Security

Information About Data Datagram Transport Layer Security

Data Datagram Transport Layer Security (DTLS) enables you to encrypt CAPWAP data packets that are sent between an access point and the controller using DTLS, which is a standards-track IETF protocol that can encrypt both control and data packets based on TLS. CAPWAP control packets are management packets that are exchanged between a controller and an access point while CAPWAP data packets encapsulate forwarded wireless frames. CAPWAP control and data packets are sent over separate UDP ports: 5246 (control) and 5247 (data).

If an access point does not support DTLS data encryption, DTLS is enabled only for the control plane, and a DTLS session for the data plane is not established.

If an access point supports Data DTLS, it enables data DTLS after receiving the new configuration from the controller . The access point performs a DTLS handshake on port 5247 and after successfully establishing the DTLS session. All the data traffic (from the access point to the controller and the controller to the access point) is encrypted.

Note The throughput is affected for some APs that have data encryption enabled.

The controller does not perform a DTLS handshake immediately after processing client-hello with a cookie, if the following incorrect settings are configured:

- ECDHE-ECDSA cipher in “ap dtls-cipher <>” and RSA-based certificate in “wireless management trustpoint”.
- RSA cipher in “ap dtls-cipher <>” and EC-based certificate in “wireless management trustpoint”.

Note This is applicable when you move from CC -> FIPS -> non-FIPS mode.

Note If the AP’s DHCP lease time is less and the DHCP pool is small, access point join failure or failure in establishing the Data Datagram Transport Layer Security (DTLS) session may occur. In such scenarios, associate the AP with a named site-tag and increase the DHCP lease time for at least 8 days.

Configuring Data DTLS (GUI)

Follow the procedure to enable DTLS data encryption for the access points on the controller :

Procedure

Step 1 Click **Configuration** > **Tags and Profile** > **AP Join**.

Step 2 Click **Add** to create a new **AP Join Profile** or click an existing profile to edit it.

Step 3 Click **CAPWAP** > **Advanced**.

Step 4 Check **Enable Data Encryption** check box to enable Datagram Transport Layer Security (DTLS) data encryption.

Step 5 Click **Update & Apply to Device**.

Configuring Data DTLS (CLI)

Follow the procedure given below to enable DTLS data encryption for the access points on the controller :

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ap profile** *ap-profile* **Example:** `Device(config)# ap profile test-ap-profile`	Configures an AP profile and enters AP profile configuration mode. **Note** You can use the default AP profile (default-ap-profile) or create a named AP profile, as shown in the example.

Step 3	**link-encryption**
	Example:
	`Device(config-ap-profile)# link-encryption`
		Enables link encryption based on the profile. Answer yes, when the system prompts you with this message:
		Note If you set stats-timer as as zero (0) under the AP profile, then the AP will not send the link encryption statistics.
		`Enabling link-encryption will reboot the APs with link-encryption.`
		`Are you sure you want to continue? (y/n)[y]:`

Step 4	**end** **Example:** `Device(config-ap-profile)# end`	Returns to privileged EXEC mode.
Step 5	**show wireless dtls connections** **Example:** `Device# show wireless dtls connections`	(Optional) Displays the DTLS session established for the AP that has joined this controller.
Step 6	**show ap link-encryption** **Example:** `Device# show ap link-encryption`	(Optional) Displays the link encryption-related statistics (whether link encryption is enabled or disabled) counter received from the AP.

Introduction to the 802.1X Authentication

IEEE 802.1X port-based authentication is configure on a device to prevent unauthorized devices from gaining access to the network. The device can combine the function of a router, switch, and access point, depending on the fixed configuration. Any device connecting to a switch port where 802.1X authentication is enabled must go through relevant EAP authentication model to start exchanging traffic.

Currently, the Cisco Wave 2 and Wi-Fi 6 (802.11AX) APs support 802.1X authentication with switch port for EAP-FAST, EAP-TLS and EAP-PEAP methods. Now, you can enable configurations and provide credentials to the AP from the controller .

Note If the AP is dot1x EAP-FAST, when the AP reboots, it should perform an anonymous PAC provision. For performing PAC provision, the ADH cipher suites should be used to establish an authenticated tunnel. If the ADH cipher suites are not supported by radius servers, AP will fail to authenticate on reload.

EAP-FAST Protocol

In the EAP-FAST protocol developed by Cisco, in order to establish a secured TLS tunnel with RADIUS, the AP requires a strong shared key (PAC), either provided via in-band provisioning (in a secured channel) or via out-band provisioning (manual).

Note The EAP-FAST type configuration requires 802.1x credentials configuration for AP, since AP will use EAP-FAST with MSCHAP Version 2 method.

Note Local EAP is not supported on the Cisco 7925 phones.

Note In Cisco Wave 2 APs, for 802.1x authentication using EAP-FAST after PAC provisioning (caused by the initial connection or after AP reload), ensure that you configure the switch port to trigger re-authentication using one of the following commands: **authentication timer restart** *num* or **authentication timer reauthenticate** *num*.

Starting from Cisco IOS XE Amsterdam 17.1.1, TLS 1.2 is supported in EAP-FAST authentication protocol.

EAP-TLS/EAP-PEAP Protocol

The EAP-TLS protocol or EAP-PEAP protocol provides certificate based mutual EAP authentication.

In EAP-TLS, both the server and the client side certificates are required, where the secured shared key is derived for the particular session to encrypt or decrypt data. Whereas, in EAP-PEAP, only the server side certificate is required, where the client authenticates using password based protocol in a secured channel.

Note The EAP-PEAP type configuration requires Dot1x credentials configuration for AP; and the AP also needs to go through LSC provisioning. AP uses the PEAP protocol with MSCHAP Version 2 method.

Limitations of the 802.1X Authentication

- 802.1X is not supported on dynamic ports or Ethernet Channel ports.
- 802.1X is not supported in a mesh AP scenario.
- There is no recovery from the controller on credential mismatch or the expiry/invalidity of the certificate on AP. The 802.1X authentication has to be disabled on the switch port to connect the AP back to fix the configurations.
- There are no certificate revocation checks implemented on the certificates installed in AP.

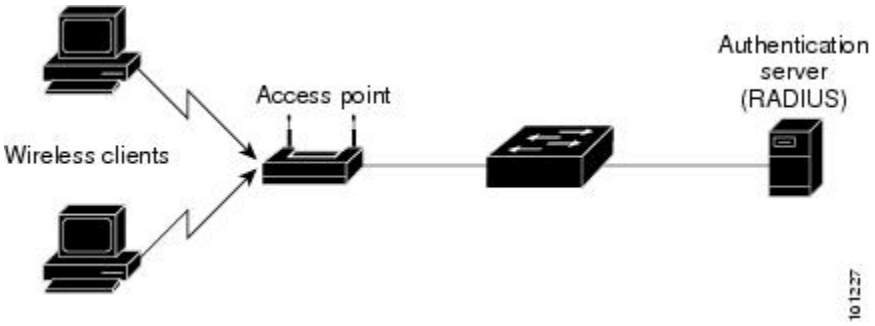
- Only one Locally Significant Certificates (LSC) can be provisioned on the AP and the same certificate must be used for CAPWAP DTLS session establishment with controller and the 802.1X authentication with the switch. If global LSC configuration on the controller is disabled; AP deletes LSC which is already provisioned.
- If clear configurations are applied on the AP, then the AP will lose the 802.1X EAP type configuration and the LSC certificates. AP should again go through staging process if 802.1X is required.
- 802.1X for trunk port APs on multi-host authentication mode is supported. Network Edge Authentication Topology (NEAT) is not supported on COS APs.
- The DHCP requests are sent in incremental periodic value of: "2, 3, 4, 6, 8, 11, 15, 20, 27, 30, 30, 30, 30, 30...". The Cisco Catalyst 9100 Access Points perform an interface reset following a 100-second timeout, which in turn resets the timers on the associated switch port to which they are connected.

Topology - Overview

The 802.1X authentication events are as follows:

1. The AP acts as the 802.1X supplicant and is authenticated by the switch against the RADIUS server which supports EAP-FAST along with EAP-TLS and EAP-PEAP. When dot1x authentication is enabled on a switch port, the device connected to it authenticates itself to receive and forward data other than 802.1X traffic.
2. In order to authenticate with EAP-FAST method, the AP requires the credentials of the RADIUS server. It can be configured at the controller , from where it will be passed on to the AP via configuration update request. For, EAP-TLS or EAP-PEAP the APs use the certificates (device/ID and CA) made significant by the local CA server.

Figure 1: Figure: 1 Topology for 802.1X Authentication



Configuring 802.1X Authentication Type and LSC AP Authentication Type (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join Profile** page, click **Add**.

The **Add AP Join Profile** page is displayed.

Step 3 In the **AP** > **General** tab, navigate to the **AP EAP Auth Configuration** section.

Step 4 From the **EAP Type** drop-down list, choose the EAP type as *EAP-FAST*, *EAP-TLS*, or *EAP-PEAP* to configure the dot1x authentication type.

Step 5 From the **AP Authorization Type** drop-down list, choose the type as either CAPWAP DTLS + or CAPWAP DTLS.

Step 6 Click **Save & Apply to Device**.

Configuring 802.1X Authentication Type and LSC AP Authentication Type

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ap profile** *profile-name* **Example:** `Device(config)# ap profile new-profile`	Specify a profile name.
Step 4	**dot1x {max-sessions | username | eap-type | lsc-ap-auth-state}** **Example:** `Device(config-ap-profile)# dot1x eap-type`	Configures the dot1x authentication type. **max-sessions**: Configures the maximum 802.1X sessions initiated per AP.

	Command or Action	Purpose
		username: Configures the 802.1X username for all Aps. **eap-type:** Configures the dot1x authentication type with the switch port. **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.
Step 5	**dot1x eap-type {EAP-FAST | EAP-TLS | EAP-PEAP}** **Example:** `Device(config-ap-profile)# dot1x eap-type`	Configures the dot1x authentication type: EAP-FAST, EAP-TLS, or EAP-PEAP.
Step 6	**dot1x lsc-ap-auth-state {CAPWAP-DTLS | Dot1x-port-auth | Both}** **Example:** `Device(config-ap-profile)#dot1x lsc-ap-auth-state Dot1x-port-auth`	Configures the LSC authentication state on the AP. **CAPWAP-DTLS**: Uses LSC only for CAPWAP DTLS. **Dot1x-port-auth**: Uses LSC only for dot1x authentication with port. **Both**: Uses LSC for both CAPWAP-DTLS and Dot1x authentication with port.
Step 7	**end** **Example:** `Device(config-ap-profile)# end`	Exits the AP profile configuration mode and enters privileged EXEC mode.

Configuring the 802.1X Username and Password (GUI)

Procedure

Step 1 Choose **Configuration** > **Tags & Profiles** > **AP Join**.

Step 2 On the **AP Join** page, click the name of the AP Join profile or click **Add** to create a new one.

Step 3 Click the **Management** tab and then click the **Credentials** tab.

Step 4 Enter the local username and password details.

Step 5 Choose the appropriate local password type.

Step 6 Enter 802.1X username and password details.

Step 7 Choose the appropriate 802.1X password type.

Step 8 Enter the time in seconds after which the session should expire.

Step 9 Enable local credentials and/or 802.1X credentials as required.

Step 10 Click **Update & Apply to Device**.

Configuring the 802.1X Username and Password (CLI)

The following procedure configures the 802.1X password for all the APs:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ap profile** *profile-name* **Example:** `Device(config)# ap profile new-profile`	Specify a profile name.
Step 4	**dot1x {max-sessions | username | eap-type | lsc-ap-auth-state}** **Example:** `Device(config-ap-profile)# dot1x eap-type`	Configures the dot1x authentication type. **max-sessions**: Configures the maximum 802.1X sessions initiated per AP. **username**: Configures the 802.1X username for all Aps. **eap-type:** Configures the dot1x authentication type with the switch port. **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.
Step 5	**dot1x username** <username> **password {0 | 8}** <password> **Example:** `Device(config-ap-profile)#dot1x username` *`username`* `password 0` *`password`*	Configures the dot1x password for all the APs. 0: Specifies an unencrypted password will follow. 8: Specifies an AES encrypted password will follow.

Enabling 802.1X on the Switch Port

The following procedure enables 802.1X on the switch port:

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**aaa new-model** **Example:** `Device(config)# aaa new-model`	Enables AAA.
Step 4	**aaa authentication dot1x {default | listname} method1[method2...]** **Example:** `Device(config)# aaa authentication dot1x default group radius`	Creates a series of authentication methods that are used to determine user privilege to access the privileged command level so that the device can communicate with the AAA server.
Step 5	**aaa authorization network group** **Example:** `aaa authorization network group`	Enables AAA authorization for network services on 802.1X.
Step 6	**dot1x system-auth-control** **Example:** `Device(config)# dot1x system-auth-control`	Globally enables 802.1X port-based authentication.
Step 7	**interface type slot/port** **Example:** `Device(config)# interface fastethernet2/1`	Enters interface configuration mode and specifies the interface to be enabled for 802.1X authentication.
Step 8	**authentication port-control {auto | force-authorized | force-unauthorized}** **Example:** `Device(config-if)# authentication port-control auto`	Enables 802.1X port-based authentication on the interface. **auto**—Enables IEEE 802.1X authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The Device requests the identity of the supplicant and begins relaying authentication messages between the supplicant and the authentication server. Each supplicant

	Command or Action	Purpose
		attempting to access the network is uniquely identified by the Device by using the supplicant MAC address. **force-authorized**—Disables IEEE 802.1X authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without IEEE 802.1X-based authentication of the client. This is the default setting. **force-unauthorized**—Causes the port to remain in the unauthorized state, ignoring all attempts by the supplicant to authenticate. The Device cannot provide authentication services to the supplicant through the port.
Step 9	**dot1x pae [supplicant | authenticator | both]** **Example:** `Device(config-if)# dot1x pae authenticator`	Enables 802.1X authentication on the port with default parameters.
Step 10	**end** **Example:** `Device(config-if)# end`	Enters privileged EXEC mode.

Verifying 802.1X on the Switch Port

The following show command displays the authentication state of 802.1X on the switch port:

```
Device# show dot1x all
Sysauthcontrol              Enabled
Dot1x Protocol Version      2
Dot1x Info for FastEthernet1
----------------------------------
PAE                       = AUTHENTICATOR
PortControl               = AUTO
ControlDirection          = Both
HostMode                  = MULTI_HOST
ReAuthentication          = Disabled
QuietPeriod               = 60
ServerTimeout             = 30
SuppTimeout               = 30
ReAuthPeriod              = 3600 (Locally configured)
ReAuthMax                 = 2
MaxReq                    = 2
TxPeriod                  = 30
RateLimitPeriod           = 0
Device#
```

Verifying the Authentication Type

The following show command displays the authentication state of an AP profile:

```
Device#show ap profile <profile-name> detailed ?
  chassis  Chassis
  |        Output modifiers
  <cr>

Device#show ap profile <profile-name> detailed

  AP Profile Name        : default-ap-profile
  Description            : default ap profile
  …
  Dot1x EAP Method       : [EAP-FAST/EAP-TLS/EAP-PEAP/Not-Configured]
  LSC AP AUTH STATE      : [CAPWAP DTLS / DOT1x port auth / CAPWAP DTLS + DOT1x port auth
```

Feature History for Access Point Client ACL Counter

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 1: Feature History for Access Point Client ACL Counter

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.13.1	Access Point Client ACL Counter	The AP Client ACL Counter feature provides a statistical count for client ACL rules. This feature allows you to count the number of packets that hit a specific rule in the client ACL.

Information About Access Point Client ACL Counter

From the Cisco IOS XE Dublin 17.13.1 release, the AP Client ACL Counter feature provides a statistical count for client ACL rules. Until the Cisco IOS XE Dublin 17.12.1 release, there was no per-rule counter to determine which rule was passing or dropping the packets.

Use this feature to enable the counter in the AP to count the number of packets that hit a specific rule in the client ACL, using the following AP commands:

- **[no] debug flexconnect access-list counter [all | vlan-acl | client-acl]**
- **[no] debug flexconnect access-list event [all | vlan-acl | client-acl]**
- To clear ACL counters use the following command:
  - **clear counters access-list client** <*MAC*> **all**

AP Client ACL Counter is supported in the FlexConnect mode and local switching central authentication sub-mode.