



MACsec Encryption

- [Information About MACsec Encryption, on page 1](#)
- [How to Configure MACsec Encryption, on page 13](#)
- [Configuration Examples for MACsec Encryption, on page 30](#)

Information About MACsec Encryption

MACsec is the IEEE 802.1AE standard for authenticating and encrypting packets between two MACsec-capable devices. These Catalyst switches support 802.1AE encryption with MACsec Key Agreement (MKA) on downlink ports for encryption between the switch and host device. The switch also supports MACsec encryption for switch-to-switch (inter-network device) security using both Cisco TrustSec Network Device Admission Control (NDAC), Security Association Protocol (SAP) and MKA-based key exchange protocol. Link layer security can include both packet authentication between switches and MACsec encryption between switches (encryption is optional).



Note MACsec is not supported with the NPE license or the LAN Base service image.

Table 1: MACsec Support on Switch Ports

| Interface | Connections | MACsec support |
|----------------|------------------|---|
| Downlink ports | Switch-to-host | MACsec MKA encryption |
| Uplink ports | Switch-to-switch | MACsec MKA encryption Cisco TrustSec NDAC MACsec |

Cisco TrustSec and Cisco SAP are meant only for switch-to-switch links and are not supported on switch ports connected to end hosts, such as PCs or IP phones. MKA is supported on switch-to-host facing links (downlink) as well as switch-to-switch links (uplink). Host-facing links typically use flexible authentication ordering for handling heterogeneous devices with or without IEEE 802.1x, and can optionally use MKA-based MACsec encryption. Cisco NDAC and SAP are mutually exclusive with Network Edge Access Topology (NEAT), which is used for compact switches to extend security outside the wiring closet.

Media Access Control Security and MACsec Key Agreement

MACsec, defined in 802.1AE, provides MAC-layer encryption over wired networks by using out-of-band methods for encryption keying. The MACsec Key Agreement (MKA) Protocol provides the required session keys and manages the required encryption keys. MKA and MACsec are implemented after successful authentication using the 802.1x Extensible Authentication Protocol (EAP-TLS) or Pre Shared Key (PSK) framework.

A switch using MACsec accepts either MACsec or non-MACsec frames, depending on the policy associated with the MKA peer. MACsec frames are encrypted and protected with an integrity check value (ICV). When the switch receives frames from the MKA peer, it decrypts them and calculates the correct ICV by using session keys provided by MKA. The switch compares that ICV to the ICV within the frame. If they are not identical, the frame is dropped. The switch also encrypts and adds an ICV to any frames sent over the secured port (the access point used to provide the secure MAC service to a MKA peer) using the current session key.

The MKA Protocol manages the encryption keys used by the underlying MACsec protocol. The basic requirements of MKA are defined in 802.1x-REV. The MKA Protocol extends 802.1x to allow peer discovery with confirmation of mutual authentication and sharing of MACsec secret keys to protect data exchanged by the peers.

The EAP framework implements MKA as a newly defined EAP-over-LAN (EAPOL) packet. EAP authentication produces a master session key (MSK) shared by both partners in the data exchange. Entering the EAP session ID generates a secure connectivity association key name (CKN). The switch acts as the authenticator for both uplink and downlink; and acts as the key server for downlink. It generates a random secure association key (SAK), which is sent to the client partner. The client is never a key server and can only interact with a single MKA entity, the key server. After key derivation and generation, the switch sends periodic transports to the partner at a default interval of 2 seconds.

The packet body in an EAPOL Protocol Data Unit (PDU) is referred to as a MACsec Key Agreement PDU (MKPDU). MKA sessions and participants are deleted when the MKA lifetime (6 seconds) passes with no MKPDU received from a participant. For example, if a MKA peer disconnects, the participant on the switch continues to operate MKA until 6 seconds have elapsed after the last MKPDU is received from the MKA peer.



Note Integrity check value (ICV) indicator in MKPDU is optional. ICV is not optional when the traffic is encrypted.

EAPoL Announcements indicate the use of the type of keying material. The announcements can be used to announce the capability of the supplicant as well as the authenticator. Based on the capability of each side, the largest common denominator of the keying material could be used.

Prior to Cisco IOS XE Fuji 16.8.1a, should-secure was supported for MKA and SAP. With should-secure enabled, if the peer is configured for MACsec, the data traffic is encrypted, otherwise it is sent in clear text. Starting with Cisco IOS XE Fuji 16.8.1a, must-secure support is enabled on both the ingress and the egress. Must-secure is supported for MKA and SAP. With must-secure enabled, only EAPoL traffic will not be encrypted. The rest of the traffic will be encrypted. Unencrypted packets are dropped.



Note Must-secure mode is enabled by default.

MKA Policies

To enable MKA on an interface, a defined MKA policy should be applied to the interface. You can configure these options:

- Policy name, not to exceed 16 ASCII characters.
- Confidentiality (encryption) offset of 0, 30, or 50 bytes for each physical interface

Virtual Ports

Use virtual ports for multiple secured connectivity associations on a single physical port. Each connectivity association (pair) represents a virtual port. In uplink, you can have only one virtual port per physical port. In downlink, you can have a maximum of two virtual ports per physical port, of which one virtual port can be part of a data VLAN; the other must externally tag its packets for the voice VLAN. You cannot simultaneously host secured and unsecured sessions in the same VLAN on the same port. Because of this limitation, 802.1x multiple authentication mode is not supported.

The exception to this limitation is in multiple-host mode when the first MACsec supplicant is successfully authenticated and connected to a hub that is connected to the switch. A non-MACsec host connected to the hub can send traffic without authentication because it is in multiple-host mode. We do not recommend using multi-host mode because after the first successful client, authentication is not required for other clients.

Virtual ports represent an arbitrary identifier for a connectivity association and have no meaning outside the MKA Protocol. A virtual port corresponds to a separate logical port ID. Valid port IDs for a virtual port are 0x0002 to 0xFFFF. Each virtual port receives a unique secure channel identifier (SCI) based on the MAC address of the physical interface concatenated with a 16-bit port ID.

MACsec and Stacking

A switch active switch running MACsec maintains the configuration files that show which ports on a member switch support MACsec. The active switch performs these functions:

- Processes secure channel and secure association creation and deletion
- Sends secure association service requests to the member switches.
- Processes packet number and replay-window information from local or remote ports and notifies the key management protocol.
- Sends MACsec initialization requests with the globally configured options to new switches that are added to the stack.
- Sends any per-port configuration to the member switches.

A member switch performs these functions:

- Processes MACsec initialization requests from the active switch.
- Processes MACsec service requests sent by the active switch.
- Sends information about local ports to the active switch.

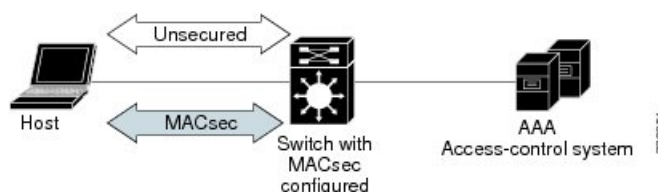
MACsec, MKA and 802.1x Host Modes

You can use MACsec and the MKA Protocol with 802.1x single-host mode, multi-host mode, or Multi Domain Authentication (MDA) mode. Multiple authentication mode is not supported.

Single-Host Mode

The figure shows how a single EAP authenticated session is secured by MACsec by using MKA

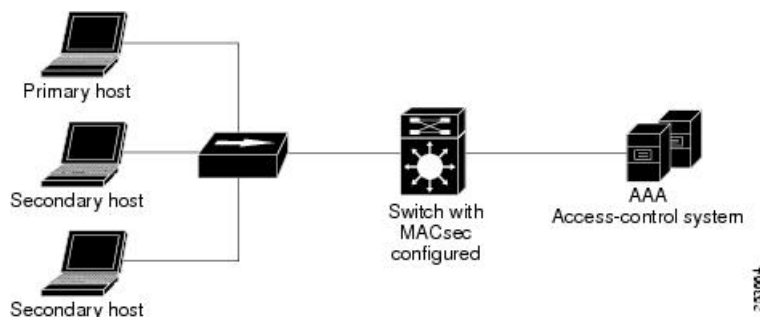
Figure 1: MACsec in Single-Host Mode with a Secured Data Session



Multiple Host Mode

In standard (not 802.1x REV) 802.1x multiple-host mode, a port is open or closed based on a single authentication. If one user, the primary secured client services client host, is authenticated, the same level of network access is provided to any host connected to the same port. If a secondary host is a MACsec supplicant, it cannot be authenticated and traffic would not flow. A secondary host that is a non-MACsec host can send traffic to the network without authentication because it is in multiple-host mode. The figure shows MACsec in Standard Multiple-Host Unsecure Mode.

Figure 2: MACsec in Multiple-Host Mode - Unsecured



Note Multi-host mode is not recommended because after the first successful client, authentication is not required for other clients, which is not secure.

In standard (not 802.1x REV) 802.1x multiple-domain mode, a port is open or closed based on a single authentication. If the primary user, a PC on data domain, is authenticated, the same level of network access is provided to any domain connected to the same port. If a secondary user is a MACsec supplicant, it cannot be authenticated and traffic would no flow. A secondary user, an IP phone on voice domain, that is a non-MACsec host, can send traffic to the network without authentication because it is in multiple-domain mode.


```
# of MACsec Capable Live Peers..... 1
# of MACsec Capable Live Peers Responded.. 1

Live Peers List:
  MI                      MN          Rx-SCI (Peer)      KS Priority
  -----
  38046BA37D7DA77E06D006A9  89560      c800.8459.e764/002a  10

Potential Peers List:
  MI                      MN          Rx-SCI (Peer)      KS Priority
  -----

Dormant Peers List:
  MI                      MN          Rx-SCI (Peer)      KS Priority
  -----
```

Device# **show mka policy**

MKA Policy Summary...

| Policy Name | KS Priority | Delay Protect | Replay Protect | Window Size | Conf Offset | Cipher Suite(s) | Interfaces Applied |
|------------------|-------------|---------------|----------------|-------------|-------------|-----------------|--------------------|
| *DEFAULT POLICY* | 0 | FALSE | TRUE | 0 | 0 | GCM-AES-128 | |
| p1 | 1 | FALSE | TRUE | 0 | 0 | GCM-AES-128 | |
| p2 | 2 | FALSE | TRUE | 0 | 0 | GCM-AES-128 | Gi1/0/1 |

Device# **show mka policy p2 detail**

```
MKA Policy Configuration ("p2")
=====
MKA Policy Name..... p2
Key Server Priority... 2
Confidentiality Offset. 0
Send Secure Announcement..DISABLED
Cipher Suite(s)..... GCM-AES-128

Applied Interfaces...
  GigabitEthernet1/0/1
```

This is an example of the **show mka statistics** command output:

Device# **show mka statistics interface G1/0/1**

```
MKA Statistics for Session
=====
Reauthentication Attempts.. 0

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

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



```

SAK Encryption/Wrap..... 0
SAK Decryption/Unwrap..... 0
SAK Cipher Mismatch..... 0

CA Failures
Group CAK Generation..... 0
Group CAK Encryption/Wrap..... 0
Group CAK Decryption/Unwrap..... 0
Pairwise CAK Derivation..... 0
CKN Derivation..... 0
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

MKPDU Failures
MKPDU Tx..... 0
MKPDU Rx Validation..... 0
MKPDU Rx Bad Peer MN..... 0
MKPDU Rx Non-recent Peerlist MN.. 0

```

Information About MACsec MKA using EAP-TLS

MACsec MKA is supported on switch-to-switch links. Using IEE 802.1X Port-based Authentication with Extensible Authentication Protocol (EAP-TLS), you can configure MACsec MKA between device uplink ports. EAP-TLS allows mutual authentication and obtains an MSK (master session key) from which the connectivity association key (CAK) is derived for MKA operations. Device certificates are carried, using EAP-TLS, for authentication to the AAA server.

Prerequisites for MACsec MKA using EAP-TLS

- Ensure that you have a Certificate Authority (CA) server configured for your network.
- Generate a CA certificate.
- Ensure that you have configured Cisco Identity Services Engine (ISE) Release 2.0.
- Ensure that both the participating devices, the CA server, and Cisco Identity Services Engine (ISE) are synchronized using Network Time Protocol (NTP). If time is not synchronized on all your devices, certificates will not be validated.
- Ensure that 802.1x authentication and AAA are configured on your device.

Limitations for MACsec MKA using EAP-TLS

- MKA is not supported on port-channels.
- MKA is not supported with High Availability and local authentication.
- MKA/EAPTLS is not supported for promiscuous PVLAN Primary port.
- While configuring MACsec MKA using EAP-TLS, MACsec secure channels encrypt counters does not increment before first Rekey.

Information About MKA/MACsec for Port Channel

MKA/MACsec can be configured on the port members of a port channel. MKA/MACsec is agnostic to the port channel since the MKA session is established between the port members of a port channel.



Note Etherchannel links that are formed as part of the port channel can either be congruent or disparate i.e. the links can either be MACsec-secured or non-MACsec-secured. MKA session between the port members is established even if a port member on one side of the port channel is not configured with MACsec.

It is recommended that you enable MKA/MACsec on all the member ports for better security of the port channel.

Information About MACsec Cipher Announcement

Cipher Announcement allows the supplicant and the authenticator to announce their respective MACsec Cipher Suite capabilities to each other. Both, the supplicant and the authenticator, calculate the largest common supported MACsec Cipher Suite and use the same as the keying material for the MKA session.



Note Only the MACsec Cipher Suite capabilities which are configured in the MKA policy are announced from the authenticator to the supplicant.

There are two types of EAPoL Announcements :

- Unsecured Announcements (EAPoL PDUs) : Unsecured announcements are EAPoL announcements carrying MACsec Cipher Suite capabilities in an unsecured manner. These announcements are used to decide the width of the key used for MKA session prior to authentication.
- Secure Announcements (MKPDUs) : Secure announcements revalidate the MACsec Cipher Suite capabilities which were shared previously through unsecure announcements.

Once the session is authenticated, peer capabilities which were received through EAPoL announcements are revalidated with the secure announcements. If there is a mismatch in the capabilities, the MKA session tears down.

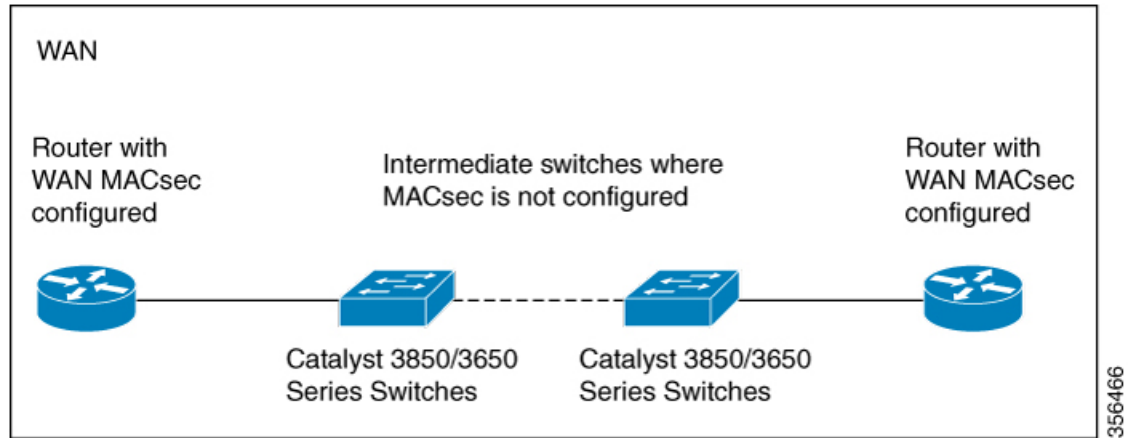
Limitations for MACsec Cipher Announcement

- If MACsec Cipher Suite Capabilities get changed in an active policy at the authenticator, the updated capabilities are not take into effect until a **shutdown/no shutdown** is performed on the interface. If you do not disable and restart the interface, EAPoL Announcement continues to announce the older capabilities.
- The MKA session between the supplicant and the authenticator does not tear down even if the MACsec Cipher Suite Capabilities configured on both do not result in a common cipher suite.

MACsec Connections Across Intermediate Switches

Prior to Cisco IOS XE Gibraltar 16.11.1, MACsec connection between end devices which have WAN MACsec configured with the intermediate switches as the Cisco Catalyst 3650 and 3850 Series Switches was not

supported. The encrypted packets were dropped if WAN MACsec was configured on the end devices with MACsec not configured on the intermediate switches. With the ClearTag feature implemented on the ASIC, the switch forwards the encrypted packet without parsing the MACsec header.



Limitations for MACsec Connections Across Intermediate Switches

- Hop-by-hop MACsec encryption with Catalyst 3650 and 3850 Series switches as intermediate switches where WAN MACsec is configured on the routers is not supported.
- WAN MACsec configured on the routers with intermediate switches as the Catalyst 3650 and 3850 Series switches is not supported on Layer 3 VPNs.
- WAN MACsec configured on the routers with intermediate switches as the Catalyst 3650 and 3850 Series switches show Cisco Discovery Protocol neighbors only in should-secure mode.

Cisco TrustSec Overview

The table below lists the TrustSec features to be eventually implemented on TrustSec-enabled Cisco switches. Successive general availability releases of TrustSec will expand the number of switches supported and the number of TrustSec features supported per switch.

| Cisco TrustSec Feature | Description |
|--------------------------|--|
| 802.1AE Tagging (MACsec) | <p>Protocol for IEEE 802.1AE-based wire-rate hop-to-hop Layer 2 encryption.</p> <p>Between MACsec-capable devices, packets are encrypted on egress from the transmitting device, decrypted on ingress to the receiving device, and in the clear within the devices.</p> <p>This feature is only available between TrustSec hardware-capable devices.</p> |

| Cisco TrustSec Feature | Description |
|---|---|
| Endpoint Admission Control (EAC) | EAC is an authentication process for an endpoint user or a device connecting to the TrustSec domain. Usually EAC takes place at the access level switch. Successful authentication and authorization in the EAC process results in Security Group Tag assignment for the user or device. Currently EAC can be 802.1X, MAC Authentication Bypass (MAB), and Web Authentication Proxy (WebAuth). |
| Network Device Admission Control (NDAC) | NDAC is an authentication process where each network device in the TrustSec domain can verify the credentials and trustworthiness of its peer device. NDAC utilizes an authentication framework based on IEEE 802.1X port-based authentication and uses EAP-FAST as its EAP method. Successful authentication and authorization in NDAC process results in Security Association Protocol negotiation for IEEE 802.1AE encryption. |
| Security Association Protocol (SAP) | After NDAC authentication, the Security Association Protocol (SAP) automatically negotiates keys and the cipher suite for subsequent MACSec link encryption between TrustSec peers. SAP is defined in IEEE 802.11i. |
| Security Group Tag (SGT) | An SGT is a 16-bit single label indicating the security classification of a source in the TrustSec domain. It is appended to an Ethernet frame or an IP packet. |
| SGT Exchange Protocol (SXP) | Security Group Tag Exchange Protocol (SXP). With SXP, devices that are not TrustSec-hardware-capable can receive SGT attributes for authenticated users and devices from the Cisco Identity Services Engine (ISE) or the Cisco Secure Access Control System (ACS). The devices can then forward a sourceIP-to-SGT binding to a TrustSec-hardware-capable device will tag the source traffic for SGACL enforcement. |

When both ends of a link support 802.1AE MACsec, SAP negotiation occurs. An EAPOL-key exchange occurs between the supplicant and the authenticator to negotiate a cipher suite, exchange security parameters, and manage keys. Successful completion of these tasks results in the establishment of a security association (SA).

Depending on your software version and licensing and link hardware support, SAP negotiation can use one of these modes of operation:

- Galois Counter Mode (GCM)—authentication and encryption
- GCM authentication (GMAC)— GCM authentication, no encryption
- No Encapsulation—no encapsulation (clear text)

- Null—encapsulation, no authentication or encryption

How to Configure MACsec Encryption

Configuring MKA and MACsec

Default MACsec MKA Configuration

MACsec is disabled. No MKA policies are configured.

Configuring an MKA Policy

SUMMARY STEPS

1. **configure terminal**
2. **mka policy** *policy name*
3. **send-secure-announcements**
4. **key-server priority**
5. **include-icv-indicator**
6. **macsec-cipher-suite** *gcm-aes-128*
7. **confidentiality-offset** *Offset value*
8. **end**
9. **show mka policy**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--------------------------------------|--|
| Step 1 | configure terminal | Enter global configuration mode. |
| Step 2 | mka policy <i>policy name</i> | Identify an MKA policy, and enter MKA policy configuration mode. The maximum policy name length is 16 characters. Note The default MACsec cipher suite in the MKA policy will always be "GCM-AES-128". If the device supports both "GCM-AES-128" and "GCM-AES-256" ciphers, it is highly recommended to define and use a user defined MKA policy to include both 128 and 256 bits ciphers or only 256 bits cipher, as may be required. |
| Step 3 | send-secure-announcements | Enabled secure announcements. Note By default, secure announcements are disabled. |

| | Command or Action | Purpose |
|--------|--|---|
| Step 4 | <code>key-server priority</code> | Configure MKA key server options and set priority (between 0-255). Note When value of key server priority is set to 255, the peer can not become the key server. The key server priority value is valid only for MKA PSK; and not for MKA EAPTLS. |
| Step 5 | <code>include-icv-indicator</code> | Enables the ICV indicator in MKPDU. Use the no form of this command to disable the ICV indicator — no include-icv-indicator . |
| Step 6 | <code>macsec-cipher-suite gcm-aes-128</code> | Configures cipher suite for deriving SAK with 128-bit encryption. |
| Step 7 | <code>confidentiality-offset Offset value</code> | Set the Confidentiality (encryption) offset for each physical interface Note Offset Value can be 0, 30 or 50. If you are using Anyconnect on the client, it is recommended to use Offset 0. |
| Step 8 | <code>end</code> | Returns to privileged EXEC mode. |
| Step 9 | <code>show mka policy</code> | Verify your entries. |

Example

This example configures the MKA policy:

```
Switch(config)# mka policy mka_policy
Switch(config-mka-policy)# key-server priority 200
Switch(config-mka-policy)# macsec-cipher-suite gcm-aes-128
Switch(config-mka-policy)# confidentiality-offset 30
Switch(config-mka-policy)# end
```

Configuring Switch-to-host MACsec Encryption

Follow these steps to configure MACsec on an interface with one MACsec session for voice and one for data:

SUMMARY STEPS

1. `enable`
2. `configureterminal`
3. `interface interface-id`
4. `switchport access vlanvlan-id`
5. `switchport mode access`
6. `macsec`
7. `authentication event linksec fail action authorize vlan vlan-id`

8. authentication host-mode multi-domain
9. authentication linksec policy must-secure
10. authentication port-control auto
11. authentication periodic
12. authentication timer reauthenticate
13. authentication violation protect
14. mka policy *policy name*
15. dot1x pae authenticator
16. spanning-tree portfast
17. end
18. show authentication session interface *interface-id*
19. show authentication session interface *interface-id* details
20. show macsec interface *interface-id*
21. show mka sessions
22. copy running-config startup-config

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|---|
| Step 1 | enable Example: Switch>enable | Enables privileged EXEC mode. Enter the password if prompted. |
| Step 2 | configureterminal Example: Switch>configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> | Identify the MACsec interface, and enter interface configuration mode. The interface must be a physical interface. |
| Step 4 | switchport access vlan <i>vlan-id</i> | Configure the access VLAN for the port. |
| Step 5 | switchport mode access | Configure the interface as an access port. |
| Step 6 | macsec | Enable 802.1ae MACsec on the interface. The macsec command enables MKA MACsec on switch-to-host links (downlink ports) only. |
| Step 7 | authentication event linksec fail action authorize vlan <i>vlan-id</i> | (Optional) Specify that the switch processes authentication link-security failures resulting from unrecognized user credentials by authorizing a restricted VLAN on the port after a failed authentication attempt. |
| Step 8 | authentication host-mode multi-domain | Configure authentication manager mode on the port to allow both a host and a voice device to be authenticated on the 802.1x-authorized port. If not configured, the default host mode is single. |

| | Command or Action | Purpose |
|---------|--|--|
| Step 9 | authentication linksec policy must-secure | Set the LinkSec security policy to secure the session with MACsec if the peer is available. If not set, the default is <i>should secure</i> . |
| Step 10 | authentication port-control auto | Enable 802.1x authentication on the port. The port changes to the authorized or unauthorized state based on the authentication exchange between the switch and the client. |
| Step 11 | authentication periodic | Enable or Disable Reauthentication for this port . |
| Step 12 | authentication timer reauthenticate | Enter a value between 1 and 65535 (in seconds). Obtains re-authentication timeout value from the server. Default re-authentication time is 3600 seconds. |
| Step 13 | authentication violation protect | Configure the port to drop unexpected incoming MAC addresses when a new device connects to a port or when a device connects to a port after the maximum number of devices are connected to that port. If not configured, the default is to shut down the port. |
| Step 14 | mka policy <i>policy name</i> | Apply an existing MKA protocol policy to the interface, and enable MKA on the interface. If no MKA policy was configured (by entering the mka policy global configuration command). |
| Step 15 | dot1x pae authenticator | Configure the port as an 802.1x port access entity (PAE) authenticator. |
| Step 16 | spanning-tree portfast | Enable spanning tree Port Fast on the interface in all its associated VLANs. When Port Fast feature is enabled, the interface changes directly from a blocking state to a forwarding state without making the intermediate spanning-tree state changes |
| Step 17 | end Example: <code>Switch(config)#end</code> | Returns to privileged EXEC mode. |
| Step 18 | show authentication session interface <i>interface-id</i> | Verify the authorized session security status. |
| Step 19 | show authentication session interface <i>interface-id</i> details | Verify the details of the security status of the authorized session. |
| Step 20 | show macsec interface <i>interface-id</i> | Verify MacSec status on the interface. |
| Step 21 | show mka sessions | Verify the established mka sessions. |
| Step 22 | copy running-config startup-config Example: <code>Switch#copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring MACsec MKA using PSK

SUMMARY STEPS

1. **configure terminal**
2. **key chain** *key-chain-name* **macsec**
3. **key** *hex-string*
4. **cryptographic-algorithm** {*gcm-aes-128* | *gcm-aes-256*}
5. **key-string** { [*0/6/7*] *pwd-string* | *pwd-string*}
6. **lifetime local** [*start timestamp {hh::mm::ss | day | month | year}*] [**duration** *seconds* | *end timestamp {hh::mm::ss | day | month | year}*]
7. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | configure terminal | Enter global configuration mode. |
| Step 2 | key chain <i>key-chain-name</i> macsec | Configures a key chain and enters the key chain configuration mode. |
| Step 3 | key <i>hex-string</i> | Configures a unique identifier for each key in the keychain and enters the keychain's key configuration mode. Note For 128-bit encryption, use 32 hex digit key-string. For 256-bit encryption, use 64 hex digit key-string. |
| Step 4 | cryptographic-algorithm { <i>gcm-aes-128</i> <i>gcm-aes-256</i> } | Set cryptographic authentication algorithm with 128-bit or 256-bit encryption. |
| Step 5 | key-string { [<i>0/6/7</i>] <i>pwd-string</i> <i>pwd-string</i> } | Sets the password for a key string. Only hex characters must be entered. |
| Step 6 | lifetime local [<i>start timestamp {hh::mm::ss day month year}</i>] [duration <i>seconds</i> <i>end timestamp {hh::mm::ss day month year}</i>] | Sets the lifetime of the pre shared key. |
| Step 7 | end | Returns to privileged EXEC mode. |

Example

Following is an indicative example:

```
Switch(config)# Key chain keychain1 macsec
Switch(config-key-chain)# key 1000
Switch(config-keychain-key)# cryptographic-algorithm gcm-aes-128
Switch(config-keychain-key)# key-string 12345678901234567890123456789012
Switch(config-keychain-key)# lifetime local 12:12:00 July 28 2016 12:19:00 July
```

28 2016

Switch(config-keychain-key)# end

Configuring MACsec MKA on an Interface using PSK



Note To avoid traffic drop across sessions, the **mka policy** command must be configured before the **mka pre-shared-key key-chain** command.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **macsec network-link**
4. **mka policy** *policy-name*
5. **mka pre-shared-key key-chain** *key-chain name*
6. **macsec replay-protection window-size** *frame number*
7. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | configure terminal | Enters global configuration mode. |
| Step 2 | interface <i>interface-id</i> | Enters interface configuration mode. |
| Step 3 | macsec network-link | Enables MACsec on the interface. Note The macsec network-link command does not block MKA sessions for downlink ports. Use the macsec command instead. |
| Step 4 | mka policy <i>policy-name</i> | Configures an MKA policy. |
| Step 5 | mka pre-shared-key key-chain <i>key-chain name</i> | Configures an MKA pre-shared-key key-chain name. Note The MKA pre-shared key can be configured on either physical interface or sub-interfaces and not on both. |
| Step 6 | macsec replay-protection window-size <i>frame number</i> | Sets the MACsec window size for replay protection. |
| Step 7 | end | Returns to privileged EXEC mode. |

Example

Following is an indicative example:

```
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# mka policy mka_policy
Switch(config-if)# mka pre-shared-key key-chain key-chain-name
Switch(config-if)# macsec replay-protection window-size 10
Switch(config-if)# end
```

What to do next

It is not recommended to change the MKA policy on an interface with MKA PSK configured when the session is running. However, if a change is required, you must reconfigure the policy as follows:

1. Disable the existing session by removing `macsec network-link` configuration on each of the participating node using the **no macsec network-link** command
2. Configure the MKA policy on the interface on each of the participating node using the **mka policy policy-name** command.
3. Enable the new session on each of the participating node by using the **macsec network-link** command.

Configuring MACsec MKA using EAP-TLS

To configure MACsec with MKA on point-to-point links, perform these tasks:

- Configure Certificate Enrollment
 - Generate Key Pairs
 - Configure SCEP Enrollment
 - Configure Certificates Manually
- Configure an Authentication Policy
- Configure EAP-TLS Profiles and IEEE 802.1x Credentials
- Configure MKA MACsec using EAP-TLS on Interfaces

Generating Key Pairs

Procedure

| | Command or Action | Purpose |
|--------|---|---|
| Step 1 | <code>configure terminal</code> | Enter global configuration mode. |
| Step 2 | <code>crypto key generate rsa label <i>label-name</i> general-keys modulus <i>size</i></code> | Generates a RSA key pair for signing and encryption. You can also assign a label to each key pair using the label keyword. The label is referenced by the trustpoint that uses the key pair. If you do not assign a label, the key pair is automatically labeled <Default-RSA-Key>. If you do not use additional keywords this command generates one general purpose RSA key pair. If the modulus |

| | Command or Action | Purpose |
|---------------|--|--|
| | | is not specified, the default key modulus of 1024 is used. You can specify other modulus sizes with the modulus keyword. |
| Step 3 | end | Returns to privileged EXEC mode. |
| Step 4 | show authentication session interface <i>interface-id</i> | Verifies the authorized session security status. |
| Step 5 | copy running-config startup-config | (Optional) Saves your entries in the configuration file. |

Configuring Enrollment using SCEP

Simple Certificate Enrollment Protocol (SCEP) is a Cisco-developed enrollment protocol that uses HTTP to communicate with the certificate authority (CA) or registration authority (RA). SCEP is the most commonly used method for sending and receiving requests and certificates.

Procedure

| | Command or Action | Purpose |
|---------------|---|---|
| Step 1 | configure terminal | Enter global configuration mode. |
| Step 2 | crypto pki trustpoint <i>server name</i> | Declares the trustpoint and a given name and enters ca-trustpoint configuration mode. |
| Step 3 | enrollment url <i>url name pem</i> | Specifies the URL of the CA on which your device should send certificate requests. An IPv6 address can be added in the URL enclosed in brackets. For example: http:// [2001:DB8:1:1::1]:80. The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request. |
| Step 4 | rsakeypair <i>label</i> | Specifies which key pair to associate with the certificate. Note The rsakeypair name must match the trust-point name. |
| Step 5 | serial-number none | The none keyword specifies that a serial number will not be included in the certificate request. |
| Step 6 | ip-address none | The none keyword specifies that no IP address should be included in the certificate request. |
| Step 7 | revocation-check crl | Specifies CRL as the method to ensure that the certificate of a peer has not been revoked. |
| Step 8 | auto-enroll percent regenerate | Enables auto-enrollment, allowing the client to automatically request a rollover certificate from the CA. If auto-enrollment is not enabled, the client must be manually re-enrolled in your PKI upon certificate expiration. |

| | Command or Action | Purpose |
|---------|---|--|
| | | <p>By default, only the Domain Name System (DNS) name of the device is included in the certificate.</p> <p>Use the percent argument to specify that a new certificate will be requested after the percentage of the lifetime of the current certificate is reached.</p> <p>Use the regenerate keyword to generate a new key for the certificate even if a named key already exists.</p> <p>If the key pair being rolled over is exportable, the new key pair will also be exportable. The following comment will appear in the trustpoint configuration to indicate whether the key pair is exportable: “! RSA key pair associated with trustpoint is exportable.”</p> <p>It is recommended that a new key pair be generated for security reasons.</p> |
| Step 9 | crypto pki authenticate <i>name</i> | Retrieves the CA certificate and authenticates it. |
| Step 10 | exit | Exits global configuration mode. |
| Step 11 | show crypto pki certificate <i>trustpoint name</i> | Displays information about the certificate for the trust point. |

Configuring Enrollment Manually

If your CA does not support SCEP or if a network connection between the router and CA is not possible. Perform the following task to set up manual certificate enrollment:

Procedure

| | Command or Action | Purpose |
|--------|---|---|
| Step 1 | configure terminal | Enter global configuration mode. |
| Step 2 | crypto pki trustpoint <i>server name</i> | Declares the trustpoint and a given name and enters ca-trustpoint configuration mode. |
| Step 3 | enrollment url <i>url name pem</i> | <p>Specifies the URL of the CA on which your device should send certificate requests.</p> <p>An IPv6 address can be added in the URL enclosed in brackets. For example: <code>http:// [2001:DB8:1:1::1]:80</code>.</p> <p>The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.</p> |
| Step 4 | rsakeypair <i>label</i> | Specifies which key pair to associate with the certificate. |
| Step 5 | serial-number none | The none keyword specifies that a serial number will not be included in the certificate request. |

| | Command or Action | Purpose |
|---------|---|--|
| Step 6 | <code>ip-address none</code> | The none keyword specifies that no IP address should be included in the certificate request. |
| Step 7 | <code>revocation-check crl</code> | Specifies CRL as the method to ensure that the certificate of a peer has not been revoked. |
| Step 8 | <code>exit</code> | Exits Global Configuration mode. |
| Step 9 | <code>crypto pki authenticate <i>name</i></code> | Retrieves the CA certificate and authenticates it. |
| Step 10 | <code>crypto pki enroll <i>name</i></code> | <p>Generates certificate request and displays the request for copying and pasting into the certificate server.</p> <p>Enter enrollment information when you are prompted. For example, specify whether to include the device FQDN and IP address in the certificate request.</p> <p>You are also given the choice about displaying the certificate request to the console terminal.</p> <p>The base-64 encoded certificate with or without PEM headers as requested is displayed.</p> |
| Step 11 | <code>crypto pki import <i>name</i> certificate</code> | <p>Imports a certificate via TFTP at the console terminal, which retrieves the granted certificate.</p> <p>The device attempts to retrieve the granted certificate via TFTP using the same filename used to send the request, except the extension is changed from “.req” to “.crt”. For usage key certificates, the extensions “-sign.crt” and “-encr.crt” are used.</p> <p>The device parses the received files, verifies the certificates, and inserts the certificates into the internal certificate database on the switch.</p> <p>Note Some CAs ignore the usage key information in the certificate request and issue general purpose usage certificates. If your CA ignores the usage key information in the certificate request, only import the general purpose certificate. The router will not use one of the two key pairs generated.</p> |
| Step 12 | <code>exit</code> | Exits global configuration mode. |
| Step 13 | <code>show crypto pki certificate <i>trustpoint name</i></code> | Displays information about the certificate for the trust point. |
| Step 14 | <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Applying the 802.1x MACsec MKA Configuration on Interfaces

To apply MACsec MKA using EAP-TLS to interfaces, perform the following task:

Procedure

| | Command or Action | Purpose |
|---------|--|--|
| Step 1 | <code>configure terminal</code> | Enters global configuration mode. |
| Step 2 | <code>interface <i>interface-id</i></code> | Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface. |
| Step 3 | <code>macsec network-link</code> | Enables MACsec on the interface. |
| Step 4 | <code>authentication periodic</code> | Enables reauthentication for this port. |
| Step 5 | <code>authentication timer reauthenticate interval</code> | Sets the reauthentication interval. |
| Step 6 | <code>access-session host-mode multi-domain</code> | Allows hosts to gain access to the interface. |
| Step 7 | <code>access-session closed</code> | Prevents preauthentication access on the interface. |
| Step 8 | <code>access-session port-control auto</code> | Sets the authorization state of a port. |
| Step 9 | <code>dot1x pae both</code> | Configures the port as an 802.1X port access entity (PAE) supplicant and authenticator. |
| Step 10 | <code>dot1x credentials profile</code> | Assigns a 802.1x credentials profile to the interface. |
| Step 11 | <code>dot1x supplicant eap profile <i>name</i></code> | Assigns the EAP-TLS profile to the interface. |
| Step 12 | <code>service-policy type control subscriber <i>control-policy name</i></code> | Applies a subscriber control policy to the interface. |
| Step 13 | <code>exit</code> | Returns to privileged EXEC mode. |
| Step 14 | <code>show macsec interface</code> | Displays MACsec details for the interface. |
| Step 15 | <code>copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Configuring Cisco TrustSec MACsec

Configuring Cisco TrustSec Switch-to-Switch Link Security in Manual Mode

Before you begin

When manually configuring Cisco TrustSec on an interface, consider these usage guidelines and restrictions:

- If no SAP parameters are defined, Cisco TrustSec encapsulation or encryption is not performed.
- If you select GCM as the SAP operating mode, you must have a MACsec Encryption software license from Cisco. If you select GCM without the required license, the interface is forced to a link-down state.

- These protection levels are supported when you configure SAP pairwise master key (sap pmk):
 - SAP is not configured—no protection.
 - **sap mode-list gcm-encrypt gmac no-encap**—protection desirable but not mandatory.
 - **sap mode-list gcm-encrypt gmac**—confidentiality preferred and integrity required. The protection is selected by the supplicant according to supplicant preference.
 - **sap mode-list gmac**—integrity only.
 - **sap mode-list gcm-encrypt**—confidentiality required.
 - **sap mode-list gmac gcm-encrypt**—integrity required and preferred, confidentiality optional.
- Before changing the configuration from MKA to Cisco TrustSec SAP and vice versa, we recommend that you remove the interface configuration.

Beginning in privileged EXEC mode, follow these steps to manually configure Cisco TrustSec on an interface to another Cisco TrustSec device:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **cts manual**
4. **sap pmk** *key* [**mode-list** *mode1* [*mode2* [*mode3* [*mode4*]]]]
5. **no propagate sgt**
6. **exit**
7. **end**
8. **show cts interface** [*interface-id* | **brief** | **summary**]

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|---|--|
| Step 1 | configure terminal Example: Switch# configure terminal | Enters global configuration mode. |
| Step 2 | interface <i>interface-id</i> Example: Switch(config)# interface tengigabitethernet 1/1/2 | Note Enters interface configuration mode. |
| Step 3 | cts manual Example: Switch(config-if)# cts manual | Enters Cisco TrustSec manual configuration mode. |

| | Command or Action | Purpose |
|--------|---|---|
| Step 4 | <p>sap pmk <i>key</i> [mode-list <i>mode1</i> [<i>mode2</i> [<i>mode3</i> [<i>mode4</i>]]]]</p> <p>Example:</p> <pre>Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap</pre> | <p>(Optional) Configures the SAP pairwise master key (PMK) and operation mode. SAP is disabled by default in Cisco TrustSec manual mode.</p> <ul style="list-style-type: none"> • <i>key</i>—A hexadecimal value with an even number of characters and a maximum length of 32 characters. <p>The SAP operation mode options:</p> <ul style="list-style-type: none"> • gcm-encrypt—Authentication and encryption • Note Select this mode for MACsec authentication and encryption if your software license supports MACsec encryption. • gmac—Authentication, no encryption • no-encap—No encapsulation • null—Encapsulation, no authentication or encryption • Note If the interface is not capable of data link encryption, no-encap is the default and the only available SAP operating mode. SGT is not supported. |
| Step 5 | <p>no propagate sgt</p> <p>Example:</p> <pre>Switch(config-if-cts-manual)# no propagate sgt</pre> | <p>Use the no form of this command when the peer is incapable of processing a SGT. The no propagate sgt command prevents the interface from transmitting the SGT to the peer.</p> |
| Step 6 | <p>exit</p> <p>Example:</p> <pre>Switch(config-if-cts-manual)# exit</pre> | <p>Exits Cisco TrustSec 802.1x interface configuration mode.</p> |
| Step 7 | <p>end</p> <p>Example:</p> <pre>Switch(config-if)# end</pre> | <p>Returns to privileged EXEC mode.</p> |
| Step 8 | <p>show cts interface [<i>interface-id</i> brief summary]</p> | <p>(Optional) Verify the configuration by displaying TrustSec-related interface characteristics.</p> |

Example

This example shows how to configure Cisco TrustSec authentication in manual mode on an interface:

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap
Switch(config-if-cts-manual)# no propagate sgt
```

```
Switch(config-if-cts-manual)# exit
Switch(config-if)# end
```

Configuring MKA/MACsec for Port Channel

Configuring MKA/MACsec for Port Channel using PSK

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **macsec network-link**
4. **mka policy** *policy-name*
5. **mka pre-shared-key key-chain** *key-chain-name*
6. **channel-group** *channel-group-number* **mode** {**auto** | **desirable**} | {**active** | **passive**} | {**on**}
7. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | configure terminal | Enters global configuration mode. |
| Step 2 | interface <i>interface-id</i> | Enters interface configuration mode. |
| Step 3 | macsec network-link | Enables MACsec on the interface. Supports layer 2 and layer 3 port channels. |
| Step 4 | mka policy <i>policy-name</i> | Configures an MKA policy. |
| Step 5 | mka pre-shared-key key-chain <i>key-chain-name</i> | Configures an MKA pre-shared-key key-chain name. Note The MKA pre-shared key can be configured on either physical interface or sub-interfaces and not on both. |
| Step 6 | channel-group <i>channel-group-number</i> mode { auto desirable } { active passive } { on } | Configures the port in a channel group and sets the mode. The channel-number range is from 1 to 4096. The port channel associated with this channel group is automatically created if the port channel does not already exist. For mode, select one of the following keywords: <ul style="list-style-type: none"> • auto — Enables PAgP only if a PAgP device is detected. This places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. Note The auto keyword is not supported when EtherChannel members are from different switches in the switch stack. |

| | Command or Action | Purpose |
|---------------|-------------------|---|
| | | <ul style="list-style-type: none"> • desirable — Unconditionally enables PAgP. This places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. <ul style="list-style-type: none"> Note The desirable keyword is not supported when EtherChannel members are from different switches in the switch stack. • on — Forces the port to channel without PAgP or LACP. In the on mode, an EtherChannel exists only when a port group in the on mode is connected to another port group in the on mode. • active — Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • passive — Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. |
| Step 7 | <code>end</code> | Returns to privileged EXEC mode. |

Configuring Port Channel Logical Interfaces for Layer 2 EtherChannels

To create a port channel interface for a Layer 2 EtherChannel, perform this task:

SUMMARY STEPS

1. `configure terminal`
2. `[no] interface port-channel channel-group-number`
3. `switchport`
4. `switchport mode {access | trunk}`
5. `end`

DETAILED STEPS

| | Command or Action | Purpose |
|---------------|--|---|
| Step 1 | <code>configure terminal</code> | Enters global configuration mode. |
| Step 2 | <code>[no] interface port-channel <i>channel-group-number</i></code> | Creates the port channel interface. <ul style="list-style-type: none"> Note Use the no form of this command to delete the port channel interface. |
| Step 3 | <code>switchport</code> | Switches an interface that is in Layer 3 mode into Layer 2 mode for Layer 2 configuration. |

| | Command or Action | Purpose |
|--------|---|---|
| Step 4 | <code>switchport mode {access trunk}</code> | Assigns all ports as static-access ports in the same VLAN, or configure them as trunks. |
| Step 5 | <code>end</code> | Returns to privileged EXEC mode. |

Configuring Port Channel Logical Interfaces for Layer 3 EtherChannels

To create a port channel interface for a Layer 3 EtherChannel, perform this task:

SUMMARY STEPS

1. `configure terminal`
2. `interface interface-id`
3. `no switchport`
4. `ip address ip-address subnet_mask`
5. `end`

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | <code>configure terminal</code> | Enters global configuration mode. |
| Step 2 | <code>interface interface-id</code> | Enters interface configuration mode. |
| Step 3 | <code>no switchport</code> | Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration. |
| Step 4 | <code>ip address ip-address subnet_mask</code> | Assigns an IP address and subnet mask to the EtherChannel. |
| Step 5 | <code>end</code> | Returns to privileged EXEC mode. |

Configuring MACsec Cipher Announcement

Configuring an MKA Policy for Secure Announcement

SUMMARY STEPS

1. `configure terminal`
2. `mka policy policy-name`
3. `key-server priority`
4. `[no] send-secure-announcements`
5. `macsec-cipher-suite {gcm-aes-128 | gcm-aes-256}`
6. `end`
7. `show mka policy`

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--|--|
| Step 1 | <code>configure terminal</code> | Enter global configuration mode. |
| Step 2 | <code>mka policy <i>policy-name</i></code> | Identify an MKA policy, and enter MKA policy configuration mode. The maximum policy name length is 16 characters. Note The default MACsec cipher suite in the MKA policy will always be "GCM-AES-128". If the device supports both "GCM-AES-128" and "GCM-AES-256" ciphers, it is highly recommended to define and use a user defined MKA policy to include both 128 and 256 bits ciphers or only 256 bits cipher, as may be required. |
| Step 3 | <code>key-server <i>priority</i></code> | Configure MKA key server options and set priority (between 0-255). Note When value of key server priority is set to 255, the peer can not become the key server. The key server priority value is valid only for MKA PSK; and not for MKA EAPTLS. |
| Step 4 | <code>[no] send-secure-announcements</code> | Enables sending of secure announcements. Use the no form of the command to disable sending of secure announcements. By default, secure announcements are disabled. |
| Step 5 | <code>macsec-cipher-suite {<i>gcm-aes-128</i> <i>gcm-aes-256</i>}</code> | Configures cipher suite for deriving SAK with 128-bit or 256-bit encryption. |
| Step 6 | <code>end</code> | Returns to privileged EXEC mode. |
| Step 7 | <code>show mka policy</code> | Verify your entries. |

Configuring Secure Announcement Globally (Across all the MKA Policies)

SUMMARY STEPS

1. `configure terminal`
2. `[no] mka defaults policy send-secure-announcements`
3. `end`

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---------------------------------|----------------------------------|
| Step 1 | <code>configure terminal</code> | Enter global configuration mode. |

| | Command or Action | Purpose |
|--------|--|---|
| Step 2 | [no] mka defaults policy send-secure-announcements | Enables sending of secure announcements in MKPDUs across MKA policies. By default, secure announcements are disabled. |
| Step 3 | end | Returns to privileged EXEC mode. |

Configuring EAPoL Announcements on an interface

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. [no] **eapol announcement**
4. **end**

DETAILED STEPS

| | Command or Action | Purpose |
|--------|--------------------------------------|--|
| Step 1 | configure terminal | Enter global configuration mode. |
| Step 2 | interface <i>interface-id</i> | Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface. |
| Step 3 | [no] eapol announcement | Enable EAPoL announcements. Use the no form of the command to disable EAPoL announcements. By default, EAPoL announcements are disabled. |
| Step 4 | end | Returns to privileged EXEC mode. |

Configuration Examples for MACsec Encryption

Configuring Switch-to-host MACsec Encryption

Follow these steps to configure MACsec on an interface with one MACsec session for voice and one for data:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport access vlan** *vlan-id*
5. **switchport mode access**
6. **macsec**
7. **authentication event linksec fail action authorize vlan** *vlan-id*

8. authentication host-mode multi-domain
9. authentication linksec policy must-secure
10. authentication port-control auto
11. authentication periodic
12. authentication timer reauthenticate
13. authentication violation protect
14. mka policy *policy name*
15. dot1x pae authenticator
16. spanning-tree portfast
17. end
18. show authentication session interface *interface-id*
19. show authentication session interface *interface-id* details
20. show macsec interface *interface-id*
21. show mka sessions
22. copy running-config startup-config

DETAILED STEPS

| | Command or Action | Purpose |
|--------|---|---|
| Step 1 | enable Example: Switch>enable | Enables privileged EXEC mode. Enter the password if prompted. |
| Step 2 | configureterminal Example: Switch>configure terminal | Enters the global configuration mode. |
| Step 3 | interface <i>interface-id</i> | Identify the MACsec interface, and enter interface configuration mode. The interface must be a physical interface. |
| Step 4 | switchport access vlan <i>vlan-id</i> | Configure the access VLAN for the port. |
| Step 5 | switchport mode access | Configure the interface as an access port. |
| Step 6 | macsec | Enable 802.1ae MACsec on the interface. The macsec command enables MKA MACsec on switch-to-host links (downlink ports) only. |
| Step 7 | authentication event linksec fail action authorize vlan <i>vlan-id</i> | (Optional) Specify that the switch processes authentication link-security failures resulting from unrecognized user credentials by authorizing a restricted VLAN on the port after a failed authentication attempt. |
| Step 8 | authentication host-mode multi-domain | Configure authentication manager mode on the port to allow both a host and a voice device to be authenticated on the 802.1x-authorized port. If not configured, the default host mode is single. |

| | Command or Action | Purpose |
|---------|--|--|
| Step 9 | authentication linksec policy must-secure | Set the LinkSec security policy to secure the session with MACsec if the peer is available. If not set, the default is <i>should secure</i> . |
| Step 10 | authentication port-control auto | Enable 802.1x authentication on the port. The port changes to the authorized or unauthorized state based on the authentication exchange between the switch and the client. |
| Step 11 | authentication periodic | Enable or Disable Reauthentication for this port . |
| Step 12 | authentication timer reauthenticate | Enter a value between 1 and 65535 (in seconds). Obtains re-authentication timeout value from the server. Default re-authentication time is 3600 seconds. |
| Step 13 | authentication violation protect | Configure the port to drop unexpected incoming MAC addresses when a new device connects to a port or when a device connects to a port after the maximum number of devices are connected to that port. If not configured, the default is to shut down the port. |
| Step 14 | mka policy <i>policy name</i> | Apply an existing MKA protocol policy to the interface, and enable MKA on the interface. If no MKA policy was configured (by entering the mka policy global configuration command). |
| Step 15 | dot1x pae authenticator | Configure the port as an 802.1x port access entity (PAE) authenticator. |
| Step 16 | spanning-tree portfast | Enable spanning tree Port Fast on the interface in all its associated VLANs. When Port Fast feature is enabled, the interface changes directly from a blocking state to a forwarding state without making the intermediate spanning-tree state changes |
| Step 17 | end Example: <code>Switch(config)#end</code> | Returns to privileged EXEC mode. |
| Step 18 | show authentication session interface <i>interface-id</i> | Verify the authorized session security status. |
| Step 19 | show authentication session interface <i>interface-id</i> details | Verify the details of the security status of the authorized session. |
| Step 20 | show macsec interface <i>interface-id</i> | Verify MacSec status on the interface. |
| Step 21 | show mka sessions | Verify the established mka sessions. |
| Step 22 | copy running-config startup-config Example: <code>Switch#copy running-config startup-config</code> | (Optional) Saves your entries in the configuration file. |

Example: Configuring MACsec MKA for Port Channel using PSK

Etherchannel Mode — Static/On

The following is a sample configuration on Device 1 and Device 2 with EtherChannel Mode on.

```
key chain KC macsec
  key 1000
    cryptographic-algorithm aes-128-cmac
    key-string FC8F5B10557C192F03F60198413D7D45
  end

mka policy POLICY
  key-server priority 0
  macsec-cipher-suite gcm-aes-128
  confidentiality-offset 0
end

interface Tel/0/1
  channel-group 2 mode on
  macsec network-link
  mka policy POLICY
  mka pre-shared-key key-chain KC
end

interface Tel/0/2
  channel-group 2 mode on
  macsec network-link
  mka policy POLICY
  mka pre-shared-key key-chain KC
end
```

Layer 2 EtherChannel Configuration

Device 1

```
interface port-channel 2
  switchport
  switchport mode trunk
  no shutdown
end
```

Device 2

```
interface port-channel 2
  switchport
  switchport mode trunk
  no shutdown
end
```

The following shows a sample output of **show etherchannel summary** command.

```
Flags: D - down          P - bundled in port-channel
       I - stand-alone  s - suspended
       H - Hot-standby (LACP only)
       R - Layer3       S - Layer2
       U - in use       f - failed to allocate aggregator

       M - not in use, minimum links not met
```

```

u - unsuitable for bundling
w - waiting to be aggregated
d - default port

```

```
A - formed by Auto LAG
```

```

Number of channel-groups in use: 1
Number of aggregators:          1

```

```

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----

```

```

2      Po2 (RU)      -      Te1/0/1 (P)  Te1/0/2 (P)

```

Layer 3 EtherChannel Configuration

Device 1

```

interface port-channel 2
no switchport
ip address 10.25.25.3 255.255.255.0
no shutdown
end

```

Device 2

```

interface port-channel 2
no switchport
ip address 10.25.25.4 255.255.255.0
no shutdown
end

```

The following shows a sample output of **show etherchannel summary** command.

```

Flags:  D - down          P - bundled in port-channel
        I - stand-alone  s - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator

```

```

M - not in use, minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port

```

```
A - formed by Auto LAG
```

```

Number of channel-groups in use: 1
Number of aggregators:          1

```

```

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----

```

```
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2      Po2 (RU)          -          Te1/0/1 (P)  Te1/0/2 (P)
```

Etherchannel Mode — LACP

The following is a sample configuration on Device 1 and Device 2 with EtherChannel Mode as LACP.

```
key chain KC macsec
  key 1000
    cryptographic-algorithm aes-128-cmac
    key-string FC8F5B10557C192F03F60198413D7D45
  end

mka policy POLICY
  key-server priority 0
  macsec-cipher-suite gcm-aes-128
  confidentiality-offset 0
end

interface Tel/0/1
  channel-group 2 mode active
  macsec network-link
  mka policy POLICY
  mka pre-shared-key key-chain KC
end

interface Tel/0/2
  channel-group 2 mode active
  macsec network-link
  mka policy POLICY
  mka pre-shared-key key-chain KC
end
```

Layer 2 EtherChannel Configuration

Device 1

```
interface port-channel 2
  switchport
  switchport mode trunk
  no shutdown
end
```

Device 2

```
interface port-channel 2
  switchport
  switchport mode trunk
  no shutdown
end
```

The following shows a sample output of **show etherchannel summary** command.

```
Flags:  D - down           P - bundled in port-channel
        I - stand-alone   s - suspended
        H - Hot-standby (LACP only)
        R - Layer3       S - Layer2
        U - in use       f - failed to allocate aggregator
```



```
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2      Po2 (RU)          LACP      Te1/1/1 (P)  Te1/1/2 (P)
```

Etherchannel Mode — PAgP

The following is a sample configuration on Device 1 and Device 2 with EtherChannel Mode as PAgP.

```
key chain KC macsec
  key 1000
    cryptographic-algorithm aes-128-cmac
    key-string FC8F5B10557C192F03F60198413D7D45
  end

mka policy POLICY
  key-server priority 0
  macsec-cipher-suite gcm-aes-128
  confidentiality-offset 0
end

interface Tel/0/1
  channel-group 2 mode desirable
  macsec network-link
  mka policy POLICY
  mka pre-shared-key key-chain KC
end

interface Tel/0/2
  channel-group 2 mode desirable
  macsec network-link
  mka policy POLICY
  mka pre-shared-key key-chain KC
end
```

Layer 2 EtherChannel Configuration

Device 1

```
interface port-channel 2
  switchport
  switchport mode trunk
  no shutdown
end
```

Device 2

```
interface port-channel 2
  switchport
  switchport mode trunk
  no shutdown
end
```

The following shows a sample output of **show etherchannel summary** command.

```
Flags:  D - down          P - bundled in port-channel
        I - stand-alone  s - suspended
        H - Hot-standby (LACP only)
        R - Layer3      S - Layer2
        U - in use      f - failed to allocate aggregator
```



```

MACsec Desired..... YES

# of MACsec Capable Live Peers..... 1
# of MACsec Capable Live Peers Responded.. 1

Live Peers List:
  MI                               MN           Rx-SCI (Peer)           KS Priority
  -----
  38046BA37D7DA77E06D006A9  89560       c800.8459.e764/002a    10

Potential Peers List:
  MI                               MN           Rx-SCI (Peer)           KS Priority
  -----

Dormant Peers List:
  MI                               MN           Rx-SCI (Peer)           KS Priority
  -----

```

The following is a sample output of the **show mka policy policy-name detail** command with secure announcement disabled.

```

Device# show mka policy p2 detail
MKA Policy Configuration ("p2")
=====
MKA Policy Name..... p2
Key Server Priority.... 2
Confidentiality Offset. 0
Send Secure Announcement..DISABLED
Cipher Suite(s)..... GCM-AES-128

Applied Interfaces...
  GigabitEthernet1/0/1

```

Example: Cisco TrustSec Switch-to-Switch Link Security Configuration

This example shows the configuration necessary for a seed and non-seed device for Cisco TrustSec switch-to-switch security. You must configure the AAA and RADIUS for link security. In this example, ACS-1 through ACS-3 can be any server names and cts-radius is the Cisco TrustSec server.

Seed Device Configuration:

```

Switch(config)#aaa new-model
Switch(config)#radius server ACS-1
Switch(config-radius-server)#address ipv4 10.5.120.12 auth-port 1812 acct-port
1813
Switch(config-radius-server)#pac key cisco123
Switch(config-radius-server)#exit
Switch(config)#radius server ACS-2
Switch(config-radius-server)#address ipv4 10.5.120.14 auth-port 1812 acct-port
1813
Switch(config-radius-server)#pac key cisco123
Switch(config-radius-server)#exit
Switch(config)#radius server ACS-3

```

```
Switch(config-radius-server)#address ipv4 10.5.120.15 auth-port 1812 acct-port
1813
Switch(config-radius-server)#pac key cisco123
Switch(config-radius-server)#exit
Switch(config)#aaa group server radius cts-radius
Switch(config-sg-radius)#server name ACS-1
Switch(config-sg-radius)#server name ACS-2
Switch(config-sg-radius)#server name ACS-3
Switch(config-sg-radius)#exit
Switch(config)#aaa authentication login default none
Switch(config)#aaa authentication dot1x default group cts-radius
Switch(config)#aaa authorization network cts-radius group cts-radius
Switch(config)#aaa session-id common
Switch(config)#cts authorization list cts-radius
Switch(config)#dot1x system-auth-control

Switch(config)#interface gil/1/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#cts manual
Switch(config-if-cts-manual)#sap pmk 0 abcd mode-list gcm-encrypt gmac

Switch(config-if-cts-manual)#exit
Switch(config-if)#exit

Switch(config)#interface gil/1/4
Switch(config-if)#switchport mode trunk
Switch(config-if)#cts manual
Switch(config-if-cts-manual)#sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
gmac
Switch(config-if-cts-manual)#no propagate sgt
Switch(config-if-cts-manual)#exit
Switch(config-if)#exit

Switch(config)#radius-server vsa send authentication
Switch(config)#end
Switch#cts credentials id cts-36 password trustsec123
```

Non-Seed Device:

```
Switch(config)#aaa new-model
Switch(config)#aaa session-id common
Switch(config)#dot1x system-auth-control

Switch(config)#interface gil/1/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#shutdown
Switch(config-if)#cts manual
Switch(config-if-cts-manual)#sap pmk 0 abcd mode-list gcm-encrypt gmac
Switch(config-if-cts-manual)#exit
Switch(config-if)#exit

Switch(config)#interface gil/1/4
Switch(config-if)#switchport mode trunk
Switch(config-if)#shutdown
Switch(config-if)#cts manual
```

```
Switch(config-if-cts-manual)#sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
gmac
Switch(config-if-cts-manual)#no propagate sgt
Switch(config-if-cts-manual)#exit
Switch(config-if)#exit

Switch(config)#radius-server vsa send authentication
Switch(config)#cts credentials id cts-72 password trustsec123
Switch(config)#end
```