Controller Self-Signed Certificate for Wireless AP Join

Use Cases

Use Case-1

Cisco Catalyst 9800-CL platform does not contain manufacturer installed SUDI certificates. You will need to configure Self-Signed Certificates on your controller.

Use Case-2

APs running on earlier versions and having Manufacturer Installed Certificate (MIC) issued by a SHA1 Cisco Trusted CA cannot join the controller with SHA2 SUDI certificate. During CAPWAP join process, the AP displays a bad certificate error and tears down the DTLS handshake.

Workaround: To upgrade APs, configure controller Self-Signed certificates. Once done, you can delete the Self-Signed certificates and revert back to the SUDI certificate.

Note This workaround does not apply to the Embedded Wireless Controller running Catalyst 9k switches. But applies to other hardware appliance controllers, such as Cisco Catalyst 9800-40, Cisco Catalyst 9800-80, and Cisco Catalyst 9800-L.

Note Certificate used in DTLS connections (AP and mobility) must use RSA key of size equal or more than 2048 bits. Otherwise, the APs and mobility connections will fail after reload. Run the **show crypto pki certificate verbose _tp-name_** command to display the key size of the device certificate.

Prerequisites

- Ensure that the VLAN interface is up and it's IP is reachable.
- Ensure that the **ip http server** is enabled. For more information, see Enabling HTTP Server.
- Set the **clock calendar-valid** command appropriately. For more information, see #unique_1539.
- Check if the PKI CA server is already configured or not. If configured, you will need to delete the existing CA server configuration.

Note The **show crypto pki server** command output should not display anything.

Configuring Clock Calendar (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**clock calendar-valid** **Example:** `Device(config)# clock calendar-valid`	Enables clock calendar.
Step 3	**exit** **Example:** `Device(config)# exit`	Exits configuration mode.

Enabling HTTP Server (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**ip http server** **Example:** `Device(config)# ip http server`	Enables the HTTP server on your IP or IPv6 system, including a Cisco web browser user interface. By default, the HTTP server uses the standard port 80.
Step 3	**ip http secure-server** **Example:** `Device(config)# ip http secure-server`	Enables the HTTP server on your IP or IPv6 system, including a Cisco web browser user interface. By default, the HTTP server uses the standard port 80.
Step 4	**exit** **Example:** `Device(config)# exit`	Exits configuration mode.

Configuring CA Server (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**crypto key generate rsa general-keys modulus** *size_of_key_module* **label** *keypair_name* **Example:** `Device(config)# crypto key generate rsa general-keys modulus 2048 label WLC_CA`	Configures a certificate for the controller. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use. **Note** The recommended key-pair name is *WLC_CA* and key modulus is *2048* bits.
Step 3	**crypto pki server** *certificate_server_name*	Enables IOS certificate server.

	Command or Action	Purpose
	Example: `Device(config)# crypto pki server WLC_CA`	**Note** The *certificate_server_name* must be the same name as the *keypair_name*.
Step 4	**issuer-name** **Example:** `Device(config)# issuer-name O=Cisco Virtual Wireless LAN Controller, CN=CA-vWLC`	Configures X.509 distinguished name for the issuer CA certificate. **Note** You need to configure the same **issuer-name** as suggested for AP join.
Step 5	**grant auto** **Example:** `Device(config)# grant auto`	Grants certificate requests automatically.
Step 6	**hash sha256** **Example:** `Device(config)# hash sha256`	(Optional) Specifies the hash function for the signature used in the granted certificates.
Step 7	**lifetime ca-certificate** *time-interval* **Example:** `Device(config)# lifetime ca-certificate 3650`	(Optional) Specifies the lifetime in days of a CA certificate.
Step 8	**lifetime certificate** *time-interval* **Example:** `Device(config)# lifetime certificate 3650`	(Optional) Specifies the lifetime in days of a granted certificate.
Step 9	**database archive pkcs12 password** *password* **Example:** `Device(config)# database archive pkcs12 password 0 cisco123`	Sets the CA key and CA certificate archive format and password to encrypt the file.
Step 10	**no shutdown** **Example:** `Device(config)# no shutdown`	Enables the certificate server. **Note** Issue this command only after you have completely configured your certificate server.
Step 11	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configuring Trustpoint (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**crypto key generate rsa exportable general-keys modulus** *size-of-the-key-modulus* **label** *label* **Example:** `Device(config)# crypto key generate rsa exportable general-keys modulus 2048 label ewlc-tp1`	When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.
Step 3	**crypto pki trustpoint** *trustpoint_name* **Example:** `Device(config)# crypto pki trustpoint ewlc-tp1`	Creates a new trust point for an external CA server. Here, *trustpoint_name* refers to the trustpoint name. **Note** Ensure that same names are used for key-pair (*label*) and *trustpoint_name*.
Step 4	**rsakeypair** *RSA_key key_size* **Example:** `Device(ca-trustpoint)# rsakeypair ewlc-tp1`	Maps RSA key with that of the trustpoint. • RSA_key—Refers to the RSA key pair label. • key_size—Refers to the signature key length. The value ranges from 360 to 4096.
Step 5	**subject-name** *subject_name* **Example:** `Device(ca-trustpoint)# subject-name O=Cisco Virtual Wireless LAN Controller, CN=DEVICE-vWLC`	Creates subject name parameters for the trustpoint.
Step 6	**revocation-check none** **Example:** `Device(ca-trustpoint)# revocation-check none`	Checks revocation.
Step 7	**hash sha256** **Example:** `Device(ca-trustpoint)# hash sha256`	Specifies the hash algorithm.

	Command or Action	Purpose
Step 8	**serial-number** **Example:** `Device(ca-trustpoint)# serial-number`	Specifies the serial number.
Step 9	**eku request server-auth client-auth** **Example:** `Device(ca-trustpoint)# eku request server-auth client-auth`	(Optional) Sets certificate key-usage purpose.
Step 10	**password** *password* **Example:** `Device(config)# password 0 cisco123`	Enables password.
Step 11	**enrollment url** *url* **Example:** `Device(config)# enrollment url http://<management-IPv4>:80`	Enrolls the URL. **Note** Replace the dummy IP with management VLAN interface IP of the controller where CA server is configured.
Step 12	**exit** **Example:** `Device(config)# exit`	Exits the configuration.

Authenticating and Enrolling the PKI TrustPoint with CA Server (CLI)

Procedure

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

Step 2	**crypto pki authenticate** *trustpoint_name*
	Example:
	`Device(config)# crypto pki authenticate ewlc-tp1`
	`Certificate has the following attributes:`
	`Fingerprint MD5: 64C5FC9A C581D827 C25FC3CF 1A7F42AC`
	`Fingerprint SHA1: 6FAFF812 7C552783 6A8FB566 52D95849 CC2FC050`
	`% Do you accept this certificate?`
		Fetches the CA certificate.

	Command or Action	Purpose
	`[yes/no]: yes` `Trustpoint CA certificate accepted.`	

Step 3	**crypto pki enroll** *trustpoint_name*
	Example:
	`Device(config)# crypto pki enroll ewlc-tp1`
	`Enter following answers for UI interaction:`
	`% Include an IP address in the subject name? [no]: no`
	`Request certificate from CA? [yes/no]: yes`
		Enrolls for client certificate.

Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Tagging Wireless Management TrustPoint Name (CLI)

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**wireless management trustpoint** *trustpoint_name* **Example:** `Device(config)# wireless management trustpoint ewlc-tp1`	Tags the wireless management trustpoint name.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verifying Controller Certificates for Wireless AP Join

To view the CA server details, use the following command:

```
Device# show crypto pki server
Certificate Server WLC_CA:
Status: enabled
```

```
State: enabled
Server's configuration is locked (enter "shut" to unlock it)
Issuer name: O=Cisco Virtual Wireless LAN Controller, CN=CA-vWLC
CA cert fingerprint: 79A3DBD5 59A7E384 73ABD152 C133F4E2
Granting mode is: auto
Last certificate issued serial number (hex): 1
CA certificate expiration timer: 12:04:00 UTC Mar 8 2029
CRL NextUpdate timer: 18:04:00 UTC Mar 11 2019
Current primary storage dir: nvram:
Database Level: Minimum - no cert data written to storage
```

To view the trustpoint details, use the following command:

```
Device# show crypto pki trustpoint ewlc-tp1 status
Trustpoint ewlc-tp1:
...
State:
Keys generated ..... Yes (General Purpose, exportable)
Issuing CA authenticated ..... Yes
Certificate request(s) ..... Yes
```

To view the wireless management trustpoint details, use the following command:

```
Device# do show wireless management trustpoint
Trustpoint Name : ewlc-tp1
Certificate Info : Available
Certificate Type : SSC
Certificate Hash : 4a5d777c5b2071c17faef376febc08398702184e
Private key Info : Available
FIPS suitability : Not Applicable
```

To view the HTTP server status, use the following command:

```
Device# show ip http server status | include server status
HTTP server status: Enabled
HTTP secure server status: Enabled
```