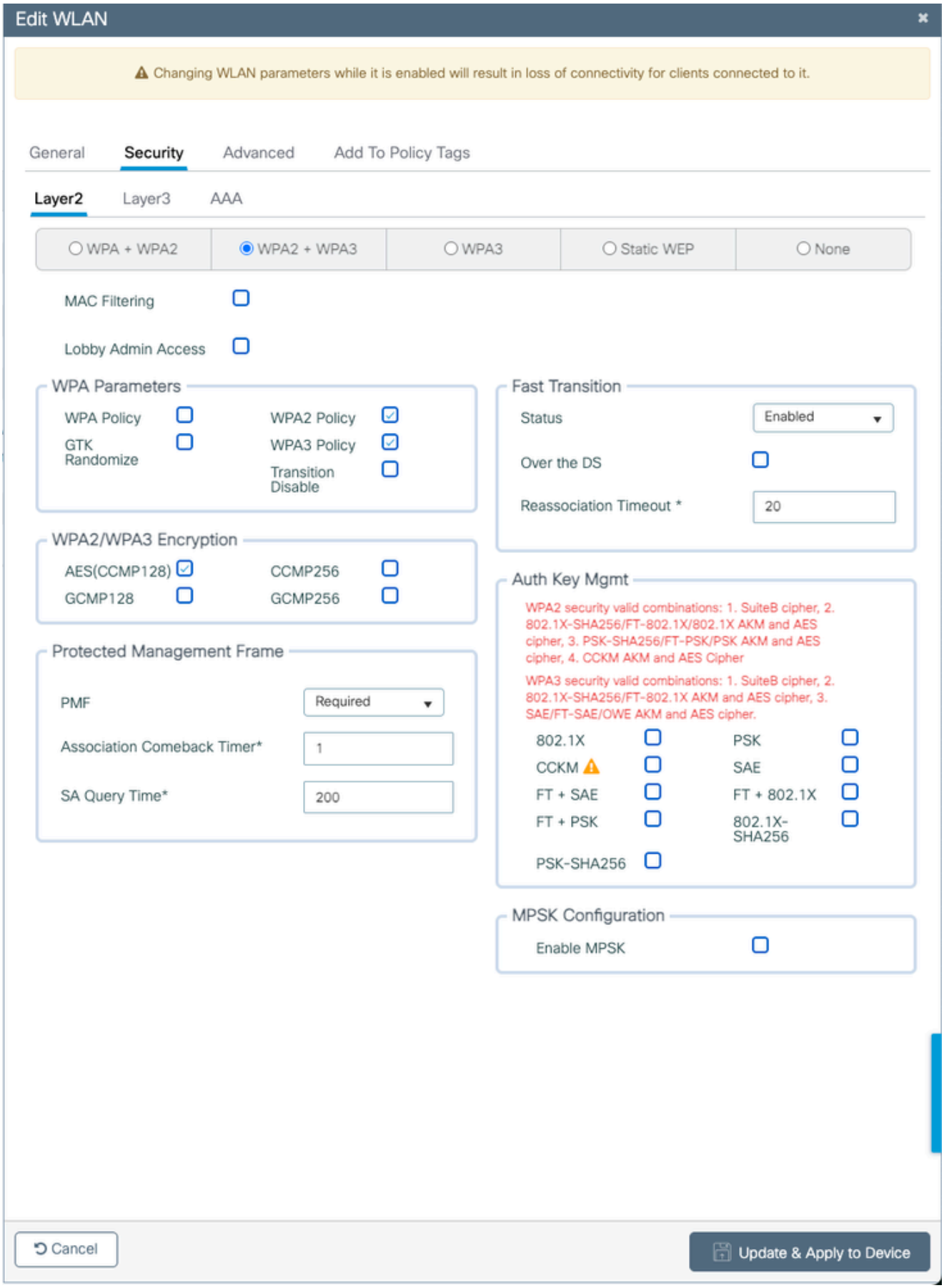

Image-13 : Error Message - Not getting OWE option in AKM

Troubleshooting

1. Check the configurations in both the WLANs, in OPEN SSID and in OWE transition SSID must have transition WLAN ID mapped.
2. Broadcasting option must be disabled in OWE transition SSID, it must be enabled only in OPEN SSID.
3. Check the supported authentication/encryption/FT methods described in this article.
4. If the configurations are fine from WLC end, than please collect the required logs and outputs to narrow down the issue,

RA Trace and EPC (Embeded PAcket Capture)

Login to WLC GUI -> Troubleshooting -> Radioactive Trace -> Add client wifi MAC address -> Click that clients checkbox -> Start

Login to WLC GUI -> Troubleshooting -> Packet Capture -> Add new file name -> Choose the uplink interface and WMI VLAN/Interface -> Start.

From Client Machine: If possible, you can install wireshark application and collect the packet capture by choosing WiFi interface.

https://www.cisco.com/c/en/us/support/docs/wireless/catalyst-9800-series-wireless-controllers/213949-wireless-debugging-and-log-collection-on.html#anc12

AIR PCAP

You can collect it by using MAC laptop or configuring one of the AP in to sniffer mode, please refer these links,

From MAC laptop:

https://www.cisco.com/c/en/us/support/docs/wireless-mobility/wireless-mobility/217042-collect-packet-captures-over-the-air-on.html

From Sniffer AP:

https://www.cisco.com/c/en/us/support/docs/wireless/catalyst-9800-series-wireless-controllers/217057-configure-access-point-in-sniffer-mode-o.html

Connect one laptop (wireshark server) to the switch port and it must have wireshark application installed in it, this wireshark server must have reachability to WLC WMI interface. Need to allow the protocol "5555 or 5000 or 5556" in firewall if it presents in between your WLC and wireshark server.

Check is there any "gscaler" installed in that PC where wireshark installed, if it is than please "turn off" and try, if it is any firewall like windows defender or anything present in it, please disable those and try to collect PCAP.

Roaming

When client roams from one AP to another, it needs to do these steps,

- Need to send re-association/association - Depends on the client request.
- Need to send DH (Diffie-Helman) details in association request.
- Client can get DH details in association response from AP, based on this PMK gets generated in both client and AP.
- 4-Way handshake can happen between AP and client.

- In OWE, you are unable to enable FT, so 802.11r is not possible.
- Each time, when client roams, it needs to do 4-way handshake after DH exchange in association.
- Client and AP using its own PMKID, it is unique for each APs and clients.
- If client connects to the same AP, than it can use the same PMKID. In some scenario, if the client got deleted than AP can generate a new PMKID however the client uses the same PMKID for 4-way handshake.

Example:

If client connects to the same AP, then you can see same PMKID in both Association-Request and Association-Response. In Association response, you cannot see DH details if it uses the same PMKID.

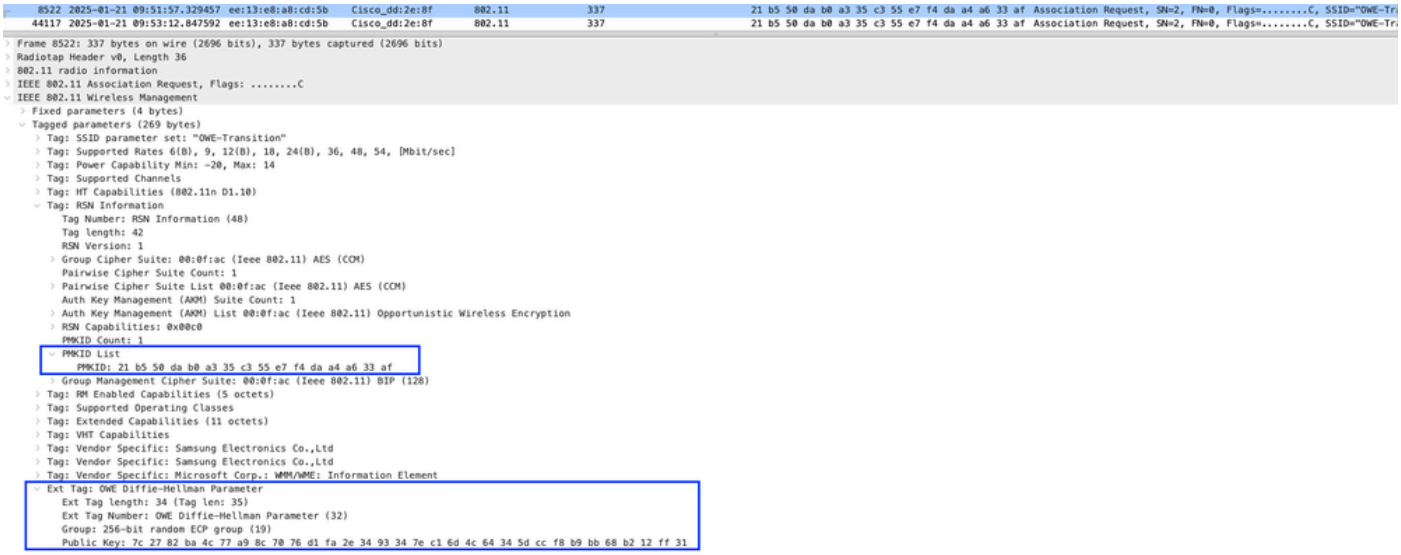

Image-14: Using Same PMKID

Image-15: Association Response with same PMKID

For testing, deleted this client manually from WLC and it got associated again to the same AP, at this time, the client sends a same PMKID however AP sends DH details in association response.

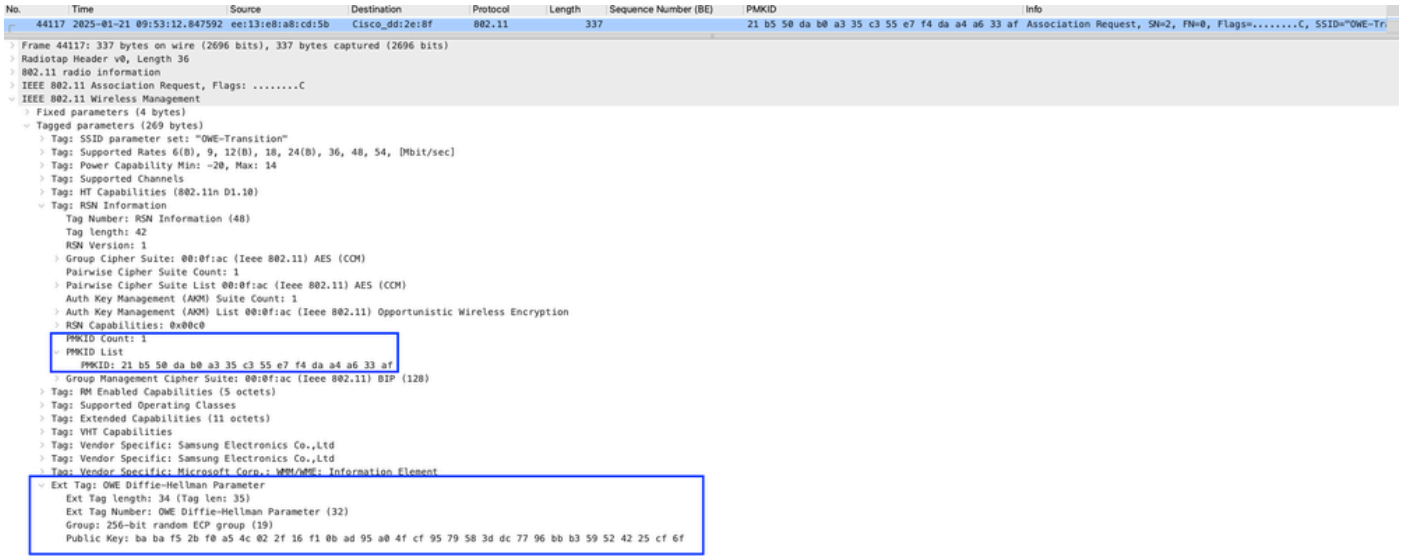

Image-16: After deleting, Client sent same PMKID with DH details

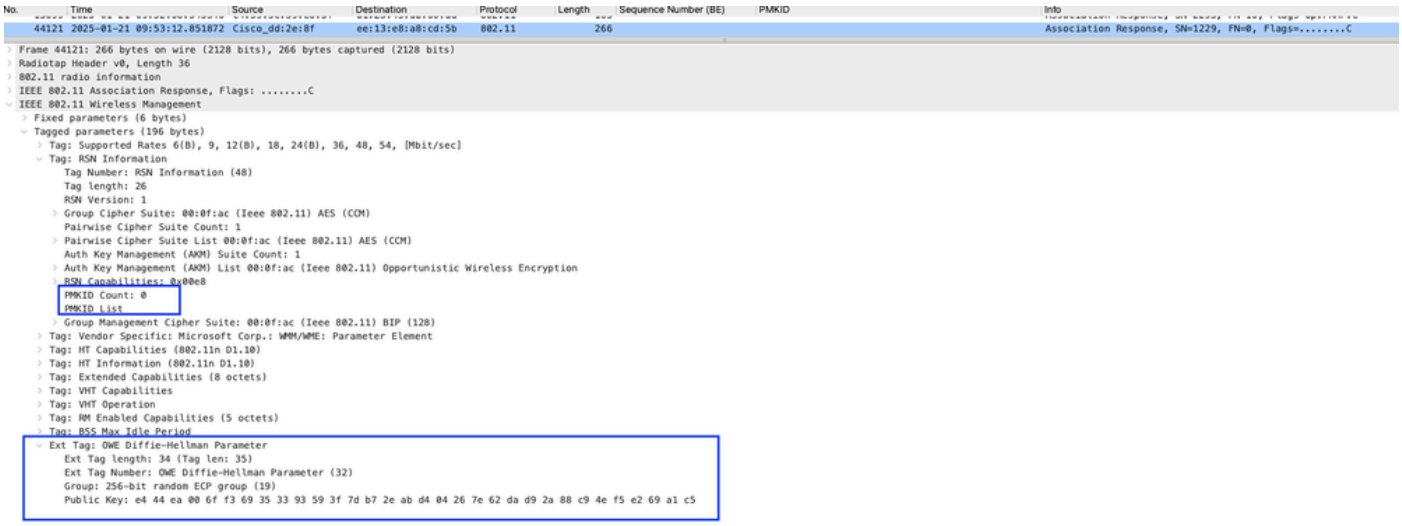

Image-17: AP uses DH values to generate its new PMKID

In this example: Both AP and client uses the same PMKID while doing 4-Way handshake, check in "M1 and in M2" messages.

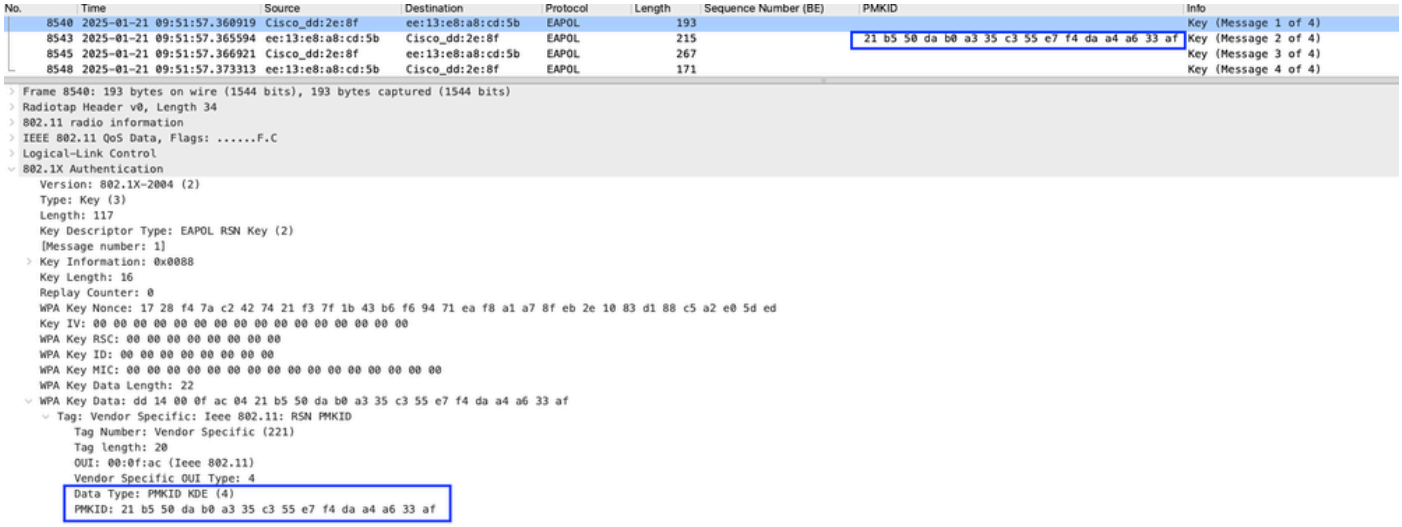

Image- 18: AP and Client using the same PMKID

In this example: Client using the same PMKID but AP using different PMKID which it generated after client got deleted, check "M1 and M2" messages.

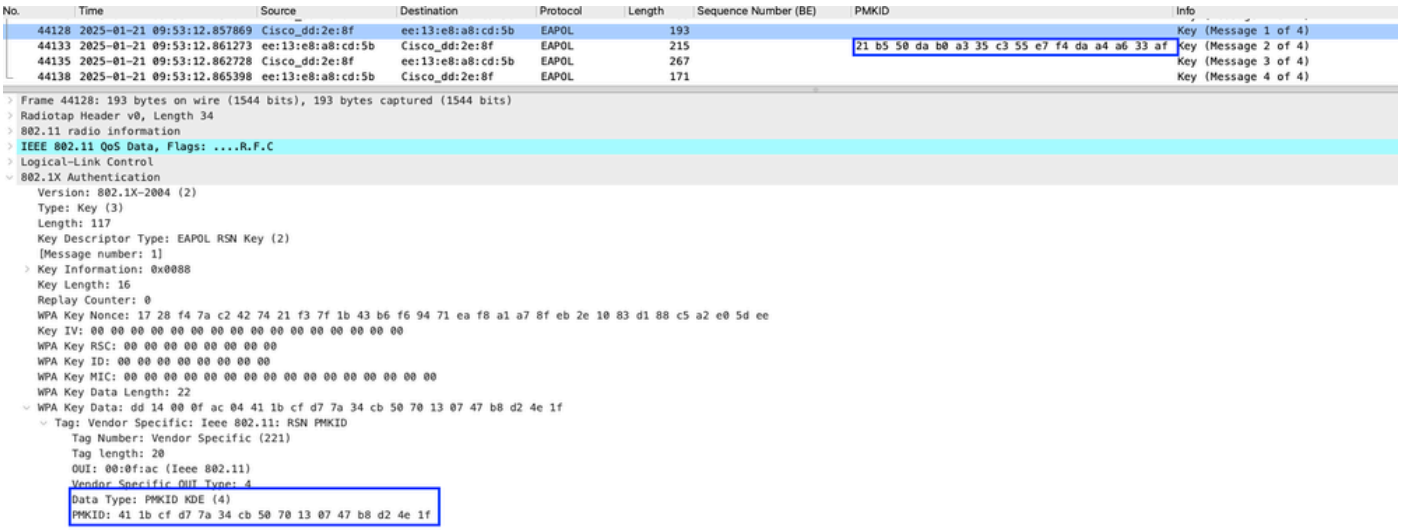

Image-19: AP and Client using different PMKID

From Internal RA Trace:

In this example: Client sent DH parameters in association-request and AP processed than generated the PMK.

```
2025/01/21 15:18:50.157081690 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:18:50.157082294 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:18:50.157523328 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:18:50.157531792 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:18:50.157532236 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:18:50.157532538 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:18:50.157841380 {wncd_x_R0-0}{1}: [dot11-frame] [21675]: (debug): MAC: ee13.e8a8.cd5b  OW
```

After this, the same client connecting to the same AP, at this time, AP did not generated new PMKID,

```
2025/01/21 15:21:57.391898613 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.391903915 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.391906073 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
2025/01/21 15:21:57.391906329 {wncd_x_R0-0}{1}: [dot11-validate] [21675]: (debug): MAC: ee13.e8a8.cd5b
```