



# CHAPTER 1

## Configuring MACsec Encryption

This chapter describes how to configure Media Access Control Security (MACsec) encryption on the Catalyst 3750-X and 3560-X switch. MACsec is the IEEE 802.1AE standard for authenticating and encrypting packets between two MACsec-capable devices. The Catalyst 3750-X and 3560-X switches support 802.1AE encryption with MACsec Key Agreement (MKA) on downlink ports for encryption between the switch and host devices. The switch also supports MACsec link layer switch-to-switch security by using Cisco TrustSec Network Device Admission Control (NDAC) and the Security Association Protocol (SAP) key exchange. Link layer security can include both packet authentication between switches and MACsec encryption between switches (encryption is optional).



### Note

MACsec is not supported on switches running the NPE or the LAN base image.

All downlink ports on the switch can run Cisco TrustSec MACsec link layer switch-to-switch security.

**Table 1**      **MACsec Support on Switch Ports**

Interface	Connections	MACsec support
User-facing downlink ports	Switch-to-host	MKA MACsec encryption
Switchports connected to other switches	Switch-to-switch	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 meant for switch-to-host facing links and is not supported on switch-to-switch links. Host-facing links typically use flexible authentication ordering for handling heterogeneous devices with or without IEEE 802.1x, and can optionally use MKA 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.

- [Understanding Media Access Control Security and MACsec Key Agreement, page 1-2](#)
- [Configuring MKA and MACsec, page 1-6](#)
- [Understanding Cisco TrustSec MACsec, page 1-8](#)
- [Configuring Cisco TrustSec MACsec, page 1-10](#)

# Understanding 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) framework. Only host facing links (links between network access devices and endpoint devices such as a PC or IP phone) can be secured using MACsec.

A switch using MACsec accepts either MACsec or non-MACsec frames, depending on the policy associated with the client. MACsec frames are encrypted and protected with an integrity check value (ICV). When the switch receives frames from the client, 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 client) 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). Because the switch is the authenticator, it is also the key server, generating a random 128-bit secure association key (SAK), which it sends it 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 client disconnects, the participant on the switch continues to operate MKA until 6 seconds have elapsed after the last MKPDU is received from the client.

These sections provide more details:

- [MKA Policies, page 1-2](#)
- [Virtual Ports, page 1-3](#)
- [MACsec and Stacking, page 1-3](#)
- [MACsec, MKA and 802.1x Host Modes, page 1-4](#)
- [MKA Statistics, page 1-5](#)

## MKA Policies

You apply a defined MKA policy to an interface to enable MKA on the interface. Removing the MKA policy disables MKA on that 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.

- Replay protection. You can configure MACsec window size, as defined by the number of out-of-order frames that are accepted. This value is used while installing the security associations in the MACsec. A value of 0 means that frames are accepted only in the correct order.

## Virtual Ports

You use virtual ports for multiple secured connectivity associations on a single physical port. Each connectivity association (pair) represents a virtual port, with a maximum of two virtual ports per physical port. Only one of the two virtual ports 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 Catalyst 3750-X stack master running MACsec maintains the configuration files that show which ports on a member switch support MACsec. The stack master performs these functions:

- Processes secure channel and secure association creation and deletion.
- Sends secure association service requests to the stack members.
- 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 stack master.
- Processes MACsec service requests sent by the stack master.
- Sends information about local ports to the stack master.

In case of a stack master changeover, all secured sessions are brought down and then reestablished. The authentication manager recognizes any secured sessions and initiates teardown of these sessions.

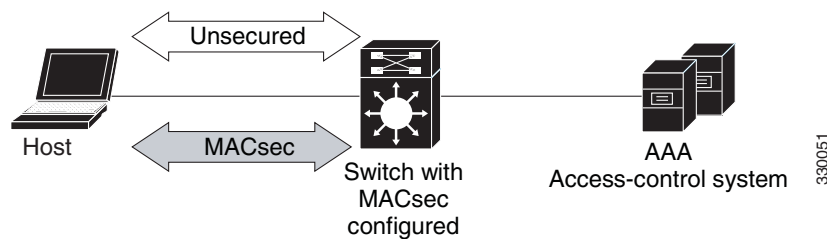
## MACsec, MKA and 802.1x Host Modes

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

### Single-Host Mode

Figure 1-1 shows how a single EAP authenticated session is secured by MACsec by using MKA.

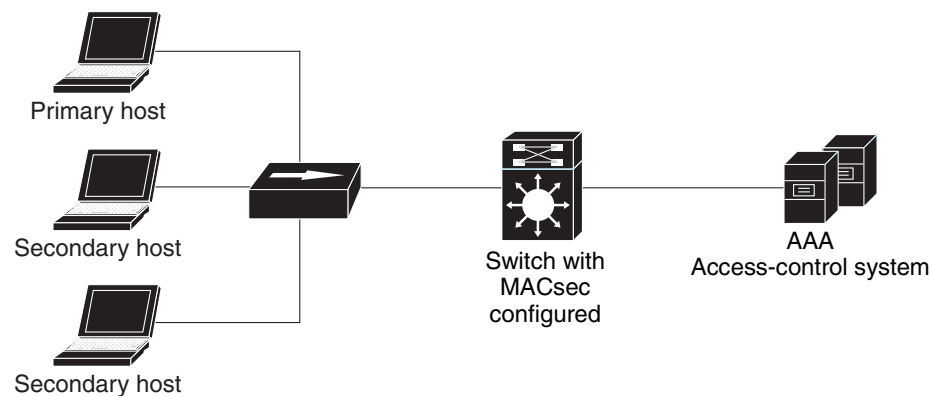
**Figure 1-1** *MACsec in Single-Host Mode with a Secured Data Session*



### 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. See Figure 1-2.

**Figure 1-2** *MACsec in Standard Multiple-Host Mode - Unsecured*



We do not recommend using multi-host mode because after the first successful client, authentication is not required for other clients, which is not secure.

## 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.

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

```
Switch# show mka statistics
MKA Global Statistics
=====
MKA Session Totals
  Secured..... 32
  Reauthentication Attempts.. 31

  Deleted (Secured)..... 1
  Keepalive Timeouts..... 0

CA Statistics
  Pairwise CAKs Derived..... 32
  Pairwise CAK Rekeys..... 31
  Group CAKs Generated..... 0
  Group CAKs Received..... 0

SA Statistics
  SAKs Generated..... 32
  SAKs Rekeyed..... 31
  SAKs Received..... 0
  SAK Responses Received..... 32

MKPDU Statistics
  MKPDUs Validated & Rx..... 580
    "Distributed SAK"..... 0
    "Distributed CAK"..... 0
  MKPDUs Transmitted..... 597
    "Distributed SAK"..... 32
    "Distributed CAK"..... 0

MKA Error Counter Totals
=====
Bring-up Failures..... 0
Reauthentication Failures..... 0

SAK Failures
  SAK Generation..... 0
  Hash Key Generation..... 0
  SAK Encryption/Wrap..... 0
  SAK Decryption/Unwrap..... 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.. 2

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

```

For description of the output fields, see the command reference for this release.

## Configuring MKA and MACsec

- [Default MACsec MKA Configuration, page 1-6](#)
- [Configuring an MKA Policy, page 1-6](#)
- [Configuring MACsec on an Interface, page 1-7](#)

### Default MACsec MKA Configuration

MACsec is disabled. No MKA policies are configured.

### Configuring an MKA Policy

Beginning in privileged EXEC mode, follow these steps to create an MKA Protocol policy. Note that MKA also requires that you enable 802.1x.

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>mka policy</b> <i>policy name</i>	Identify an MKA policy, and enter MKA policy configuration mode. The maximum policy name length is 16 characters.
Step 3	<b>replay-protection window-size</b> <i>frames</i>	Enable replay protection, and configure the window size in number of frames. The range is from 0 to 4294967295. The default window size is 0.  Entering a window size of 0 is not the same as entering the <b>no replay-protection</b> command. Configuring a window size of 0 uses replay protection with a strict ordering of frames. Entering <b>no replay-protection</b> turns off MACsec replay-protection.
Step 4	<b>end</b>	Return to privileged EXEC mode.
Step 5	<b>show mka policy</b>	Verify your entries.
Step 6	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

This example configures the MKA policy *relay-policy*:

```

Switch(config)# mka policy relay-policy
Switch(config-mka-policy)# replay-protection window-size 300
Switch(config-mka-policy)# end

```

## Configuring MACsec on an Interface

Beginning in privileged EXEC mode, follow these steps to configure MACsec on an interface with one MACsec session for voice and one for data:

	Command	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>interface</b> <i>interface-id</i>	Identify the MACsec interface, and enter interface configuration mode. The interface must be a physical interface.
Step 3	<b>switchport access vlan</b> <i>vlan-id</i>	Configure the access VLAN for the port.
Step 4	<b>switchport mode access</b>	Configure the interface as an access port.
Step 5	<b>macsec</b>	Enable 802.1ae MACsec on the interface.
Step 6	<b>authentication event linksec fail action authorize vlan</b> <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	<b>authentication host-mode multi-domain</b>	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	<b>authentication linksec policy must-secure</b>	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	<b>authentication port-control auto</b>	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	<b>authentication violation protect</b>	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 11	<b>mka policy</b> <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 <b>mka policy</b> global configuration command), you must apply the MKA default policy to the interface by entering the <b>mka default-policy</b> interface configuration command.
Step 12	<b>dot1x pae authenticator</b>	Configure the port as an 802.1x port access entity (PAE) authenticator.
Step 13	<b>spanning-tree portfast</b>	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 14	<b>end</b>	Return to privileged EXEC mode.
Step 15	<b>show authentication session interface</b> <i>interface-id</i>	Verify the authorized session security status.
Step 16	<b>copy running-config startup-config</b>	(Optional) Save your entries in the configuration file.

This is an example of configuring and verifying MACsec on an interface:

```
Switch(config)# interface GigabitEthernet1/0/25
Switch(config-if)# switchport access vlan 10
Switch(config-if)# switchport mode access
Switch(config-if)# macsec
Switch(config-if)# authentication event linksec fail action authorize vlan 2
Switch(config-if)# authentication host-mode multi-domain
Switch(config-if)# authentication linksec policy must-secure
Switch(config-if)# authentication port-control auto
Switch(config-if)# authentication violation protect
Switch(config-if)# mka policy replay-policy
Switch(config-if)# dot1x pae authenticator
Switch(config-if)# spanning-tree portfast
Switch(config-if)# end
Switch# show authentication sessions interface gigabitethernet1/0/25
Interface: GigabitEthernet1/0/25
MAC Address: 001b.2140.ec3c
IP Address: 1.1.1.103
User-Name: ms1
Status: Authz Success
Domain: DATA
Security Policy: Must Secure β--- New
Security Status: Secured β--- New
Oper host mode: multi-domain
Oper control dir: both
Authorized By: Authentication Server
Vlan Policy: 10
Session timeout: 3600s (server), Remaining: 3567s
Timeout action: Reauthenticate
Idle timeout: N/A
Common Session ID: 0A05783B0000001700448BA8
Acct Session ID: 0x00000019
Handle: 0x06000017
Runnable methods list:
Method State
dot1x Authc Success
```

## Understanding Cisco TrustSec MACsec



### Note

Cisco TrustSec MACsec for switch-to-switch security is supported only on switches running the IP base or IP services feature set. It is not supported on switches running the NPE or LAN base feature set.

Table 1-2 summarizes the Cisco TrustSec features supported on the switch. For more detailed explanations, see the *Cisco TrustSec Switch Configuration Guide*:  
[http://www.cisco.com/en/US/docs/switches/lan/trustsec/configuration/guide/arch\\_over.html#wp1054561](http://www.cisco.com/en/US/docs/switches/lan/trustsec/configuration/guide/arch_over.html#wp1054561)



Table 1-2 Cisco TrustSec Features

Cisco TrustSec Feature	Description
802.1AE Encryption (MACsec)	<p>Protocol for 802.1AE-based wire-rate hop-to-hop Layer 2 encryption.</p> <p>Between MACsec-capable devices, packets are encrypted on egress from the sending device, decrypted on ingress to the receiving device, and in the clear within the devices.</p> <p>This feature is only available between 802.1AE-capable devices.</p>
Network Device Admission Control (NDAC)	<p>NDAC is an authentication process by which each network device in the TrustSec domain can verify the credentials and trustworthiness of its peer device. NDAC uses an authentication framework based on IEEE 802.1x port-based authentication and uses Extensible Authentication Protocol Flexible Authentication via Secure Tunnel (EAP-FAST) as its EAP method. Authentication and authorization by NDAC results in Security Association Protocol negotiation for 802.1AE encryption.</p>
Security Association Protocol (SAP)	<p>SAP is a Cisco proprietary key exchange protocol between switches. After NDAC switch-to-switch authentication, SAP automatically negotiates keys and the cipher suite for subsequent switch-to-switch MACsec encryption between TrustSec peers. The protocol description is available under a nondisclosure agreement.</p>
Security Group Tag (SGT) <b>Note</b> SGT is not supported in this release.	<p>An SGT is a 16-bit single label showing 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), including SXPv2	<p>With SXP, devices that are not TrustSec-hardware capable can receive SGT attributes for authenticated users or devices from the Cisco Access Control System (ACS). The devices then forward the source IP-to-SGT binding to a TrustSec-hardware capable device for tagging and security group ACL (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

Cisco TrustSec uses AES-128 GCM and GMAC and is compliant with the 802.1AE standard. GCM is not supported on switches running the NPE or the LAN base image.

Cisco TrustSec NDAC SAP is supported on trunk ports because it is intended only for network device to network device links, that is, switch-to-switch links. It is not supported on:

- Host facing access ports (these ports support MKA MACsec)
- Switch virtual interfaces (SVIs)
- SPAN destination ports

The switch also does not support security group ACLs.

You must set the Cisco TrustSec credentials to create the Cisco TrustSec network.

You can configure Cisco TrustSec link layer security in 802.1x mode or manual mode.

## Configuring Cisco TrustSec MACsec

- [Configuring Cisco TrustSec Credentials on the Switch, page 1-10](#)
- [Configuring Cisco TrustSec Switch-to-Switch Link Security in 802.1x Mode, page 1-11](#)
- [Configuring Cisco TrustSec Switch-to-Switch Link Security in Manual Mode, page 1-12](#)
- [Cisco TrustSec Switch-to-Switch Link Security Configuration Example, page 1-14](#)



### Note

The sample configuration in the last section shows the AAA and the RADIUS configuration. Use this example to configure RADIUS and AAA before configuring switch-to-switch security.

## Configuring Cisco TrustSec Credentials on the Switch

To enable Cisco TrustSec features, you must create Cisco TrustSec credentials on the switch to use in other TrustSec configurations. Beginning in privilege EXEC mode, follow these steps to configure Cisco TrustSec credentials.

	Command	Purpose
Step 1	<b>cts credentials id <i>device-id</i> password <i>cts-password</i></b>	Specifies the Cisco TrustSec credentials for this switch to use when authenticating with other Cisco TrustSec devices with EAP-FAST. <ul style="list-style-type: none"> <li>• <b>id <i>device-id</i></b>—Specifies a Cisco TrustSec device ID for the switch. The <i>device-id</i> argument has a maximum length of 32 characters and is case sensitive.</li> <li>• <b>password <i>cts-password</i></b>—Specifies the Cisco TrustSec password for the device.</li> </ul>
Step 2	<b>show cts credentials</b>	(Optional) Displays Cisco TrustSec credentials configured on the switch.
Step 3	<b>copy running-config startup-config</b>	(Optional) Saves your entries in the configuration file.

To delete the Cisco TrustSec credentials, enter the **clear cts credentials** privileged EXEC command.

This example shows how to create Cisco TrustSec credentials.

```
Switch# cts credentials id trustsec password mypassword
CTS device ID and password have been inserted in the local keystore. Please make
sure that the same ID and password are configured in the server database.
```

```
Switch# show cts credentials
CTS password is defined in keystore, device-id = trustsecchange-password Initiate
password change with AAA server
```



	Command	Purpose
Step 7	<b>show cts interface</b> [ <i>interface-id</i>   <b>brief</b>   <b>summary</b> ]	(Optional) Verify the configuration by displaying TrustSec-related interface characteristics.
Step 8	<b>copy running-config startup-config</b>	(Optional) Saves your entries in the configuration file.

This example shows how to enable Cisco TrustSec authentication in 802.1x mode on an interface using GCM as the preferred SAP mode:

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts dot1x
Switch(config-if-cts-dot1x)# sap mode-list gcm-encrypt null no-encap
Switch(config-if-cts-dot1x)# exit
Switch(config-if)# end
```

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

If your switch does not have access to an authentication server or if 802.1X authentication is not needed, you can manually configure Cisco TrustSec on an interface. You must manually configure the interface on each end of the connection.

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

- If no SAP parameters are defined, MACsec encapsulation or encryption will not be 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.

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

	Command	Purpose
Step 1	<b>configure terminal</b>	Enters global configuration mode.
Step 2	<b>interface</b> <i>interface-id</i>	<b>Note</b> Enters interface configuration mode.
Step 3	<b>cts manual</b>	Enters Cisco TrustSec manual configuration mode.

	Command	Purpose
Step 4	<code>sap pmk key [mode-list mode1 [mode2 [mode3 [mode4]]]]</code>	<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"> <li><i>key</i>—A hexadecimal value with an even number of characters and a maximum length of 32 characters.</li> </ul> <p>The SAP operation <i>mode</i> options:</p> <ul style="list-style-type: none"> <li><b>gcm-encrypt</b>—Authentication and encryption</li> </ul> <p><b>Note</b> Select this mode for MACsec authentication and encryption if your software license supports MACsec encryption.</p> <ul style="list-style-type: none"> <li><b>gmac</b>—Authentication, no encryption</li> <li><b>no-encap</b>—No encapsulation</li> <li><b>null</b>—Encapsulation, no authentication or encryption</li> </ul> <p><b>Note</b> If the interface is not capable of data link encryption, <b>no-encap</b> is the default and the only available SAP operating mode. SGT is not supported.</p>
Step 5	<code>no propagate sgt</code>	Use the <b>no</b> form of this command when the peer is incapable of processing a SGT. The <b>no propagate sgt</b> command prevents the interface from transmitting the SGT to the peer and is required in manual mode.
Step 6	<code>exit</code>	Exits Cisco TrustSec 802.1X interface configuration mode.
Step 7	<code>end</code>	Returns to privileged EXEC mode.
Step 8	<code>show cts interface [interface-id   brief   summary]</code>	(Optional) Verify the configuration by displaying TrustSec-related interface characteristics.
Step 9	<code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

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
```

## 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 address ipv4 10.5.120.12 auth-port 1812 acct-port 1813
pac key cisco123
Switch(config)# radius server ACS-2 address ipv4 10.5.120.14 auth-port 1812 acct-port 1813
pac key cisco123
Switch(config)# radius server ACS-3 address ipv4 10.5.120.15 auth-port 1812 acct-port 1813
pac key cisco123
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 authentication network cts-radius group radius
Switch(config)# aaa session-id common
Switch(config)# cts authorization list cts-radius
Switch(config)# dot1x system-auth-control

Switch(config)# interface g1/1/2
Switch(config-if)# switchport trunk encapsulation dot1q
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 g1/1/4
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# cts manual
Switch(config-if-cts-dot1x)# sap pmk 033445AABBCDDEEFF 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 g1/1/2
Switch(config-if)# switchport trunk encapsulation dot1q
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 gi1/1/4
Switch(config-if)# switchport trunk encapsulation dot1q
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# cts credentials id cts-72 password trustsec123
```

