Fundamentals of MACsec encryption

This chapter provides a comprehensive overview of MACsec encryption fundamentals, including key concepts, deployment models, configuration steps, and verification procedures. Users can leverage this chapter to understand MACsec benefits, set up secure Layer 2 encryption on their routers, and follow best practices for configuration and key management.

MACsec encryption

MACsec encryption is a Layer 2 security technology that

- protects data on physical media from common attacks such as MAC address spoofing, ARP spoofing, Denial of Service (DoS) attacks targeting DHCP servers, and VLAN hopping
- provides data confidentiality and integrity by encrypting traffic at the physical layer,
- precedence over higher-layer encryption methods such as IPsec and SSL, and
- deploys on Customer Edge (CE) router interfaces that connect to Provider Edge (PE) routers and on all provider router interfaces.

Benefits of MACsec encryption

- Data integrity check: Uses an Integrity Check Value (ICV) sent with the protected data unit. The receiver recalculates and compares the ICV to detect any data modification.
- Data encryption: Enables a port to encrypt outbound frames and decrypt inbound frames encrypted with MACsec.
- Replay protection: Provides a configurable window that accepts a specified number of out-of-sequence frames to handle frames transmitted out of order.
- Support for clear traffic: Allows unencrypted data to transit through the port if configured accordingly.

Hardware support for MACsec encryption

The table lists the compatibility between specific Cisco IOS XR Software Releases and the corresponding hardware Product IDs (PIDs) that support MACsec encryption.

Table 1: Hardware support for MACsec encryption

Cisco IOS XR Software Release	Product ID (PID)
Release 25.1.1	8712-MOD-M
Release 24.4.1	8711-32FH-M
Release 24.3.1	• 8212-48FH-M • 88-LC1-52Y8H-EM
Release 7.10.1	Cisco 8608: • 86-MPA-14H2FH-M • 86-MPA-4FH-M • 86-MPA-24Z-M
Release 7.5.2	8202-32FH-M
Release 7.3.3	88-LC0-34H14FH
Release 7.3.15	88-LC0-36FH-M
Release 7.0.12	8800-LC-48H

MACsec encryption by interface type

- Physical interfaces (Standard MACsec): Applies security directly to a physical Ethernet port. This provides standard link-layer security within a LAN or between directly connected devices.
- L3 subinterfaces (WAN MACsec): Designed for service provider networks. It preserves the provider's outer VLAN tag in clear text while encrypting the customer's data payload. This allows the provider's network to switch traffic correctly and ensures end-to-end security.

Both physical interfaces and L3 subinterfaces support point-to-point (P2P) and point-to-multipoint (P2MP) MACsec encryption deployment models.

MACsec encryption deployment models

MACsec encryption supports two primary deployment models:

1. Point-to-Point (P2P): Secures a direct link between two endpoints.
2. Point-to-Multipoint (P2MP): Enables a single device to establish secure communications with multiple remote devices.

P2P MACsec encryption deployments

- LAN: Establishes secure Ethernet connectivity between two devices on the same local network.

- Over L2VPN (Pseudowire): Extends MACsec protection across a service provider network by encapsulating encrypted traffic over Layer 2 VPNs or pseudowires.

P2MP MACsec encryption deployments

- LAN: Establishes separate secure sessions from a central device to multiple peers on the same local network segment.
- Over VPLS (Virtual Private LAN Service):

  Establishes encrypted multipoint connectivity by creating a secure hub-and-spoke topology over a provider's VPLS network, connecting a central site with multiple branch locations.

P2P is suitable for securing direct links on LANs and across service provider networks. P2MP is ideal when a single device must securely communicate with multiple endpoints, especially in hub-and-spoke topologies over VPLS. Both deployment models, P2P and P2MP MACsec encryption, are supported on physical interfaces and L3 subinterfaces.

Key concepts for MACsec encryption

MACsec Key Agreement protocol

MACsec Key Agreement (MKA) is a protocol that manages the secure exchange of cryptographic keys for MACsec. It establishes and maintains secure associations between devices, enabling encrypted communication over Ethernet links. MKA handles key distribution, authentication, and rekeying processes to ensure continuous data confidentiality and integrity.

MACsec Pre-shared Key

MACsec Pre-shared Key (PSK) is a static key shared between devices before communication begins. It serves as a basis for authenticating devices and deriving session keys in MACsec. PSK simplifies deployment in environments where dynamic key management is not feasible but requires secure key distribution and management practices.

- Connectivity Association Key Name (CKN): CKN is an identifier used to associate devices within a MACsec connectivity association. It uniquely identifies the keying material group and helps devices recognize peers that share the same security context. CKN ensures that only authorized devices participate in the secure communication.
- Connectivity Association Key (CAK): CAK is the primary cryptographic key shared among devices in a MACsec connectivity association. It is used to derive session keys for encrypting and authenticating data frames. CAK must be securely distributed and protected to maintain the integrity and confidentiality of the MACsec session.

Fallback PSK and active fallback

Fallback PSK is a session recovery mechanism that activates when the primary PSK fails to establish a secured MKA session, ensuring a PSK is always available for MACsec encryption and decryption. Cisco IOS XR software enhances fallback PSK with the active fallback, which initiates a fallback MKA session when fallback configuration is present on the interface. Active fallback ensures faster session convergence on fallback during primary key deletion, expiry, or mismatch. It also accelerates traffic recovery under the should-secure security policy when both primary and fallback keys mismatch.

Secure Association Key

The actual encryption key that the key server generates and distributes to the key client. Each secure channel uses a new Secure Association Key (SAK) for data encryption.

- Key server: A router selected during the MKA process that is responsible for generating and distributing the SAK. Its selection is based on configured priority values, where a numerically lower value indicates higher preference.
- Key client: The peer router that receives the SAK from the key server.

MACsec frame format

The MACsec frame format defines the structure of a frame after Media Access Control Security (MACsec) encryption. It consists of specific components that ensure data confidentiality, integrity, and authenticity at Layer 2.

Figure 1: MACsec frame format

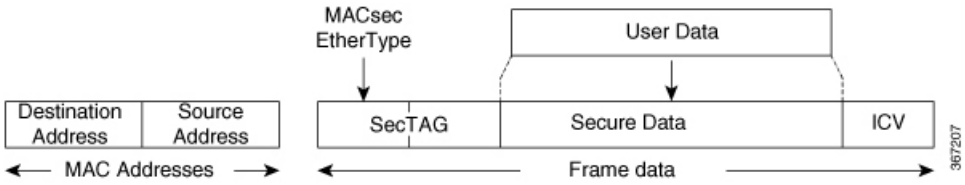


Table 2: MACsec frame components

MACsec frame component	What it is	Used for
SecTAG	A security tag, 8 to 16 bytes in length (16 bytes if Secure Channel Identifier (SCI) encoding is used, otherwise 8 bytes). It also provides replay protection.	Identifying the Secure Association Key (SAK) used for the frame and detecting out-of-sequence frames.
Secure Data	The portion of the frame containing data encrypted using MACsec, with a length of 2 or more octets.	Carrying encrypted data within the frame.
ICV (Integrity Check Value)	A value that provides an integrity check for the entire frame, typically ranging from 8 to 16 bytes in length.	Ensuring the integrity of the frame; frames with an ICV that does not match the expected value are dropped at the receiving port.

MACsec keychain

A MACsec keychain is a collection of cryptographic keys used to authenticate peers that need to exchange encrypted information. It defines the keys, their associated key strings (passwords), the cryptographic algorithm to be used, and the validity period for each key.

Table 3: MACsec keychain elements

MACsec keychain element	What it is	Used for
Key (CKN)	An identifier for the MACsec secret key.	Identifying each key entry in a MACsec keychain.
Key-string (CAK)	The actual secret key in the MACsec encryption.	Encrypting data based on the cryptographic algorithm used.
Cryptographic Algorithm	Specifies the encryption algorithm.	Determining how the key-string (CAK) is used for encryption.
Lifetime	Defines the validity period of the key, either as a duration or indefinitely.	Ensuring the key is used only within its valid time frame for security purposes.

MACsec policy

A MACsec policy defines the security parameters and behaviors for Media Access Control Security (MACsec) encryption in routers. It specifies the cryptographic algorithms, key management preferences, and traffic handling rules for secure Layer 2 communication.

MACsec policy encompasses several key parameters that govern MACsec operation:

Table 4: MACsec policy parameters

MACsec policy parameter	What it is	What it does
Cipher Suite	The encryption algorithm used for MACsec.	Provides the cryptographic strength and method for MACsec data encryption.
Confidentiality Offset	An offset value for MACsec encryption.	Modifies the starting point of encryption within a frame. Changes are recommended only when the port is administratively down to prevent traffic loss.
Key Server Priority	A value that determines a router's preference to be selected as the key server in an MKA session. A numerically lower value indicates higher preference.	Influences which router becomes the key server, responsible for generating and maintaining the Secure Association Key (SAK).
Security Policy	Defines the traffic handling behavior based on MACsec encryption status.	Controls whether unencrypted traffic is allowed before the MKA session secures, or if only encrypted traffic is permitted.
Data Delay Protection	A feature that ensures MACsec-protected data frames do not exceed a specific delay threshold.	Rejects MACsec-protected traffic that experiences excessive delay (over 2 seconds) to maintain real-time performance.
Replay Protection Window Size	The maximum number of out-of-sequence frames that are accepted.	Protects against replay attacks by defining the acceptable window for frame reordering.

MACsec policy parameter	What it is	What it does
Include ICV Indicator	A configuration option for including an optional Integrity Check Value (ICV) Indicator in the transmitted MACsec Key Agreement PDU (MKPDU).	Ensures interoperability with other vendor MACsec implementations that expect this specific indicator in the MKPDU.
SAK Rekey Interval	A timer value for periodically rekeying the MACsec Secure Association Key (SAK).	Periodically updates the data encryption key (SAK) to enhance security by limiting the lifespan of a single key. This configuration is effective on the node acting as the key server.

How MACsec encryption works

MACsec is a Layer 2 IEEE 802.1AE standard that secures data on physical media by encrypting packets between two MACsec routers.

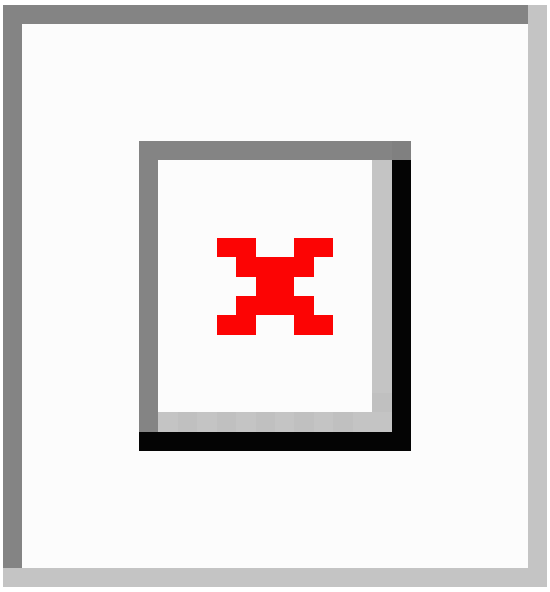
Summary

The key components involved in MACsec encryption are:

- MACsec routers: Devices that implement the MACsec standard to encrypt and decrypt traffic.
- MACsec Key Agreement (MKA) protocol: Manages the exchange of session keys and encryption keys.
- Pre-shared Key (PSK): A shared secret used for mutual peer authentication.
- Secure Association Key (SAK): The actual encryption key used for data encryption.
- MACsec frame format: The structure of encrypted packets, including SecTAG, Secure Data, and ICV.

Workflow

Figure 2: MACsec encryption process



These stages describe how MACsec encryption works:

1. Link establishment and peer authentication: When two MACsec routers first connect, they establish a peer relationship. Both devices perform mutual authentication using a pre-shared key (PSK).
2. Connectivity association formation: After successful peer authentication, the routers create a connectivity association. They exchange a secure connectivity association key name (CKN) and validate the media key agreement (MKA) integrity check value (ICV) using the connectivity association key (CAK).
3. Key server selection: The routers select a key server based on their configured priorities.

   Rules that apply to key server selection include:

   - Lower numerical values of key server priority and SCI receive the highest preference.
   - A lower priority value increases the preference for the router to become the key server, while the other router functions as a key client. If no value is configured, the default value of 16 is taken to be the key server priority value for the router.
   - Each router selects a peer advertising the highest preference as its key server if peer has not selected another router as its key server or is not willing to function as the key server.
   - If two routers tie for the highest preference, a router with the highest priority SCI becomes the key server (KS).

4. Security association and SAK distribution: The selected key server generates and distributes the secure association key (SAK). Each secure channel relies on a series of overlapping security associations (SA), with each SA utilizing a new SAK.
5. Encrypted data exchange: Once the routers distribute the SAKs and establish security associations, they begin exchanging encrypted data. The data frames include a MACsec header with a SecTAG (for SAK identification and replay protection), the secure data (the encrypted payload), and an ICV (for integrity checking). Once assembled, both devices transmit the encrypted data.

Result

The MACsec process secures data on physical media, making it impossible for data to be compromised at higher layers. It provides data integrity checks, data encryption, and replay protection. This enhances the overall security of the network.

Guidelines for MACsec encryption

To ensure secure and reliable MACsec encryption:

- Use strong keychains to protect MACsec credentials and keys.
- Safely manage fallback preshared keys (PSK) and configure active fallback settings to support redundancy.
- Consistently apply MACsec encryption configuration to all relevant interfaces to prevent security gaps.

Guidelines for configuring MACsec keychains

Follow these guidelines to effectively and securely manage MACsec keychains:

- Ensure that the MACsec Connectivity Association Key Name (CKN) and Connectivity Association Key (CAK) match exactly on both ends. If the CKN or CAK do not match, the MKA session cannot be established, resulting in failed secure communication.
- Use unique, case-insensitive key IDs for each MACsec key to prevent session instability. MACsec key IDs are case-insensitive and stored in uppercase (for example, 'FF' and 'ff' are treated the same), so duplicate IDs may cause session instability. This case insensitivity does not apply to Netconf protocol configurations.
- Use MACsec keys of even length, up to 64 characters. Odd-length keys cause the system to exit MACsec configuration mode, preventing key setup.
- Always use the latest key in the keychain for MKA protocol operations. The key with the most recent Start Time among active keys is automatically used. You can verify key details with the **show key chain** command.
- Activate new MACsec keys in advance to ensure at least a one-minute overlap with the current key, ensuring seamless CAK rollover and preventing session interruptions.
- Set Start and Expiry times with future timestamps to automate CAK rotation. Automating key rotation enables bulk configuration for daily CAK rotation without manual intervention, improving operational efficiency and security.
- Do not delete or allow the current active key to expire. Deleting or allowing the active key to expire will terminate the MKA session and disrupt traffic. To prevent service interruption, configure keys with an infinite lifetime. If fallback is enabled, traffic will continue by switching to the fallback key upon expiry or deletion of the primary active key.
- Monitor key status regularly and take action before a key expires.When a key expires, the MACsec session terminates and secure connectivity is lost. Use the following commands to check status:
  - **show macsec mka session**: Displays no session information if key expires.

- **show macsec mka interface detail**: Displays `*** No Active Keys Present ***` in the PSK information.

Guidelines for managing fallback PSK and active fallback

Follow these guidelines to ensure seamless and secure key management during MACsec operations:

- Ensure the system performs a hitless rollover from the current active key to the fallback key during CAK rollover of primary keys if the latest active keys mismatch and the fallback keys match.
- Ensure the system performs a hitless rollover back to the primary latest active key when a session is active with the fallback key and the primary latest active key mismatch is resolved between peers.
- Enable active fallback to include the fallback PSK entry in MACsec show commands. When the session is secured with the primary key, the fallback session status must display as ACTIVE.
- Configure a valid fallback PSK (CKN and CAK) with an infinite lifetime.
- Do not configure the fallback PSK with a CAK mismatch. If a mismatch happens, resolve it by pushing a new set of PSK configurations across all association members—first on the fallback PSK keychain, then on the primary PSK keychain.
- Configure the enable-legacy-fallback command under the macsec-policy to maintain backward compatibility if the peer device runs an older software release that does not support active fallback.
- In point-to-point (P2P) topologies, rollover to the fallback PSK occurs when either node in the Secure Association (SA) cannot establish a session with the primary PSK.
- In point-to-multipoint (P2MP) topologies, fallback occurs only when the primary key expires or is deleted on all peers, not just one. If the primary PSK is deleted or expires on a single node (e.g., R1), a new key server is selected among the remaining peers to perform a SAK rekey. This process excludes that node from the SA. All traffic to and from that node is dropped.

Guidelines to configure MACsec interface

Follow these guidelines to ensure optimal configuration and performance of MACsec interfaces:

- Configure separate keychains for primary and fallback PSKs. Do not update both PSKs at the same time. Use the fallback PSK only to recover a MACsec session if the primary key fails.
- Adjust the interface MTU to account for MACsec overhead. For example, if the default MTU is 1514 bytes, set it to 1546 bytes (1514 + 32 bytes overhead). For IS-IS, ensure a minimum MTU of 1546 bytes.
- Enable MACsec on all members of a bundle.
  - If MACsec peers use IOS-XR version 24.1.1 or higher, configure **impose-overhead-on-bundle** in the MACsec policy to adjust the bundle interface MTU for routing protocols running on the bundle interface.
  - If using IOS-XR versions prior to 24.1.1, configure the maximum MTU on the bundle interface to accommodate the protocol packet size plus 32 bytes MACsec overhead. Disable hello-padding for IS-IS running on the bundle interface.

- Define the MACsec keychain before applying the MACsec configuration to the interface. If you apply the keychain without specifying a policy, the default MACsec policy is used.
- Use the `openconfig-macsec.yang` OpenConfig data model to programmatically view the MACsec configuration. For more information, see the *Programmability Configuration Guide for Cisco 8000 Series Routers*.

Configure MACsec encryption

Configure MACsec encryption on network interfaces to secure data.

To enable secure communication on physical interfaces, configure MACsec encryption with specific settings.

Procedure

Step 1 Create a MACsec keychain to manage encryption keys.

Step 2 Define a user-defined MACsec policy to specify security requirements.

Step 3 Apply the MACsec configuration to physical interfaces.

MACsec encryption is successfully configured on the specified physical interfaces.

Configure a MACsec keychain

Enable MACsec encryption and decryption on routers by configuring a MACsec keychain, ensuring secure communication between peers using the MACsec protocol.

Follow these steps to configure a MACsec keychain:

Before you begin

- Ensure you have administrative access to the router.
- Verify that the router supports MACsec encryption.

Procedure

Step 1 Create a keychain, specifying a unique keychain name.

Example:

```
Router# configure
Router(config)# key chain kc
```

Step 2 Enable MACsec mode for the keychain.

Example:

```
Router(config-kc)# macsec
```

Step 3 Configure a MACsec key for the keychain.

Example:

```
Router(config-kc-MacSec)# key key1
```

Step 4 Specify the key string and the cryptographic algorithm.

Example:

```
Router(config-kc-MacSec-KEY1)# key-string
112233445566778899900AABBCCDDEEFF00112233445566778899AABBCCDDEEFF cryptographic-algorithm AES-128-CMAC-96
```

- Key-string range: The key-string range is 32 characters for AES-128 and 64 characters for AES-256. Ensure that the string length matches the requirements of the selected algorithm.
- Cryptographic algorithm options: AES-128-CMAC-96 or AES-256-CMAC.

Step 5 Define the validity period for the key.

Example:

```
Router(config-kc-MacSec-KEY1)# lifetime 05:00:00 01 January 2019 infinite
Router(config-kc-MacSec-KEY1)# commit
```

Lifetime range: You can specify a lifetime range by providing a fixed timeframe (including start and expiry), or set it as infinite.

Step 6 Verify the keychain settings in the running configuration.

Example:

```
Router# show running-config key chain kc1
key chain kc1
 macsec
  key key1
    key-string 112233445566778899900AABBCCDDEEFF00112233445566778899AABBCCDDEEFF cryptographic-algorithm
 AES-128-CMAC-96
     lifetime 05:00:00 01 January 2019 infinite
  !
 !
```

The MACsec keychain is created and ready for use with MACsec encryption.

What to do next

Apply the keychain to the router interface configuration when required.

Create a user-defined MACsec policy

Define and configure a custom MACsec policy to secure network traffic. Specify encryption, key server priority, security parameters, and additional protections.

Follow these steps to create a user-defined MACsec policy:

Before you begin

- Ensure you have administrative access to the router.

- Verify that the router supports MACsec encryption.

Procedure

Step 1 Create a MACsec policy, specifying a unique policy name.

Example:

```
Router# configure
Router(config)# macsec-policy mp1
```

Step 2 Configure the cipher suite for MACsec encryption.

Example:

```
Router(config-macsec-policy)# cipher-suite GCM-AES-XPN-128
```

The GCM encryption method, which uses the AES encryption algorithm, supports the following encryption suites:

- GCM-AES-XPN-128
- GCM-AES-XPN-256

Step 3 Set the confidentiality offset value.

Example:

```
Router(config-macsec-policy)# conf-offset CONF-OFFSET-30
```

Step 4 Configure the key server priority.

Example:

```
Router(config-macsec-policy)# key-server-priority 10
```

Range: `0` to `255` (A lower value indicates higher priority for key server selection. Default value is `16`).

Step 5 Set the security policy:

Example:

```
Router(config-macsec-policy)# security-policy should-secure
```

- `must-secure`: Allows only MACsec-encrypted traffic. The router drops traffic until the MKA session is secured.
- `should-secure`: Allows unencrypted traffic until the MKA session is secured, then only encrypted traffic is allowed.

Step 6 Enable data delay protection.

Example:

```
Router(config-macsec-policy)# delay-protection
```

Step 7 Configure the replay protection window size.

Example:

```
Router(config-macsec-policy)# window-size 64
```

Range: `0` to `1024`

Step 8 Include the Integrity Check Value (ICV) indicator in frames that arrive on the port and commit the configuration to save the MACsec policy settings.

Example:

```
Router(config-macsec-policy)# include-icv-indicator
Router(config-macsec-policy)# commit
```

To set the rekey interval, use the **sak-rekey-interval** command in macsec-policy configuration mode. The timer ranges from 60 to 2,592,000 seconds, the default being OFF.

Step 9 Verify the MACsec policy settings in the running configuration.

Example:

```
Router# show running-config macsec-policy mp1

macsec-policy mp1
 conf-offset CONF-OFFSET-30
 security-policy should-secure
 cipher-suite GCM-AES-XPN-128
 window-size 64
 include-icv-indicator
 delay-protection
 key-server-priority 10
!
```

The user-defined MACsec policy is created and ready for use with MACsec encryption.

What to do next

Apply the user-defined MACsec policy to the router interface configuration when required.

Configure MACsec encryption on an interface

Secure network communication on a host-facing interface using MACsec encryption.

he MACsec PSK (keychain and user-defined policy) configuration is applied to a host-facing interface of a CE router. This establishes a secure connection.

Follow these steps to configure MACsec on an interface:

Before you begin

Ensure the interface is a host-facing interface on a CE router.

Procedure

Step 1 Access interface configuration mode.

Example:

```
Router# configure
Router(config)# interface hundredGigE Hu0/1/0/10
```

Step 2 Configure the IPv4 address for the interface.

Example:

```
Router(config-if)# ipv4 address 192.168.30.1 255.255.255.0
```

Step 3 Apply the MACsec keychain and user-defined MACsec policy to the interface.

Example:

```
Router(config-if)# macsec psk-keychain kc1 policy mp1
```

Step 4 Commit the configuration to save changes.

Example:

```
Router(config-if)# commit
```

Step 5 Verify the MACsec configuration applied to the interface.

Example:

```
Router# show running-config interface HundredGigE 0/1/0/10
interface HundredGigE 0/1/0/10
 ipv4 address 192.168.30.1 255.255.255.0
 macsec psk-keychain kc1 policy mp1
!
```

MACsec encryption is applied to the specified interface, securing communication.

Verify MACsec session status

Confirm that MACsec encryption is correctly configured and operational on your network devices.

After configuring MACsec on your routers, perform this task to ensure security and connectivity.

Follow these steps to verify MACsec encryption:

Before you begin

- Ensure MACsec is configured on the relevant interfaces.
- Access the executive mode on your router.

Procedure

Step 1 Verify the MACsec policy configuration using the **show macsec policy detail** command.

Example:

```
Router# show macsec policy mp1 detail
Policy Name                 : mp1
      Cipher Suite          : GCM-AES-XPN-128
      Key-Server Priority   : 10
      Window Size           : 64
      Conf Offset           : 30
      Replay Protection     : TRUE
      Delay Protection      : FALSE
      Security Policy       : Should Secure
      Vlan Tags In Clear    : 1
```

```
LACP In Clear         : FALSE
LLDP In Clear         : FALSE
Pause Frame In Clear  : FALSE
Sak Rekey Interval    : FALSE
Include ICV Indicator : TRUE
Use Eapol PAE in ICV  : FALSE
Disable Suspend On Request    : FALSE
Disable Suspend For           : FALSE
Enable legacy fallback        : FALSE
SKS Profile                   : N/A
Max AN                        : 3
```

If the displayed values do not match your expected settings, run the **show run macsec-policy** command review your configuration.

Step 2 View summary of the MACsec sessions using the **show macsec mka summary** command.

Example:

```
Router# show macsec mka summary

NODE: node0_1_CPU0
================================================================================================
Interface-Name        Status  Cipher-Suite       KeyChain   PSK/EAP   CKN
================================================================================================
Hu0/1/0/10            Secured GCM-AES-XPN-128    kc         PRIMARY   1234

Total MACSec Sessions : 1 : 1 Secured Sessions
Pending Sessions      : 1
Suspended Sessions    : 0
```

Step 3 Verify interface peering using the **show macsec mka session** command.

Example:

```
Router# show macsec mka session

NODE: node0_1_CPU0
================================================================================================
Interface-Name        Local-TxSCI           #Peers Status  Key-Server PSK/EAP CKN
================================================================================================
Hu0/1/0/10            7872.5d1a.e7d4/0001   1      Secured NO         PRIMARY 1234
```

Step 4 View details of the MKA session using the **show macsec mka session detail** command.

Example:

```
Router# show macsec mka session detail

NODE: node0_1_CPU0

MKA Detailed Status for MKA Session
===================================
Status: Secured - Secured MKA Session with MACsec

Local Tx-SCI                    : 7872.5d1a.e7d4/0001
Local Tx-SSCI                   : 1
Interface MAC Address           : 7872.5d1a.e7d4
MKA Port Identifier             : 1
Interface Name                  : Hu0/1/0/10
CAK Name (CKN)                  : 1234
CA Authentication Mode          : PRIMARY-PSK
Keychain Member Identifier (MI) : kc
Message Number (MN)             : C12A70FEE1212B835BDDDCBA
Authenticator                   : 3009
Key Server                      : NO
```

```
MKA Cipher Suite                 : NO : AES-128-CMAC
Configured MACSec Cipher Suite   : GCM-AES-XPN-128

Latest SAK Status                : Rx & Tx
Latest SAK AN                    : 0
Latest SAK KI (KN)               : 018E2F0D63FF2ED6A5BF270E00000001 (1)
Old SAK Status                   : FIRST-SAK
Old SAK AN                       : 0
Old SAK KI (KN)                  : FIRST-SAK (0)

SAK Transmit Wait Time           : 0s (Not waiting for any peers to respond)
SAK Retire Time                  : 0s (No Old SAK to retire)
Time to SAK Rekey                : NA
Time to exit suspension          : NA

MKA Policy Name                  : mp-SF
Key Server Priority              : 10
Delay Protection                 : TRUE
Replay Window Size               : 64
Include ICV Indicator            : TRUE
Confidentiality Offset           : 30
Algorithm Agility                : 80C201
SAK Cipher Suite                 : 0080C20001000003 (GCM-AES-XPN-128)
MACsec Capability                : 3 (MACsec Integrity, Confidentiality, & Offset)
MACsec Desired                   : YES

# of MACsec Capable Live Peers   : 1
# of MACsec Capable Live Peers Responded : 0

Live Peer List:
-----------------------------------------------------------------------------
SSCI               MI                  MN   Rx-SCI               KS-Priority
-----------------------------------------------------------------------------
018E2F0D63FF2ED6A5BF270E 2699 008a.962d.7400/0001 2 16

Potential Peer List:
-----------------------------------------------------------------------------
SSCI               MI                  MN   Rx-SCI               KS-Priority
-----------------------------------------------------------------------------

Peers Status:
Last Tx MKPDU                    : 2019 Oct 08 09:07:06.475
Peer Count                       : 1

RxSCI                            : 008A962D74000001
MI                               : 018E2F0D63FF2ED6A5BF270E
Peer CAK                         : Match
Latest Rx MKPDU                  : 2019 Oct 08 09:07:06.032
```

Step 5 View detailed MKA session information for a specific interface using the **show macsec mka session interface** command.

Example:

```
Router# show macsec mka session interface hundredGigE 0/1/0/10

===============================================================================================
Interface-Name        Local-TxSCI          #Peers Status Key-Server PSK/EAP CKN
===============================================================================================
Hu0/1/0/10            7872.5d1a.e7d4/0001   1      Secured NO         PRIMARY 1234
Hu0/1/0/10            7872.5d1a.e7d4/0001   1      Secured NO         FALLBACK 5678
```

The `Status` field should indicate `Secured` for the MKA session. A status of `Pending` or `INITIALIZING` means MACsec encryption is not successfully configured.

Step 6 Verify MACsec session counter statistics using the **show macsec mka statistics** command.

Example:

```
Router# show macsec mka statistics interface hundredGigE 0/1/0/10

MKA Statistics for Session on interface (Hu0/1/0/10)
=====================================================
Reauthentication Attempts.. 0

CA Statistics
Pairwise CAKs Derived... 0
Pairwise CAK Rekeys..... 0
Group CAKs Generated.... 0
Group CAKs Received..... 0

SA Statistics
SAKs Generated.......... 0
SAKs Rekeyed............ 0
SAKs Received........... 1
SAK Responses Received.. 0

MKPDU Statistics
MKPDUs Transmitted...... 3097
"Distributed SAK".. 0
"Distributed CAK".. 0
MKPDUs Validated & Rx... 2788
"Distributed SAK".. 1
"Distributed CAK".. 0

MKA IDB Statistics
MKPDUs Tx Success.......... 3097
MKPDUs Tx Fail............. 0
MKPDUS Tx Pkt build fail... 0
MKPDUS No Tx on intf down.. 3
MKPDUS No Rx on intf down.. 0
MKPDUs Rx CA Not found..... 0
MKPDUs Rx Error............ 0
MKPDUs Rx Success.......... 2788
MKPDUs Rx Invalid Length... 0
MKPDUs Rx Invalid CKN...... 0
MKPDUs Rx force suspended.. 0
MKPDUs Tx force suspended.. 0

MKPDU Failures
MKPDU Rx Validation (ICV).................. 0
MKPDU Rx Bad Peer MN....................... 0
MKPDU Rx Non-recent Peerlist MN............ 0
MKPDU Rx Drop SAKUSE, KN mismatch......... 0
MKPDU Rx Drop SAKUSE, Rx Not Set.......... 0
MKPDU Rx Drop SAKUSE, Key MI mismatch..... 0
MKPDU Rx Drop SAKUSE, AN Not in Use....... 0
MKPDU Rx Drop SAKUSE, KS Rx/Tx Not Set.... 0
MKPDU Rx Drop Packet, Ethertype Mismatch.. 0
MKPDU Rx Drop Packet, Source MAC NULL..... 0
MKPDU Rx Drop Packet, Destination MAC NULL 0
MKPDU Rx Drop Packet, Payload NULL........ 0

SAK Failures
SAK Generation................... 0
Hash Key Generation.............. 0
SAK Encryption/Wrap.............. 0
SAK Decryption/Unwrap............ 0

CA Failures
```

```
ICK Derivation.................. 0
KEK Derivation.................. 0
Invalid Peer MACsec Capability... 0

MACsec Failures
Rx SC Creation.................. 0
Tx SC Creation.................. 0
Rx SA Installation.............. 0
Tx SA Installation.............. 0
```

Review the counters for MACsec PDUs transmitted, validated, and received, and check for transmission errors.

Step 7 Verify MACsec encryption and hardware interface descriptor block (IDB) information using the **show macsec ea idb interface** command.

Example:

```
Router# show macsec ea idb interface hundredGigE 0/1/0/10

IDB Details:
if_sname                    : Hu0/1/0/10
if_handle                   : 0x8001e0
MacSecControlledIfh         : 0x800330
MacSecUnControlledIfh       : 0x800338
Replay window size          : 64
Local MAC                   : 78:72:5d:1a:e7:d4
Rx SC Option(s)             : Validate-Frames Replay-Protect
Tx SC Option(s)             : Protect-Frames Always-Include-SCI
Security Policy             : SHOULD SECURE
Delay Protection            : TRUE
Sectag offset               : 0

Rx SC 1
Rx SCI                      : 008a962d74000001
Peer MAC                    : 00:8a:96:2d:74:00
Stale SAK Data              : NO
SAK[0]                      : ***
SAK Len                     : 16
SAK Version                 : 1
HashKey[0]                  : ***
HashKey Len                 : 16
Conf offset                 : 30
Cipher Suite                : GCM-AES-XPN-128
CtxSalt[0]                  : 01 8f 2f 0f 63 ff 2e d6 a5 bf 27 0e
ssci                        : 2
Rx SA Program Req[0]: 2019 Oct 08 07:37:14.870
Rx SA Program Rsp[0]: 2019 Oct 08 07:37:14.902

Tx SC
Tx SCI                      : 78725d1ae7d40001
Active AN                   : 0
Old AN                      : 255
Next PN                     : 1, 0, 0, 0
SAK Data                    : ***
SAK[0]                      : 16
SAK Len                     : 1
SAK Version                 : ***
HashKey[0]                  : 16
HashKey Len                 : 30
Conf offset                 : GCM-AES-XPN-128
Cipher Suite                : 01 8f 2f 0c 63 ff 2e d6 a5 bf 27 0e
CtxSalt[0]                  : 1
ssci                        : 2019 Oct 08 07:37:14.908
Tx SA Program Req[0]: 2019 Oct 08 07:37:14.931
Tx SA Program Rsp[0]: 2019 Oct 08 07:37:14.931
```

Step 8 Verify hardware programming using the **show macsec platform hardware sa interface** command.

Example:

```
Router# show macsec platform hardware sa interface hundredGigE 0/1/0/10
----------------------------------------------------------------
Tx SA Details:
----------------------------------------------------------------
SCI : 7872.5d1a.e7d4/0001
Crypto Algo : GCM-AES-XPN-128
AES Key Len : 128 bits
AN : 0
Initial Packet Number : 1
Current Packet Number : 1
Maximum Packet Number : 3221225400
XForm in Use : YES
Action Type : SA Action Egress
Direction : Egress
Conf Offset : 00000030
Drop Type : 0x00000003
SA In Use : YES
ConfProtect : YES
IncludeSCI : YES
ProtectFrame : YES
UseEs : NO
UseSCB : NO
----------------------------------------------------------------
Rx SA Details:
----------------------------------------------------------------
SCI : 008a.962d.7400/0001
Replay Window : 64
Crypto Algo : GCM-AES-XPN-128
AES Key Len : 128 bits
AN : 0
Initial Packet Number : 1
```