Troubleshoot Identity PSK on Wireless LAN Controllers

Contents

Introduction

This document describes how to troubleshoot Identity Pre-Shared Key (PSK) connection issues on the Cisco Wireless LAN Controller (WLC).

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Cisco WLC that runs code 8.5 and higher and Identity Services Engine (ISE).
- Identity PSK configuration on the WLC and ISE. This can be found in this link:

https://www.cisco.com/c/en/us/td/docs/wireless/controller/technotes/8-5/b_Identity_PSK_Feature_Deployment_Guide.html

Components Used

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

- The Cisco 5508 Series WLC that runs software Release 8.5.103.0.
- The Cisco ISE that runs version 2.2.

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, make sure that you understand the potential impact of any command.

Understand Flow of Identity PSK

Step 1. Client sends an association request to the Service Set Identifier (SSID) enabled with PSK+MAC authentication.

Step 2. Since MAC authentication has enabled the WLC contacts, the radius server is to verify the MAC address of the client.

Step 3. Radius server verifies the client details and sends the Cisco av-pairs for which it specifies PSK as the authentication type to be used as well as the key value to be used for the client.

Step 4. Once this is received the WLC sends the association response to the client. It is important to be aware of this step, as if there is a delay in the communication between the WLC and radius server, clients can get stuck in an association loop, where they send a second association request before the response is received from the radius server.

Step 5. The WLC uses the key value sent by the radius server as the master key. The Access Point (AP) then proceeds with the four way handshake which verifies that the password configured on the client matches the value sent by the radius server.

Step 6. The client then completes the DHCP process and moves into RUN state as well.

Troubleshoot Scenarios

These debugs are required to troubleshoot Identity PSK issues:

Debugs on the WLC:

- **debug client client_mac,** where client _mac is the MAC address of the client's test.
- **debug aaa detail enable**

Scenario 1. Pass Scenario where Client Connects Successfully

The client sends the association request to the AP:

```
*apfMsConnTask_6: Sep 21 15:01:43.496: e8:50:8b:64:4f:45 Association received from mobile on
BSSID 28:6f:7f:e2:24:cf AP AP_2802-1
```

The WLC then contacts the radius server to verify the client MAC address:

```
*aaaQueueReader: Sep 21 15:01:43.498: AuthenticationRequest: 0x2b8c8a9c
*apfMsConnTask_6: Sep 21 15:01:43.498: e8:50:8b:64:4f:45 apfProcessAssocReq (apf_80211.c:11440)
Changing state for mobile e8:50:8b:64:4f:45 on AP 28:6f:7f:e2:24:c0 from Associated to AAA
Pending

*aaaQueueReader: Sep 21 15:01:43.498:
Callback.....................................0x10762018

*aaaQueueReader: Sep 21 15:01:43.498:
protocolType.................................0x40000001
```

The radius server responds with the Access-Accept message which also contains the PSK method type and key which is used for authentication:

```
*radiusTransportThread: Sep 21 15:01:43.794: AuthorizationResponse: 0x171b5c00

*radiusTransportThread: Sep 21 15:01:43.794:
structureSize.............................313

*radiusTransportThread: Sep 21 15:01:43.794:
resultCode.................................0


*radiusTransportThread: Sep 21 15:01:43.794:         Packet contains 5 AVPs:

*radiusTransportThread: Sep 21 15:01:43.794:             AVP[01] User-
Name..............................E8-50-8B-64-4F-45 (17 bytes)

*radiusTransportThread: Sep 21 15:01:43.794:             AVP[02]
State..................................ReauthSession:0a6a20770000000059c346ed (38 bytes)

*radiusTransportThread: Sep 21 15:01:43.794:             AVP[03]
Class..................................CACS:0a6a20770000000059c346ed:ISE/291984633/6 (45
bytes)

*radiusTransportThread: Sep 21 15:01:43.794:             AVP[04] Cisco / PSK-
Mode.......................ascii (5 bytes)

*radiusTransportThread: Sep 21 15:01:43.794:             AVP[05] Cisco /
PSK............................cisco123 (8 bytes)
```

Once this is received you can see that the WLC sends the association response and a four way handshake happens:

```
*apfReceiveTask: Sep 21 15:01:43.924: e8:50:8b:64:4f:45 Sending assoc-resp with status 0
station:e8:50:8b:64:4f:45 AP:28:6f:7f:e2:24:c0-01 on apVapId 1
```

The Four way handshake:

```
*Dot1x_NW_MsgTask_5: Sep 21 15:01:43.994: e8:50:8b:64:4f:45 Sending EAPOL-Key Message to mobile
e8:50:8b:64:4f:45
   state INITPMK (message 1), replay counter 00.00.00.00.00.00.00.00
*Dot1x_NW_MsgTask_5: Sep 21 15:01:43.998: e8:50:8b:64:4f:45 Received EAPOL-key in PTK_START
state (message 2) from mobile e8:50:8b:64:4f:45
*Dot1x_NW_MsgTask_5: Sep 21 15:01:43.998: e8:50:8b:64:4f:45 Received valid MIC in EAPOL Key
Message M2!!!!!
*Dot1x_NW_MsgTask_5: Sep 21 15:01:43.999: e8:50:8b:64:4f:45 Sending EAPOL-Key Message to mobile
e8:50:8b:64:4f:45
   state PTKINITNEGOTIATING (message 3), replay counter 00.00.00.00.00.00.00.01
*Dot1x_NW_MsgTask_5: Sep 21 15:01:44.003: e8:50:8b:64:4f:45 Received EAPOL-key in
PTKINITNEGOTIATING state (message 4) from mobile e8:50:8b:64:4f:45
```

Once this is done, client completes the DHCP process and goes into RUN state (output is clipped to show the important sections):

```
(WLC_1) >show client detail e8:50:8b:64:4f:45
Client MAC Address.............................. e8:50:8b:64:4f:45
Client Username ................................ E8-50-8B-64-4F-45
Hostname: ...................................... S6-edge
Device Type: ................................... Android-Samsung-Galaxy-Phone
AP MAC Address.................................. 28:6f:7f:e2:24:c0
AP Name......................................... AP_2802-1
Wireless LAN Network Name (SSID)................ Identity PSK
Wireless LAN Profile Name....................... Identity PSK
Security Policy Completed....................... Yes
```

```
Policy Manager State.......................... RUN
```

Scenario 2. Client Tries to Connect with Incorrect Password

The initial sequence of steps stays the same as that of a passed authentication.

- The client sends an association request.
- Once the WLC receives this, it initiates communication with the radius server to verify the client MAC address.
- If the radius server has the client details it sends an access-accept with the key value and authentication type which is PSK.
- The useful section where the failure can be noticed is in the four way handshake.

The AP sends message 1, to which the client responds with message 2:

```
*Dot1x_NW_MsgTask_7: Sep 21 15:12:47.661: 50:8f:4c:9d:ef:87 Received EAPOL-key in PTK_START
state (message 2) from mobile 50:8f:4c:9d:ef:87
```

However, due to different master key values (password) the AP and client derive different keys which results in an invalid MIC receipt in message 2:

```
*Dot1x_NW_MsgTask_7: Sep 21 15:12:47.662: 50:8f:4c:9d:ef:87 Received EAPOL-key M2 with invalid
MIC from mobile 50:8f:4c:9d:ef:87 version 2
*osapiBsnTimer: Sep 21 15:12:48.824: 50:8f:4c:9d:ef:87 802.1x 'timeoutEvt' Timer expired for
station 50:8f:4c:9d:ef:87 and for message = M2
*Dot1x_NW_MsgTask_7: Sep 21 15:12:48.824: 50:8f:4c:9d:ef:87 Retransmit 1 of EAPOL-Key M1 (length
121) for mobile 50:8f:4c:9d:ef:87

The client then is then de-authenticated by the WLC:
*Dot1x_NW_MsgTask_7: Sep 21 15:12:50.825: 50:8f:4c:9d:ef:87 Sent Deauthenticate to mobile on
BSSID 28:6f:7f:e2:24:c0 slot 0(caller 1x_ptsm.c:655)
```

Another useful output to check is the 'show client detail'. Here you can see the client is stuck in START state:

```
*Dot1x_NW_MsgTask_7: Sep 21 15:12:47.662: 50:8f:4c:9d:ef:87 Received EAPOL-key M2 with invalid
MIC from mobile 50:8f:4c:9d:ef:87 version 2
*osapiBsnTimer: Sep 21 15:12:48.824: 50:8f:4c:9d:ef:87 802.1x 'timeoutEvt' Timer expired for
station 50:8f:4c:9d:ef:87 and for message = M2
*Dot1x_NW_MsgTask_7: Sep 21 15:12:48.824: 50:8f:4c:9d:ef:87 Retransmit 1 of EAPOL-Key M1 (length
121) for mobile 50:8f:4c:9d:ef:87

The client will then be de-authenticated by the WLC:
*Dot1x_NW_MsgTask_7: Sep 21 15:12:50.825: 50:8f:4c:9d:ef:87 Sent Deauthenticate to mobile on
BSSID 28:6f:7f:e2:24:c0 slot 0(caller 1x_ptsm.c:655)
```

Scenario 3. Radius Server Unreachable

The WLC tries to contact the radius server once it receives the association request. In case the radius server is unreachable, the WLC repeatedly tries to contact the radius server (until the retry count is reached). Once the radius server is detected to be unreachable after the configured number of retries (default value is 5) the WLC sends an association response with status code 1 as shown here:

```
*apfReceiveTask: Sep 21 15:28:55.777: 50:8f:4c:9d:ef:87 Sending assoc-resp with status 1
station:50:8f:4c:9d:ef:87 AP:a0:e0:af:62:f3:c0-00 on apVapId 1
*apfReceiveTask: Sep 21 15:28:55.777: 50:8f:4c:9d:ef:87 Sending Assoc Response (status:
'unspecified failure') to station on AP AP_2802-2 on BSSID a0:e0:af:62:f3:c0 ApVapId 1 Slot 0,
mobility role 0
```

You can also see the number of retry requests and timeout requests which grows in the radius server statistics, for which you can navigate to **Monitor >Statistics >RADIUS Servers** as shown in the image:

Scenario 4. Incorrect Override Parameter Sent by Radius Server

There are several parameters that can be pushed along with PSK and the key, such as VLAN, ACL and User Role. However, if the ACL entry sent by the radius server is not configured then the WLC rejects the client, even if the radius server approves the authentication request. This can be clearly seen in the client debugs:

```
*radiusTransportThread: Sep 22 14:39:05.499: AuthorizationResponse: 0x171b5c00

*radiusTransportThread: Sep 22 14:39:05.499:
structureSize.............................376

*radiusTransportThread: Sep 22 14:39:05.499:
resultCode.................................0

*radiusTransportThread: Sep 22 14:39:05.499:
protocolUsed.............................0x00000001

*radiusTransportThread: Sep 22 14:39:05.499:         Packet contains 7 AVPs:
```

```
*radiusTransportThread: Sep 22 14:39:05.499:            AVP[01] User-
Name.............................E8-50-8B-64-4F-45 (17 bytes)

*radiusTransportThread: Sep 22 14:39:05.499:            AVP[02]
State.................................ReauthSession:0a6a20770000002659c493e9 (38 bytes)

*radiusTransportThread: Sep 22 14:39:05.499:            AVP[03]
Class.................................CACS:0a6a20770000002659c493e9:ISE/291984633/78 (46
bytes)

*radiusTransportThread: Sep 22 14:39:05.499:            AVP[04] Cisco / PSK-
Mode.......................ascii (5 bytes)

*radiusTransportThread: Sep 22 14:39:05.499:            AVP[05] Cisco /
PSK............................cisco123 (8 bytes)

*radiusTransportThread: Sep 22 14:39:05.499:            AVP[06] Unknown Cisco / Attribute
19............teacher (7 bytes)

*radiusTransportThread: Sep 22 14:39:05.499:            AVP[07] Airespace / ACL-
Name.....................testing (7 bytes)
```

Client debugs:

```
*apfReceiveTask: Sep 22 14:39:05.564: e8:50:8b:64:4f:45  ACL received from RADIUS does not exist
in WLC de-authenticating the client
*apfReceiveTask: Sep 22 14:39:05.628: e8:50:8b:64:4f:45 Sending assoc-resp with status 12
station:e8:50:8b:64:4f:45 AP:28:6f:7f:e2:24:c0-01 on apVapId 1
```

Scenario 5. Client Policy Not Configured on Radius Server

When the radius server is reachable but there is no policy configured on the radius server for the client, it can get connected only if it uses the PSK, configured globally under the WLAN. Any other entries would fail. There is nothing specific to differentiate between a working global PSK authentication and a working identity PSK authentication except in the debug Authentication, Authorization, and Accounting (AAA) output which won't have any override parameters that is pushed:

```
*radiusTransportThread: Sep 22 14:32:13.734: AuthorizationResponse: 0x171b5c00

*radiusTransportThread: Sep 22 14:32:13.734:
structureSize.............................269

*radiusTransportThread: Sep 22 14:32:13.734:
resultCode.................................0

*radiusTransportThread: Sep 22 14:32:13.734:
protocolUsed.............................0x00000001

*radiusTransportThread: Sep 22 14:32:13.734:
proxyState.................................50:8F:4C:9D:EF:87-00:00

*radiusTransportThread: Sep 22 14:32:13.734:         Packet contains 3 AVPs:

*radiusTransportThread: Sep 22 14:32:13.734:            AVP[01] User-
Name.............................50-8F-4C-9D-EF-87 (17 bytes)

*radiusTransportThread: Sep 22 14:32:13.734:            AVP[02]
State.................................ReauthSession:0a6a20770000002359c49240 (38 bytes)

*radiusTransportThread: Sep 22 14:32:13.734:            AVP[03]
Class.................................CACS:0a6a20770000002359c49240:ISE/291984633/74 (46
```

bytes)