Configure IPsec on Catalyst 9000X Series Switches

Contents

Introduction

Prerequisites

- Requirements
- Components Used

Background Information

- Terminology

Configure

- Network Diagram
- Install HSEC license
- SVTI Tunnel Protection

Verify

- IPsec Tunnel
- IOSd Control Plane
- PD Control Plane

Troubleshoot

- IOSd
- PD Control Plane
- PD Data Plane
- Dataplane Packet-tracer
- PD Dataplane Debugging

Related Information

Introduction

This document describes how to verify Internet Protocol Security (IPsec) feature on Catalyst 9300X switches.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- IPsec

Components Used

The information in this document is based on these software and hardware versions:

- C9300X
- C9400X

- Cisco IOS® XE 17.6.4 and later

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Background Information

Starting in Cisco IOS® XE 17.5.1, Catalyst 9300-X series switches support IPsec. IPsec provides high levels of security through encryption and authentication, as well as protecting data from unauthorized access. The IPsec implementation on the C9300X provides secure tunnels between two peers using the sVTI (Static Virtual Tunnel Interface) configuration.

IPsec support on the Catalyst 9400X series switches was introduced in Cisco IOS® XE 17.10.1.

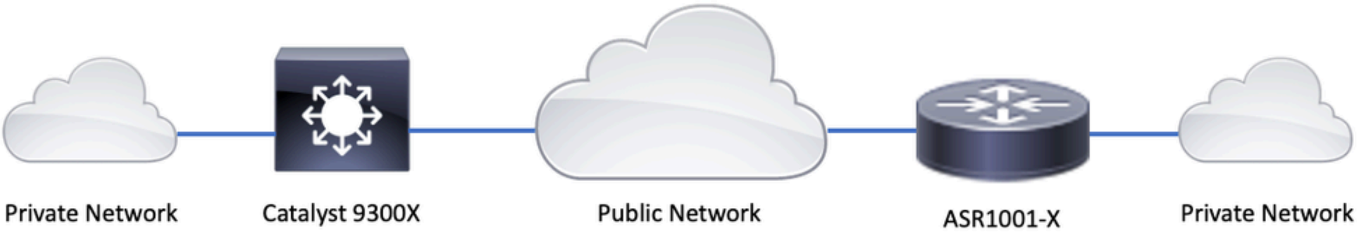
Terminology

IOSd	IOS daemon	This is the Cisco IOS daemon that runs on the Linux kernel. It is run as a software process within the kernel.IOSdprocesses CLI commands and protocols that build up state and configuration.
PD	Platform Dependent	Data and commands that are specific to the platform they are run on
IPsec	Internet Protocol Security	A secure network protocol suite that authenticates and encryptspackets of data to provide secure encrypted communication between two computers over an Internet Protocol network.
SVTI	Static Virtual Tunnel Interface	A statically configured virtual interface to which you can apply security features
SA	Security Association	An SA is a relationship between two or more entities that describes how the entities use security services to communicate securely
FED	Forwarding Engine Driver	The switch component responsible for hardware programming of UADP ASIC

Configure

Network Diagram

For the purpose of this example, the Catalyst 9300X and ASR1001-X function as IPsec peers with IPsec Virtual Tunnel Interfaces.



Install HSEC license

Enable the IPsec feature on the Catalyst 9300X platform, an HSEC license (C9000-HSEC) is required. This is different from other Cisco IOS XE based routing platforms that support IPsec, where an HSEC license is only needed to increase the allowed encryption throughtput. On the Catalyst 9300X platform, the **tunnel mode** and **tunnel protection** CLI is blocked if an HSEC license is not installed:

<#root>

```
C9300X(config)#

int tunnel1


C9300X(config-if)#

tunnel mode ipsec ipv4



%'tunnel mode' change not allowed



*Sep 19 20:54:41.068: %PLATFORM_IPSEC_HSEC-3-INVALID_HSEC: HSEC

license not present: IPSec mode configuration is rejected
```

Install the HSEC license when the switch is connected to CSSM or CSLU using Smart Licensing:

<#root>

```
C9300X#

license smart authorization request add hseck9 local


*Oct 12 20:01:36.680: %SMART_LIC-6-AUTHORIZATION_INSTALL_SUCCESS: A new licensing authorization code wa
```

Verify HSEC license is correctly installed:

<#root>

```
C9300X#

show license summary


Account Information:
  Smart Account: Cisco Systems, TAC As of Oct 13 15:50:35 2022 UTC
  Virtual Account: CORE TAC

License Usage:
  License                 Entitlement Tag               Count Status
  -----------------------------------------------------------------------------
  network-advantage       (C9300X-12Y Network Adv...)       1 IN USE
  dna-advantage           (C9300X-12Y DNA Advantage)        1 IN USE
  C9K HSEC                (Cat9K HSEC)                      0

NOT IN USE
```

Enable IPsec as the tunnel mode on the tunnel interface:

<#root>

```
C9300X(config)#

interface tunnel1


C9300X(config-if)#

tunnel mode ipsec ipv4


C9300X(config-if)#

end
```

Once IPsec is enabled, the HSEC license becomes **IN USE**

<#root>

```
C9300X#

show license summary


Account Information:
  Smart Account: Cisco Systems, TAC As of Oct 13 15:50:35 2022 UTC
  Virtual Account: CORE TAC

License Usage:
  License                 Entitlement Tag               Count Status
  -----------------------------------------------------------------------------
  network-advantage       (C9300X-12Y Network Adv...)       1 IN USE
  dna-advantage           (C9300X-12Y DNA Advantage)        1 IN USE
  C9K HSEC                (Cat9K HSEC)                      1

IN USE
```

SVTI Tunnel Protection

IPsec configuration on the C9300X uses the standard Cisco IOS XE IPsec configuration. This is a simple SVTI configuration using IKEv2 Smart Defaults, where we are using the default IKEv2 policy, IKEv2 proposal, IPsec transform, and IPsec profile for IKEv2.

C9300X Configuration

<#root>

```
ip routing

!
crypto ikev2 profile default

 match identity remote address 192.0.2.2 255.255.255.255
 authentication remote pre-share key cisco123
 authentication local pre-share key cisco123
!
interface Tunnel1

 ip address 192.168.1.1 255.255.255.252
 tunnel source 198.51.100.1
 tunnel mode ipsec ipv4
 tunnel destination 192.0.2.2

 tunnel protection ipsec profile default
```

Note: Since Catalyst 9300X is essentially an access layer switch, **ip routing** has to be explicitly enabled for routing based features like VTI to work.

Peer Configuration

<#root>

```
crypto ikev2 profile default

 match identity remote address 198.51.100.1 255.255.255.255
 authentication remote pre-share key cisco123
 authentication local pre-share key cisco123
!
interface Tunnel1

 ip address 192.168.1.2 255.255.255.252
 tunnel source 192.0.2.2
 tunnel mode ipsec ipv4
 tunnel destination 198.51.100.1

 tunnel protection ipsec profile default
```

For a more detailed discussion of the various IKEv2 and IPsec configuration constructs, please see C9300X IPsec Configuration Guide.

Verify

IPsec Tunnel

IPsec implementation on the C9300X platform is architecturally different that on the routing platforms (ASR1000, ISR4000, Catalyst 8200/8300, etc), where the IPsec feature processing is implemented in the QFP (Quantum Flow Processor) microcode.

The C9300X forwarding architecture is based on the UADP ASIC, so most of the QFP feature FIA implementation does not apply here.

Here are some of the key differences:

- **show crypto ipsec sa peer x.x.x.x platform** does not show the platform programming information from the FMAN down to the QFP.
- Packet-trace also does not work (more on this below).
- UADP ASIC does not support crypto traffic classification, so **show crypto ruleset platform** does not apply

IOSd Control Plane

IPsec control plane verification is exactly the same as that for the routing platforms, see . To display the IPsec SA installed in IOSd:

<#root>

```
C9300X#

show crypto ipsec sa


interface: Tunnel1
    Crypto map tag: Tunnel1-head-0, local addr 198.51.100.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
   remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
   current_peer 192.0.2.2 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 200, #pkts encrypt: 200, #pkts digest: 200
    #pkts decaps: 200, #pkts decrypt: 200, #pkts verify: 200
    #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.: 198.51.100.1, remote crypto endpt.: 192.0.2.2
     plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb TwentyFiveGigE1/0/1
     current outbound spi: 0x42709657(1114674775)
     PFS (Y/N): N, DH group: none


     inbound esp sas:

      spi: 0x4FE26715(1340237589)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2098,

flow_id: CAT9K:98

, sibling_flags FFFFFFFF80000048, crypto map: Tunnel1-head-0
         sa timing: remaining key lifetime (k/sec): (26/1605)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:


     outbound esp sas:

      spi: 0x42709657(1114674775)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2097,

flow_id: CAT9K:97

, sibling_flags FFFFFFFF80000048, crypto map: Tunnel1-head-0
         sa timing: remaining key lifetime (k/sec): (32/1605)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     outbound ah sas:

     outbound pcp sas:
```

Note the **flow_id** in the output, this must match the flow id installed in the forwarding plane.

PD Control Plane

Statistics between IOSd and PD control plane

<#root>

```
C9300X#

show platfor software ipsec policy statistics


                PAL CMD    REQUEST   REPLY OK  REPLY ERR      ABORT
        SADB_INIT_START          3          3          0          0
      SADB_INIT_COMPLETED        3          3          0          0
            SADB_DELETE          2          2          0          0
       SADB_ATTR_UPDATE          4          4          0          0
```

```
         SADB_INTF_ATTACH          3          3          0          0
         SADB_INTF_UPDATE          0          0          0          0
         SADB_INTF_DETACH          2          2          0          0
               ACL_INSERT          4          4          0          0
               ACL_MODIFY          0          0          0          0
               ACL_DELETE          3          3          0          0
              PEER_INSERT          7          7          0          0
              PEER_DELETE          6          6          0          0
               SPI_INSERT         39         37          2          0
               SPI_DELETE         36         36          0          0
             CFLOW_INSERT          5          5          0          0
             CFLOW_MODIFY         33         33          0          0
             CFLOW_DELETE          4          4          0          0
          IPSEC_SA_DELETE         76         76          0          0
              TBAR_CREATE          0          0          0          0
              TBAR_UPDATE          0          0          0          0
              TBAR_REMOVE          0          0          0          0
                                   0          0          0          0

               PAL NOTIFY    RECEIVE   COMPLETE   PROC ERR     IGNORE
                NOTIFY_RP          0          0          0          0
                  SA_DEAD          0          0          0          0
             SA_SOFT_LIFE         46         46          0          0
               IDLE_TIMER          0          0          0          0
                DPD_TIMER          0          0          0          0
              INVALID_SPI          0          0          0          0
                                   0          5          0          0
                 VTI SADB          0         33          0          0
                  TP SADB          0         40          0          0

IPSec PAL database summary:
                       DB NAME    ENT ADD    ENT DEL      ABORT
                      PAL_SADB          3          2          0
                   PAL_SADB_ID          3          2          0
                      PAL_INTF          3          2          0
                     PAL_SA_ID         76         74          0
                       PAL_ACL          0          0          0
                      PAL_PEER          7          6          0
                       PAL_SPI         39         38          0
                     PAL_CFLOW          5          4          0
                      PAL_TBAR          0          0          0
```

SADB Object Table

<#root>

```
C9300X#

show plat software ipsec switch active f0 sadb all

IPsec SADB object table:

  SADB-ID     Hint         Complete    #RefCnt     #CfgCnt     #ACL-Ref
  -----------------------------------------------------------------------
  3           vir-tun-int  true        2           0           0
```

SADB entry

<#root>

```
C9300X#

show plat software ipsec switch active f0 sadb identifier 3

=========== SADB id: 3
               hint: vir-tun-int
          completed: true
    reference count: 2
    configure count: 0
      ACL reference: 0

        SeqNo (Static/Dynamic)         ACL id
        ----------------------------------------
```

IPsec Flow Information

<#root>

```
C9300X#

show plat software ipsec switch active f0 flow all


===========

Flow id: 97


               mode: tunnel
          direction: outbound
           protocol: esp
                SPI: 0x42709657
      local IP addr: 198.51.100.1
     remote IP addr: 192.0.2.2
      crypto map id: 0
             SPD id: 3
         cpp SPD id: 0
    ACE line number: 0
      QFP SA handle: INVALID
   crypto device id: 0
IOS XE interface id: 65
     interface name: Tunnel1
       use path MTU: FALSE
       object state: active
  object bind state: new
===========

Flow id: 98


               mode: tunnel
          direction: inbound
           protocol: esp
                SPI: 0x4fe26715
      local IP addr: 198.51.100.1
     remote IP addr: 192.0.2.2
      crypto map id: 0
             SPD id: 3
         cpp SPD id: 0
    ACE line number: 0
      QFP SA handle: INVALID
   crypto device id: 0
IOS XE interface id: 65
     interface name: Tunnel1
       object state: active
```

Troubleshoot

IOSd

These debug and show commands are commonly collected:

<#root>

```
show crypto eli all

show crypto socket

show crypto map

show crypto ikev2 sa detail

show crypto ipsec sa

show crypto ipsec internal
```

<#root>

```
debug crypto ikev2

debug crypto ikev2 error

debug crypto ikev2 packet

debug crypto ipsec

debug crypto ipsec error

debug crypto kmi
debug crypto socket
```

```
debug tunnel protection
```

PD Control Plane

To verify the PD Control Plane operations, use the verification steps shown previously. To debug any issues related to the PD control plane, enable PD control plane debugs:

1. **Increase** the btrace logging level to verbose:

<#root>

```
C9300X#

set platform software trace forwarding-manager switch active f0 ipsec verbose


C9300X#

show platform software trace level forwarding-manager switch active f0 | in ipsec

ipsec

Verbose
```

2. **Enable** PD controlplane conditional debugging:

<#root>

```
C9300X#

debug platform condition feature ipsec controlplane submode level verbose


C9300X#

show platform conditions


Conditional Debug Global State: Stop

Feature      Type           Submode                                                             Level
------------|-------------|--------------------------------------------------------------------|-----

IPSEC

         controlplane  N/A

verbose
```

3. **Collect** the debug output from fman_fp btrace output:

<#root>

```
C9300X#

show logging process fman_fp module ipsec internal

Logging display requested on 2022/10/19 20:57:52 (UTC) for Hostname: [C9300X], Model: [C9300X-24Y], Ver

Displaying logs from the last 0 days, 0 hours, 10 minutes, 0 seconds
executing cmd on chassis 1 ...
Unified Decoder Library Init .. DONE
Found 1 UTF Streams

2022/10/19 20:50:36.686071658 {fman_fp_F0-0}{1}: [ipsec] [22441]: (ERR): IPSEC-PAL-IB-Key::
2022/10/19 20:50:36.686073648 {fman_fp_F0-0}{1}: [ipsec] [22441]: (ERR): IPSEC-b0 d0 31 04 85 36 a6 08
```

PD Data Plane

Verify the dataplane IPsec tunnel statistics including common IPsec drops such as HMAC or replay failures

<#root>

```
C9300X#

show platform software fed sw active ipsec counters if-id all

########################################
Flow Stats for if-id 0x41
########################################
-----------------------------------
Inbound Flow Info for

flow id: 98

------------------------------

SA Index: 1

--------------------
Asic Instance 0: SA Stats
  Packet Format Check Error:    0
  Invalid SA:           0
  Auth Fail:           0
  Sequence Number Overflows:    0
  Anti-Replay Fail:        0
  Packet Count:            200
  Byte Count:           27600
-----------------------------------
Outbound Flow Info for

flow id: 97

------------------------------

SA Index: 1025

--------------------
Asic Instance 0: SA Stats
  Packet Format Check Error:    0
  Invalid SA:           0
  Auth Fail:           0
  Sequence Number Overflows:    0
  Anti-Replay Fail:        0
  Packet Count:            200
  Byte Count:            33600
```

Note: the flow id matches the flow id in the **show crypto ipsec sa** output. Individual flow statistics can also be obtained with the command **show platform software fed switch active ipsec counters sa <sa_id>** where the sa_id the **SA Index** in the previous output.

Dataplane Packet-tracer

Packet-tracer on the UADP ASIC platform behaves very differently from that on the QFP based system. It can be enabled with either a manual trigger or a PCAP based trigger. Here is an example of using PCAP (EPC) based trigger.

1. **Enable** EPC and start capture:

<#root>

```
C9300X#
```

```
monitor capture test interface twentyFiveGigE 1/0/2 in match ipv4 10.1.1.2/32 any
```

<#root>

```
C9300X#
```

```
show monitor capture test
```

```
Status Information for Capture test
  Target Type:
 Interface: TwentyFiveGigE1/0/2, Direction: IN
   Status : Inactive
  Filter Details:
   IPv4
    Source IP:  10.1.1.2/32
    Destination IP:  any
   Protocol: any
  Buffer Details:
   Buffer Type: LINEAR (default)
   Buffer Size (in MB): 10
  File Details:
   File not associated
  Limit Details:
   Number of Packets to capture: 0 (no limit)
   Packet Capture duration: 0 (no limit)
   Packet Size to capture: 0 (no limit)
   Maximum number of packets to capture per second: 1000
   Packet sampling rate: 0 (no sampling)
```

2. **Run** the rest and stop the capture:

<#root>

```
C9300X#
```

```
monitor capture test start
```

```
Started capture point : test
*Oct 18 18:34:09.656: %BUFCAP-6-ENABLE: Capture Point test enabled.
<run traffic test>
```

```
C9300X#
```

```
monitor capture test stop
```

```
Capture statistics collected at software:
    Capture duration - 23 seconds
    Packets received - 5
    Packets dropped - 0
    Packets oversized - 0

Bytes dropped in asic - 0

Capture buffer will exists till exported or cleared

Stopped capture point : test
```

3. **Export** the capture into flash

<#root>

```
C9300X#

show monitor capture test buff

*Oct 18 18:34:33.569: %BUFCAP-6-DISABLE
Starting the packet display ........ Press Ctrl + Shift + 6 to exit

    1   0.000000     10.1.1.2 -> 10.2.1.2     ICMP 114 Echo (ping) request  id=0x0003, seq=0/0, ttl=255
    2   0.000607     10.1.1.2 -> 10.2.1.2     ICMP 114 Echo (ping) request  id=0x0003, seq=1/256, ttl=2
    3   0.001191     10.1.1.2 -> 10.2.1.2     ICMP 114 Echo (ping) request  id=0x0003, seq=2/512, ttl=2
    4   0.001760     10.1.1.2 -> 10.2.1.2     ICMP 114 Echo (ping) request  id=0x0003, seq=3/768, ttl=2
    5   0.002336     10.1.1.2 -> 10.2.1.2     ICMP 114 Echo (ping) request  id=0x0003, seq=4/1024, ttl=

C9300X#

monitor capture test export location flash:test.pcap
```

4. **Run** packet-tracer:

<#root>

```
C9300X#

show platform hardware fed switch 1 forward interface TwentyFiveGigE 1/0/2 pcap flash:test.pcap number 1

Show forward is running in the background. After completion, syslog will be generated.

C9300X#
*Oct 18 18:36:56.288: %SHFWD-6-PACKET_TRACE_DONE: Switch 1 F0/0: fed: Packet Trace Complete:  Execute (
*Oct 18 18:36:56.288: %SHFWD-6-PACKET_TRACE_FLOW_ID: Switch 1 F0/0: fed: Packet Trace Flow id is 131077
C9300X#
C9300X#show plat hardware fed switch 1 forward last summary
Input Packet Details:
###[ Ethernet ]###
  dst       = b0:8b:d0:8d:6b:d6
  src=78:ba:f9:ab:a7:03
  type      = 0x800
###[ IP ]###
     version   = 4
     ihl       = 5
     tos       = 0x0
     len       = 100
     id        = 15
     flags     =
     frag      = 0
     ttl       = 255
     proto     = icmp
     chksum    = 0xa583
     src=10.1.1.2
     dst       = 10.2.1.2
     options   = ''
###[ ICMP ]###
        type      = echo-request
        code      = 0
        chksum    = 0xae17
        id        = 0x3
        seq       = 0x0
###[ Raw ]###
           load      = '00 00 00 00 01 1B CF 14 AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD AB CD A
Ingress:
   Port                        : TwentyFiveGigE1/0/2
   Global Port Number          : 2
   Local Port Number           : 2
   Asic Port Number            : 1
   Asic Instance               : 1
   Vlan                        : 4095
   Mapped Vlan ID              : 1
   STP Instance                : 1
   BlockForward                : 0
   BlockLearn                  : 0
   L3 Interface                : 38
       IPv4 Routing            : enabled
       IPv6 Routing            : enabled
       Vrf Id                  : 0
   Adjacency:
       Station Index           : 179
       Destination Index       : 20754
       Rewrite Index           : 24
       Replication Bit Map     : 0x1    ['remoteData']
   Decision:
       Destination Index       : 20754  [DI_RCP_PORT3]
       Rewrite Index           : 24
       Dest Mod Index          : 0      [IGR_FIXED_DMI_NULL_VALUE]
       CPU Map Index           : 0      [CMI_NULL]
       Forwarding Mode         : 3      [Other or Tunnel]
       Replication Bit Map     :        ['remoteData']
       Winner                  :        L3FWDIPV4 LOOKUP
       Qos Label               : 1
       SGT                     : 0
       DGTID                   : 0
Egress:
   Possible Replication        :
       Port                    : RCP
       Asic Instance           : 0
       Asic Port Number        : 0
   Output Port Data            :
     Port                      : RCP
       Asic Instance           : 0
       Asic Port Number        : 90
       Unique RI               : 0
       Rewrite Type            : 0      [Unknown]
       Mapped Rewrite Type     : 229    [IPSEC_TUNNEL_MODE_ENCAP_FIRSTPASS_OUTERV4_INNERV4]
       Vlan                    : 0
       Mapped Vlan ID          : 0
        RCP, mappedRii.fdMuxProfileSet = 1 , get fdMuxProfile from MappedRii
       Qos Label               : 1
       SGT                     : 0
********************************************************************************
Input Packet Details:
N/A: Recirculated Packet
Ingress:
   Port                        : Recirculation Port
   Asic Port Number            : 90
   Asic Instance               : 0
   Vlan                        : 0
   Mapped Vlan ID              : 2
   STP Instance                : 0
   BlockForward              : 0
   BlockLearn                : 0
   L3 Interface              : 38
       IPv4 Routing          : enabled
       IPv6 Routing          : enabled
       Vrf Id                : 0
   Adjacency:
       Station Index         : 177
       Destination Index     : 21304
       Rewrite Index         : 21
       Replication Bit Map   : 0x1    ['remoteData']
   Decision:
       Destination Index     : 21304
       Rewrite Index         : 21
       Dest Mod Index        : 0      [IGR_FIXED_DMI_NULL_VALUE]
       CPU Map Index         : 0      [CMI_NULL]
       Forwarding Mode       : 3      [Other or Tunnel]
       Replication Bit Map   :        ['remoteData']
       Winner                :        L3FWDIPV4 LOOKUP
       Qos Label             : 1
       SGT                   : 0
       DGTID                 : 0
Egress:
   Possible Replication      :
       Port                  : TwentyFiveGigE1/0/1
   Output Port Data          :
     Port                    : TwentyFiveGigE1/0/1
       Global Port Number    : 1
       Local Port Number     : 1
       Asic Port Number      : 0
       Asic Instance         : 1
       Unique RI             : 0
       Rewrite Type          : 0      [Unknown]
       Mapped Rewrite Type   : 13     [L3_UNICAST_IPV4_PARTIAL]
       Vlan                  : 0
       Mapped Vlan ID        : 0

Output Packet Details:
   Port                      : TwentyFiveGigE1/0/1
###[ Ethernet ]###
  dst       = 00:62:ec:da:e0:02
  src=b0:8b:d0:8d:6b:e4
  type      = 0x800
###[ IP ]###
     version   = 4
     ihl       = 5
     tos       = 0x0
     len       = 168
     id        = 2114
     flags     = DF
     frag      = 0
     ttl       = 254
     proto     = ipv6_crypt
     chksum    = 0x45db
     src=198.51.100.1
     dst       = 192.0.2.2
     options   = ''
###[ Raw ]###        load      = '
```

6D 18 45 C9

```
 00 00 00 06 09 B0 DC 13 11 FA DC F8 63 98 51 98 33 11 9C C0 D7 24 BF C2 1C 45 D3 1B 91 0B 5F B4 3A C0
********************************************************************************
```

```
C9300X#

show crypto ipsec sa | in current outbound


 current outbound spi:

0x6D1845C9

(1830307273)

<-- Matches the load result in packet trace
```

Note: in the previous output, the packet forwarded egress is the ESP packet with the current outbound SA SPI. For a more detailed FED forwarding decision analysis, the **detail** variant of the same command. Example: **show plat hardware fed switch 1 forward last detail** can be used**.**

PD Dataplane Debugging

Note: PD dataplane debugging must only be enabled with assistance from TAC. These are very low level traces that engineering needs if the issue cannot be identified via normal CLIs/Debugs.

<#root>

```
C9300X#

set platform software trace fed switch active ipsec verbose

C9300X#

debug platform condition feature ipsec dataplane submode all level verbose

C9300X#

show logging process fed module ipsec internal
```

IPsec PD SHIM Debugs

<#root>

```
debug platform software ipsec info
```

```
debug platform software ipsec error
```

```
debug platform software ipsec verbose
```

```
debug platform software ipsec all
```

Related Information

- Configure IPsec on Catalyst 9300 Switches
- Configure IPsec on Catalyst 9400 Switches
- Cisco IOS XE 17.12.1 for Catalyst Switching