Troubleshoot Bandwidth and Throughput Issues

For supported software information, click [here](#).

This article describes steps for troubleshooting bandwidth and through issues.

Check for Half-Duplex Issues and Link Speed

To check the link speed and to check for half-duplex issues, issue the following CLI command:

```
admin@cli > show interfaces detail interface-name
```

For example:

```
admin@CPE3-cli> show interfaces detail vni-0/0
Interface: vni-0/0
  Tenant
  Vlan-Id
  Administrative status : up
  Operational status    : n/a
  Protocols Down        : up
  Interface index       : 1061
  Interface Role        : external
  MAC address           : 00:90:0b:54:e9:08
  IP address            : n/a
  Obtained from DHCP    : False
  DHCP Server IP        : n/a
  DHCP Lease Time       : n/a
  DHCP Leaase Expiry    : n/a
  Name Server 1 Address : n/a
  Name Server 2 Address : n/a
  Routing instance      : Interenet-Transport-VR (10)
  Host interface        : eth1
  MTU                   : 1500
  Duplex / Speed        : half-duplex / 100mbps
    RX packets:12767402441  errors:0
    RX bytes:15555320808343
    TX packets:22243079055  errors:2
    TX bytes:27602690732046
```

The highlighted text in the command output shows that the interface is in half-duplex mode and operating at 100 Mbps. The interface should not be in half-duplex mode and, here, 100 Mbps is incorrect.

Fix any half-duplex and link speed issues by correcting the configuration on the device to which the VOS device is connected. For the transmission odes configured on the ISP side, check for an auto/auto configuration.

Check for Asymmetrical SD-WAN Paths

Check that there are no asymmetrical SD-WAN paths. An example is a path on which traffic is transmitted on one transport network and returns on another transport that has different bandwidth.

To check for traffic traverses an asymmetrical SD-WAN path, issue the following CLI command:

> admin@cli > **show orgs org** *organization-name* **sessions sdwan brief**

For example:

```
admin@cli > show orgs org Tenant-Common sessions sdwan brief

VSN VSN  SESS                DESTINATION     SOURCE  DESTINATION
ID  VID  ID    SOURCE IP       IP             PORT    PORT       PROTOCOL NATTED SDWAN APPLICATION RX
WAN CKT        TX WAN CKT
0   2    33287 192.168.50.100  192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE50
0   2    33293 192.168.55.100  192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE55
0   2    33295 192.168.59.100  192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE59
0   2    33306 192.168.2.100   192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE2
0   2    33306 192.168.3.100   192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE3
0   2    33307 192.168.5.100   192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE5
0   2    33308 192.168.12.100  192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE12
0   2    33320 192.168.19.100  192.168.106.100 1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE19
0   2    33362 192.168.106.100 192.168.31.100  1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE31
0   2    33363 192.168.106.100 192.168.33.100  1024    1024       17       No     Yes   unknown_udp
Internet:Internet MPLS:MPLS         Silver-Customer-CPE33
0   2    33364 192.168.106.100 192.168.36.100  1024    1024       17       No     Yes   unknown_udp
Internet:Internet MPLS-v6:MPLS-v6   Silver-Customer-CPE36
0   2    33365 192.168.106.100 192.168.61.100  1024    1024       17       No     Yes   unknown_udp MPLS-
v6:MPLS-v6   Internet:Internet Silver-Customer-CPE61
0   2    33366 192.168.106.100 192.168.62.100  1024    1024       17       No     Yes   unknown_udp MPLS-
v6:MPLS-v6   MPLS:MPLS         Silver-Customer-CPE62
0   2    33367 192.168.106.100 192.168.66.100  1024    1024       17       No     Yes   unknown_udp MPLS-
v6:MPLS-v6   MPLS-v6:MPLS-v6   Silver-Customer-CPE66
0   2    33368 192.168.106.100 192.168.75.100  1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Silver-Customer-CPE75
0   2    33369 192.168.106.100 192.168.81.100  1024    1024       17       No     Yes   unknown_udp
Internet:Internet Internet:Internet Gold-Customer-CPE81
```

If the SD-WAN paths are symmetrical, the circuits shown in the RX WAN CKT and TX WAN CKT fields must be the same on the local and remote sites. If, for example, a packet is transmitted to a remote branch on the Internet circuit and returns on an MPLS circuit, you may see different throughput based on the bandwidth available on the Internet and MPLS circuits.

If you do not enable FEC and replication, the overhead added to packets is 82 bytes. Enabling FEC adds 12 bytes and enabling replication adds 12 bytes, for a total of 24 bytes in addition to the 82 bytes. To add headers while doing performance testing of sending traffic over tunnels, it is recommended that you use the Spirent or IXIA tools to add 82 bytes to the packets.

For traffic going from one LAN to another over SD-WAN, the packet overhead is based on the packet size and type of traffic.

Check that Packets Are not Dropped by CoS

If a CoS shaper or rate limiter is configured on the VOS device, it may drop packets when the number of packets exceeds the configured shaping rate.

To check whether CoS is dropping packets, issue the following CLI commands:

```
admin@cli> show class-of-services interfaces brief
admin@cli> show class-of-services interfaces detail interface-name
admin@cli> show orgs org-services organization-name class-of-service qos-policies
admin@cli> show orgs org-services organization-name class-of-service app-qos-policies
```

For example:

```
admin@cli> show class-of-services interfaces brief
                                        TX
         TX       TX   TX                   BYTES     QUEUE
NAME     PACKETS  PPS  DROPPED  TX BYTES   TX BPS   DROPPED   LEN
---------------------------------------------------------------------
vni-0/0  130475   726  465      1183016979  5069504  562734    0
vni-0/1  0        0    0        0           0        0
```

```
admin@cli> show class-of-services interfaces detail vni-0/0

Intereface: vni-0/0
  Traffic Stats:
    TX Packets         : 133214
    TX PPS             : 253
    TX Packets Dropped : 465
    TX Bytes           : 119712785
    TX bps             : 834688
    TX Bytes Dropped   : 562734
  Port Stats:
       Traffic Class       TX Pkts     TX Dropped     TX Bytes    Bytes Dropped
     tc0 network-control        16289            0     11136743              0
     tc1   expedited-fwd            0            0            0              0
     tc2    assured-fwd          0          0        0            0
     tc3    best-effort       116925        465   108576042       562734
   Pipe Stat:
     Pipe ID    : 0
     Users      : [ vni-0/0.0 ]
       Traffic Class      TX Pkts    TX Dropped     TX Bytes    Bytes Dropped
     tc0 network-control        16289          0    11136743           0
     tc1   expedited-fwd          0          0        0            0
     tc2    assured-fwd          0          0        0            0
     tc3    best-effort       116925        465   108576042       562734
```

```
admin@cli> show orgs org-services Tenant-Common class-of-service qos-policies
                 QOS                      QOS      QOS      QOS     PPS     PPS       KBPS    KBPS
            QOS    DROP                  FORWARD   FORWARD   SESSION  POLICER  POLICER
POLICER  POLICER
            RULE   HIT    PACKET   QOS DROP   PACKET    BYTE      DENY     PKTS     BYTES       PKTS
BYTES
NAME             NAME   COUNT   COUNT    BYTE COUNT  COUNT     COUNT      COUNT    DROPPED
DROPPED     DROPPED  DROPPED
---------------------------------------------------------------------------------------------------------------------------------------
Default-Policy  VOICE  123281  5193940  6210916995  201902    319018559  0       5193940  6210916994
0       0
```

```
admin@cli> show orgs org-services Tenant-Common class-of-service app-qos-policies
                  APP     APP QOS  APP QOS   APP QOS   APP QOS
                  QOS      DROP     DROP      FORWARD   FORWARD
            RULE    HIT     PACKET   BYTE      PACKET    BYTE
NAME             NAME    COUNT   COUNT    COUNT     COUNT     COUNT
---------------------------------------------------------------------
Default-Policy  STREAM  30240   36371    50620084   998       1660980
```

To avoid packet drops correct the CoS shaper and rate limiter configurations. If you are running throughput tests in a lab environment, remove the CoS configuration and verify the throughput.

Check that Traffic Sessions Use All Worker Cores

For the best throughout from a VOS device, all the CPU cores must be used.

The VOS software allocates separate CPU cores for control daemons, worker threads, and poller threads. The CPUs assigned to control cores control daemons such as BGP and DHCPD. CPUs assigned for worker threads are responsible for things such as the forwarding plane, encryption, and decryption. CPUs assigned for the poller thread are responsible for reading packets from NICs and passing them to worker threads, and also and writing packets to NICs during transmission. The following figure illustrates the functioning of the CPU cores.

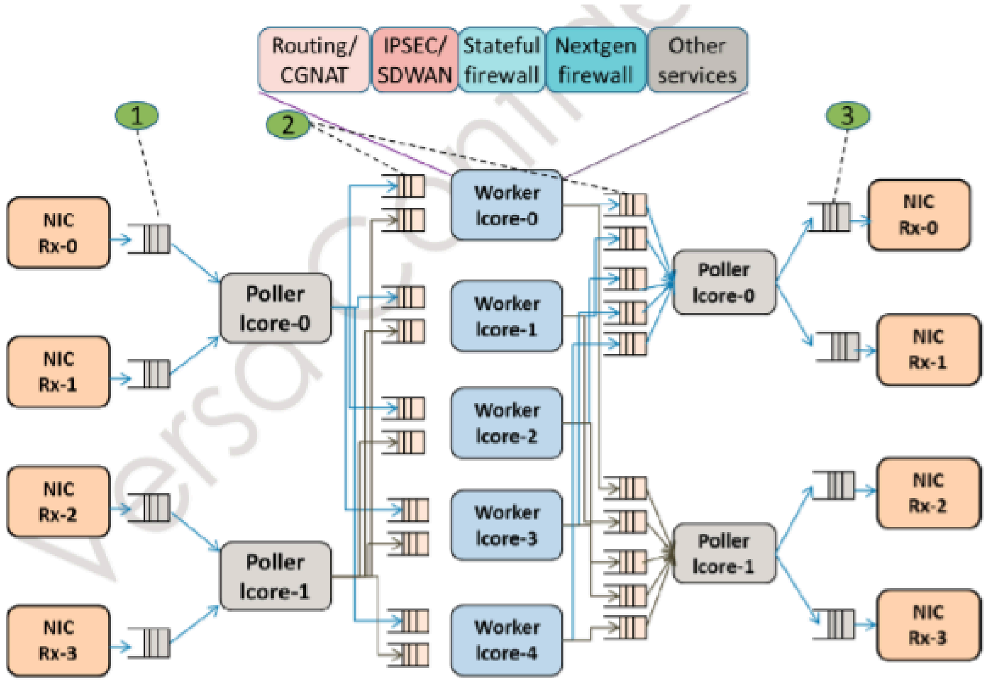

Figure 1 – FlexVNF High level Architecture

A session, as defined by 5-tuple consisting of a source IP address, a destination IP address, a source port number, a destination port number, and a protocol, is processed by a single worker CPU.

To check that all the worker CPUs are being used for sessions, send enough sessions so that at least few are processed by each worker CPU. It is recommended that you send traffic for at least 100 sessions while running throughput tests on eight core CPUs so that at least a few sessions are processed by each core. The following sample output shows that the usage each of the five core CPUs is approximately balanced, although CPU 4 is being used less than the others.

```
$ vsh connect vsmd
vsm-vcsn0> show vsf per-thread nfp stats summary
 Thr-Id    Sess-Active    Sess-Created    Sess-Closed
--------   -------------   --------------   -------------
   0          25        1397618       1396726
   1          18        1396258       1395344
   2          20        1394215       1393289
   3          15        1395297       1394376
   4          22        1266916       1266045
```

Check for Underlay Throughput Issues

Ensure that the underlay is not dropping the packets. For example, if a customer is trying to measure a 10-Gbps throughput but the underlay switches are not capable of switching at speeds of 10 Gbps, packets drop.

Check that the SLA is not experience a PDU loss of 100 percent. For information about PDUs, see Configure SLA Profiles for SD-WAN Traffic Steering.

Check the input rate (in pps and bps) and output rate (in pps/bps). To confirm, verify that packets transmitted out one side of a transport interface reach on other side of the transport.

To display TCP/IP and other packets being transmitted or received over a network, use the tcpdump utility on the WAN interface:

```
admin@cli> tcpdump vni-x/x filter remote-host
```

To check for drops on the circuit side, use the rapid ping utility on the VOS device, with a large count value, such as 1000:

```
admin@cli> ping ip-address rapid enable ?
Possible completions:
  count            - Number of pings to send
  df-bit           - Enable Do Not Fragment bit in IP header
  interface        - Source  interface from where to send the ping
  packet-size      - Packet size to send
  record-route     - Displays the route buffer on returned packets
  routing-instance - Routing instance
  source           - Source IP address
```

Then check the interface statistics to ensure that the TX pps and bps counts on the local site match the RX pps and bps counts on the remote site. For example:

```
admin@cli> show interfaces port statistics brief
             IF
        HOST  OPER   RX        RX               RX      RX    TX        TX              TX      TX    RX     TX
NAME     INF   STATUS PACKETS   PPS RX BYTES     ERRORS  BPS   PACKETS   PPP  TX BYTES
ERRORS  BPS   USAGE  USAGE
-------------------------------------------------------------------------------------------------------------------
vni-0/0  eth1  up     22578104  1   3663729635  0      1376  23241202  1    4048636473  0      1056  0.0    0.0
vni-0/1  eth2  up     13188574  1   890447986   0      2216  1514288   1    112008904   0      160   0.0    0.0
vni-0/2  eth3  up     8959110   1   646170340   0      1192  8092802   1    566530672   0      1352  0.0    0.0
vni-0/3  eth4  down   0         1   0           0      0     0         1    0           0      0     0.0    0.0
```

To run ping and tcpdump from the Director GUI, see Access Monitoring Tools.

Check whether Application Offload Is Enabled

First, if NGFW or UTM is not configured, check whether application offload is enabled:

```
admin@cli> show configuration orgs org-services organization-name application-identification
application-generic-options
```

Note that if NGFW or UTM is enabled, it is recommended that you disable application offload. If you enable it, with HTTP

Version 1.1 or later, different transactions of a connection may be identified as different applications. For example, if a Facebook session later reuses the same connection to exchange chat messages, it might be identified as Facebook Messenger instead of as Facebook.

If application offload is not enabled, enable it:

```
admin@cli> set orgs org-services tenant-id application-identification application-generic-options offload
enabled
```

Then, check whether isolcpu is enabled, to isolate the CPUs from the kernel scheduler. When you are doing performance throughput testing, if you wants to achieve close to no packet loss (that is, a packet loss of < 0.01%), it is recommended that you enable isolcpu.

Check whether isolcpu is enabled:

```
admin@cli> request system isolate-cpu status
status  isolcpu disabled
```

If isolcpu is not enabled, enable it:

```
admin@cli> request system isolate-cpu enable
status  GRUB PARAMETERS HAVE CHANGED. PLEASE REBOOT THE SYSTEM FOR VERSA-FLEXVNF
TO FUNCTION CORRECTLY.
admin@cli> request system isolcpu status
status  isolcpu enabled with num-control-cpus 1
```

Check that Sessions Are Load-Balanced on All Workers

To check that traffic sessions are load balanced equally across all the worker cores, issue the **vsh connect vsmd** and **show vsf per-thread nfp stats summary** commands. For more information, see Check that Traffic Sessions Use All Worker Cores, above.

If sessions are not load-balanced across worker threads, issue the following commands to check that class of traffic being received:

```
$ vsh connect vsmd
vsm-vcsn0> show vsm anchor core map
+--------+--------+---------+--------+---------+
|H-Index | NC Core| EF Core | AF Core| BE Core |
+--------+--------+---------+--------+---------+
|      0 |      0 |       1 |      1 |       0 |
|      1 |      0 |       2 |      2 |       1 |
|      2 |      0 |       3 |      3 |       2 |
|      3 |      0 |       4 |      4 |       3 |
|      4 |      0 |       5 |      5 |       4 |
|      5 |      0 |       1 |      1 |       5 |
+--------+--------+---------+--------+---------+
vsm-vcsn0> show vsm cq stats
+-------+--------+-----------+-------+-------+
| W TID | CTRL   | DATA      | EF    | AF    |
+-------+--------+-----------+-------+-------+
|     0 | 528669 | 356169364 |     0 |     0 |
|     1 |    199 | 330210649 |     0 |     0 |
|     2 |    160 | 339295575 |     0 |     0 |
|     3 |    200 | 337426918 |     0 |     0 |
|     4 |    189 | 313042396 |     0 |     0 |
|     5 |    157 | 301416739 |     0 |     0 |
+-------+--------+---------+---------+--------+
```

By default, the VOS software maps traffic for given class is mapped to worker cores. You can configure changes to these mappings.

If the sessions are not equally distributed across worker cores and throughput is less than expected, contact Versa Network Customer Support.

Check for Fragmented Packets

Check that there are not too many fragmented packets. Fragmentation and reassembly are CPU-intensive tasks, so throughput decreases if there are too many fragments.

A tunnel overhead is added to traffic transiting an SD-WAN tunnel. If larger packets are sent to the SD-WAN LAN network before they are sent to the WAN, the VOS device may fragment the packets before sending them over the SD-WAN tunnel. Fragmented packets are reassembled at the remote site before they are sent to the customer LAN.

The Director node adjusts the MSS for TCP packets transiting SD-WAN tunnels. If TCP MSS adjust is set for the tunnel, TCP packets are not fragmented. Instead, only larger UDP packets that may not fit into the SD-WAN tunnel are fragmented.

To check whether TCP MSS adjust is enabled, issue the following command:

```
admin@cli> show configuration system session tcp-adjust-mss
enable          true
interface-types all;
```

To check the number of packets that have been fragmented and reassembled, issue the following commands:

```
$ vsh connect vsmd
vsm-vcsn0> show vsm statistics datapath
# Packets Punt to WT                              : 63784
# Fragments Received for Reassembly                      : 47692
# Packets Reassembled                               : 23846
# Packets FDT Action Error                          : 22
# Pipeline Session Lookup - 2nd time local              : 1681
# Allowed - Filter Lookup                         : 333533
# Forward - NNon-local tunneled pkt, decaps not done   : 24
# Forwarded - Filter Lookup                         : 333533
# Forwarded - SFW No Match                        : 333533
# Sent - ARP to CT                             : 263
# Passed - Host-bound rate limit                   : 333533
# Injected - into VUNET                         : 46685
# Packets FDT Action Error                        : 22
# Packets Dropped - Interface disabled                : 3327
# Packets Dropped - Filter Lookup Module Action Denied : 46761
# Packets Dropped - Tunnel Decaps pkt processing error : 5988
```

For customer traffic whose DF bit is set, when the traffic arrives on a LAN or WAN network but fragmentation is needed to send it over an SD-WAN tunnel, the VOS device sends the ICMP error message “DF bit set but fragmentation needed” to the sender. Most network devices react to this message by sending future packets in which the DF bit is not set. However, some SIP phones and legacy devices, such as RADIUS servers, do not respond to this ICMP error message and continue to send packets with the DF bit set. As a result, these packets are dropped. To handle these situations, configure the **override-df-bit tunnels** option. Then, when traffic requires fragmentation but the DF bit is set and the sender does not respond to the ICMP error message, the VOS device clears the DF bit, fragments the packets, and sends them over the SD-WAN tunnels. At the other end of the tunnel, the fragments are reassembled and the DF bit is reset.

To check whether the **override-df-bit tunnels** option is set, issue the following command:

```
admin@cli> show configuration orgs org-services organization-name options override-df-bit
```

For example:

```
admin@cli> show configuration orgs org-services Tenant-Common options override-df-bit
options override-df-bit
override-df-bit tunnels;
```

To check packet fragmentation on the SD-WAN tunnel, issue the following commands:

```
$ vsh connect vsmd
vsm-vcsn0> show vsf tunnel stats
----------------------------------------------------------------
                   Tunnel encap stats
----------------------------------------------------------------
Tunnel Encap Processing successful:              246827441
Tunnel Encap Processing dropped:                      978
Tunnel IP-UDP transport encap forwarded:           246827441
Tunnel MPLSoGRE encap forwarded:                   246827441
Tunnel VXLAN-GPE encap forwarded:                  246827441
Tunnel IPsec-ESP encap forwarded:                234464119
Tunnel IPsec-ESP encap scheduled:                234464119
Tunnel Encap packt map not found, dropped:                 24
Tunnel Encap Pre-processing dropped:                    978
Tunnel Encap Pre-processing pre-fragmented:           14580958
Tunnel Encap Pre-processing Fragments:              29161916
Tunnel Encap Send completed:                   246827441
Tunnel Encap ether output completed:             246827441
Tunnel Encap Invalid Access circuit dropped:               3
Tunnel Overhead Calculation failed:                  951
Tunnel Pkts switched to valid AC:               112
Tunnel Pkts switched to mgmt tenant:            1304651
------------------------------------------------------------
```

Check for Packet Punting across Worker Threads

The traffic for a session is processed by a single worker core. To anchor a session to a worker core, a 5-tuple is used, consisting of a source IP address, a destination IP address, a source port number, a destination port number, and a protocol. All the traffic between the local site and a remote site travels over a single SD-WAN tunnel that has same 5-tuple for all customer sessions carried in the tunnel.

To anchor a session on a core, th worker thread must perform decapsulation on the tunnel, which is a CPU-intensive operation. To achieve load balancing among worker threads at the remote end, the local site sends to the remote site a CRC of the 5-tuple in the encapsulation header. The remote site then anchors the session based on the CRC. It is possible that some sessions may be anchored on an incorrect core and is then later punted to correct core. If a large number of packets are being punted or if the rate of punting is high, the throughput might decrease.

To check the number of packets being punted to a different worker thread (WT), issue the following commands:

```
$ vsh connect vsmd
vsm-vcsn0> show vsm statistics datapath
# Packets Punt to WT                                  : 1662602
# Fragments Received for Reassembly                   : 226354
# Packets Reassembled                                 : 113177
# Packets FDT Action Error                            : 49
# Pipeline Session Lookup - 2nd time punt             : 16
# Pipeline Session Lookup - 2nd time local            : 2259530
# Allowed - Filter Lookup                             : 6945526
# Forward - Non-local tunneled pkt, decaps not done   : 59
# Forwarded - Filter Lookup                           : 6945526
# Forwarded - SFW No Match                            : 81058422
# Sent - ARP to CT                                    : 1479
# Passed - Host-bound rate limit                      : 6945526
# Injected - into VUNET                               : 904280
# Packets FDT Action Error                            : 49
# Packets Dropped - Interface disabled                : 127521
# Packets Dropped - Interface disabled Reinject       : 27
# Packets Dropped - Tuple Extract Failure             : 22
# Packets Dropped - Filter Lookup Module Action Denied : 439633
# Packets Dropped - Tunnel Decaps pkt processing error : 68162
# Packets Dropped - Packet reinject ttl expired       : 142307
```

If you have enabled NAT, firewall and HA, it is expected that packets are punted between worker threads and so the output may report a large number of fragmented packets.

If packets are punted to different worker threads at high rate and throughput is less than expected, contact Versa Networks Customer Support.

Check the Poller Count

Typically, a VOS devices allocates one poller CPU for each 10 GB of a Tx/Rx link. For example, if there are 6x1G and 2x10G interfaces, the VOS device may assign three poller CPUs. The poller CPUs are assigned when Versa services come up during a boot, reboot, or restart. Even though some NICs may not be connected or used, poller CPUs are assigned based on number of NICs present in the VOS device.

To check number of poller CPUs assigned, issue the following commands:

```
$ vsh connect vsmd
vsm-vcsn0> show vsm cpu info
VSM CPU into:
-------------------------------------
# of CPUs              :  8
# of poller threads   :  1
# of worker threads   :  6
# of control threads  :  1
  Used CPUs           :  [ 0 1 2 3 4 5 6 7  ]
  Poller CPUs         :  [ 7  ]
  Worker CPUs         :  [ 1 2 3 4 5 6  ]
  Control CPUs        :  [ 0  ]
```

If some of NICs are not used, you can reduce the number of poller CPUs assigned to make more CPUs worker cores available. To change the number of poller CPUs, issue the following CLI command:

```
admin@cli> set system service-options poller-count number
```

Check Worker and Poller CPU Utilization and Drops

Check the CPU usage by the worker and poller CPU usage. If it is already running at 100 percent, the VOS device has reached its maximum throughput even if you have enabled all optimizations.

By default, VOS devices run in performance mode. However, if you change the run mode to hyper, the CPUs run at 100 percent even if there are no packets.

To check worker and poller CPU utilization, issue the following commands, pressing 1 to sort by process ID:

```
admin@vos$ htop
admin@vos$ top -H
```

To check for high memory usage, issue the following command:

```
admin@vos$ top -o %MEM
```

To check for high CPU usage, issue the following command:

```
admin@vos$ top -o %CPU
```

For example:

```
admin@vos$ htop

1  [||||||||||                                    ]  Tasks: 89, 69  thr; 3 running
2  [||||||||||||                                  ]  Load average:       0.68 0.72
3  [||||||||||||                                  ]  Uptime:  46 days,  23:41:30
4  [|||||||                                       ]
Mem[||||||||||||||||||||||||||||||||||||||||||||||||2713/3007MB]
Swp[                                                ]
```

```
admin@vos$ top -H
top - 14:29:09 up 46 days, 23:43,  2 users,  load average: 0.92, 0.77, 0.74
Threads: 259 total,  5 running, 254 sleeping,  0 stopped,  0 zombie
%Cpu(s): 14.8 us,  2.7 sy,  0.0 ni,  82.6 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0st
KiB Mem:   3080100 total,  2998636 used,    81464 free,    109412 buffers
KiB Swap:        0 total,         0 used,         0 free,    111256 cached Mem

  PID USER     PR  NI    VIRT    RES    SHR S %CPU %MEM      TIME+ COMMAND
20630 root     20   0 2609412 0.983g  8904 S 20.9 33.4  10374:38 worker-0
20631 root     20   0 2609412 0.983g  8904 S 20.6 33.4  10124:04 worker-1
20803 root     20   0 2609412 0.983g  8904 R 14.3 33.4   7090:53 ipsec-control
20632 root     20   0 2609412 0.983g  8904 S 10.0 33.4   4871:50 poller-0
19649 versa    20   0   50708  12248      0 S  2.0  0.4 668:51.32 versa-certd
19509 root     20   0 2609412 0.983g  8904 R  1.0 33.4 398:33.67 versa-vsmd
20798 root     20   0 2609412 0.983g  8904 R  1.0 33.4 501:34.86 ctrl-data-0
20779 root     20   0 2609412 0.983g  1524 R  0.7 33.4 306:05.67 vunet-timer
 2238 root     20   0   89320  15268  8904 S  0.3  0.5  40:09.24 vmtoolsd
19790 root     20   0  107656   2068  1384 S  0.3  0.1  35:50.74 monit
20507 root     20   0   39384   7724  1572 S  0.3  0.3 110:27.30 redis-server
    1 root     20   0   34096   3428  1436 S  0.0  0.1   0:07.61 init
    2 root     20   0       0      0      0 S  0.0  0.0   0:00.32 kthreadd
...
```

```
admin@vos$ top -o %MEM
top - 14:30:43 up 46 days, 23:44,  2 users,  load average: 0.58, 0.69, 0.72
Tasks: 187 total,  1 running, 186 sleeping,  0 stopped,  0 zombie
%Cpu(s): 13.1 us,  2.8 sy,  0.0 ni,  84.1 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0st
KiB Mem:   3080100 total,  2995984 used,    84116 free,    109500 buffers
KiB Swap:        0 total,         0 used,         0 free,    111272 cached Mem

  PID USER     PR  NI    VIRT    RES    SHR S %CPU %MEM      TIME+ COMMAND
19509 root     20   0 2609412 0.983g  8904 S 69.9 33.4  33669:05 versa-vsmd
19509 root     20   0 2609412 0.983g  8904 S 69.9 33.4  59:27.61 confd
19509 versa    20   0 2609412 0.983g  8904 S 69.9 33.4  11:57.38 versa-vmod
19509 versa    20   0 2609412 0.983g  8904 S 69.9 33.4 102:04.73 versa-acctmgrd
19509 versa    20   0 2609412 0.983g  8904 S 69.9 33.4  29:07.86 versa-rtd
19509 versa    20   0 2609412 0.983g  8904 S 69.9 33.4  0:00.02 versa-fltrmgr
19509 versa    20   0 2609412 0.983g  8904 S 69.9 33.4   11:47.17 versa-dhcpd
19509 versa    20   0 2609412 0.983g  8904 S 69.9 33.4   0:00.74 nodejs
...
```

```
admin@vos$ top -o %CPU
top - 02:34:38 up 195 days, 6:09,  r users,  load average: 3.22, 3.28, 3.32
```

```
Tasks: 208 total,  1 running, 207 sleeping,  0 stopped,  0 zombie
%Cpu(s): 0.4 us,  0.2 sy,  0.0 ni,  52.3 id,  0.0 wa,  0.0 hi,  0.0 si,  47.1 st
KiB Mem:  4045984 total,  3904600 used,   141384 free,   34712 buffers
Ki Swap:        0 total,        0 used,        0 free,   76992 cached Mem

  PID USER     PR  NI    VIRT    RES   SHR S %CPU %MEM      TIME+ COMMAND
31529 root     20   0 3649752 1.969g  6288 S 66.8 51.0 174511:005 versa-vsmd
31671 versa    20   0   85912  40488  2048 S  1.7  1.0    1740:15 versa-acctmgrd
17560 admin    20   0   25208   2924  2368 R  1.3  0.1   00:00.20 top
32555 versa    20   0   47584  13796  1840 S  1.3  0.3    1738:28 redis-server
    7 root     20   0       0      0     0 S  1.0  0.0    4460:37 rcu_sched
   13 root     20   0       0      0     0 S  0.7  0.0  873:23.18 ksoftirqd/1
31693 versa    20   0   50708  13620  1364 S  0.7  0.3    4102:51 versa-certd
31655 versa    20   0  149312  36492  3592 S  0.3  0.9  205:57.88 versa-rtd
    1 root     20   0   33920   3392  1580 S  0.0  0.1    1:01.28 init
    2 root     20   0       0      0     0 S  0.0  0.0    0:18.98 kthreadd
    3 root     20   0       0      0     0 S  0.0  0.0  568:49.96 ksoftirqd/0
    5 root      0 -20       0      0     0 S  0.0  0.0    0:00.00 kworker/0:0H
    8 root     20   0       0      0     0 S  0.0  0.0    0:00.00 rcu_bh
    9 root     rt   0       0      0     0 S  0.0  0.0  198:10.12 migration/0
   10 root     rt   0       0      0     0 S  0.0  0.0    1:49.48 watchdog/0
   11 root     rt   0       0      0     0 S  0.0  0.0    1:33.12 watchdog/1
```

```
admin@vos$ top -H
Threads: 283 total,  9 running, 273 sleeping,  0 stopped,  1 zombie
%Cpu(s): 18.3 us,  4.1 sy,  0.0 ni,  77.2 id,  0.1 wa,  0.0 hi,  0.4 si,  0.0st
KiB Mem:  16405444 total, 10084552 used,  6320892 free,   257940 buffers
KiB Swap: 16748540 total,        0 used, 16748540 free,  1741824 cached Mem

  PID USER     PR  NI    VIRT    RES   SHR S %CPU %MEM     TIME+ COMMAND
29813 root     20   0 7949212 2.889g 85732 R 98.3 18.5 114:43.34 worker-1
29815 root     20   0 7949212 2.889g 85732 R 98.3 18.5 112:21.57 worker-3
29816 root     20   0 7949212 2.889g 85732 R 98.3 18.5 108:11.48 worker-4
29817 root     20   0 7949212 2.889g 85732 R 98.3 18.5 119:33.45 worker-5
29812 root     20   0 7949212 2.889g 85732 R 98.3 18.5 125:59.33 worker-0
29814 root     20   0 7949212 2.889g 85732 R 98.3 18.5 105:43.43 worker-2
29818 root     20   0 7949212 2.889g 85732 R 17.3 18.5 112:34.01 poller-0
```

The following flags are present in the output of the **top –H** command:

- us—User. Time running un-niced user processes.
- sy—System. Time running kernel processes.
- ni—Nice. Time running niced user processes.
- wa—IO-wait. Time waiting for I/O completion.
- hi—Time spent servicing hardware interrupts.
- si—Time spent servicing software interrupts.
- st—Time stolen from this VM by the hypervisor. If KVM or Hypervisor is oversubscribed or has high CPU usage, this number is high.

If the worker and poller are running at 100 percent, packet drops may occur at worker and poller. To check the worker and poller drops, issue the following commands:

```
$ vsh connect vsmd
vsm-vcsn0> show vsm statistics dropped
DPDK ERROR STATISTICS
~~~~~~~~~~~~~~~~~~~~~~~~~~~~
DATAPATH ERROR STATISTICS
~~~~~~~~~~~~~~~~~~~~~~~~~~~~
# Packets FDT Action Error                : 22
# Packets Dropped - Interface disabled     : 3382
# Packets Dropped - Stale Fragment Entry   : 1634
# Packets Dropped - Filter Lookup Module Action Denied : 47571
# Packets Dropped - Tunnel Decaps pkt processing error : 5988

THRM ERROR STATISTICS
~~~~~~~~~~~~~~~~~~~~~~~
POLLER PID : 29818
# Drop Packets RX                        : 12694125
# Drop Packets TX                        : 40

NFP ERROR STATISTICS
~~~~~~~~~~~~~~~~~~~~~~
# Packets Dropped - Invalid session handle : 1
# Number of calls to icmp_error            : 294

VSF ERROR STATISTICS
~~~~~~~~~~~~~~~~~~~~~~
# Sess Create Denied (mbuf sanity fail)    : 58
# Route lookup failure (ip-out)            : 24

VUNET ERROR STATISTICS
~~~~~~~~~~~~~~~~~~~~~~~~
# VN_MOD_IP_ERR_NO_ROUTE_CNT               : 266
# VN_MOD_ETH_ERR_BAD_TYPE_CNT              : 6764

COS DROPS
~~~~~~~~~~~~~
# Shaper drops                             : 0
```

For more details about where the packets are dropped, issue the following command:

```
admin@vos$ show vsm statistics thrm detail
```

If the worker and poller are not running at 100 percent, if you are still seeing packet drops in the poller or worker, if you have performed all the checks above, and if the throughput is less than expected, contact Versa Networks Customer Support to debug further.

Verify Link Bandwidth

To verify link bandwidth, run the automatic bandwidth test. For more information, see Troubleshoot Link Bandwidth Issues.

Supported Software Information

Releases 20.2 and later support all content described in this article.

Additional Information

Troubleshoot Link Bandwidth Issues