Understand and Troubleshoot IPsec and DTLS Offloading in Secure Firepower 3100 and 4200

Contents

Introduction

This document describes troubleshooting common issues in Firepower architecture responsible for handling flow offloading.

Prerequisites

IPSec configuration either route-based or policy-based or both.

Requirements

Cisco recommends that you have knowledge of these topics:

- Site to Site VPN
- Remote Access VPN

Components Used

The information in this document is based on:

- Cisco Secure Firewall Threat Defense 7.2.0+
- Cisco Secure Firewall 3K/4K

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.

Feature Information

Supporting device models use IPsec flow offload where after the initial negotiation of an IPsec site-to-site VPN or remote access VPN security association (SA), IPsec connections are offloaded to the field-programmable gate array (FPGA) in the device, which improves device performance.

Offloaded operations are specifically related to the pre-decryption and decryption processing on ingress, and pre-encryption and encryption processing on egress. The system software handles the inner flow to apply your security policies.

Supported Platforms

IPsec flow offload is enabled by default, and applies to these device types so far:

- Secure Firewall 3100
- Secure Firewall 4200

IPsec flow offload is also used when the VTI is sourced from the loopback interface.

IPsec offloading is available on supported platforms starting:

- Secure Firewall FTD 7.2
- Secure Firewall ASA 9.18

While DTLS offloading is available on supported platforms starting:

- Secure Firewall FTD 7.6
- Secure Firewall ASA 9.22

Limitation

IPSec Offloading

These are the limitations of IPsec offloading:

- IKEv1
- Transport Mode
- Compression
- Post-fragmentation
- Anti-replay with window size other than 64-bits
- Firewall-filters for tunneled traffic
- Multi-context

DTLS Offloading

These are the limitations of DTLS offloading:

- DTLS 1.0
- Compression
- Multi-context
- Multi-instance

- Cluster

Configuration

Flow offload is enabled by default on supported platforms for both IPSEC and DTLS. Cli / flex-config can be leveraged for enabling or disabling it.

<#root>

```
FPR(config)#flow-offload-ipsec
FPR(config)#no flow-offload-ipsec
```

<<<<<<< disable flow-offload for ipsec

```
FPR(config)#flow-offload-ipsec egress-optimization
FPR(config)#no flow-offload-ipsec egress-optimization
```

<<<<<<< disable egress optimization for ipsec

```
FPR(config)#flow-offload-dtls
FPR(config)#no flow-offload-dtls
```

<<<<<<< disable flow-offload for DTLS

```
FPR(config)#flow-offload-dtls egress-optimization
FPR(config)#no flow-offload-dtls egress-optimization
```

<<<<<<< disable egress optimization for DTLS

Troubleshooting

Before proceeding further, please understand that offloading does not kick in until the negotiation is completed and you have SA established. The case is pretty much same for DTLS too so problems during initial handshakes or negotiations are possibly unrelated to offloading and can have the traditional troubleshooting approach with debugs and necessary captures. Specific problems related to flow offloading can occur in the form of traffic disruption.

Here are few important commands which can be executed to elicit a confirmation if you have flow offload enabled and problem is with the packet handling due to flow offload.

- Verify the **show crypto ipsec sa** command to check if offload is enabled.

<#root>

firepower# show crypto ipsec sa peer 203.0.113.2

```
peer address: 203.0.113.2
    Crypto map tag: CSM_dmz_a_001_map, seq num: 1, local addr: 203.0.113.1
```

```
      access-list CSM_IPSEC_ACL_1 extended permit ip 192.0.2.0 255.255.255.252 192.0.2.4 255.255.255.25
      Protected vrf (ivrf):
      local ident (addr/mask/prot/port): (192.0.2.0/255.255.255.252/0/0)
      remote ident (addr/mask/prot/port): (192.0.2.4/255.255.252.252/0/0)
      current_peer: 203.0.113.2


      #pkts encaps: 443, #pkts encrypt: 443, #pkts digest: 443
      #pkts decaps: 10254, #pkts decrypt: 10254, #pkts verify: 10254
      #pkts compressed: 0, #pkts decompressed: 0
      #pkts not compressed: 443, #pkts comp failed: 0, #pkts decomp failed: 0
      #pre-frag successes: 0, #pre-frag failures: 0, #fragments created: 0
      #PMTUs sent: 0, #PMTUs rcvd: 0, #decapsulated frgs needing reassembly: 0
      #TFC rcvd: 0, #TFC sent: 0
      #Valid ICMP Errors rcvd: 0, #Invalid ICMP Errors rcvd: 0
      #send errors: 886, #recv errors: 0

      local crypto endpt.: 203.0.113.1/500, remote crypto endpt.: 203.0.113.2/500
      path mtu 1500, ipsec overhead 86(44), media mtu 1500
      PMTU time remaining (sec): 0, DF policy: copy-df
      ICMP error validation: disabled, TFC packets: disabled
      current outbound spi: XXXXXXXX
      current inbound spi : YYYYYYYY

    inbound esp sas:
      spi: 0xYYYYYYYY (YYYYYYYY)
        SA State: active
        transform: esp-aes-256 esp-sha-384-hmac no compression
        in use settings ={L2L, Tunnel, PFS Group 14, IKEv2,
CAN_BE_OFFLOADED, OFFLOADED, } <<<<<<<


        slot: 0, conn_id: 80438, crypto-map: CSM_cisco_map
        sa timing: remaining key lifetime (kB/sec): (32808888/26585)
        IV size: 16 bytes
        replay detection support: Y
        Anti replay bitmap:
         0xFFFFFFFF 0xFFFFFFFF
    outbound esp sas:
      spi: 0xXXXXXXXX (XXXXXXXX)
        SA State: active
        transform: esp-aes-256 esp-sha-384-hmac no compression
        in use settings ={L2L, Tunnel, PFS Group 14, IKEv2,
CAN_BE_OFFLOADED, OFFLOADED, } <<<<<<<


        slot: 0, conn_id: 80438, crypto-map: CSM_cisco_map
        sa timing: remaining key lifetime (kB/sec): (34652026/26584)
        IV size: 16 bytes
        replay detection support: Y
        Anti replay bitmap:
         0x00000000 0x00000001
```

- The **show ipsec stats** command can also be referred for offloading confirmation.

<#root>

```
firepower# show ipsec stats
```

```
IPsec Global Statistics
-----------------------
Active tunnels: 1
Previous tunnels: 54
Inbound
    Bytes: 3396848
    Decompressed bytes: 0
    Packets: 30329
    Dropped packets: 0
    Replay failures: 0
    Authentications: 30329
    Authentication failures: 0
    Decryptions: 30329
    Decryption failures: 0
    TFC Packets: 0
    Decapsulated fragments needing reassembly: 0
    Valid ICMP Errors rcvd: 0
    Invalid ICMP Errors rcvd: 0
 Outbound
    Bytes: 3431248
    Uncompressed bytes: 1837548
    Packets: 30585
    Dropped packets: 0
    Authentications: 30584
    Authentication failures: 0
    Encryptions: 30584
    Encryption failures: 0
    TFC Packets: 0
    Fragmentation successes: 0
        Pre-fragmentation successes: 0
        Post-fragmentation successes: 0
    Fragmentation failures: 0
        Pre-fragmentation failures: 0
        Post-fragmentation failures: 0
    Fragments created: 0
    PMTUs sent: 0
    PMTUs rcvd: 0

 Offloaded Inbound, Pipe 0 <<<<<<<<<<<<<<<<


    Bytes: 3336256


    Packets: 29788


    Authentications: 29788


    Decryptions: 29788


 Offloaded Outbound, Pipe 0 <<<<<<<<<<<<<<<<


    Bytes: 3358480
```

```
        Packets: 29918


        Authentications: 29918


        Encryptions: 29918


      Protocol failures: 0
      Missing SA failures: 0
      System capacity failures: 0
      Inbound SA delete requests: 89
      Outbound SA delete requests: 0
      Inbound SA destroy calls: 75
      Outbound SA destroy calls: 71
```

- Moving forward FTD 7.6 / ASA 9.22 when DTLS offload was also introduced, you can use **show flow-offload info detail** to check status of both DTLS and IPSEC offload.

<#root>

```
firepower# show flow-offload info detail



Current running state : Enabled


  Dynamic flow offload : Disabled
  flow table refresh count : 7866 [246]
  HW flow table refresh count : 246
  Invalid Precomputation Request: 2755106
  instance_id:2304 interface_id:105 action:0 logic_id_opt:0 subinterface_id_opt:1 service_map: 0000000b
  instance_id:256 interface_id:105 action:0 logic_id_opt:0 subinterface_id_opt:1 service_map: 0000000b
  ..........
  instance_id:256 interface_id:1011 action:0 logic_id_opt:0 subinterface_id_opt:1 service_map: 0000000b
  Services Registered:
  Flow_Offload: Registered

  IPSec Offload: Registered <<<<<<<<<<<


  DTLS Offload: Registered  <<<<<<<<<<<
```

- Check for possible drop / error counters in the **show flow-offload statistics** output. The output can vary depending on the problem. In some cases, 6-tuple miss count was seen very high. But best practice is these outputs are supposed to be gathered multiple times during the problem to have a comparative analysis on the appropriate counters. Simplest explanation for CAM stats counters are each counter has specific purpose and mostly they act as classifier responsible for identifying the packet and forwarding it correctly. The two sets of outputs represents IPSEC and DTLS stats.

<#root>

These outputs can also be fetched separately for DTLS and IPSEC by

show flow-offload-ipsec statistics

and

show flow-offload-dtls statistics

.

firepower# show flow-offload info detail

```
  Packet stats of Pipe 0
  ----------------------
  Rx Packet count : 50736432
  Tx Packet count : 45999280
  Error Packet count : 0 <<<<<<<<<


  Drop Packet count : 0 <<<<<<<<<



  NOTE: The CAM counters displayed are cumulative counters
  for all offload applications and indicates the total packets offloaded

  CAM stats of Pipe 0
  ----------------------
  Option ID Table CAM Hit Count : 9675832699
  Option ID Table CAM Miss Count : 0
  Tunnel Table CAM Hit Count : 0
  Tunnel Table CAM Miss Count : 74
  6-Tuple CAM Hit Count : 177440969
  6-Tuple CAM Miss Count : 9498391657

  NOTE: The counters displayed are cumulative counters
  for all offload applications and indicates the total packets offloaded

  Packet stats of Pipe 0
  ----------------------
  Rx Packet count : 48444809
  Tx Packet count : 44575287
  Error Packet count : 0 <<<<<<<<<


  Drop Packet count : 41 <<<<<<<<<



  NOTE: The CAM counters displayed are cumulative counters
  for all offload applications and indicates the total packets offloaded

  CAM stats of Pipe 0
  ----------------------
  Option ID Table CAM Hit Count : 9675832699
  Option ID Table CAM Miss Count : 0
  Tunnel Table CAM Hit Count : 0
  Tunnel Table CAM Miss Count : 74
```

```
  6-Tuple CAM Hit Count : 177440969
  6-Tuple CAM Miss Count : 9498391657

  NOTE: The counters displayed are cumulative counters
  for all offload applications and indicates the total packets offloaded
```

- The **show counters** command can also be referred for offload counters and advised to be gathered multiple times for the sake of comparative analysis.

<#root>

For IPSEC offload

```
firepower# show counters
    IPSEC   OFFLOAD_IB_PKT_PROCESS                46201663    Summary
    IPSEC   OFFLOAD_IB_PKT_PROCESS_SUCCESS        46201663    Summary
    IPSEC   OFFLOAD_OB_PKT_PROCESS                44580990    Summary
    IPSEC   OFFLOAD_OB_PKT_PROCESS_SUCCESS        44580990    Summary
    IPSEC   OFFLOAD_EGRESS_OPTIMIZE_PKT           44580990    Summary
    IPSEC   OFFLOAD_FLOW_INBOUND_ADD_RULE         296         Summary
    IPSEC   OFFLOAD_FLOW_OUTBOUND_ADD_RULE        296         Summary
    IPSEC   OFFLOAD_FLOW_INBOUND_DEL_RULE         286         Summary
    IPSEC   OFFLOAD_FLOW_OUTBOUND_DEL_RULE        286         Summary
    IPSEC   OFFLOAD_FLOW_INBOUND_UPDATE_SUCCESS   253         Summary
```

For DTLS offload

```
firepower# show counters
    CRYPTO   DTLS_OFFLOAD_IB_PKT_PROCESS          11122701    Summary
    CRYPTO   DTLS_OFFLOAD_IB_PKT_SUCCESS          11122701    Summary
    CRYPTO   DTLS_OFFLOAD_OB_PKT_PROCESS          27269819    Summary
    CRYPTO   DTLS_OFFLOAD_OB_PKT_SUCCESS          27269819    Summary
    CRYPTO   DTLS_OFFLOAD_FLOW_IB_ADD_RULE        4189        Summary
    CRYPTO   DTLS_OFFLOAD_FLOW_OB_ADD_RULE        4189        Summary
    CRYPTO   DTLS_OFFLOAD_FLOW_IB_UPDATE_SUCCESS  3730        Summary
    CRYPTO   DTLS_OFFLOAD_RX_ALERT                621         Summary
    CRYPTO   DTLS_OFFLOAD_CONTROL_IN_PKT          226951      Summary
    CRYPTO   DTLS_OFFLOAD_EGRESS_OPTIMIZE_PKT     27269819    Summary
```

- IPSEC or DTLS offload captures can be gathered to make sure you are receiving the encrypted packets if nothing is seen in LINA captures. LINA captures only prints the outputs if FPGA handled the incoming packet correctly and injected it into datapath. If packet was not handled correctly by FPGA, then there are chances nothing is visible in the LINA captures but this does not imply that you have not received any packet at all. Any tool can be used to restore the dumps to readable format.

<#root>

```
firepower# capture TAC ipsec-offload match spi 0x7XXXXXX9 203.0.113.1 203.0.113.2
```

```
<<< for IPSEC

firepower# capture TAC-DTLS dtls-offload match udp 203.0.113.1 eq <src port> 203.0.113.2 eq <dst port>
<<< for DTLS

firepower# show capture TAC
<<<< this is extracted for ipsec-offload


2 packets captured

   1: 13:54:40.883758       20db.ea88.ce95 c860.8f37.f614 0xc008 Length: 202

                          xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
                          xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
                          83a8 7c14 3c64 594f 951d ca36 0e4d ca7e
                          2d34 d4ea 3515 0202 ce36 ace9 59a5 6f69
                          04c6 8ff9 ddf7 9e82 f6c2 11c5
  2: 13:54:42.877014       20db.ea88.ce95 c860.8f37.f614 0xc008 Length: 202

                          xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
                          xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
                          3e83 a9b4 63b1 41cb 2408 0de1 4819 288b
                          9df8 fade 611e a338 98e5 74ec 552f c37d
                          8aa0 42d9 0b68 e5e7 7876 8bab
2 packets shown
```

- You also have option to check switch level captures to make sure that traffic is received and forwarded to FPGA correctly. These captures are taken from lab environment, please make sure to apply appropriate filters to minimize the impact on production environments. Details can be referred in Secure Firewall Captures.

```
firepower# capture TAC switch interface <interface name> match ip 203.0.113.1 203.0.113.2
OR
firepower# capture TAC switch real-time
6 packets captured using switch real-time capture

    1: 09:10:29.298126 bc5a.56ac.6702 e4a4.0400.11bc 0x2057 Length: 150

               xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
               xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
               c685 5d8e c938 1617 c72e 7028 af65 aeba
               04b8 d2d5 db53 783f afed a8ee 9dcd 5938
               f198 e89f 5555 5555
    2: 09:10:39.298751 bc5a.56ac.6702 e4a4.0400.11bc 0x2057 Length: 150

               xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
               xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
               a340 8252 d626 6cd8 f16a c6f7 3460 0e5a
               290a 5ca7 8f9b 864c ef76 cdad 1839 8020
               2590 804b 5555 5555
    3: 09:10:49.298766 bc5a.56ac.6702 e4a4.0400.11bc 0x2057 Length: 150

               xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
```

```
              xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
              7ebc d4f3 c706 55ac 1358 ab7c 6363 9827
              ec29 47fe 4f91 4967 73a3 b646 7499 9269
              0816 f463 5555 5555

    4: 09:10:59.303405 bc5a.56ac.6702 e4a4.0400.11bc 0x2057 Length: 150

              xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
              xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
              d15c 1115 3042 72b4 3b81 88ea 7548 c7e4
              3401 b7ba 5555 5555
    5: 09:11:09.308165 bc5a.56ac.6702 e4a4.0400.11bc 0x2057 Length: 150

              xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
              xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
              752b 0ed4 1f2d 3429 0a09 bda5 2c68 1acd
              64e9 7e5e 5555 5555
    6: 09:11:19.313139 bc5a.56ac.6702 e4a4.0400.11bc 0x2057 Length: 150

              xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
              xxxx xxxx xxxx xxxx xxxx xxxx xxxx xxxx
              0631 4b9d 0a08 52b5 d084 cb39 d55a ad91
              777c cfe4 5555 5555
    6 packets shown
```

- For DTLS specific outputs, along with the previous **show** outputs, this can be verified for session specific data. These can also be fetched multiple times for the sake of analysis, especially the marked counters confirming if packets are processed and forwarded correctly.

<#root>

```
firepower# show asp table socket offloaded


Protocol   Socket    State      Local Address               Foreign Address               IB-Pipe#

SVC_UDP    104d40e8  CONNECTED

  203.0.113.5:443              198.51.100.5:3875             0            0
SVC_UDP    0f435518  CONNECTED  203.0.113.5:443             198.51.100.6:13265            0


firepower# show asp table socket 104d40e8 detail


Statistics for socket

0x104d40e8

:

3) AM Module
   Mod handle: 0x00000000104d40eb
   Rx:  0/3  ( 0 queued), Flow-Ctrl:      0,  Tot:      1
   Tx:  0/3  ( 0 queued), Flow-Ctrl:      0,  Tot:      0
        App Flow-Ctrl Tx: 0
        Stack:         0x000014a89473bb80
        New Conn Cb:   0x00005559542f6130
        Notify   Cb:   0x00005559542f62a0
```

```
        App Hdl:       0x000000000549358a
        Shared  Lock:  0x000014a7e010d848
          Group Lock:  0x000014a7e010d848
          Async Lock:  0x000014a84a270b40
        Closed Mod Rx: -1, Tx: 4
        Push Module:   INVALID
        State:         CONNECTED
        Flags: 0x500003
          Inbound
          Accepted
          New Conn App Notify Success
          Stack Ref count

2) SVC_UDP Module
   Mod handle: 0x000014a8921aa180
   Rx:  0/1  ( 0 queued), Flow-Ctrl:      0,  Tot:      1
   Tx:  0/1  ( 0 queued), Flow-Ctrl:      0,  Tot:    785
        Idle (ms):              0
        DF-Bit Ignore:          Disable
        MTU:                    1150
        Fragmented Packets:     0
        Downstream:
           Data Pkts/Bytes:     768/481092


Drop Pkts/Bytes:     0/0


           Ctrl Pkts/Bytes:     15/10347
        Upstream:
           Data Pkts/Bytes:     1093/536093


Drop Pkts/Bytes:     0/0


           Ctrl Pkts/Bytes:     21/102
        Offload Stats:


#pkts in: 1093, #bytes in: 536093, #pkts decrypt: 1093 <<<<<<< this is expected to match with vpn-sessio


           #pkts out: 767, #bytes out: 480393, #pkts encrypt: 767


<<<<<<< this is expected to match with vpn-sessiondb det output counters


           #send errors: 0, #recv errors: 0
           #pkts failed (send): 0, #pkts failed (rcv): 0
           #pkts replay failed (rcv): 0


1) DTLS Module
   Mod handle: 0x000014a89030f300
   Rx:  0/128 ( 0 queued), Flow-Ctrl:      0,  Tot:      0
   Tx:  0/128 ( 0 queued), Flow-Ctrl:      0,  Tot:    786
        Upstream Active/peak/total:     0/0/0
        Downstream Active/peak/total:   0/1/785
        Inbound bytes rx/tx:    303/0
        Inbound packets rx/tx:  2/0
        Inbound packets lost:   0
```

```
        Outbound bytes rx/tx:   427737/444392
        Outbound packets rx/tx: 785/786
        Outbound packets lost:  0
        Upstream Close Attempt:         0
        Upstream Close Forced:          0
        Upstream Close Next:            0
        Upstream Close Handshake:       0
        Downstream Close Attempt:       0
        Downstream Close Forced:        0
        Downstream Close Next:          0
        Inbound discard empty buf:      0
        Empty downstream buf:   0
        Encrypt call:           0
        Encrypt call error:     0
        Encrypt handoff:        0
        Encrypt CB success:     0
        Encrypt CB fail:        0
        Flowed Off:             0
        Stats Last State:       0x20 (TRFIN)
        Pending crypto cmds:    0
        Socket Last State:      0x1 (SSLOK )
        Socket Read State:      0xf0 (read header)
        Handle Read State:      0xf0 (read header)
        References:             2
        In Rekey:               0x0
        Flags:                  0x2000000
        Header Len:             13
        Record Type:            0x0
        Record Len:             0
        Queued Blocks:          0
        Queued Bytes:           0

0) TM Module
   Mod handle: 0x00000000104d40e8
   Rx:  0/1  (

0 queued

), Flow-Ctrl:      0,  Tot:      2
   Tx:  0/1  (

0 queued

), Flow-Ctrl:      0,  Tot:    786
        Transp Flow-Ctrl Rx: 0
        UDP handle: 0x000014a890217500
        Conn Timeout: 1800000 ms
        Local host: 203.0.113.5, Local port: 443
        Foreign host:198.51.100.5, Foreign port: 3875
          Rcvd:                 2
            with data:          2
            total data bytes:   303
          Sent:                 786
            with data:          786
            total data bytes:   444392

          Dropped:


            Rcv queue full:     0 <<<<<<<<<
```

- There are few additional CLIs which can be executed depending on the requirement.

<#root>

Global stats

- show flow-offload-dtls statistics
- show crypto protocol ssl statistics

(aggregate of offloaded/ non-offloaded stats)

- show ssl mib

(aggregate of offloaded/ non-offloaded stats)

- show crypto accelerator statistics

(separate Offloaded statistics added)

Clearing stats

- clear flow-offload-dtls statistics

- Along with this, for both DTLS and IPSEC offload, the **show npu-accel statistics** can also be gathered from the fxos CLI multiple times during the issue to verify few important counters. This output varies, depending on the problem type and the environment.

<#root>

>show npu-accel statistics

Output is cropped and gathered from one of the affected devices.

ilk_tx_good_pkt_cnt = 133997299
ilk_rx_good_pkt_cnt = 129123883

ilk_tx_err_pkt_cnt = 0 <<<<<<<<<<

ilk_tx_taildrop_pkt_cnt = 4867559 <<<<<<<<<<

ilk_tx_fifo_sbit_err_cnt = 0 <<<<<<<<<<

ilk_tx_fifo_dbit_err_cnt = 0 <<<<<<<<<<

ilk_rx_fifo_sbit_err_cnt = 0 <<<<<<<<<<

ilk_rx_fifo_dbit_err_cnt = 0 <<<<<<<<<<

ilk_rx_err_pkt_cnt = 0 <<<<<<<<<<

ilk_rx_seg_sop_cnt = 129123883
ilk_rx_seg_eop_cnt = 129123883

module: nvppu, pipe: 0
----
nvppu_ipsec_in_pkt_count = 46201704
nvppu_ipsec_in_byte_count = 5970198256
nvppu_ipsec_in_decrypt_pkt_count = 46201704
nvppu_ipsec_in_decrypt_byte_count = 4122130096
nvppu_ipsec_in_hash_pkt_count = 46201704
nvppu_ipsec_in_hash_byte_count = 5230970992
nvppu_ipsec_out_pkt_count = 44575287
nvppu_ipsec_out_byte_count = 31277069992
nvppu_ipsec_out_encrypt_pkt_count = 44575287
nvppu_ipsec_out_encrypt_byte_count = 29494058512
nvppu_ipsec_out_hash_pkt_count = 44575287
nvppu_ipsec_out_hash_byte_count = 30563865400

nvppu_ipsec_drop_pkt_count = 0 <<<<<<<<<<

nvppu_dtls_in_pkt_count = 11122815
nvppu_dtls_in_byte_count = 2810772142
nvppu_dtls_out_pkt_count = 27223995
nvppu_dtls_out_byte_count = 17111805764

nvppu_dtls_in_drop_pkt_count = 82   <<<<<<<<<<

nvppu_dtls_out_drop_pkt_count = 0 <<<<<<<<<<

nvppu_filtering_total_cnt = 46201704

nvppu_tfc_drop_cnt = 0 <<<<<<<<<<

nvppu_filtering_drop_cnt = 41   <<<<<<<<<<

nvppu_anti_drop_cnt = 0 <<<<<<<<<<

nvppu_dtls_anti_drop_cnt = 114 <<<<<<<<<<

- Generally, it is recommended to have troubleshooting file of FXOS and FTD both collected along with show tech support from FTD CLI from both devices in case if they are running in HA for the

analysis along with the previous outputs.

Conclusion

The purpose of this document is to explain in depth how to gather offload specific outputs as this is challenging in terms of limited visibility because of the architectural changes done in newer FPGA based platforms.