Internet Protocol Security

Internet Protocol Security (IPsec)

Internet Protocol Security (IPsec) ia framework of open standards that

- ensures confidentiality by encrypting data packets before transmission
- guarantees data integrity by authenticating packets, and
- provides data origin authentication to verify the source

IPsec includes an anti-replay feature that helps you detect and reject replayed packets.

An IPsec tunnel is a secure communication channel connecting two IPsec peers. It encrypts selected network traffic, uses security associations (SAs) to determine protocols, algorithms, and keying material for traffic protection, and applies access lists and crypto map sets to define which packets are encrypted and routed through the tunnel.

A security association (SA) is an agreement between two IPsec peers that defines the security protocols and algorithms to apply to protected packets, specifies the keying material used, and operates unidirectionally for each protocol.

Key Features of IPsec

IPsec provides the following security features:

- Confidentiality: IPsec encrypts packets to protect sensitive information during transmission.
- A framework of open standards, IPsec verifies the source of the packets, enhancing security.
- Anti-replay: The IPsec receiver can detect and reject replayed packets.

The controller supports IPsec configuration, securing syslog traffic and enabling a robust network-wide security policy.

If an you wants to encrypt all traffic between two sites, you can:

- Create access lists specifying relevant IP address ranges
- Apply those access lists to a crypto map set, and

- Bind the crypto map set to the device interface facing the remote site.

All matching packets is encrypted and transmitted through the IPsec tunnel.

Think of access lists as the guest list at a secure building.

Crypto maps are the guard's special instructions for who receives extra screening.

Transform sets are the security checks both guard and guest agree on.

Only guests who pass the right checks get inside; others are turned away.

IKEv1 transform sets

An IKEv1 transform set is a cryptographic configuration set that

- specifies a combination of security protocols and algorithms
- enables peers to negotiate security parameters for IPsec Security Associations (SAs), and
- is assigned to specific data flows through crypto map entries.

Transform set behavior during IPsec SA negotiation

During IPsec SA negotiation:

- Both peers search for a transform set that matches on each side.
- If a matching transform set is found, it is applied to the relevant protected traffic as part of the IPsec SA.

Additional reference information

- Administrators can define multiple transform sets and assign one or more to crypto map entries.
- The transform set associated with a crypto map entry determines the security protocols and algorithms used to protect specified data flows.
- Changing a transform set during operation does not affect existing SAs.
  - The new configuration applies only to future SA negotiations.
  - To apply changes immediately, you can clear all or part of the SA database using the **clear crypto sa** command.

Example

The configuration snippet shows how to configure an IPsec IKEv1 transform set using AES-CBC-128 for payload encryption. To use AES-CBC-256, specify **encryption aes 256**.

```
Device # conf t
Device (config)#crypto isakmp policy 1
Device (config-isakmp)# hash sha
Device (config-isakmp)# encryption aes
```

Configure IPsec using IKEv1

Follow the procedure given below to configure IPsec IKEv1 to use AES-CBC-128 for payload encryption.

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Enters global configuration mode.

Step 2 Define an Internet Key Exchange (IKE) policy and assign a priority to the policy.

Example:

```
Device(config)# crypto isakmp policy priority
```

- *priority*: Uniquely identifies the IKE policy and assigns a priority to the policy. Valid values: 1 to 10,000; 1 is the highest priority.

Step 3 Specify the hash algorithm.

Example:

```
Device(config-isakmp)# hash sha
```

Specifies the hash algorithm.

Step 4 Configure IPsec IKEv1 to use AES-CBC-128 for payload encryption.

Example:

```
Device(config-isakmp)# encryption aes
```

AES-CBC-256 can be selected with ‘encryption aes 256’.

Note
The authorized administrator must ensure that the keysize for this setting is greater than or equal to the keysize selected for ESP in section IPsec Transforms and Lifetimes. If AES 128 is selected here, then the highest keysize that can be selected on the device for ESP is AES 128 (either CBC or GCM).

Both confidentiality and integrity are configured with the **hash sha** and **encryption aes** commands respectively. As a result, confidentiality-only mode is disabled.

Step 5 Configure IPsec to use the specified preshared keys as the authentication method.

Example:

```
Device(config-isakmp)# authentication pre-share
```

Preshared keys require that you separately configure these preshared keys.

Step 6 Exit config-isakmp configuration mode.

Example:

```
Device(config-isakmp)# exit
```

Exits config-isakmp configuration mode.

Step 7 Configure a preshared authentication key.

Example:

```
Device(config)# crypto isakmp key keystring address peer-address
```

Note
To ensure a secure configuration, we recommend that you enter the pre-shared keys with at least 22 characters in length and can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”).

The device supports pre-shared keys up to 127 characters in length. While longer keys increase the difficulty of brute-force attacks, longer keys increase processing time.

Step 8 Specify the Diffie-Hellman (DH) group identifier as 2048-bit DH group 14 and selects DH Group 14 (2048-bit MODP) for IKE.

Example:

```
Device(config-isakmp)# group 14
```

However, 19 (256-bit Random ECP), 24 (2048-bit MODP with 256-bit POS), 20 (384-bit Random ECP), 15 (3072 bit MODP), and 16 (4096-bit MODP) are also allowed and supported.

Step 9 Specify the lifetime of the IKE SA.

Example:

```
Device(config-isakmp)# lifetime seconds
```

The default time value for Phase 1 SAs is 24 hours (86400 seconds), but this setting can be changed using the command above with different values.

- *seconds*: Time, in seconds, before each SA expires. Valid values: 60 to 86,400; default value: 86,400.

Note
The shorter the lifetime (up to a point), the more secure your IKE negotiations will be. However, with longer lifetimes, future IPsec SAs can be set up more quickly.

Step 10 Ensure all IKEv1 Phase 1 exchanges will be handled in the default main mode.

Example:

```
Device(config-isakmp)# crypto isakmp aggressive-mode disable
```

Step 11 Exit config-isakmp configuration mode.

Example:

```
Device(config-isakmp)# exit
```

IKEv2 transform sets

An IKEv2 transform set is a security proposal that

- consists of a set of cryptographic transforms (encryption, integrity, and Diffie-Hellman group) used during IKEv2 negotiation,
- is required for a valid IKEv2 Security Association (SA), and

- enables the devices to negotiate compatible security parameters during the IKE_SA_INIT phase.

Expanded explanation

If you do not attach a user-configured IKEv2 proposal to an IKEv2 policy, the system uses the default proposal for negotiation.

Example

This configuration snippet demonstrates how to define and apply an IKEv2 transform set and related parameters:

```
device # conf t
device(config)#crypto ikev2 proposal sample
device(config-ikev2-proposal)# integrity sha1
device (config-ikev2-proposal)# encryption aes-cbc-128
device(config-ikev2-proposal)# group 14
device(config-ikev2-proposal)# exit
device(config)# crypto ikev2 keyring keyring-1
device (config-ikev2-keyring)# peer peer1
device (config-ikev2-keyring-peer)# address 192.0.2.4 255.255.255.0
device (config-ikev2-keyring-peer)# pre-shared-key cisco123!cisco123!CISC
device (config-ikev2-keyring-peer)# exit
device(config)#crypto ikev2 keyring keyring-1
device (config-ikev2-keyring)# peer peer1
device (config-ikev2-keyring-peer)# address 192.0.2.4 255.255.255.0
device (config-ikev2-keyring-peer)# pre-shared-key cisco123!cisco123!CISC
device (config-ikev2-keyring-peer)# exit
device(config)#crypto logging ikev2
```

Configure IPsec using IKEv2

Follow the procedure given below to configure the IPsec with IKEv2:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Define an IKEv2 proposal name.

Example:

```
Device(config)# crypto ikev2 proposal name
```

Step 3 Define the integrity algorithm for the IKEv2 proposal.

Example:

```
Device(config-ikev2-proposal)# integrity sha1
```

Step 4 Configure IPsec IKEv2 to use AES-CBC-128 for payload encryption.

Example:

```
Device(config-ikev2-proposal)# encryption aes-cbc-128
```

AES-CBC-256 can be selected with encryption aes-cbc-256. AES-GCM-128 and AES-GCM-256 can also be selected similarly.

Note
The authorized administrator must ensure that the keysize for this setting is greater than or equal to the keysize selected for ESP in section IPsec Transforms and Lifetimes. If AES 128 is selected here, then the highest keysize that can be selected on the device for ESP is AES 128 (either CBC or GCM).

Both confidentiality and integrity are configured with the **hash sha** and **encryption aes** commands respectively. As a result, confidentiality-only mode is disabled.

Step 5 Selects DH Group 14 (2048-bit MODP) for IKE.

Example:

```
Device(config-ikev2-proposal)# group 14
```

However, 19 (256-bit Random ECP), 24 (2048-bit MODP with 256-bit POS), 20 (384-bit Random ECP), 15 (3072 bit MODP), and 16 (4096-bit MODP) are also allowed and supported.

Step 6 Exit IKEv2 proposal configuration mode.

Example:

```
Device(config-ikev2-proposal)# exit
```

Step 7 Define an IKEv2 keyring.

Example:

```
Device(config)# crypto ikev2 keyring keyring-name
```

Step 8 Define the peer or peer group.

Example:

```
Device(config-ikev2-keyring)# peer peer-name
```

Step 9 Specify an IPv4 or IPv6 address or range for the peer.

Example:

```
Device(config-ikev2-keyring)# address 192.0.2.4 255.255.255.0
```

Note
This IP address is the IKE endpoint address and is independent of the identity address.

Step 10 Specify the preshared key for the peer.

Example:

```
Device(config-ikev2-keyring)# pre-shared-key cisco123!cisco123!CISC
```

You can enter the local or remote keyword to specify an asymmetric preshared key. By default, the preshared key is symmetric.

Note
To ensure a secure configuration, we recommend that you enter the pre-shared keys with at least 22 characters in length and can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”).

The device supports pre-shared keys up to 127 characters in length. While longer keys increase the difficulty of brute-force attacks, longer keys increase processing time.

HEX keys generated off system can also be input for IKEv2 using the **pre-shared-key hex** *[hex key]* command instead of the pre-shared-key command above:. For example: pre-shared-key hex 0x6A6B6C. This configures IPsec to use pre-shared keys.

Step 11 Exit IKEv2 keyring peer configuration mode.

Example:

```
Device(config-ikev2-keyring)# exit
```

Step 12 Enable IKEv2 syslog messages.

Example:

```
Device(config)# crypto logging ikev2
```

Note
The configuration above is not a complete IKE v2 configuration, and that additional settings will be needed.

IPsec transforms and lifetimes

Regardless of the IKE version, the device must be configured with appropriate IPsec ESP transforms (encryption and integrity algorithms) and SA lifetimes. The tables and commands below show how to set these parameters.

This configuration sets IPsec ESP to use HMAC-SHA-1 and AES-CBC-128. To change this to any of the other allowed algorithms, replace **esp-aes 128** in the command

IPsec transform command example

To configure an IPsec ESP transform set with AES-CBC-128 and HMAC-SHA-1:

```
device(config)# crypto ipsec transform-set example esp-aes 128 esp-sha-hmac
```

This command sets ESP to use HMAC-SHA-1 for integrity and AES-CBC-128 for encryption.

Supported Encryption Algorithms

The table lists supported encryption algorithms and their commands:

Encryption Algorithm	Command
AES-CBC-256	esp-aes 256
AES-GCM-128	esp-gcm 128
AES-GCM-256	esp-gcm 256

Note The key size for the algorithm in the transform set must not exceed the key size selected for IKE encryption. For example, if AES-CBC-128 is selected for IKE, then only AES-CBC-128 or AES-GCM-128 may be selected here.

Configuring IPsec modes

Set the operational mode for IPsec using one of these commands:

```
device(config-crypto)# mode tunnel
```

- Requests and accepts only tunnel mode for IPsec. Tunnel is the default mode.

```
device(config-crypto)# mode transport
```

- Configures transport mode for IPsec.

Configuring SA lifetimes

By time (seconds):

The default lifetime for Phase 2 SAs is 3600 seconds (1 hour). To change it (for example to 8 hours or 28800 seconds) use this command:

```
device(config)# crypto ipsec security-association lifetime seconds 28800
```

By data volume (kilobytes):

Default is 2560 KB (minimum configurable). The maximum is 4 GB. For example, to set a 100 MB (100000 KB) limit use this command:

```
device(config)# crypto ipsec security-association lifetime kilobytes 100000
```

Additional notes

- No configuration is needed for default values; configure only for custom values to meet policy or compliance.
- Always ensure that your IPsec transforms and lifetimes are compatible between your IPsec peers to prevent negotiation failures.

Use of X.509 with IKE version

Enable X.509v3 certificate authentication for secure IKEv1 negotiation.

The controller supports RSA and ECDSA based certificates. Once X.509v3 keys are installed on the device, they can be set for use with IKEv1 with the commands:

Procedure

Step 1 Enter global configuration mode.

Example:

```
Device# configure terminal
```

Step 2 Define an Internet Key Exchange (IKE) policy and assign a priority to the policy.

Example:

```
Device(config)# crypto isakmp policy policy-name
```

Step 3 Use RSA based certificates for IKEv1 authentication.

Example:

```
Device(config-isakmp)# authentication rsa-sig
```

Step 4 Use ECDSA based certificates for IKEv1 authentication.

Example:

```
Device(config-isakmp)# authentication ecdsa-sig
```

The X.509 certificates are now configured for use with IKEv1.

IPsec session interruption and recovery

If an IPsec session with a peer is unexpectedly interrupted, the connection will be broken. In this scenario, no administrative interaction is required. The IPsec session will be reestablished (a new SA set up) once the peer is back online.

Example: Configure IPsec using ISAKMP

This example displays the IPSec **isakmp** configuration.

The sample outputs display the IPSec **isakmp** configuration:

```
crypto isakmp policy 1
                encr aes 256
                hash sha256
                authentication pre-share
                group 14
                lifetime 28800
               crypto isakmp key 0 Cisco!123 address 192.0.2.4
               crypto isakmp peer address 192.0.2.4
               crypto ipsec transform-set aes-gcm-256 esp-gcm 256
                mode tunnel
               crypto map IPSEC_ewlc_to_syslog 1 ipsec-isakmp
                set peer 192.0.2.4
                set transform-set aes-gcm-256
                match address acl_ewlc_to_syslog
               interface Vlan15
                crypto map IPSEC_ewlc_to_syslog
               end
```

Verify IPSec traffic

This example shows how to verify the IPSec traffic configuration in isakmp configuration:

```
Device# show crypto map
Crypto Map IPv4 "IPSEC_ewlc_to_syslog" 1 ipsec-isakmp
        Peer = 192.0.2.4
        Extended IP access list acl_ewlc_to_syslog
            access-list acl_ewlc_to_syslog permit ip host 192.0.2.2 host 192.0.2.4
        Current peer: 192.0.2.4
        Security association lifetime: 4608000 kilobytes/3600 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): N
        Mixed-mode : Disabled
        Transform sets={
```

```
                aes-gcm-256:  { esp-gcm 256  } ,
        }
        Interfaces using crypto map IPSEC_ewlc_to_syslog:
                Vlan15
```

```
Device# show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
192.0.2.5       192.0.2.4       QM_IDLE           1011 ACTIVE

IPv6 Crypto ISAKMP SA
```

```
Device# show crypto ipsec sa

interface: Vlan15
    Crypto map tag: IPSEC_ewlc_to_syslog, local addr 192.0.2.5

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.0.2.5/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (192.0.2.4/255.255.255.255/0/0)
   current_peer 192.0.2.4 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 1626, #pkts encrypt: 1626, #pkts digest: 1626
    #pkts decaps: 1625, #pkts decrypt: 1625, #pkts verify: 1625
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 192.0.2.5, remote crypto endpt.: 192.0.2.4
     plaintext mtu 1446, path mtu 1500, ip mtu 1500, ip mtu idb Vlan15
     current outbound spi: 0x17FF2F4C(402599756)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0x4B77AD78(1266134392)
        transform: esp-gcm 256 ,
        in use settings ={Tunnel, }
        conn id: 2041, flow_id: HW:41, sibling_flags FFFFFFFF80004048, crypto map:
IPSEC_ewlc_to_syslog
        sa timing: remaining key lifetime (k/sec): (4607904/1933)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x17FF2F4C(402599756)
        transform: esp-gcm 256 ,
        in use settings ={Tunnel, }
        conn id: 2042, flow_id: HW:42, sibling_flags FFFFFFFF80004048, crypto map:
IPSEC_ewlc_to_syslog
        sa timing: remaining key lifetime (k/sec): (4607904/1933)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     outbound ah sas:
     outbound pcp sas:
```

```
Device# show ip access-lists acl_ewlc_to_syslog
Extended IP access list acl_ewlc_to_syslog
    10 permit ip host 192.0.2.5 host 192.0.2.4 (17 matches)
```

Example: Configure IPsec using IKEv2

The sample outputs display the IPsec **IKEv2** configuration:

```
topology : [192.0.2.6]DUT — (infra) — PEER[192.0.2.9]
ikev2 config in 192.0.2.6 (peer is 192.0.2.9)
hostname for 192.0.2.9: Edison-M1
hostname for 192.0.2.6: prsna-nyquist-192.0.2.6
ip access-list extended ikev2acl
 permit ip host 192.0.2.6 host 192.0.2.9
crypto ikev2 proposal PH1PROPOSAL
 encryption aes-cbc-256
 integrity sha256
 group 14
!
crypto ikev2 policy PH1POLICY
 proposal PH1PROPOSAL
crypto ikev2 keyring PH1KEY
  peer Edison-M1
  address 192.0.2.9
  pre-shared-key Cisco!123Cisco!123Cisco!123
crypto ikev2 profile PH1PROFILE
 match identity remote address 192.0.2.9 255.255.255.255
 authentication remote pre-share
 authentication local pre-share
 keyring local PH1KEY
crypto ipsec transform-set aes256-sha1 esp-aes 256 esp-sha-hmac
 mode tunnel
crypto map ikev2-cryptomap 1 ipsec-isakmp
 set peer 192.0.2.9
 set transform-set aes256-sha1
 set ikev2-profile PH1PROFILE
 match address ikev2acl
interface Vlan15
 ip address 192.0.2.6 255.255.255.0
 crypto map ikev2-cryptomap
```

Verify IPSec with IKEv2 traffic

The following example shows how to verify the IPSec traffic configuration in IKEv2 configuration:

```
Device# show ip access-lists
               Extended IP access list ikev2acl
                   10 permit ip host 192.0.2.6 host 192.0.2.9 (80 matches)
               Device#show crypto map
               Crypto Map IPv4 "ikev2-cryptomap" 1 ipsec-isakmp
                       Peer = 192.0.2.9
                       IKEv2 Profile: PH1PROFILE
                       Extended IP access list ikev2acl
                           access-list ikev2acl permit ip host 192.0.2.6 host 192.0.2.9
                       Current peer: 192.0.2.9
                       Security association lifetime: 4608000 kilobytes/3600 seconds
                       Responder-Only (Y/N): N
                       PFS (Y/N): N
                       Mixed-mode : Disabled
                       Transform sets={
                               aes256-sha1:  { esp-256-aes esp-sha-hmac  } ,
```

```
                        }
                        Interfaces using crypto map ikev2-cryptomap:
                                Vlan15
                Device# show crypto ikev2 sa detailed
                 IPv4 Crypto IKEv2  SA
                Tunnel-id Local                 Remote                fvrf/ivrf
 Status
                1         192.0.2.6/500         192.0.2.9/500          none/none
  READY
                      Encr: AES-CBC, keysize: 256, PRF: SHA256, Hash: SHA256, DH Grp:14,
Auth sign: PSK, Auth verify: PSK
                      Life/Active Time: 86400/1002 sec
                      CE id: 1089, Session-id: 2
                      Status Description: Negotiation done
                      Local spi: 271D20169FE91074       Remote spi: 13895472E3B910AF
                      Local id: 192.0.2.6
                      Remote id: 192.0.2.9
                      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
                      Fragmentation not  configured.
                      Dynamic Route Update: disabled
                      Extended Authentication not configured.
                      NAT-T is not detected
                      Cisco Trust Security SGT is disabled
                      Initiator of SA : Yes
                Device# show crypto ipsec sa detail
                interface: Vlan15
                    Crypto map tag: ikev2-cryptomap, local addr 192.0.2.6
                   protected vrf: (none)
                   local  ident (addr/mask/prot/port): (192.0.2.6/255.255.255.255/0/0)
                   remote ident (addr/mask/prot/port): (192.0.2.9/255.255.255.255/0/0)
                   current_peer 192.0.2.9 port 500
                     PERMIT, flags={origin_is_acl,}
                    #pkts encaps: 80, #pkts encrypt:80, #pkts digest: 80
                    #pkts decaps: 80, #pkts decrypt: 80, #pkts verify: 80
                    #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.: 192.0.2.6, remote crypto endpt.: 192.0.2.9
                     plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb Vlan15
                     current outbound spi: 0xB546157A(3041269114)
                     PFS (Y/N): N, DH group: none
                     inbound esp sas:
                      spi: 0x350925BC(889791932)
                        transform: esp-256-aes esp-sha-hmac ,
                        in use settings ={Tunnel, }
                        conn id: 838, flow_id: 838, sibling_flags FFFFFFFF80000040, crypto
 map: ikev2-cryptomap
                        sa timing: remaining key lifetime (k/sec): (4287660676/2560)
                        IV size: 16 bytes
                        replay detection support: Y
                        Status: ACTIVE(ACTIVE)
```

```
                        inbound ah sas:
                        inbound pcp sas:
                        outbound esp sas:
                         spi: 0xB546157A(3041269114)
                           transform: esp-256-aes esp-sha-hmac ,
                           in use settings ={Tunnel, }
                           conn id: 837, flow_id: 837, sibling_flags FFFFFFFF80000040, crypto
map: ikev2-cryptomap
                           sa timing: remaining key lifetime (k/sec): (4287660672/2560)
                           IV size: 16 bytes
                           replay detection support: Y
                           Status: ACTIVE(ACTIVE)
                        outbound ah sas:
                        outbound pcp sas:
```