



MACsec Encryption

- [Finding Feature Information, on page 1](#)
- [Prerequisites for MACsec Encryption, on page 1](#)
- [Restrictions for MACsec Encryption, on page 2](#)
- [Information About MACsec Encryption, on page 2](#)
- [How to Configure MACsec Encryption, on page 8](#)
- [Configuration Examples for MACsec Encryption, on page 29](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for MACsec Encryption

Prerequisites for Certificate-Based MACsec

- 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.

Restrictions for MACsec Encryption

- MACsec is supported only on the first 16 downlink ports and on all the uplink ports of C9300-48UXM switch model.
- MACsec configuration is not supported on EtherChannel ports.
- Bidirectional Forwarding and Detection (BFD) is not supported on both MKA pre-shared keys (PSK) and MKA Extensible Authentication Protocol (EAP-TLS).
- The MACsec Cipher announcement is not supported for MACsec XPN Ciphers.
- Certificated based MACSec (EAP-TLS) is not supported if the access-session mode is configured as open.
- MACsec XPN Cipher Suites are not supported in MACsec connections.
- If the **dot1q tag vlan native** command is configured globally, the dot1x reauthentication will fail on trunk ports.
- MACsec with Precision Time Protocol (PTP) is not supported.
- MACsec is not supported on Locator ID Separation Protocol (LISP) interfaces and Cisco Software-Defined Access (SD-Access) solution.
- MACsec is not supported with Multicast VPN (mVPN).

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.



Note When switch-to-switch MACSec is enabled, all traffic is encrypted, except the EAP-over-LAN (EAPOL) packets.

Link layer security can include both packet authentication between switches and MACsec encryption between switches (encryption is optional).

Table 1: MACsec Support on Switch Ports

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



Note Switch-to-host connection is not supported on downlink ports in Cisco IOS XE Everest 16.5.1a.

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 only on switch-to-switch links (uplink). 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.

Recommendations for MACsec Encryption

This section lists the recommendations for configuring MACsec encryption:

- Use the confidentiality (encryption) offset as 0 in switch-to-host connections.
- Use Bidirectional Forwarding and Detection (BFD) timer value as 750 milliseconds for 10Gbps ports and 1.25 seconds for any port with speed above 10Gbps.
- Execute the **shutdown** command, and then the **no shutdown** command on a port, after changing any MKA policy or MACsec configuration for active sessions, so that the changes are applied to active sessions.
- Use Extended Packet Numbering (XPN) Cipher Suite for port speeds of 40Gbps and above.
- Set the connectivity association key (CAK) rekey overlap timer to 30 seconds or more.
- Do not use Cisco TrustSec Security Association Protocol (SAP) MACsec encryption for port speeds above 10Gbps.
- Do not enable both Cisco TrustSec SAP and uplink MKA at the same time on any interface.

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 the uplink. 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.

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

MKA Statistics

Some MKA counters are aggregated globally, while others are updated both globally and per session. You can also obtain information about the status of MKA sessions. See [Example: Displaying MKA Information, on page 32](#) for further information.

Key Lifetime and Hitless Key Rollover

A MACsec key chain can have multiple pre-shared keys (PSK) each configured with a key id and an optional lifetime. A key lifetime specifies at which time the key expires. In the absence of a lifetime configuration, the default lifetime is unlimited. When a lifetime is configured, MKA rolls over to the next configured pre-shared key in the key chain after the lifetime is expired. Time zone of the key can be local or UTC. Default time zone is UTC.

You can Key rolls over to the next key within the same key chain by configuring a second key in the key chain and configuring a lifetime for the first key. When the lifetime of the first key expires, it automatically rolls over to the next key in the list. If the same key is configured on both sides of the link at the same time, then the key rollover is hitless, that is, key rolls over without traffic interruption.

On all participating devices, the MACsec key chain must be synchronised by using Network Time Protocol (NTP) and the same time zone must be used. If all the participating devices are not synchronized, the connectivity association key (CAK) rekey will not be initiated on all the devices at the same time.



Note The lifetime of the keys need to be overlapped in order to achieve hitless key rollover.

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.

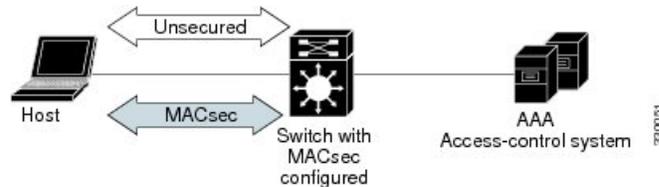


Note Switch-to-host connection is not supported on downlink ports in Cisco IOS XE Everest 16.5.1a. Support will be added in future releases.

Single-Host Mode

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

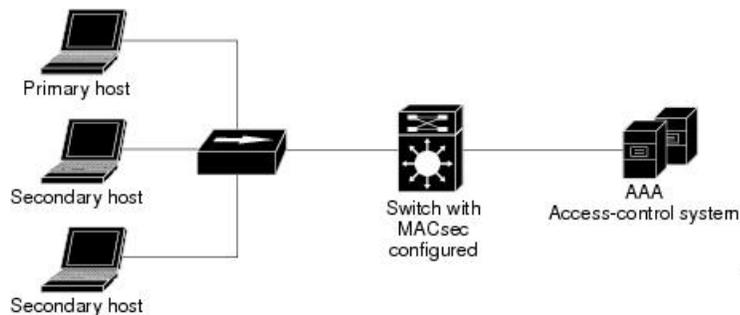
Figure 1: MACsec in Single-Host Mode with a Secured Data Session



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.

Understanding 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.
- MKA/EAPTLS does not block the unauthenticated supplicant from sending bridge protocol data units (BPDUs) under Spanning Tree Protocol (STP). But the BPDUs can be rate-limited using QoS profile.
- Certificated based MACSec (EAPTLS) is not supported if the access-session mode is configured as open.

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>
Endpoint Admission Control (EAC)	<p>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).</p>
Network Device Admission Control (NDAC)	<p>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.</p>
Security Association Protocol (SAP)	<p>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.</p>
Security Group Tag (SGT)	<p>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.</p>
SGT Exchange Protocol (SXP)	<p>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.</p>

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

Procedure

	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.
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.
Step 4	<code>macsec-cipher-suite <i>gcm-aes-128</i></code>	Configures cipher suite for deriving SAK with 128-bit encryption.
Step 5	<code>confidentiality-offset <i>Offset value</i></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.

	Command or Action	Purpose
Step 6	end	Returns to privileged EXEC mode.
Step 7	show mka policy	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:

Procedure

	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	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 7	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 8	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 9	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 10	authentication periodic	Enable or Disable Reauthentication for this port .
Step 11	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 12	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 13	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 14	dot1x pae authenticator	Configure the port as an 802.1x port access entity (PAE) authenticator.
Step 15	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 16	end Example: <code>Switch(config)#end</code>	Returns to privileged EXEC mode.
Step 17	show authentication session interface <i>interface-id</i>	Verify the authorized session security status.
Step 18	show authentication session interface <i>interface-id</i> details	Verify the details of the security status of the authorized session.
Step 19	show macsec interface <i>interface-id</i>	Verify MacSec status on the interface.

	Command or Action	Purpose
Step 20	<code>show mka sessions</code>	Verify the established mka sessions.
Step 21	<code>copy running-config startup-config</code> Example: <code>Switch#copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring MACsec MKA using PSK

Procedure

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enter global configuration mode.
Step 2	<code>key chain key-chain-name macsec</code>	Configures a key chain and enters the key chain configuration mode.
Step 3	<code>key hex-string</code>	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.
Step 4	<code>cryptographic-algorithm {gcm-aes-128}</code>	Set cryptographic authentication algorithm with 128-bit encryption.
Step 5	<code>key-string { [0/6/7] pwd-string pwd-string }</code>	Sets the password for a key string. Only hex characters must be entered.
Step 6	<code>lifetime local [start timestamp {hh::mm::ss / day / month / year}] [duration seconds end timestamp {hh::mm::ss / day / month / year}]</code>	Sets the lifetime of the pre shared key.
Step 7	<code>end</code>	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

Procedure

	Command or Action	Purpose
Step 1	<code>configure terminal</code>	Enters global configuration mode.
Step 2	<code>interface <i>interface-id</i></code>	Enters interface configuration mode.
Step 3	<code>macsec network-link</code>	Enables MACsec on the interface.
Step 4	<code>mka policy <i>policy-name</i></code>	Configures an MKA policy.
Step 5	<code>mka pre-shared-key key-chain <i>key-chain name</i></code>	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	<code>macsec replay-protection window-size <i>frame number</i></code>	Sets the MACsec window size for replay protection.
Step 7	<code>end</code>	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

Remote Authentication

Generating Key Pairs

Procedure

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	crypto key generate rsa label <i>label-name</i> general-keys modulus <i>size</i>	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 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 <i>percent regenerate</i>	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. By default, only the Domain Name System (DNS) name of the device is included in the certificate. Use the percent argument to specify that a new certificate will be requested after the

	Command or Action	Purpose
		<p>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: http://[2001:DB8:1:1::1]:80.</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.

	Command or Action	Purpose
Step 5	<code>serial-number none</code>	The none keyword specifies that a serial number will not be included in the certificate request.
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 “.cert”. For usage key certificates, the extensions “-sign.cert” and “-encr.cert” 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>

	Command or Action	Purpose
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.

Enabling 802.1x Authentication and Configuring AAA

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>aaa new-model</code>	Enables AAA.
Step 4	<code>dot1x system-auth-control</code>	Enables 802.1X on your device.
Step 5	<code>radius server <i>name</i></code>	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode.
Step 6	<code>address <i>ip-address</i> auth-port <i>port-number</i></code> <code>acct-port <i>port-number</i></code>	Configures the IPv4 address for the RADIUS server accounting and authentication parameters.
Step 7	<code>automate-tester username <i>username</i></code>	Enables the automated testing feature for the RADIUS server. With this practice, the device sends periodic test authentication messages to the RADIUS server. It looks for a RADIUS response from the server. A success message is not necessary - a failed authentication suffices, because it shows that the server is alive.
Step 8	<code>key <i>string</i></code>	Configures the authentication and encryption key for all RADIUS communications between the device and the RADIUS server.
Step 9	<code>radius-server deadtime <i>minutes</i></code>	Improves RADIUS response time when some servers might be unavailable and skips unavailable servers immediately.
Step 10	<code>exit</code>	Returns to global configuration mode.

	Command or Action	Purpose
Step 11	aaa group server radius <i>group-name</i>	Groups different RADIUS server hosts into distinct lists and distinct methods, and enters server group configuration mode.
Step 12	server <i>name</i>	Assigns the RADIUS server name.
Step 13	exit	Returns to global configuration mode.
Step 14	aaa authentication dot1x default group <i>group-name</i>	Sets the default authentication server group for IEEE 802.1x.
Step 15	aaa authorization network default group <i>group-name</i>	Sets the network authorization default group.

Configuring EAP-TLS Profile and 802.1x Credentials

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	eap profile <i>profile-name</i>	Configures EAP profile and enters EAP profile configuration mode.
Step 4	method tls	Enables EAP-TLS method on the device.
Step 5	pki-trustpoint <i>name</i>	Sets the default PKI trustpoint.
Step 6	exit	Returns to global configuration mode.
Step 7	dot1x credentials <i>profile-name</i>	Configures 802.1x credentials profile and enters dot1x credentials configuration mode.
Step 8	username <i>username</i>	Sets the authentication user ID.
Step 9	pki-trustpoint <i>name</i>	Sets the default PKI trustpoint.
Step 10	end	Returns to privileged EXEC mode.

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	configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<code>interface interface-id</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 name</code>	Assigns the EAP-TLS profile to the interface.
Step 12	<code>service-policy type control subscriber control-policy name</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.

Local Authentication

Configuring the EAP Credentials using Local Authentication

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>aaa new-model</code>	Enables AAA.

	Command or Action	Purpose
Step 4	aaa local authentication default authorization default	Sets the default local authentication and default local authorization method.
Step 5	aaa authentication dot1x default local	Sets the default local username authentication list for IEEE 802.1x.
Step 6	aaa authorization network default local	Sets an authorization method list for local user.
Step 7	aaa authorization credential-download default local	Sets an authorization method list for use of local credentials.
Step 8	exit	Returns to privileged EXEC mode.

Configuring the Local EAP-TLS Authentication and Authorization Profile

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	aaa new-model	Enables AAA.
Step 4	dot1x credentials <i>profile-name</i>	Configures the dot1x credentials profile and enters dot1x credentials configuration mode.
Step 5	username <i>name</i> password <i>password</i>	Sets the authentication user ID and password.
Step 6	exit	Returns to global configuration mode.
Step 7	aaa attribute list <i>list-name</i>	(Optional) Sets the AAA attribute list definition and enters attribute list configuration mode.
Step 8	aaa attribute type linksec-policy must-secure	(Optional) Specifies the AAA attribute type.
Step 9	exit	Returns to global configuration mode.
Step 10	username <i>name</i> aaa attribute list <i>name</i>	(Optional) Specifies the AAA attribute list for the user ID.
Step 11	end	Returns to privileged EXEC mode.

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	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 4	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 5	rsakeypair <i>label</i>	Specifies which key pair to associate with the certificate. Note The rsakeypair name must match the trust-point name.
Step 6	serial-number none	The none keyword specifies that a serial number will not be included in the certificate request.
Step 7	ip-address none	The none keyword specifies that no IP address should be included in the certificate request.
Step 8	revocation-check <i>crl</i>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 9	auto-enroll <i>percent regenerate</i>	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. By default, only the Domain Name System (DNS) name of the device is included in the certificate.

	Command or Action	Purpose
		<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 10	crypto pki authenticate <i>name</i>	Retrieves the CA certificate and authenticates it.
Step 11	exit	Exits global configuration mode.
Step 12	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	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>server name</i>	Declares the trustpoint and a given name and enters ca-trustpoint configuration mode.
Step 4	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: http://[2001:DB8:1:1::1]:80.</p> <p>The pem keyword adds privacy-enhanced mail (PEM) boundaries to the certificate request.</p>

	Command or Action	Purpose
Step 5	<code>rsakeypair <i>label</i></code>	Specifies which key pair to associate with the certificate.
Step 6	<code>serial-number none</code>	The none keyword specifies that a serial number will not be included in the certificate request.
Step 7	<code>ip-address none</code>	The none keyword specifies that no IP address should be included in the certificate request.
Step 8	<code>revocation-check <i>crl</i></code>	Specifies CRL as the method to ensure that the certificate of a peer has not been revoked.
Step 9	<code>exit</code>	Exits Global Configuration mode.
Step 10	<code>crypto pki authenticate <i>name</i></code>	Retrieves the CA certificate and authenticates it.
Step 11	<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 12	<code>crypto pki import <i>name certificate</i></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>

	Command or Action	Purpose
		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.
Step 13	<code>exit</code>	Exits Global Configuration mode.
Step 14	<code>show crypto pki certificate <i>trustpoint name</i></code>	Displays information about the certificate for the trust point.
Step 15	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring EAP-TLS Profile and 802.1x Credentials

Procedure

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	<code>configure terminal</code>	Enters global configuration mode.
Step 3	<code>eap profile <i>profile-name</i></code>	Configures EAP profile and enters EAP profile configuration mode.
Step 4	<code>method tls</code>	Enables EAP-TLS method on the device.
Step 5	<code>pki-trustpoint <i>name</i></code>	Sets the default PKI trustpoint.
Step 6	<code>exit</code>	Returns to global configuration mode.
Step 7	<code>dot1x credentials <i>profile-name</i></code>	Configures 802.1x credentials profile and enters dot1x credentials configuration mode.
Step 8	<code>username <i>username</i></code>	Sets the authentication user ID.
Step 9	<code>pki-trustpoint <i>name</i></code>	Sets the default PKI trustpoint.
Step 10	<code>end</code>	Returns to privileged EXEC mode.

Applying the 802.1x MKA MACsec Configuration on Interfaces

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

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Identifies the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 4	macsec	Enables MACsec on the interface.
Step 5	authentication periodic	Enables reauthentication for this port.
Step 6	authentication timer reauthenticate interval	Sets the reauthentication interval.
Step 7	access-session host-mode multi-domain	Allows hosts to gain access to the interface.
Step 8	access-session closed	Prevents preauthentication access on the interface.
Step 9	access-session port-control auto	Sets the authorization state of a port.
Step 10	dot1x pae both	Configures the port as an 802.1X port access entity (PAE) supplicant and authenticator.
Step 11	dot1x credentials profile	Assigns a 802.1x credentials profile to the interface.
Step 12	dot1x authenticator eap profile <i>name</i>	Assigns the EAP-TLS authenticator profile to the interface.
Step 13	dot1x supplicant eap profile <i>name</i>	Assigns the EAP-TLS supplicant profile to the interface.
Step 14	service-policy type control subscriber <i>control-policy name</i>	Applies a subscriber control policy to the interface.
Step 15	exit	Returns to privileged EXEC mode.
Step 16	show macsec interface	Displays MACsec details for the interface.
Step 17	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Verifying MACsec MKA using EAP-TLS

Use the following **show** commands to verify the configuration of MACsec MKA using EAP-TLS. Given below are the sample outputs of the **show** commands.


```

                Status: Authorized
                Domain: DATA
    Oper host mode: multi-host
    Oper control dir: both
    Session timeout: N/A
    Common Session ID: 00000000000000BB72E8AFA
    Acct Session ID: Unknown
                Handle: 0xc3000001
    Current Policy: MUSTS_1

```

```

Local Policies:
  Security Policy: Must Secure
  Security Status: Link Secured

```

```
Server Policies:
```

```

Method status list:
  Method          State
  dot1xSup        Authc Success
  dot1x            Authc Success

```

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.
- When CTS is configured on an interface and the System MTU is set to a value greater than 9191, the resulting packet size is limited to 9190.
- 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:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 2	interface interface-id 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.
Step 4	sap pmk key [mode-list mode1 [mode2 [mode3 [mode4]]]] Example: Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap	<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 • gmac—Authentication, no encryption • no-encap—No encapsulation • null—Encapsulation, no authentication or encryption <p>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.</p>

	Command or Action	Purpose
Step 5	no propagate sgt Example: Switch(config-if-cts-manual)# no propagate sgt	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.
Step 6	exit Example: Switch(config-if-cts-manual)# exit	Exits Cisco TrustSec 802.1x interface configuration mode.
Step 7	end Example: Switch(config-if)# end	Returns to privileged EXEC mode.
Step 8	show cts interface [<i>interface-id</i> brief summary]	(Optional) Verify the configuration by displaying TrustSec-related interface characteristics.

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
```

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:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. Enter the password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Switch> configure terminal	Enters the global configuration mode.
Step 3	interface interface-id	Identify the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 4	switchport access vlan vlan-id	Configure the access VLAN for the port.
Step 5	switchport mode access	Configure the interface as an access port.
Step 6	authentication event linksec fail action authorize vlan vlan-id	(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 7	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.
Step 8	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 9	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 10	authentication periodic	Enable or Disable Reauthentication for this port .
Step 11	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 12	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 13	mka policy policy name	Apply an existing MKA protocol policy to the interface, and enable MKA on the interface. If no MKA policy was configured (by entering

	Command or Action	Purpose
		the mka policy global configuration command).
Step 14	dot1x pae authenticator	Configure the port as an 802.1x port access entity (PAE) authenticator.
Step 15	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 16	end Example: <code>Switch(config)#end</code>	Returns to privileged EXEC mode.
Step 17	show authentication session interface <i>interface-id</i>	Verify the authorized session security status.
Step 18	show authentication session interface <i>interface-id</i> details	Verify the details of the security status of the authorized session.
Step 19	show macsec interface <i>interface-id</i>	Verify MacSec status on the interface.
Step 20	show mka sessions	Verify the established mka sessions.
Step 21	copy running-config startup-config Example: <code>Switch#copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuration Examples for MACsec MKA using EAP-TLS

Example: Enrolling the Certificate

Configure Crypto PKI Trustpoint:

```
crypto pki trustpoint POLESTAR-IOS-CA
  enrollment terminal
  subject-name CN=ASR1000x1@polestar.com, C=IN, ST=KA, OU=ENG,O=Polestar
  revocation-check none
  rsakeypair mkaioscarsa
  storage nvram
!
```

Manual Installation of Root CA certificate:

```
crypto pki authenticate POLESTAR-IOS-CA
```

Example: Enabling 802.1x Authentication and AAA Configuration

```

aaa new-model
dot1x system-auth-control
radius server ISE
  address ipv4 <ISE ipv4 address> auth-port 1645 acct-port 1646
  automate-tester username dummy
  key dummy123
  radius-server deadtime 2
!
aaa group server radius ISEGRP
  server name ISE
!
aaa authentication dot1x default group ISEGRP
aaa authorization network default group ISEGRP

```

Example: Configuring EAP-TLS Profile and 802.1X Credentials

```

eap profile EAPTLS-PROF-IOSCA
  method tls
  pki-trustpoint POLESTAR-IOS-CA
!

dot1x credentials EAPTLSCRED-IOSCA
  username asr1000@polestar.company.com
  pki-trustpoint POLESTAR-IOS-CA
!

```

Example: Applying 802.1X, PKI, and MACsec Configuration on the Interface

```

interface TenGigabitEthernet0/1
  macsec network-link
  authentication periodic
  authentication timer reauthenticate <reauthentication interval>
  access-session host-mode multi-host
  access-session closed
  access-session port-control auto
  dot1x pae both
  dot1x credentials EAPTLSCRED-IOSCA
  dot1x supplicant eap profile EAPTLS-PROF-IOSCA
  service-policy type control subscriber DOT1X_POLICY_RADIUS

```

Example: Displaying MKA Information

The following is sample output from the **show mka sessions** command.

```

Device# show mka sessions

Total MKA Sessions..... 1
  Secured Sessions... 1
  Pending Sessions... 0

```



```

Latest SAK AN..... 0
Latest SAK KI (KN)..... D46CBEC05D5D67594543CEAE00000001 (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)

MKA Policy Name..... p2
Key Server Priority..... 2
Delay Protection..... NO
Replay Protection..... YES
Replay Window Size..... 0
Confidentiality Offset... 0
Algorithm Agility..... 80C201
Send Secure Announcement.. DISABLED
SAK Cipher Suite..... 0080C20001000001 (GCM-AES-128)
MACsec Capability..... 3 (MACsec Integrity, Confidentiality, & Offset)
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 sample output from the **show mka policy** command:

```
Device# show mka policy
```

MKA Policy Summary...

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

Example: Displaying MKA Information

```
p2                2                FALSE    TRUE    0        0        GCM-AES-128
  Gil/0/1
```

The following is sample output from the **show mka policy *policy-name*** command:

```
Device# show mka policy p2
```

```
MKA Policy Summary...
```

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

The following is sample output from the **show mka policy *policy-name* detail** command:

```
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
```

The following is sample output from the **show mka statistics interface *interface-name*** command:

```
Device# show mka statistics interface GigabitEthernet 1/0/1
```

```
MKA Statistics for Session
```

```
=====
```

```
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..... 1
SAKs Rekeyed..... 0
SAKs Received..... 0
SAK Responses Received.. 1
```

```
MKPDU Statistics
```



```

"Distributed SAK"..... 1
"Distributed CAK"..... 0

MKA Error Counter Totals
=====
Session Failures
  Bring-up Failures..... 0
  Reauthentication Failures..... 0
  Duplicate Auth-Mgr Handle..... 0

SAK Failures
  SAK Generation..... 0
  Hash Key Generation..... 0
  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

```

Cisco TrustSec Switch-to-Switch Link Security Configuration Example

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 dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac

Switch(config-if-cts-dot1x)#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-dot1x)#sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
gmac
Switch(config-if-cts-dot1x)#no propagate sgt
Switch(config-if-cts-dot1x)#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 dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac
```

```
Switch(config-if-cts-dot1x)#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-dot1x)#sap pmk 033445AABBCCDDEEFF mode-list gcm-encrypt
gmac
Switch(config-if-cts-dot1x)#no propagate sgt
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit

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