

CloudSec Encryption

- Cisco ACI CloudSec Encryption, on page 1
- Requirements and Guidelines, on page 2
- CloudSec Encryption Terminology, on page 4
- CloudSec Encryption and Decryption Handling, on page 5
- CloudSec Encryption Key Allocation and Distribution, on page 6
- Configuring Cisco APIC for CloudSec Encryption, on page 8
- Enabling CloudSec Encryption Using Cisco ACI Multi-Site Orchestrator GUI, on page 11
- Rekey Process During Spine Switch Maintenance, on page 12

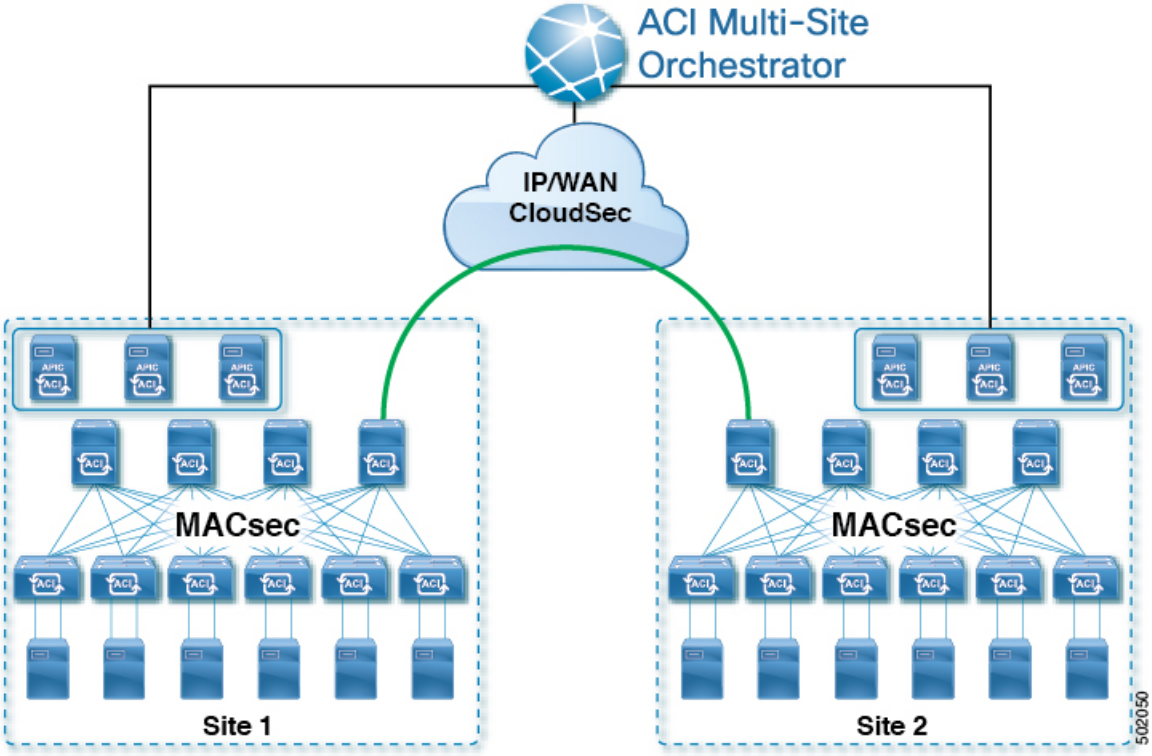
Cisco ACI CloudSec Encryption

As most Cisco ACI deployments are adopting the Cisco ACI Multi-Site architecture to address disaster recovery and scale, the current security implementation using MACsec encryption within local site is becoming insufficient to guarantee data security and integrity across multiple sites connected by insecure external IP networks interconnecting separate fabrics. Cisco ACI Multi-Site Orchestrator Release 2.0(1) introduces the CloudSec Encryption feature designed to provide inter-site encryption of traffic.

Cisco ACI Multi-Site topology uses three tunnel end-point (TEP) IP addresses to provide connectivity between sites. These TEP addresses are configured by the admin on Cisco ACI Multi-Site Orchestrator and pushed down to each site's Cisco APIC, which in turn configures them on the spine switches. These three addresses are used to determine when traffic is destined for a remote site, in which case an encrypted CloudSec tunnel is created between the two spine switches that provide physical connectivity between the two sites through the Inter-Site Network (ISN).

The following figure illustrates the overall encryption approach that combines MACsec for local site traffic and CloudSec for inter-site traffic encryption.

Figure 1: CloudSec Encryption



Requirements and Guidelines

When configuring CloudSec encryption, the following guidelines apply:

- CloudSec has been validated using a Nexus 9000 Inter-Site Network (ISN) infrastructure. If your ISN infrastructure is made up of different devices, or the devices are unknown (such as in the case of circuits purchased from a service provider), it is required that an ASR1K router is the first hop device directly connected to the ACI spine, or the Nexus 9000 ISN network. The ASR1K router with padding-fixup enabled allows the CloudSec traffic to traverse any IP network between the sites.

  To configure an ASR1K router:

  1. Log in to the device.
  2. Configure the UDP ports.

     ```
     ASR1K(config)# platform cloudsec padding-fixup dst-udp-port 9999
     ```

  3. Verify the configuration.

     ```
     ASR1K# show platform software ip rp active cloudsec
       CloudSec Debug: disabled
       CloudSec UDP destination port: enabled
       1st UDP destination port: 9999
       2nd UDP destination port: 0
       3rd UDP destination port: 0

     ASR1K# show platform software ip fp active cloudsec
     ```

```
CloudSec Debug: disabled
CloudSec UDP destination port: enabled
1st UDP destination port: 9999
2nd UDP destination port: 0
3rd UDP destination port: 0
```

- If one or more spine switches are down when you attempt to disable CloudSec encryption, the disable process will not complete on those switches until the switches are up. This may result in packet drops on the switches when they come back up.

  We recommend you ensure that all spine switches in the fabric are up or completely decommissioned before enabling or disabling CloudSec encryption.

- The CloudSec Encryption feature is not supported with the following features:
  - Precision Time Protocol (PTP)
  - Remote Leaf Direct
  - Virtual Pod (vPOD)
  - SDA
  - Intersite L3Out
  - Other routable TEP configurations

Requirements

The CloudSec encryption capability requires the following:

- Cisco ACI spine-leaf architecture with a Cisco APIC cluster for each site
- Cisco ACI Multi-Site Orchestrator to manage each site
- One **Advantage** or **Premier** license per each device (leaf only) in the fabric
- An add-on license **ACI-SEC-XF** per device for encryption if the device is a fixed spine
- An add-on license **ACI-SEC-XM** per device for encryption if the device is a modular spine

The following table provides the hardware platforms and the port ranges that are capable of CloudSec encryption.

Hardware Platform	Port Range
N9K-C9364C spine switches	Ports 49-64
N9K-C9332C spine switches	Ports 25-32
N9K-X9736C-FX line cards	Ports 29-36

If CloudSec is enabled for a site, but the encryption is not supported by the ports, a fault is raised with `unsupported-interface` error message.

CloudSec encryption's packet encapsulation is supported if Cisco QSFP-to-SFP Adapters (QSA), such as CVR-QSFP-SFP10G, is used with a supported optic. The full list of supported optics is available from the

following link: https://www.cisco.com/c/en/us/support/interfaces-modules/transceiver-modules/products-device-support-tables-list.html.

CloudSec Encryption Terminology

CloudSec Encryption feature provides a secure upstream symmetric key allocation and distribution method for initial key and rekey requirements between sites. The following terminology is used in this chapter:

- `Upstream device` – The device that adds the CloudSec Encryption header and does the encryption of the VXLAN packet payload on transmission to a remote site using a locally generated symmetric cryptography key.
- `Downstream device` – The device that interprets the CloudSec Encryption header and does the decryption of the VXLAN packet payload on reception using the cryptography key generated by the remote site.
- `Upstream site` – The datacenter fabric that originates the encrypted VXLAN packets.
- `Downstream site` – The datacenter fabric that receives the encrypted packets and decrypts them.
- `TX Key` – The cryptography key used to encrypt the clear VXLAN packet payload. In ACI only one TX key can be active for all the remote sites.
- `RX Key` – The cryptography key used to decrypt the encrypted VXLAN packet payload. In ACI two RX keys can be active per remote site.

  Two RX keys can be active at the same time because during the rekey process, the downstream sites will keep the old and the new RX keys after the new key deployment is finished for some duration to ensure that out of order packet deliveries with either key can be properly decrypted.
- `Symmetric Keys` – When the same cryptography key is used to encrypt (`TX Key`) and decrypt (`RX Key`) a packet stream by the upstream and downstream devices respectively.
- `Rekey` – The process initiated by the upstream site to replace its old key with a newer key for all downstream sites after the old key expires.
- `Secure Channel Identifier (SCI)` – A 64-bit identifier that represents a security association between the sites. It is transmitted in encrypted packet in CloudSec header and is used to derive the RX key on the downstream device for packet decryption.
- `Association Number (AN)` – A 2-bit number (`0, 1, 2, 3`) that is sent in the CloudSec header of the encrypted packet and is used to derive the key at the downstream device in conjunction with the SCI for decryption. This allows multiple keys to be active at the downstream device to handle out of order packet arrivals with different keys from the same upstream device following a rekey operation.

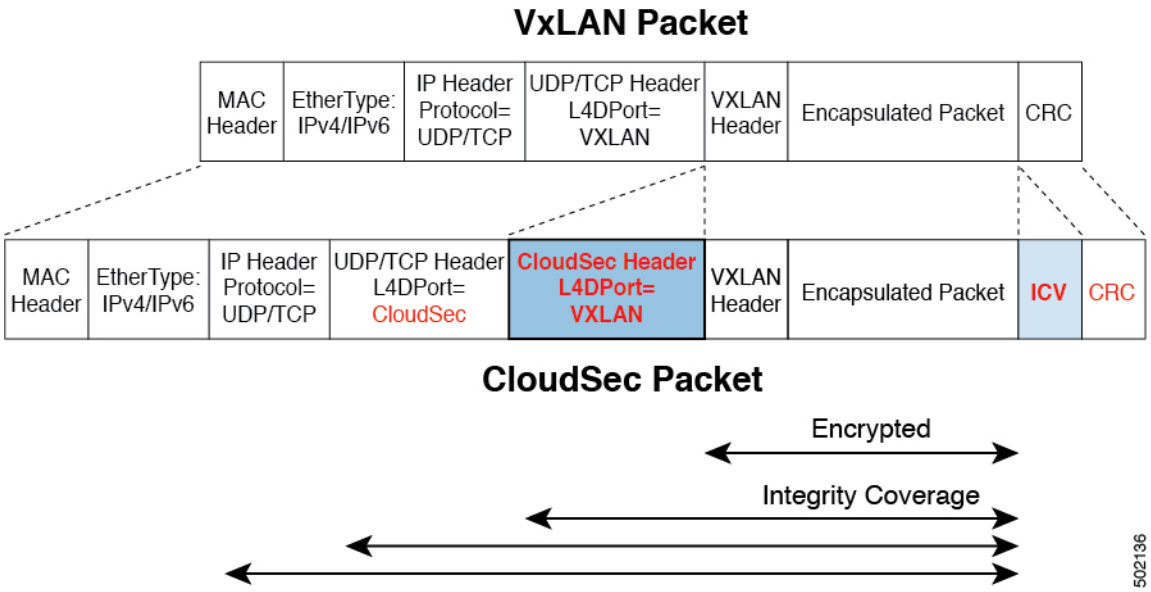
  In ACI only two association number values (`0` and `1`) are used for the two active RX keys and only one association number value (`0` or `1`) is used for the TX Key at any point in time.
- `Pre-shared key (PSK)` – One ore more keys must be configured in the Cisco APIC GUI to be used as a random seed for generating the CloudSec TX and RX keys. If multiple PSK are configured, each rekey process will use the next PSK in order of their indexes; if no higher index PSK is available, a PSK with the lowest index will be used. Each PSK must be a hexadecimal string 64 characters long. Cisco APIC supports up to 256 pre-shared keys.

CloudSec Encryption and Decryption Handling

In order to provide a fully integrated, simple, and cost-effective solution that addresses both, data security and integrity, starting with Release 2.0(1), Cisco ACI Multi-Site provides a CloudSec Encryption feature that allows for complete source-to-destination packet encryption between Multi-Site fabrics.

The following figure shows packet diagram before and after CloudSec encapsulation, followed by descriptions of the encryption and decryption processes:

Figure 2: CloudSec Packet



Packet Encryption

The following is a high level overview of how CloudSec handles outgoing traffic packets:

- The packets are filtered using the outer IP header and Layer-4 destination port information and matching packets are marked for encryption.
- The offset to use for encryption is calculated according to the fields of the packet. For example, the offset may vary based on whether there is a 802.1q VLAN or if the packet is an IPv4 or IPv6 packet.
- The encryption keys are programmed in the hardware tables and are looked up from the table using the packet IP header.

Once the packet is marked for encryption, the encryption key is loaded, and the offset from the beginning of the packet where to start the encryption is known, the following additional steps are taken:

- The UDP destination port number is copied from the UDP header into a CloudSec field for recovery when the packet is decrypted.
- The UDP destination port number is overwritten with a Cisco proprietary Layer-4 port number (Port `9999`) indicating that it is a CloudSec packet.
- The UDP length field is updated to reflect the additional bytes that are being added.

- The CloudSec header is inserted directly after the UDP header.
- The Integrity Check Value (ICV) is inserted at the end of the packet, between the payload and the CRC.
- The ICV requires construction of a 128-bit initialization vector. For CloudSec, any use of the source MAC address for ICV purposes is replaced by a programmable value per SCI.
- CRC is updated to reflect the change in the contents of the packet.

Packet Decryption

The way CloudSec handles incoming packets is symmetric to the outgoing packets algorithm described above:

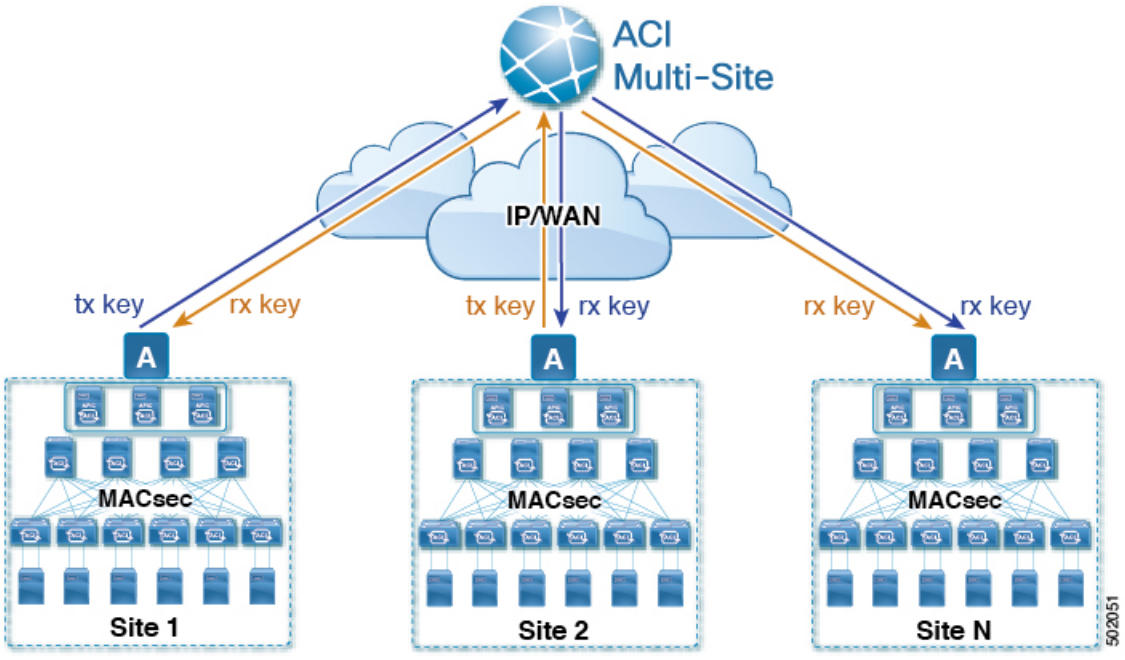
- If the received packet is a CloudSec packet, it is decrypted and the ICV is verified.

  If ICV verification passed, the extra fields are removed, the UDP destination port number is moved from the CloudSec header to the UDP header, the CRC is updated, and the packet is forwarded to destination after decryption and CloudSec header removal. Otherwise the packet is dropped.
- If the key store returns two or more possible decryption keys, the Association Number (AN) field of the CloudSec header is used to select which key to use.
- If the packet is not a CloudSec packet, the packet is left unchanged.

CloudSec Encryption Key Allocation and Distribution

Initial Key Configuration

Figure 3: CloudSec Key Distribution



The following is a high level overview of the CloudSec encryption key initial allocation and distribution process illustrated by the figure above:

- The upstream site's Cisco APIC generates a local symmetric key intended to be used for data encryption of VXLAN packets transmitted from its site. The same key that is used by the upstream site for encryption is used for decryption of the packets on the downstream remote receiving sites.

  Every site is an upstream site for the traffic it transmits to other sites. If multiple sites exist, each site generates its own site-to-site key and use that key for encryption before transmitting to the remote site.
- The generated symmetric key is pushed to the Cisco ACI Multi-Site Orchestrator (MSO) by the upstream site's Cisco APIC for distribution to downstream remote sites.
- The MSO acts as a message broker and collects the generated symmetric key from the upstream site's Cisco APIC, then distributes it to downstream remote sites' Cisco APICs.
- Each downstream site's Cisco APIC configures the received key as RX key on the local spine switches which are intended to receive the traffic from the upstream site that generated the key.
- Each downstream site's Cisco APIC also collects the deployment status of the RX Key from the local spine switches and then pushes it to the MSO.
- The MSO relays the key deployment status from all downstream remote sites back to the upstream site's Cisco APIC.
- The upstream site's Cisco APIC checks if the key deployment status received from all downstream remote sites is successful.
  - If the deployment status received from a downstream device is successful, the upstream site deploys the local symmetric key as its TX key on the spine switches to enable encryption of the VXLAN packets that are sent to the downstream site.
  - If the deployment status received from a downstream device is failed, a fault is raised on the Cisco APIC site where it failed and it is handled based on the "secure mode" setting configured on the MSO. In "must secure" mode the packets are dropped and in the "should secure" mode the packets are sent clear (unencrypted) to the destination site.



Note In current release, the mode is always set to “should secure” and cannot be changed.

Rekey Process

Each generated TX/RX key expires after a set amount of time, by default key expiry time is set to 15 minutes. When the initial set of TX/RX keys expires, a rekey process takes place.

The same general key allocation and distribution flow applies for the rekey process. The rekey process follows the "make before break" rule, in other words all the RX keys on the downstream sites are deployed before the new TX key is deployed on the upstream site. To achieve that, the upstream site will wait for the new RX key deployment status from the downstream sites before it configures the new TX key on the local upstream site's devices.

If any downstream site reports a failure status in deploying the new RX key, the rekey process will be terminated and the old key will remain active. The downstream sites will also keep the old and the new RX keys after

the new key deployment is finished for some duration to ensure that out of order packet deliveries with either key can be properly decrypted.

Note Special precautions must be taken in regards to rekey process during spine switch maintenance, see Rekey Process During Spine Switch Maintenance, on page 12 for details.

Rekey Process Failure

In case of any downstream site failing to deploy the new encryption key generated by the rekey process, the new key is discarded and the upstream device will continue to use the previous valid key as TX key. This approach keeps the upstream sites from having to maintain multiple TX keys per set of downstream sites. However, this approach may also result in the rekey process being delayed if the rekey deployment failures continue to occur with any one of the downstream sites. It is expected that the Multi-Site administrator will take action to fix the issue of the key deployment failure for the rekey to succeed.

Cisco APIC's Role in Key Management

The Cisco APIC is responsible for key allocation (both, initial key and rekey distribution), collection of the key deployment status messages from the spine switches, and notification of the Cisco ACI Multi-Site Orchestrator about each key's status for distribution to other sites.

Cisco ACI Multi-Site Orchestrator's Role in Key Management

The Cisco ACI Multi-Site Orchestrator is responsible for collecting the TX keys (both, initial key and subsequent rekeys) from the upstream site and distributing it to all downstream sites for deployment as RX keys. The MSO also collects the RX key deployment status information from the downstream sites and notifies the upstream site in order for it to update the TX key on successful RX key deployment status.

Upstream Model

In contrast to other technologies, such as MPLS, that use downstream key allocation, CloudSec's upstream model provides the following advantages:

- The model is simple and operationally easier to deploy in the networks.
- The model is preferred for Cisco ACI Multi-Site use cases.
- It provides advantages for multicast traffic as it can use the same key and CloudSec header for each copy of the replicated packet transmitted to multiple destination sites. In downstream model each copy would have to use a different security key for each site during encryption.
- It provides easier troubleshooting in case of failures and better traceability of packets from the source to destination consistently for both, unicast and multicast replicated packets.

Configuring Cisco APIC for CloudSec Encryption

You must configure one or more Pre-Shared Keys (PSK) to be used by the Cisco APIC for generating the CloudSec encryption and decryption keys. The PSK are used as a random seed during the re-key process. If multiple PSK are configured, each re-key process will use the next PSK in order of their indexes; if no higher index PSK is available, a PSK with the lowest index will be used.

Because PSK is used as a seed for encryption key generation, configuring multiple PSK provides additional security by lowering the over-time vulnerability of the generated encryption keys.

Note If no pre-shared key is configured on the Cisco APIC, CloudSec will not be enabled for that site. In that case, turning on CloudSec setting in Cisco ACI Multi-Site will raise a fault.

If at any time you wish to refresh a previously added PSK with a new one, simply repeat the procedure as if you were adding a new key, but specify an existing index.

You can configure one or more pre-shared keys in one of three ways:

- Using the Cisco APIC GUI, as described in Configuring Cisco APIC for CloudSec Encryption Using GUI, on page 9
- Using the Cisco APIC NX-OS Style CLI, as described in Configuring Cisco APIC for CloudSec Encryption Using NX-OS Style CLI, on page 9
- Using the Cisco APIC REST API, as described in Configuring Cisco APIC for CloudSec Encryption Using REST API, on page 10

Configuring Cisco APIC for CloudSec Encryption Using GUI

This section describes how to configure one or more pre-shared keys (PSK) using the Cisco APIC GUI.

Procedure

Step 1 Log in to APIC.

Step 2 Navigate to **Tenants** > **infra** > **Policies** > **CloudSec Encryption**

Step 3 Specify the **SA Key Expiry Time**.

This option specifies how long each key is valid (in minutes). Each generated TX/RX key expires after the specified amount of time triggering a re-key process. The expiration time can be between 5 and 1440 minutes.

Step 4 Click the + icon in the **Pre-Shared Keys** table.

Step 5 Specify the **Index** of the pre-shared key you are adding and then the **Pre-Shared Key** itself.

The **Index** field specifies the order in which the pre-shared keys are used. After the last (highest index) key is used, the process will continue with the first (lowest index) key. Cisco APIC supports up to 256 pre-shared keys, so the PSK index value must be between 1 and 256.

Each **Pre-Shared Key** must be a hexadecimal string 64 characters long.

Configuring Cisco APIC for CloudSec Encryption Using NX-OS Style CLI

This section describes how to configure one or more pre-shared keys (PSK) using the Cisco APIC NX-OS Style CLI.

Procedure

Step 1 Log in to the Cisco APIC NX-OS style CLI.

Step 2 Enter configuration mode.

Example:

```
apic1# configure
apic1 (config)#
```

Step 3 Enter configuration mode for the default CloudSec profile.

Example:

```
apic1(config)# template cloudsec default
apic1(config-cloudsec)#
```

Step 4 Specify the Pre-Shared Keys (PSK) expiration time.

This option specifies how long each key is valid (in minutes). Each generated TX/RX key expires after the specified amount of time triggering a re-key process. The expiration time can be between 5 and 1440 minutes.

Example:

```
apic1(config-cloudsec)# sakexpirytime <duration>
```

Step 5 Specify one or more Pre-Shared Keys.

In the following command, specify the index of the PSK you're configuring and the PSK string itself.

Example:

```
apic1(config-cloudsec)# pskindex <psk-index>
apic1(config-cloudsec)# pskstring <psk-string>
```

The <*psk-index*> parameter specifies the order in which the pre-shared keys are used. After the last (highest index) key is used, the process will continue with the first (lowest index) key. Cisco APIC supports up to 256 pre-shared keys, so the PSK index value must be between 1 and 256.

The <*psk-string*> parameter specifies the actual PSK, which must be a hexadecimal string 64 characters long.

Step 6 (Optional) View the current PSK configuration.

You can view how many PSK are currently configured and their duration using the following command:

Example:

```
apic1(config-cloudsec)# show cloudsec summary
```

Configuring Cisco APIC for CloudSec Encryption Using REST API

This section describes how to configure one or more pre-shared keys (PSK) using the Cisco APIC REST API.

Procedure

Configure PSK expiration time, index, and string.

In the following XML POST, replace:

- The value of **sakExpiryTime** with the expiration time of each PSK.

  This **sakExpiryTime** parameter specifies how long each key is valid (in minutes). Each generated TX/RX key expires after the specified amount of time triggering a re-key process. The expiration time can be between 5 and 1440 minutes.

- The value of **index** with the index of the PSK you're configuring.

  The **index** parameter specifies the order in which the pre-shared keys are used. After the last (highest index) key is used, the process will continue with the first (lowest index) key. Cisco APIC supports up to 256 pre-shared keys, so the PSK index value must be between 1 and 256.

- The value of **pskString** with the index of the PSK you're configuring.

  The **pskString** parameter specifies the actual PSK, which must be a hexadecimal string 64 characters long.

Example:

```
<fvTenant annotation="" descr="" dn="uni/tn-infra" name="infra" nameAlias="" ownerKey="" ownerTag="">

   <cloudsecIfPol descr="cloudsecifp" name="default" sakExpiryTime="10" stopRekey= "false" status=""
 >
        <cloudsecPreSharedKey index="1"
pskString="1234567812345678123456781234567812345678123456781234567812345678" status=""/>
    </cloudsecIfPol>
</fvTenant>
```

Enabling CloudSec Encryption Using Cisco ACI Multi-Site Orchestrator GUI

The CloudSec encryption can be enabled or disabled for each site individually. However, the communications between two sites will be encrypted only if the feature is enabled on both sites.

Before you begin

Before you enable the CloudSec encryption between two or more sites, you must have completed the following tasks:

- Installed and configured the Cisco APIC clusters in multiple sites, as described in *Cisco APIC Installation, Upgrade, and Downgrade Guide*
- Installed and configured Cisco ACI Multi-Site Orchestrator, as described in *Cisco ACI Multi-Site Orchestrator Installation and Upgrade Guide*.
- Added each Cisco APIC site to the Cisco ACI Multi-Site Orchestrator, as described in *Cisco ACI Multi-Site Configuration Guide*.

Procedure

Step 1 Log in to the Cisco ACI Multi-Site Orchestrator.

Step 2 From the left-hand sidebar, select the **Sites** view.

Step 3 Click on the **Configure Infra** button in the top right of the main window.

Step 4 From the left-hand sidebar, select the site for which you want to change the CloudSec configuration.

Step 5 In the right-hand sidebar, toggle the **CloudSec Encryption** setting to enable or disable the CloudSec Encryption feature for the site.

Rekey Process During Spine Switch Maintenance

The following is a summary of the CloudSec rekey process during typical maintenance scenarios for the spine switches where the feature is enabled:

- **Normal Decommissioning** – CloudSec rekey process stops automatically whenever a CloudSec-enabled spine switch is decommissioned. Rekey process will not start again until the decommissioned node is commissioned back or the decommissioned node ID is removed from the Cisco APIC
- **Spine Switch Software Upgrade** – CloudSec rekey process stops automatically if a spine switch is reloaded due to software upgrade. Rekey process will resume after the spine switch comes out of reload.
- **Maintenance (GIR mode)** – CloudSec rekey process must be manually stopped using the instructions provided in Disabling and Re-Enabling Re-Key Process Using NX-OS Style CLI, on page 12. Rekey can be enabled back only after the node is ready to forward traffic again.
- **Decommissioning and Removal from Cisco APIC** – CloudSec rekey process must be manually stopped using the instructions provided in Disabling and Re-Enabling Re-Key Process Using NX-OS Style CLI, on page 12. Rekey can be enabled back only after the node is removed from Cisco APIC.

Disabling and Re-Enabling Re-Key Process Using NX-OS Style CLI

It is possible to manually stop and restart the re-key process. You may be required to manually control the re-key process in certain situations, such as switch decommissioning and maintenance. This section describes how to toggle the setting using Cisco APIC NX-OS Style CLI.

Procedure

Step 1 Log in to the Cisco APIC NX-OS style CLI.

Step 2 Enter configuration mode.

Example:

```
apic1# configure
apic1(config)#
```

Step 3 Enter configuration mode for the default CloudSec profile.

Example:

```
apic1(config)# template cloudsec default
apic1(config-cloudsec)#
```

Step 4 Stop or restart the re-key process.

To stop the re-key process:

Example:

```
apic1(config-cloudsec)# stoprekey yes
```

To restart the re-key process:

Example:

```
apic1(config-cloudsec)# stoprekey no
```

Disabling and Re-Enabling Re-Key Process Using REST API

It is possible to manually stop and restart the re-key process. You may be required to manually control the re-key process in certain situations, such as switch decommissioning and maintenance. This section describes how to toggle the setting using Cisco APIC REST API.

Procedure

Step 1 You can disable the rekey process using the following XML message.

Example:

```
<fvTenant annotation="" descr="" dn="uni/tn-infra" name="infra" nameAlias="" ownerKey="" ownerTag="">

    <cloudsecIfPol descr="cloudsecifp" name="default" sakExpiryTime="10" stopRekey= "true" status=""
 />
</fvTenant>
```

Step 2 You can enable the rekey process using the following XML message.

Example:

```
<fvTenant annotation="" descr="" dn="uni/tn-infra" name="infra" nameAlias="" ownerKey="" ownerTag="">

   <cloudsecIfPol descr="cloudsecifp" name="default" sakExpiryTime="10" stopRekey= "false" status=""
 />
</fvTenant>
```