Configuring IPsec Inline Tagging for TrustSec

The IPsec Inline Tagging for TrustSec feature enables IPsec to carry the Cisco TrustSec (CTS) Security Group Tag (SGT) between IPsec peers.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see 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 document.

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

Prerequisites for Configuring IPsec Inline Tagging for TrustSec

Internet Key Exchange Version 2 (IKEv2) and IPsec must be configured on the router. For more information, see the "Configuring Internet Key Exchange Version 2" and "Configuring Security for VPNs with IPsec" modules.

This feature is supported only on the Cisco ISR G2 890, 1900, 2900, 3900, and 3900E routers.

Restrictions for Configuring IPsec Inline Tagging for TrustSec

The IPsec Inline Tagging for TrustSec feature can be negotiated only with IKEv2 and supports the following with IKEv2:

- DMVPN
- Dynamic Virtual Tunnel Interface (dVTI)
- GRE with Tunnel Protection
- Site-to-site VPNs
- Static crypto maps
- Static Virtual Tunnel Interface (sVTI)

The IPsec Inline Tagging for TrustSec feature does not support the following:

- Cisco AnyConnect
- Cisco VPNClient
- DMVPN with IKEv1
- EasyVPN
- FlexVPN
- GetVPN
- IKEv1 IPsec methods
- SSLVPN

Information About Configuring IPsec Inline Tagging for TrustSec

Cisco TrustSec

The Cisco TrustSec (CTS) architecture helps to build secure networks by establishing a domain of trusted network devices by combining identity, trust, and policy to protect user transactions and enforce role-based policies. CTS uses the user and the device identification information acquired during the authentication phase to classify packets as they enter the network. CTS maintains a classification of each packet by tagging packets on ingress to the CTS network so that they can be properly identified for applying security and other policy criteria along the data path. The packets or frames are tagged using the Security Group Tag (SGT), which allows network intermediaries such as switches and firewalls, to enforce an access control policy based on the classification.

The IPsec Inline Tagging for TrustSec feature is used to propagate the SGT to other network devices.

Note If this feature is not supported, you can use the SGT Exchange Protocol over TCP (SXP) feature.

For more information on CTS and SXP, see the Cisco TrustSec Switch Configuration Guide.

SGT and IPsec

IPsec uses the IKE protocol for negotiating algorithms, keys, and capabilities. IKEv2 is used to negotiate and inform IPsec about the SGT capability. Once the peers acknowledge the SGT tagging capability, an SGT tag number (a 16-bit) is added as the SGT Cisco Meta Data (CMD) payload into IPsec and sent to the receiving peer.

The access layer device authenticates the incoming packets. The access layer device receives an SGT from the authentication server and assigns the SGT along with an IP address to the incoming packets. In other words, an IP address is bound to an SGT. This IP address/SGT binding is propagated to upstream devices to enforce SGT-based policy and inline tagging.

If IKEv2 is configured to negotiate the SGT capability in the initiator, the initiator proposes the SGT capability information in the SA_INIT request. If IKEv2 is configured to negotiate the SGT capability in the responder, the responder acknowledges in the SA_INIT response and the initiator and the responder inform IPsec to use inline tagging for all packets to the peer.

During egress, IPsec adds the SGT capability and prefixes to the IPsec payload if the peer supports inline tagging; otherwise the packet is not tagged.

During ingress, IPsec inspects the packet for the SGT capability. If a tag is available, IPsec extracts the tag information and passes the information to the device only if inline tagging is negotiated. If there is no tag, IPsec processes the packet as a normal packet.

The tables below describe how IPsec behaves during egress and ingress.

Table 1: IPsec Behavior on the Egress Path

Inline Tagging Negotiated	CTS Provides SGT	IPsec Behavior
Yes	Yes	An SGT CMD is added to the packet.
Yes	No	The packet is sent without the SGT CMD.
No	Yes or no	The packet is sent without the SGT CMD.

Table 2: IPsec Behavior on the Ingress Path

Packet Is Tagged	Inline Tagging Negotiated	IPsec Behavior
Yes	Yes	The SGT CMD in the packet is processed.
Yes	No	The SGT CMD in the packet is not processed.

Packet Is Tagged	Inline Tagging Negotiated	IPsec Behavior
No	Yes or no	The packet is processed as a normal IPsec packet.

SGT on the IKEv2 Initiator and Responder

To enable SGT on an IKEv2 session, the SGT capability support must be sent to the peers using the **crypto ikev2 cts** command. SGT is a Cisco proprietary capability; hence, it is sent as a Vendor ID (VID) payload in the SA_INIT exchange.

The table below explains the scenarios when SGT capability is configured on the initiator and the responder:

Table 3: SGT Capability on IKEv2 Initiator and Responder

SGT Enabled on Initiator	SGT Enabled on Responder	What Happens . . .
Yes	Yes	The VID is exchanged between the initiator and the responder, and IPsec SA is enabled with the SGT inline tagging capability.
Yes	No	The initiator proposes the VID, but the responder ignores the VID. IPsec SA is not enabled with the SGT inline tagging capability.
No	Yes	The initiator does not propose the VID, and the responder does not send the VID payload. IPsec SA is not enabled with the SGT inline tagging capability.
No	No	The initiator does not propose the VID, and responder also does not send the VID payload. IPsec SA is not enabled with the SGT inline tagging capability.

Handling Fragmentation

Fragmentation is handled in the following two ways:

- Fragmentation before IPsec—If IPsec receives fragmented packets, each fragment is tagged.
- Fragmentation after IPsec—If IPsec packets are fragmented after encryption, the first fragment will be tagged.

How to Configure IPsec Inline Tagging for TrustSec

Enabling IPsec Inline Tagging

Before You Begin

IKEv2 and IPsec must be configured.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ikev2 cts sgt**
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**crypto ikev2 cts sgt** **Example:** `Router(config)# crypto ikev2 cts sgt`	Enables SGT IPsec inline tagging globally. **Note** This command applies to all sessions on the router. If the IPsec Inline Tagging for TrustSec feature is disabled, then the new negotiated sessions do not negotiate the VID. However, the current SA and the subsequent SA rekey is enabled with the feature until the lifetime of the SA. This applies to the new IPsec SA and the rekey of IPsec SA established using the parent or rekeyed IKE SA.
Step 4	**exit** **Example:** `Router(config)# exit`	Exits global configuration mode.

Monitoring and Verifying IPsec Inline Tagging for TrustSec

To monitor and verify the IPsec Inline Tagging for TrustSec configuration, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **debug crypto ipsec metadata sgt**
3. **debug crypto ikev2 internal**
4. **show crypto ikev2 sa detailed**
5. **show crypto ipsec sa detail**

DETAILED STEPS

Step 1 **enable**

Example:
```
Router> enable
```

Enables privileged EXEC mode.

Step 2 **debug crypto ipsec metadata sgt**

Example:
```
Router# debug crypto ipsc metadata sgt

*Oct 17 05:29:17.067: IPsec SGT:: extracted SGT = 400 for src ip 10.0.0.1
*Oct 17 05:29:17.067: IPsec SGT:: inserted SGT = 333 for src ip 3.3.3.1
```

Use this command to verify whether packets from an IP address is tagged correctly.

Step 3 **debug crypto ikev2 internal**

Example:
```
Router# debug cryto ikev2 internal

Mar 28 06:11:19.159:
*Mar 28 06:11:19.159: IKEv2:Construct Vendor Specific Payload: DELETE-REASON
*Mar 28 06:11:19.159: IKEv2:(1): Sending custom vendor id : CISCO-CTS-SGT
*Mar 28 06:11:19.159: IKEv2:Construct Vendor Specific Payload: (CUSTOM)
Mar 28 06:11:19.203:
*Mar 28 06:11:19.203: IKEv2:(1): Received custom vendor id : CISCO-CTS-SGT
*Mar 28 06:11:19.203: IKEv2:(1): SM Trace-> SA: I_SPI=89D5FB4E99C03FC1 R_SPI=8
```

Use this command to log the IKEv2 events. The presence of the Vendor ID payload for the SGT capability in SA_INIT exchange indicates the proposal or acceptance of the feature.

Step 4 **show crypto ikev2 sa detailed**

Example:
```
Router# show crypto ikev2 sa detailed

IPv4 Crypto IKEv2  SA
Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         1.1.1.1/500           1.1.1.2/500           none/none            READY
```

```
Encr: 3DES, Hash: SHA96, DH Grp:1, Auth sign: PSK, Auth verify: PSK
Life/Active Time: 120/12 sec
CE id: 1001, Session-id: 1
Status Description: Negotiation done
Local spi: 2BD8B64C579F0C6E       Remote spi: 2CC7EB7D1862500F
Local id: 1.1.1.1
Remote id: 1.1.1.2
Local req msg id:  2              Remote req msg id:  0
Local next msg id: 2              Remote next msg id: 0
Local req queued:  2              Remote req queued:  0
Local window:      5              Remote window:      5
DPD configured for 0 seconds, retry 0
NAT-T is not detected
Cisco Trust Security SGT is enabled
```

Use this command to display the status of the SGT capability.

Step 5 **show crypto ipsec sa detail**

Example:

```
Router# show crypto ipsec sa detail

interface: GigabitEthernet0/0
    Crypto map tag: cmap, local addr 10.1.1.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (209.165.201.1/255.255.255.224/0/0)
   remote ident (addr/mask/prot/port): (209.165.200.1/255.255.255.224/0/0)
   current_peer 172.16.0.1 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 20, #pkts encrypt: 20, #pkts digest: 20
    #pkts decaps: 20, #pkts decrypt: 20, #pkts verify: 20
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #pkts no sa (send) 0, #pkts invalid sa (rcv) 0
    #pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0
    #pkts invalid prot (recv) 0, #pkts verify failed: 0
    #pkts invalid identity (recv) 0, #pkts invalid len (rcv) 0
    #pkts replay rollover (send): 0, #pkts replay rollover (rcv) 0
    ##pkts replay failed (rcv): 0
    #pkts tagged (send): 20, #pkts untagged (rcv): 20
    #pkts not tagged (send): 0, #pkts not untagged (rcv): 0
    #pkts internal err (send): 0, #pkts internal err (recv) 0

     local crypto endpt.: 10.1.1.2, remote crypto endpt.: 172.160.1.1
     path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/0
     current outbound spi: 0x9F0DFA17(2668493335)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0xD026B7DD(3492198365)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2043, flow_id: Onboard VPN:43, sibling_flags 80000040, crypto map: cmap
        sa timing: remaining key lifetime (k/sec): (4228802/3367)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x9F0DFA17(2668493335)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2044, flow_id: Onboard VPN:44, sibling_flags 80000040, crypto map: cmap
```

```
      sa timing: remaining key lifetime (k/sec): (4228802/3367)
      IV size: 8 bytes
      replay detection support: Y
      Status: ACTIVE

    outbound ah sas:

    outbound pcp sas:
  protected vrf: (none)
  local  ident (addr/mask/prot/port): (192.168.12.1/255.255.0.0/0/0)
  remote ident (addr/mask/prot/port): (209.165.201.1/255.255.255.224/0/0)
  current_peer 172.160.1.1 port 500
    PERMIT, flags={origin_is_acl,}
   #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
   #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
   #pkts compressed: 0, #pkts decompressed: 0
   #pkts not compressed: 0, #pkts compr. failed: 0
   #pkts not decompressed: 0, #pkts decompress failed: 0
   #pkts no sa (send) 0, #pkts invalid sa (rcv) 0
   #pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0
   #pkts invalid prot (recv) 0, #pkts verify failed: 0
   #pkts invalid identity (recv) 0, #pkts invalid len (rcv) 0
   #pkts replay rollover (send): 0, #pkts replay rollover (rcv) 0
   ##pkts replay failed (rcv): 0
   #pkts tagged (send): 0, #pkts untagged (rcv): 0
   #pkts not tagged (send): 0, #pkts not untagged (rcv): 0
   #pkts internal err (send): 0, #pkts internal err (recv) 0

    local crypto endpt.: 10.1.1.2, remote crypto endpt.: 172.160.1.1
    path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/0
    current outbound spi: 0x0(0)
    PFS (Y/N): N, DH group: none

    inbound esp sas:

    inbound ah sas:

    inbound pcp sas:

    outbound esp sas:

    outbound ah sas:

    outbound pcp sas:
```

Displays the settings used by current security associations (SAs).

Configuration Examples for IPsec Inline Tagging for TrustSec

Example: Enabling IPsec Inline Tagging

The following example shows how to enable IPsec inline tagging on a static VTI initiator and dynamic VTI responder. You can use this configuration for configuring crypto maps and VTIs.

Static VTI Initiator Configuration

```
crypto ikev2 proposal p1
 encryption 3des
 integrity md5
```

```
 group 2
!
crypto ikev2 policy policy1
 proposal p1
!
crypto ikev2 keyring key
 peer peer
  address ::/0
  pre-shared-key cisco
 !
 peer v4
  address 0.0.0.0 0.0.0.0
  pre-shared-key cisco
 !
!
!
crypto ikev2 profile prof3
 match identity remote address 0.0.0.0
 authentication local pre-share
 authentication remote pre-share
 keyring key
!
crypto ikev2 cts sgt
!
crypto ipsec transform-set trans esp-3des esp-sha-hmac
!
crypto map cmap 1 ipsec-isakmp
 set peer 10.1.1.2
 set transform-set trans
 set ikev2-profile prof3
 match address ipv4acl
!
!
interface Loopback1
 ip address 209.165.201.1 255.255.255.224
 ipv6 address 2001::4:1/112
!
interface Loopback2
 ip address 209.165.200.1 255.255.255.224
 ipv6 address 2001::40:1/112
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 192.168.210.74 255.255.255.0
 duplex auto
 speed auto
!
interface GigabitEthernet0/1
 ip address 172.16.0.1 255.240.0.0
 duplex auto
 speed auto
 ipv6 address 2001::5:1/112
 ipv6 enable
 crypto map cmap
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 172.16.0.2
ip route 10.12.255.200 255.0.0.0 172.31.255.254
!
ip access-list extended ipv4acl
 permit ip host 209.165.201.1host 192.168.12.125
 permit ip host 209.165.200.1 host 172.18.0.1
 permit ip host 172.28.0.1 host 10.10.10.1
 permit ip host 10.12.255.200 host 192.168.14.1
!
logging esm config
```

```
ipv6 route ::/0 2001::5:2
!
!
!
!
!!
control-plane
!
!
!
line con 0
 exec-timeout 0 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport input all
 transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input all
!
exception data-corruption buffer truncate
scheduler allocate 20000 1000
```

Dynamic VTI Responder Configuration

```
crypto ikev2 proposal p1
 encryption 3des
 integrity md5
 group 2
!
crypto ikev2 policy policy1
 proposal p1
!
crypto ikev2 keyring key
 peer peer
  address 172.160.1.1 255.240.0.0
  pre-shared-key cisco
 !
 peer v4_p2
  address 172.31.255.1 255.240.0.0
  pre-shared-key cisco
 !
crypto ikev2 profile prof
 match identity remote address 0.0.0.0
 authentication local pre-share
 authentication remote pre-share
 keyring key
 virtual-template 25
!
crypto ikev2 cts sgt
!
crypto ipsec transform-set trans esp-null esp-sha-hmac
!
crypto ipsec profile prof_ipv4
 set transform-set trans
 set ikev2-profile prof1_ipv4
!
!
interface Loopback0
 ip address 192.168.12.1 255.255.0.0
!
interface Loopback1
 no ip address
!
interface Loopback2
 ip address 172.18.0.1 255.240.0.0
!
```

```
interface Loopback10
 no ip address
 ipv6 address 2001::8:1/112
!
interface Loopback11
 no ip address
 ipv6 address 2001::80:1/112
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.1.1.2 255.0.0.0
 duplex auto
 speed auto
 ipv6 address 2001::7:1/112
 ipv6 enable
!
interface GigabitEthernet0/1
 ip address 10.10.10.2 255.255.255.0
 duplex auto
 speed auto
!
interface GigabitEthernet0/2
 ip address 192.168.210.144 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/0/0
 no ip address
 shutdown
!
interface FastEthernet0/0/1
 no ip address
!
interface FastEthernet0/0/2
 no ip address
!
interface FastEthernet0/0/3
 no ip address
!
!
interface Virtual-Template25 type tunnel
 ip unnumbered GigabitEthernet0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile prof_ipv4
!
interface Vlan1
 no ip address
!
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
ip route 172.17.0.0 255.240.0.0 10.10.10.1
!
logging esm config
ipv6 route ::/0 2001::7:2
!
control-plane
!
!
!
line con 0
 exec-timeout 0 0
line aux 0
line 2
 no activation-character
 no exec
```

```
 transport preferred none
 transport input all
 transport output lat pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input all
!
exception data-corruption buffer truncate
scheduler allocate 20000 1000
end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Security commands	• Cisco IOS Security Command Reference Commands A to C • Cisco IOS Security Command Reference Commands D to L • Cisco IOS Security Command Reference Commands M to R • Cisco IOS Security Command Reference Commands S to Z
Cisco TrustSec and SXP configuration	Cisco TrustSec Switch Configuration Guide
IPsec configuration	Configuring Security for VPNs with IPsec
IKEv2 configuration	Configuring Internet Key Exchange Version 2 (IKEv2) and FlexVPN Site-to-Site
Cisco Secure Access Control Server	Configuration Guide for the Cisco Secure ACS

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring IPsec Inline Tagging for TrustSec

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

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

Table 4: Feature Information for Configuring IPsec Inline Tagging for TrustSec

Feature Name	Releases	Feature Information
IPsec Inline Tagging for TrustSec	15.2(2)T	The IPsec Inline Tagging for TrustSec feature enables IPsec to carry Cisco Trust Sec (CTS) Security Group Tag (SGT) between IPsec peers. In Cisco IOS Release 15.2(2)T, this feature was introduced. The following commands were introduced or modified: **crypto ikev2 cts sgt, debug crypto ikev2 detail, debug crypto ipsec, show crypto ipsec sa.**