Troubleshooting Guide

VeloCloud SD-WAN

Version 6.0

Arista.com

Headquarters	Support	Sales
5453 Great America Parkway Santa Clara, CA 95054 USA +1-408-547-5500	+1-408-547-5502 +1-866-476-0000	+1-408-547-5501 +1-866-497-0000
www.arista.com/en/	support@arista.com	sales@arista.com

Contents

Chapter 1

VeloCloud SD-WAN Troubleshooting Guide

The Arista VeloCloud SD-WAN™ Troubleshooting Guide provides details about the logging capabilities available in Orchestrator. It also provides details about the most common issues and troubleshooting information for the VeloCloud SD-WAN product.

Chapter 2

Intended Audience

The Arista VeloCloud SD-WAN Troubleshooting Guide is intended for network administrators, network analysts, and IT administrators responsible for deploying, monitoring, and managing Enterprise branch network.

Chapter 3

About Remote Diagnostics

VeloCloud Orchestrator supports bi-directional communication with the VeloCloud Edge by using WebSockets. WebSocket is a full-duplex communication protocol over a single TCP connection. WebSockets easily support communication between a Web browser (or other client applications) and a Web server with much lower overhead than HTTP polling. Remote Diagnostics uses a bi-directional WebSocket connection instead of the live-mode heartbeat mechanism to improve the responsiveness of the Remote Diagnostics in the VeloCloud Orchestrator.

The WebSocket communication involves the following two WebSocket connections for passing WebSocket messages from a Web browser to a VeloCloud Edge and vice versa:

- A WebSocket connection between a Web browser (Orchestrator UI portal) and an Orchestrator. This connection is responsible for all communications with the Web browser and for setting up the system properties needed for establishing a WebSocket connection.
- Another WebSocket connection between an Orchestrator and an Edge. This connection is persistent and setup on Edge activation for processing heartbeats from the Edge and sending back responses to the Orchestrator.

While establishing WebSocket connections between a Web browser and an Edge, in order to ensure Web security against Distributed Denial-of-Service (DDoS) and Cross site request forgery (CSRF) attacks, the browser origin address that is used to access the Orchestrator UI is validated for incoming requests.

In most Orchestrators, the browser origin address/DNS hostname is the same as the value of the `network.public.address` system property. To support scenarios where the address used to access the Orchestrator UI from the browser is different from the value of the `network.public.address` system property, the following system properties are added newly for WebSocket connections:

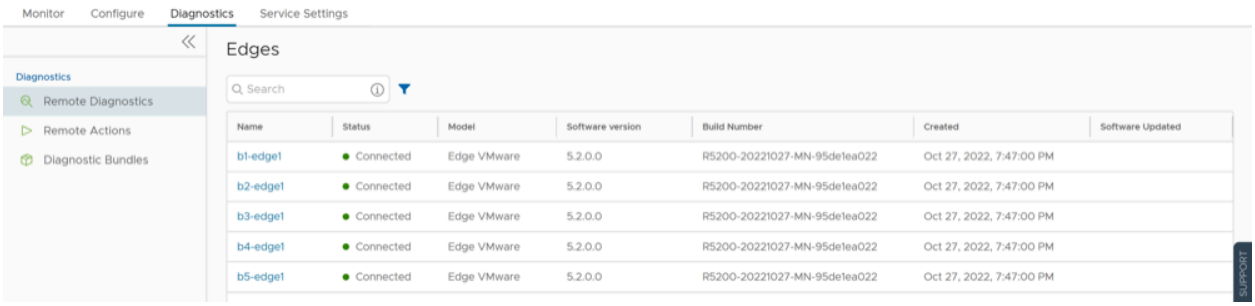
- `network.portal.websocket.address`- Allows to set an alternate address/ DNS hostname to access the UI from a browser if the browser address is not the same as the value of `network.public.address` system property. By default, the `network.portal.websocket.address` system property is not set.
- `session.options.websocket.portal.idle.timeout`- Allows to set the total amount of time (in seconds) the browser WebSocket connection is active in an idle state. By default, the browser WebSocket connection is active for 300 seconds in an idle state.

Run Remote Diagnostic Tests on Edges

VeloCloud Orchestrator allows you to run various Remote Diagnostic tests on a selected Edge.

To run Remote Diagnostics on an Edge:

1. In the Orchestrator UI, select the **Diagnostics** tab.
   The **Remote Diagnostics** page displays the existing Edges.
   Figure 4-1: Remote Diagnostics



2. Select the link to an Edge.

> **Note:** You can also run the Remote Diagnostics test using the **Shortcuts** option available in the **Configure** > **Edges** or **Monitor** > **Edges** pages.

   A connection is established to the Edge and the **Remote Diagnostics** window displays all the possible Remote Diagnostics tests than you can run on the Edge.
3. Choose an appropriate Remote Diagnostics test to run on the Edge and select **Run**.
   The diagnostic information is fetched from the Edge and displayed in the screen.

Chapter 5

Remote Diagnostic Tests on Edges

VeloCloud Orchestrator allows you to run various remote diagnostics test on a selected Edge. The remote diagnostic information contains Edge-specific logs for analysis.

The following sections describe the possible remote diagnostics tests that you can run on an Edge:

Network Reachability

- ARP Table Dump
- Clear ARP Cache
- Ping IPv6 Test
- Ping Test
- Traceroute

DNS and DHCP

- DNS Test
- DNS/DHCP Service Restart

Interfaces

- EVDSL Modem Status
- Interface Status
- LTE Modem Information

Firewall

- Flush Firewall Sessions
- List Active Firewall Sessions
- Get IP Threat Reputation Test
- Get URL Category and Reputation Test

IPv6 Settings

- IPv6 Clear ND Cache
- IPv6 ND Table Dump
- IPv6 Route Table Dump

Flow and NAT Table

- Flush Flows
- Flush NAT
- NAT Table Dump
- List Active Flows

BFD Sessions

- Show BFD/BFDv6 Peer Status
- Show BFD/BFDv6 Peer Counters
- Show BFD Settings
- Show BFDv6 Settings

BGP Sessions

- List BGP Redistributed Routes
- List BGP Routes
- List Routes per Prefix
- Show BGP Neighbor Advertised Routes
- Show BGP Neighbor Learned Routes
- Show BGP Neighbor Received Routes
- Show BGP Neighbor Details
- Show BGP Routes per Prefix
- Show BGP Summary
- Show BGP Table
- Show BGPv6 Neighbor Advertised Routes
- Show BGPv6 Neighbor Learned Routes
- Show BGPv6 Neighbor Received Routes
- Show BGPv6 Neighbor Details
- Show BGPv6 Routes per Prefix
- Show BGPv6 Summary
- Show BGPv6 Table

OSPF Areas

- List OSPF Redistributed Routes
- List OSPF Routes
- Show OSPF Database

- Show OSPF Database for E1 Self-Originate Routes
- Show OSPF Neighbors
- Show OSPF Route Table
- Show OSPF Setting

OSPFv3 Areas

- List OSPFv3 Redistributed Routes
- List OSPFv3 Routes
- Show OSPFv3 Database
- Show OSPFv3 Database for E1 Self-Originate Routes
- Show OSPFv3 Neighbors
- Show OSPFv3 Route Table
- Show OSPFv3 Setting

Miscellaneous

- Dump Context Logging Information
- Enable or Disable Context Logging
- Gateway
- HA Info
- List Clients
- List Paths
- MIBs for Edge
- NTP Dump
- Reset USB Modem
- System Information
- Route Table Dump
- VPN Test
- WAN Link Bandwidth Test

5.1 ARP Table Dump

What is the Purpose of This Test

Run this test to:

- Verify the IP address to MAC address binding on the LAN/WAN interfaces that are connected to the Edge on which the test is run.
- View the status of the ARP cache.

When Can You Run This Test

The following are the scenarios when you can run this test:

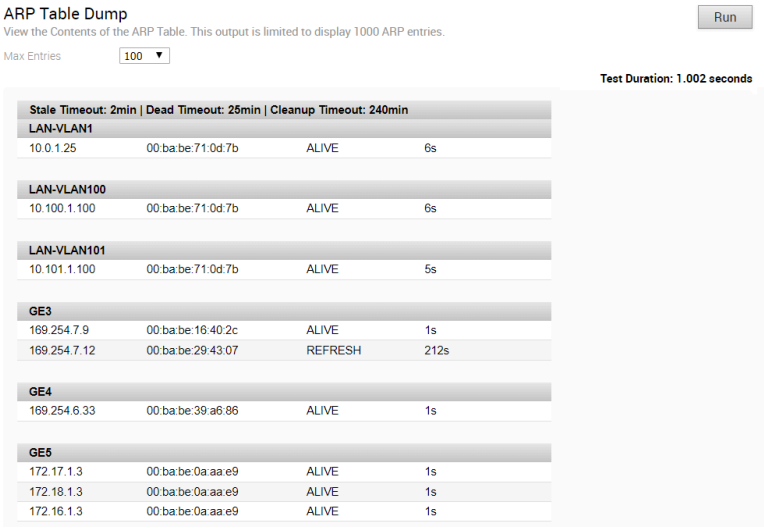
- Clients in the same broadcast domain are not reachable.
- Interface device not reachable.
- Next hop is not reachable.

What to Check in the Test Output

Run the **ARP Table Dump** test on the required Edge. For instructions, see Remote Diagnostic Tests on Edges.

Following is an example of the test output:

Figure 5-1: ARP Table Dump Sample Output



The output is limited to displaying 1000 ARP entries. Following are the possible ARP cache statuses:

- Alive—Interface is reachable.
- Dead—Interface is not reachable.
- Refresh—Interface is trying to relearn the ARP.

To identify the cause for the device reachability issue:

1. In the output of the **ARP Table Dump** test, verify the ARP cache status of the interface. If the status is of any value other than **Alive**, run the **Clear ARP Cache** test to clear the cache of the previously stored ARP value.

> **Note:** You can clear the ARP cache for only one interface at a time.

2. Run the **ARP Table Dump** test again to verify the IP address and the MAC address mapping on the LAN/WAN interface.
3. If the problem persists, collect Diagnostic and Packet Capture bundles and contact Arista Customer Support. See Diagnostic Bundles for Edges.

5.2 Clear ARP Cache

What is the Purpose of This Test

Clears specific interface-learned ARP entries from the ARP table.

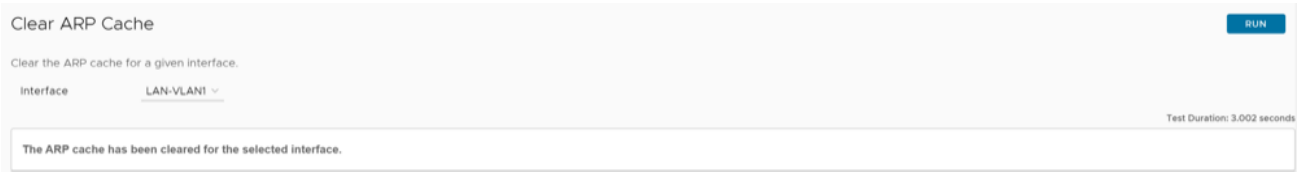
When Can You Run This Test

This test is run when there are stale entries in the ARP table dump test. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Clear ARP Cache** test on the required Edge. Following is an example of the test output:

Figure 5-2: Clear ARP Cache Sample Output



5.3 Ping IPv6 Test

What is the Purpose of This Test

Verifies if the next hop or an IPv6 address is reachable from the interface.

When Can You Run This Test

Run this test to check the following:

- IPv6 reachability
- Packet Loss
- RTT

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Ping IPv6** test on the required Edge. Check if the output displays **Reachable**. For a successful PING test, packet transmitted must be equal to packet received. If not, capture the Interface and Destination IP addresses to see where the packet is getting dropped. Following is an example of the test output:

Figure 5-3: Ping IPv6 Test Sample Output

5.4 Ping Test

What is the Purpose of This Test

Run this test if the next hop or an IPv4 address is reachable from the interface.

When Can You Run This Test

Run this test to check the following:

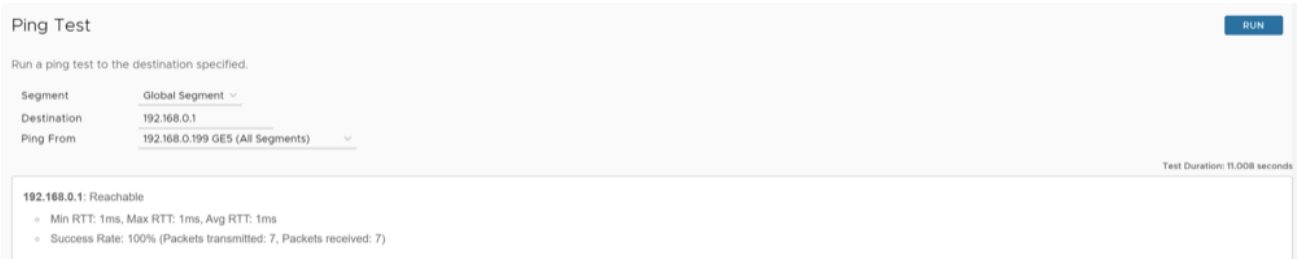
- IPv4 reachability
- Packet Loss
- RTT

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Ping** test on the required Edge. Check if the output displays `Reachable`. For a successful PING test, packet transmitted must be equal to packet received. If not, capture the Interface and Destination IP addresses to see where the packet is getting dropped. Following is an example of the test output:

Figure 5-4: Ping Test Sample Output



5.5 Traceroute

What is the Purpose of This Test

- Displays path and nodes to destination from Edge interface.
- Measures transit delays of packets at each hop across an Internet Protocol (IP) network.

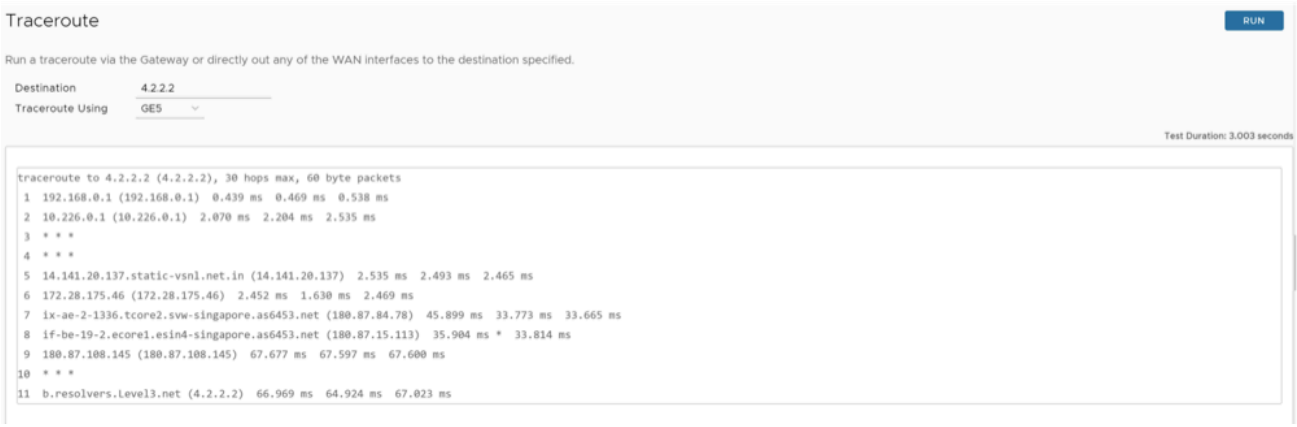
When Can You Run This Test

Run this test in parallel with the PING test to understand routing or network reachability issues. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Traceroute** test on the required Edge. Following is an example of the test output:

Figure 5-5: Traceroute Sample Output



- Verify if the Destination IP address is seen as last hop.
- To measure transit delay, check the time (ms) which is displayed next to the IP address.
- If the Destination IP address is not seen, check the last hop seen for route reachability to both source and destination to resolve the routing issue.

5.6 DNS Test

What is the Purpose of This Test

Run this test to confirm whether the Edge can resolve the Domain name to IP address.

When Can You Run This Test

The following are the scenarios when you can run this test:

- Domain name is not resolved.

- Name lookup error occurs.
- Name lookup error logs are seen.

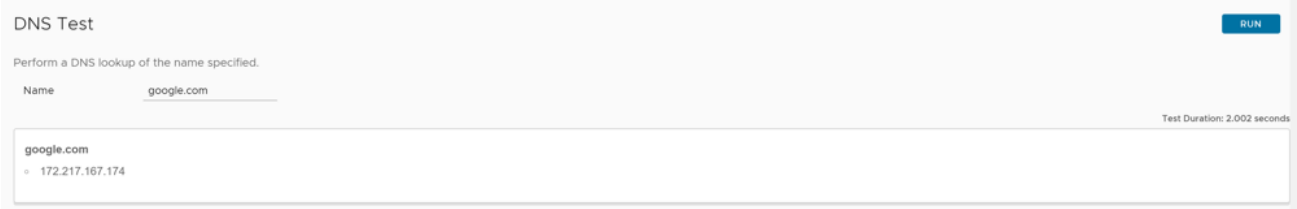
> **Note:** This test is specific to Edge DNS, not for clients located in the LAN side.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **DNS Test** on the required Edge. Following is an example of the test output:

Figure 5-6: DNS Test Sample Output



Verify that the name provided is resolved to an IP address. If it is not resolved, kindly check the DNS setting under the **Device** tab of Edge configuration, and then try restarting the DNS server or check if DNS Server is reachable or not.

5.7 DNS/DHCP Service Restart

What is the Purpose of This Test

Run this test to restart both DNS and DHCP processes in the Edge.

When Can You Run This Test

The following are the scenarios when you can run this test:

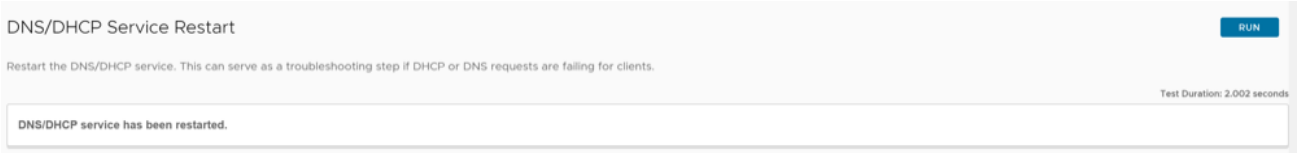
- The Edge is acting as DHCP and DNS server to LAN network devices.
- The DNS and DHCP servers fail.
- DNS and DHCP processes get stalled.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **DNS/DHCP Service Restart** test on the required Edge. Following is an example of the test output:

Figure 5-7: DNS/DHCP Service Restart Sample Output



Note: Any output other than the above indicates issues with either DNS/DHCP processes or Orchestrator API call issue. Kindly contact the support team.

5.8 EVDSL Modem Status

What is the Purpose of This Test

Run this test to check the status of the ADSL (Asymmetric Digital Subscriber Line) or VDSL (Very-high-bit-rate Digital Subscriber Line) link.

When Can You Run This Test

Run this test to check the following details of the ADSL/VDSL link:

- Mode
- Uptime
- Peer MAC Address
- Status
- Link rate

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **EVDSL Modem Status** test on the required Edge. Following is an example of the test output:

Figure 5-8: EVDSL Modem Status Sample Output

Table 1: EVDSL Modem Status- Options and Descriptions

Option	Description
Name	The SFP interface name where the link is connected to.
Mode	The mode in which the interface is operating. If the interface is used for DSL connection, then the mode will be listed as **DSL**.
Vendor MAC	The mac address of the DSL peer device.
xDSL Mode	VDSL or ADSL.
Link Time	Uptime of the link.
Status	**Showtime** indicates that the line is in sync.

5.9 Get IP Threat Reputation Test

What is the Purpose of This Test

Run this test on the required Edge by providing the IP address to view the threat category of the given IP.

When Can You Run This Test

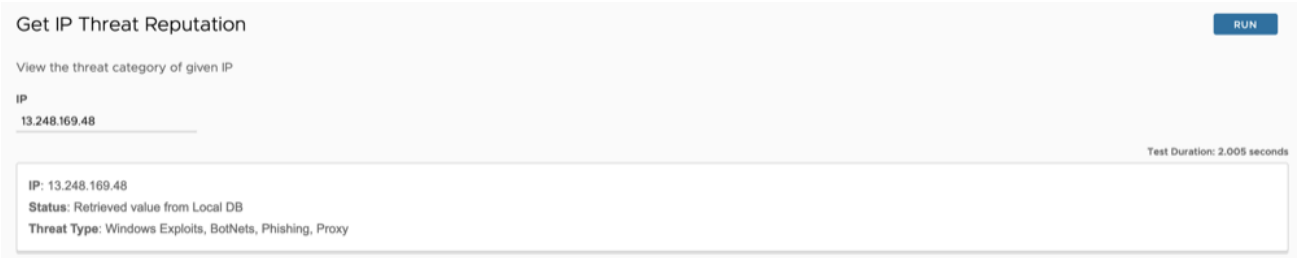
You can run this test when IP Threats are detected and to obtain the threat category of the given IP.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Get IP Threat Reputation** on the required Edge. Following is an example of the test output:

Figure 5-9: Get IP Threat Reputation Test Sample Output



5.10 Get URL Category and Reputation Test

What is the Purpose of This Test

Run this test on the required Edge by providing the URL to view the category and reputation score of a given URL.

When Can You Run This Test

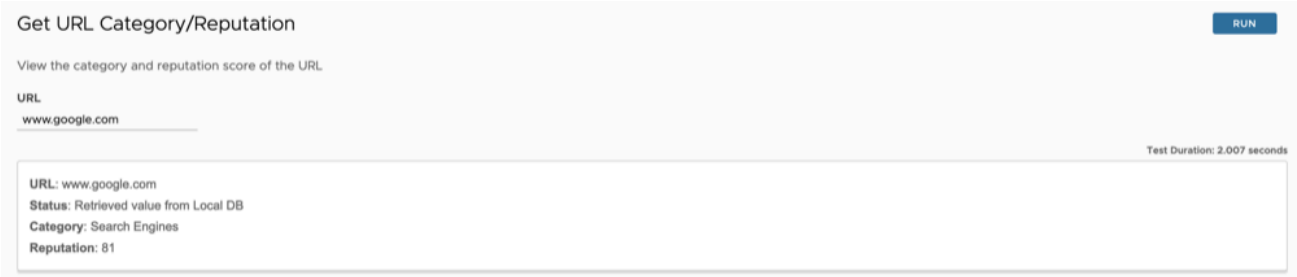
You can run this test to check the category and reputation score of a threat URL.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Get URL Category and Reputation** on the required Edge. Following is an example of the test output:

Figure 5-10: Get URL Category and Reputation Test Sample Output



5.11 Interface Status

What is the Purpose of This Test

Run this test to:

- Verify the configuration and status of the interface.
- Check packet loss situation.

When Can You Run This Test

Run this test to check the following:

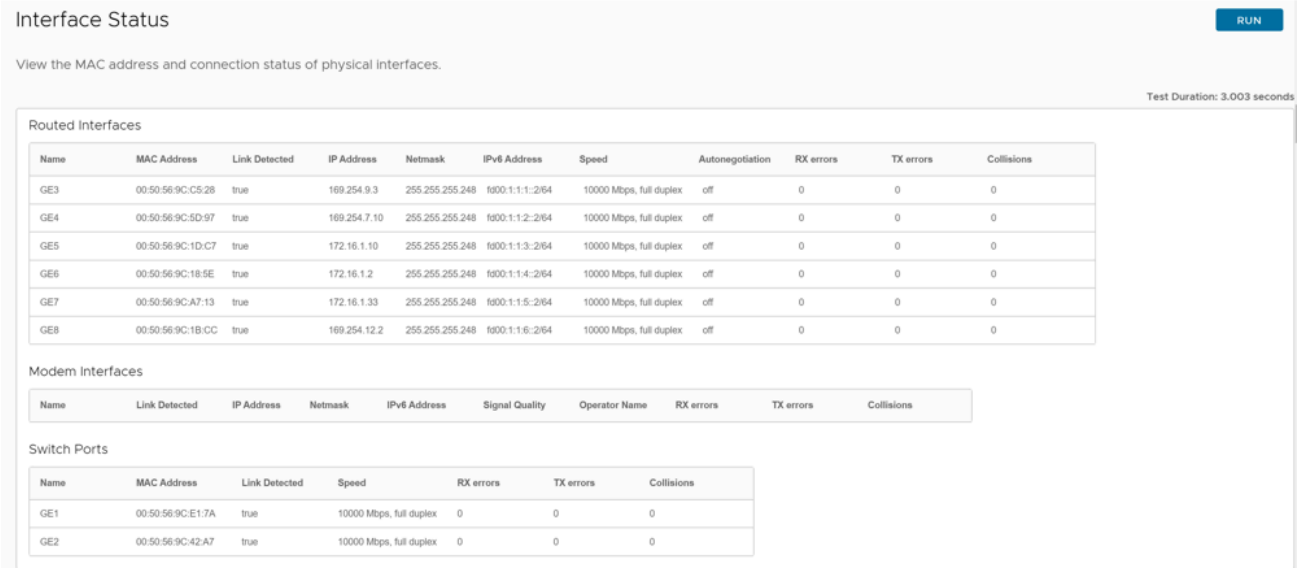
- The link detected must be true. If it is false, then you must check the physical cabling and peer end configuration.
- Ideally RX and TX should be "0". If the RX and TX errors are incrementing, then try replacing the cables/ SFP's. The values do not increase when this test is run twice.
- Negotiation parameter for interface issue.
- For Modem interface, verify if the signal quality strength is above 80% for a good wireless WAN link.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Interface Status** test on the required Edge. Following is an example of the test output:

Figure 5-11: Interface Status Sample Output



5.12 LTE Modem Information

What is the Purpose of This Test

Run this test to get details about the following:

- Modem
- Connection
- Location
- Signal Strength
- Status of the LTE

When Can You Run This Test

The following are the scenarios when you can run this test:

- Issues with LTE disconnections.
- Failure to understand the current state of the LTE.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **LTE Modem Information** test on the required Edge. Following are the examples of the test outputs:

- **Modem Information**

  Figure 5-12: Modem Information Sample Output

Modem Information


```
{
    "Manufacturer": "Sierra Wireless, Incorporated",
    "Model": "EM7455",
    "Modem identifier": "359073060235219",
    "Firmware Revision": "SWI9X30C_02.33.03.00 r8209 CARMD-EV-FRMWR2 2019/08/28 20 59 30",
    "Hardware Revision": "1.0",
    "Supported capabilites": "gsm-umts, lte",
    "Current capabilites": "gsm-umts, lte",
    "own number": "NA",
    "state": "connected",
    "Failed reason": "--",
    "Power state": "on",
    "current modes": "allowed 3g, 4g; preferred 4g",
    "imei": "359073060235219",
    "operator code": "311480",
    "operator name": "VZW",
    "registration state": "home",
    "signal quality(%)": "100"
}
```


  Verify the following in the output:

  - Operator name must be correct.
  - Signal quality must be above 70 % for better reception.
  - Registration state must display home for better speed.
  - LTE must be in a connected state. If disconnected, check for the failure reason.

- **Connection Information**

  Figure 5-13: Connection Information Sample Output

Connection Information


```
{
    "Bearer": "Available",
    "Connected": "yes",
    "Suspended": "no",
    "Interface": "wwan0",
    "APN": "",
    "IP type": "ipv4v6",
    "user": "--",
    "password": "NA",
    "IP method": "static",
    "IP address": "100.88.59.190",
    "Gateway": "100.88.59.189",
    "DNS": "198.224.173.135",
    "MTU": "1428",
    "Stats Duration": "45390",
    "Rx bytes": "6636796",
    "Tx bytes": "34505527"
}
```


  Verify the following in the output:

  - Rx and Tx are incrementing as expected.
  - IPv4 and IPv6 address and interface must be assigned with WAN information.

- Connected state must be "yes".
- If Suspended state is "yes", try resetting the LTE Modem.

- **Location Information**

  Figure 5-14: Location Information Sample Output

```
Location Information

{
    "Operator code": "311",
    "Operator name": "480",
    "Location area code": "FFFE",
    "tracking area code": "1F07",
    "cell id": "00792D02"
}
```

- **Signal Information**

  Figure 5-15: Signal Information Sample Output

```
Signal Information

{
    "Serving": {
        "EARFCN": "5230",
        "MCC": "311",
        "MNC": "480",
        "TAC": "7943",
        "CID": "00792D02",
        "Bd": "13",
        "D": "3",
        "U": "3",
        "SNR": "22",
        "PCI": "433",
        "RSRQ": "-6.2",
        "RSRP": "-71.4",
        "RSSI": "-48.7",
        "RXLV": "--"
    },
    "IntraFreq": {
        "PCI": "433",
        "RSRQ": "-6.2",
        "RSRP": "-71.4",
        "RSSI": "-48.7",
        "RXLV": "--"
    },
    "InterFreq": {
        "EARFCN": "2075",
        "ThresholdLow": "0",
        "ThresholdHi": "0",
        "Priority": "0",
        "PCI": "433",
        "RSRQ": "-18.7",
        "RSRP": "-128.1",
        "RSSI": "-101.6",
        "RXLV": "0"
    }
}
```

Verify whether the mode is online, and then check for "last error code", which indicates that the LTE is in failed state.

- **Status Information**

  Figure 5-16: Status Information Sample Output

```
Status Information

response: '!GSTATUS:
Current Time:  46460            Temperature: 55
Reset Counter: 1                Mode:        ONLINE
System mode:   LTE              PS state:    Attached
LTE band:      B13              LTE bw:      10 MHz
LTE Rx chan:   5230             LTE Tx chan: 23230
LTE CA state:  INACTIVE                 LTE Scell band:B4
LTE Scell bw:15 MHz             LTE Scell chan:2075
EMM state:     Registered       Normal Service
RRC state:     RRC Connected
IMS reg state: Full Srv         IMS mode:    Normal

PCC RxM RSSI:  -53              RSRP (dBm):  -71
PCC RxD RSSI:  -54              RSRP (dBm):  -72
SCC RxM RSSI:  -101             RSRP (dBm):  -130
SCC RxD RSSI:  -101             RSRP (dBm):  -132
Tx Power:      --               TAC:         1F07 (7943)
RSRQ (dB):     -6.1             Cell ID:     00792D02 (7941378)
SINR (dB):     20.4'
```

- **Debug Information**

  Figure 5-17: Debug Information Sample Output

```
Debug Information

{
    "response": "'LTE Engineering",
    "IMSI": "311 480 753787712",
    "State": "Connected",
    "PLMN ID": "311 480",
    "PCI": "433",
    "Band": "13",
    "UL Channel": "23230",
    "DL Channel": "5230",
    "RSRP": "-71.4 dBm",
    "RSRQ": "-6.0 dB",
    "RS-SINR": "21.8 dB",
    "Tx Pwr": "0 dBm",
    "IP - internet": "IPv4 100.88.59.190 IPv6 2600 1010 B010 35B5 0 13 30A0 4A01",
    "APN": "Internet VZWINTERNET OTA vzwadmin",
    "Technology": "LTE",
    "1x Diversity": "NA",
    "QLIC": "NA",
    "PRL": "NA",
    "Chipset": "MDM9x35",
    "AMSS Version": "MPSS.BO.2.5.1.c1-00168-M9635TAAAANAZM-1",
    "Device Version": "SWI9X30C_02.33.03.00 r8209 CARMD-EV-FRMWR2 2019/08/28 20 59 30",
    "Hardware Version": "40",
    "IP Address": "EVDO NA 1x NA",
    "Last Error code": "NA'"
}
```

5.13 Flush Firewall Sessions

What is the Purpose of This Test

This is similar to the Flush Flows options, but is specifically for the Stateful Firewall. This causes the Edge to not just flush the sessions, but actively send a TCP RST for the TCP-based sessions.

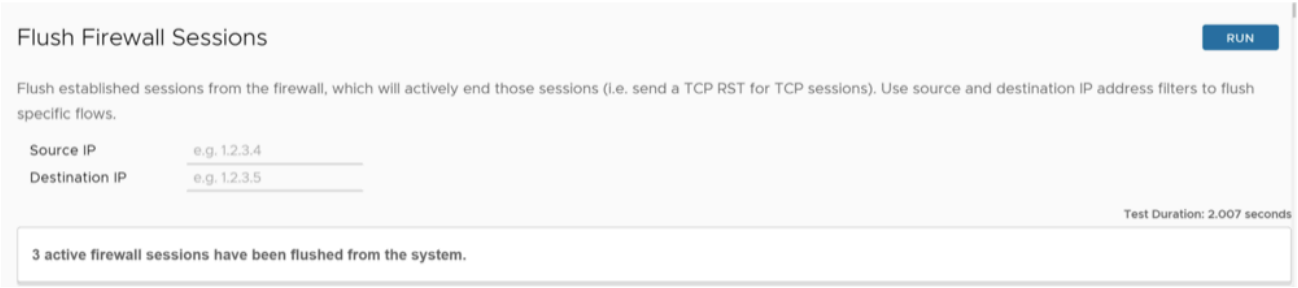
When Can You Run This Test

Run the **Flush Firewall Sessions** test on the required Edge by providing the Source and Destination IP addresses to flush the active firewalls session which needs to be reset. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-18: Flush Firewall Sessions



5.14 List Active Firewall Sessions

What is the Purpose of This Test

This allows you to see the current state of the active firewall sessions (up to a maximum of 1000 sessions). You can filter by Source and Destination IP and Port as well as Segment to limit the number of sessions returned.

Note: IPv6 firewall session information can be viewable from the New Orchestrator UI. To view IPv6 firewall session information, you must run the **List Active Firewall Sessions** test from the New Orchestrator UI.

When Can You Run This Test

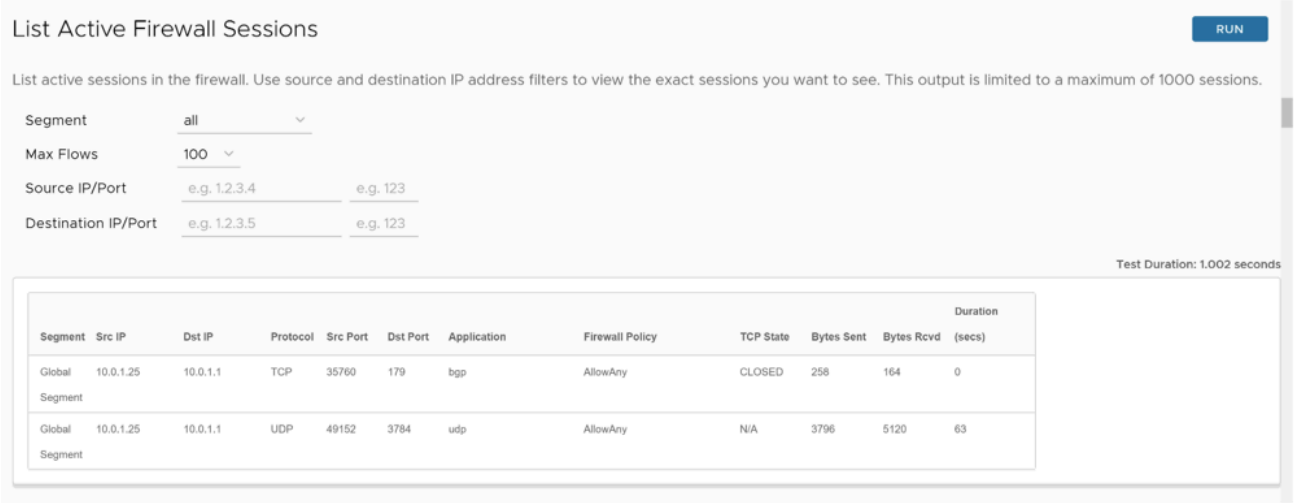
To verify if the session is allowed or blocked. If it is allowed, it would be seen in the output. Also, you can see the current state of the session.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

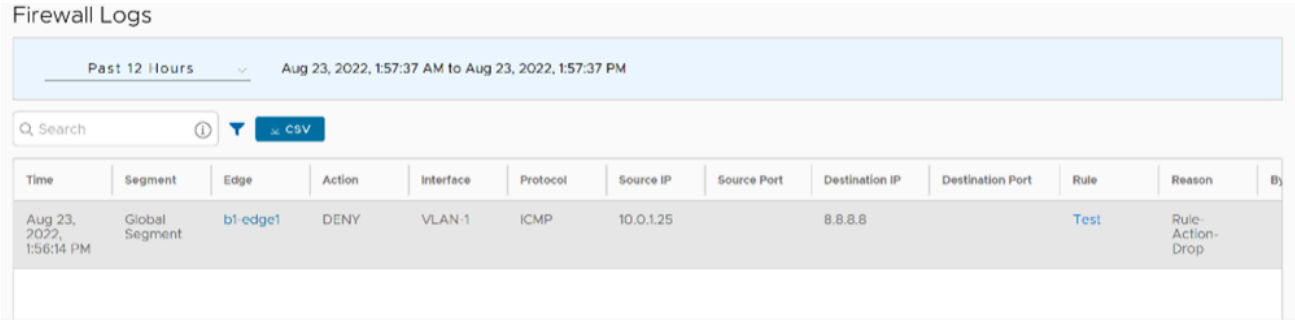
Run the **List Active Firewall Sessions** test on the required Edge. Following is an example of the test output:

Figure 5-19: List Active Firewall Sessions Sample Output



You can verify denied traffic of firewall under **Monitor** > **Firewall Logs**.

Figure 5-20: Firewall Logs



The Remote Diagnostics output displays the following information:

Table 2: List Active Firewall Sessions Sample Output- Options and Descriptions

Option	Description
Segment	Specifies the segment in which the firewall session is processed by the Edge. You can also filter the output based on specific segment.
Src IP	Specifies the source IP which initiated the firewall session.
Dst IP	Specifies the destination IP of the firewall session.
Protocol	Specifies the protocol that the firewall session traffic is using.
Src Port	Specifies the source port of the firewall session traffic.
Dst Port	Specifies the destination port of the firewall session traffic.
Application	Specifies the application that is identified by the Application engine/DPI engine.
Firewall Policy	Specifies the firewall rule which is being matched by the session among the configured firewall rules.
TCP State	Specifies the current TCP state of the session. In the output you will see the current TCP state of any flows. There are 11 distinct TCP states as defined in *RFC 793*: • SERVER_LISTEN- represents the initial state of the TCP FSM on the Edge. This state is not shown in the Remote Diagnostic output, as this is the default state as soon as the session is created for the first packet of the flow. If it is SYN, then it is immediately moved to SYN_SENT state. • SYN_SENT- Session moves to this state, when you see a connection request SYN from the Client to the Server. • SYN_RECEIVED- represents a state where SYN+ACK is received from the Server side. • ESTABLISHED- represents a state after 3-away handshake completing ACK from the Client side. Session is now ready for the data transfer phase. • CLIENT_FIN- From the ESTABLISHED state, transition happens to the CLIENT_FIN state after FIN is received from the Client side. In this state, only FIN or ACK retransmits are allowed from the Client side. But from the Server side, all packets are allowed, with an exception to FIN which moves the state to CLOSING. • SERVER_ FIN- From the ESTABLISHED state, transition happens to the SERVER_FIN state after FIN received from the Server side. In this state, only FIN or ACK retransmits are allowed from the Server side. But from the Client side, all packets are allowed, with an exception to FIN which moves the state to CLOSING. • CLOSING- represents a state when FIN was received from both the Server and Client ends. In this state, only SYN packets are allowed to reopen the session. • CLOSED- represents a state where RST packet received from either the Server or the Client end. In this state, only SYN packets are allowed to reopen the session, any other packets are dropped.
Bytes Sent	Specifies the firewall session traffic from source IP to destination IP in Bytes.
Bytes Received	Specifies the firewall session traffic from destination IP to source IP in Bytes.
Duration	Specifies the age of the firewall session in seconds.

5.15 IPv6 Clear ND Cache

What is the Purpose of This Test

Clear Neighbor Discovery (ND) cache command is used to clear specific interface learned neighbor entries and their corresponding Layer 2 information.

When Can You Run This Test

Run this test if there is a change of neighbor IP addressing or Layer 2 information. Choose the interface for which the change has taken place.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

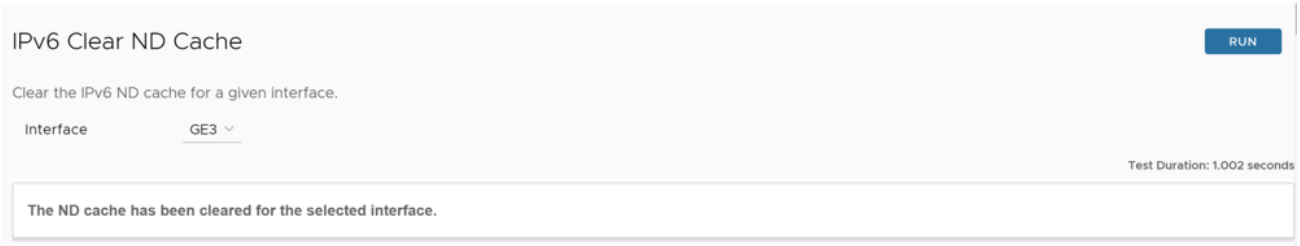
The query should run and provide an output stating as follows:

```
The ND cache has been cleared for the selected interface
```

If you see any other result like for example, "Timed out", check the Edge status.

Following is an example of the test output:

Figure 5-21: IPv6 Clear ND Cache Sample Output



5.16 IPv6 ND Table Dump

What is the Purpose of This Test

IPv6 Neighbor Discovery (ND) Table Dump command shows IPv6 neighbor entries and their corresponding Layer 2 information for each interface.

When Can You Run This Test

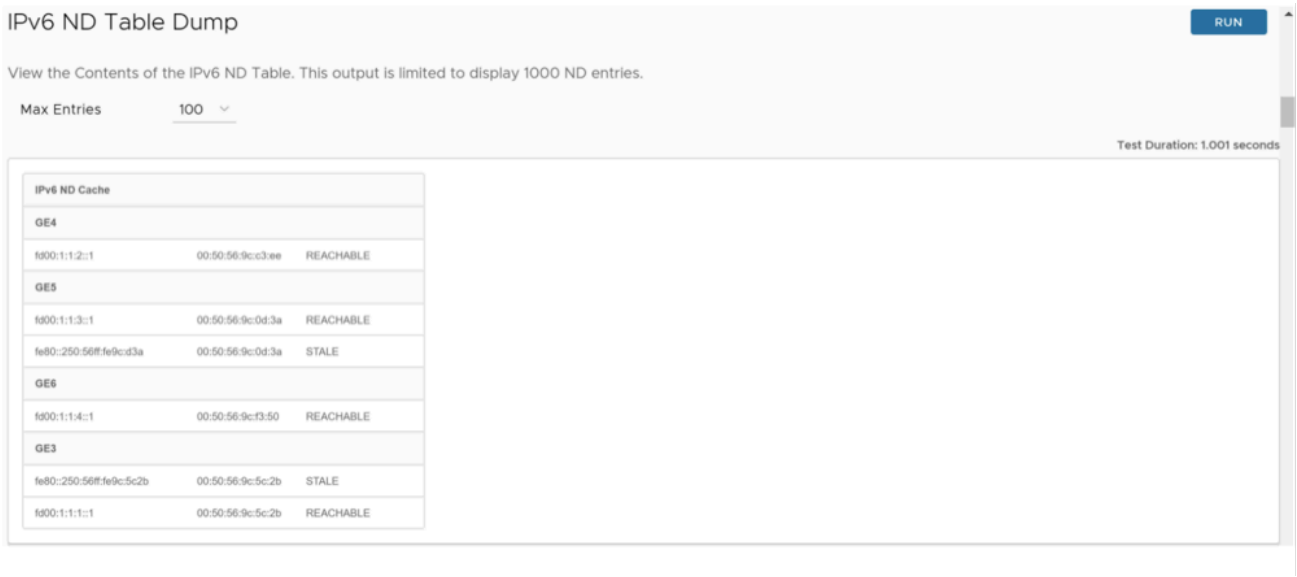
Run this test if you want to verify IPv6 neighbors and the corresponding Layer 2 MAC mapping information.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-22: IPv6 ND Table Dump Sample Output



The Remote Diagnostics output displays the following information for each interface:

- IPv6 address of neighbor
- Corresponding Layer 2 MAC address
- Status of neighbor. The status can be any one of the following:
  - Reachable- This shows the neighbor information has been updated recently.
  - Stale- This shows the neighbor information was collected in past but not used currently.

5.17 IPv6 Route Table Dump

What is the Purpose of This Test

IPv6 Route Table Dump command lists the complete Routing table in IPv6.

When Can You Run This Test

Run this test if you want to verify the Route in the FIB table of IPv6. You can run the test by specifying any of the following options:

- **Segment**- Select the segment for which routes must be displayed. Select "all" for all segments.
- **Prefix**- Specify a particular prefix for which routes must be displayed.
- **Routes**- Select any of the following options from the drop-down menu:
  - **all**- Display all the routes for every prefix.

- **preferred**- Display the most preferred route alone for every prefix (this is the route being used for data forwarding).

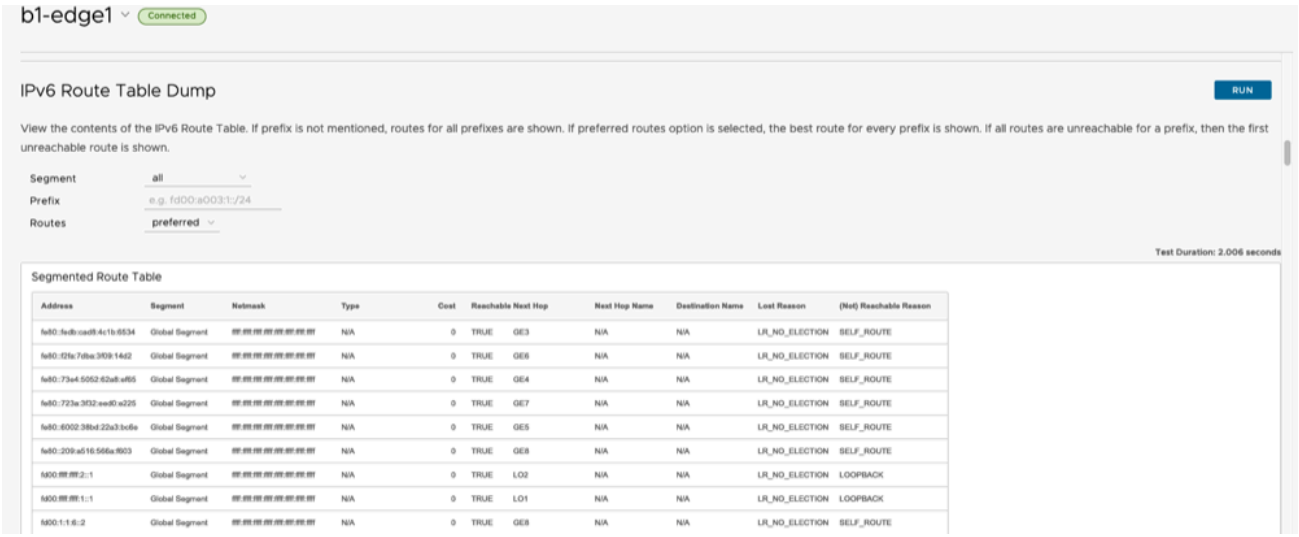
For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

Note: The Route Table Dump command output has a limit of 16000 routes.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-23: IPv6 Route Table Dump Sample Output



The Remote Diagnostics output displays the following information:

Table 3: IPv6 Route Table Dump Field Descriptions

Field	Description
Address	Specifies the IPv6 Routes available in the table.
Segment	Specifies the segment in which the Routes are available and handled by the Edge.
Netmask	Specifies the range of addresses in IPv6.
Type	Specifies the Route type, such as Cloud, Edge2Edge, any (Underlay or Connected), and so on.
Cost	Specifies the Route Cost or Metric used in selection of Route criteria.
Reachable	Specifies the Status of the Route: • True- Reachable • False- Not Reachable
Next Hop	Indicates the local exit interface in case of local routes. In case of overlay/remote routes, it indicates the type of next hop. For example, "Cloud gateway" in case of cloud routes, "Cloud VPN" in case of data center, or "edge to edge" routes etc,
Next Hop Name	Specifies the name of the next hop device.
Destination Name	Specifies the name of the destination device.
Lost Reason	Specifies the codes for different reasons for the routes being lost to next preferred route on the Edge.
(Not) Reachable Reason	Specifies the reason for the route being reachable or not reachable.

Note: An unresolved route, learned over multi-hop BGP, might point to an intermediate interface.

The following table lists the reason codes and the corresponding description:

Table 4: Reason Code Descriptions

Reason Code	Description
PR_UNREACHABLE	In case of overlay routes, the remote peer, which is either Gateway or Edge, is not reachable.
IF_DOWN	Egress Interface is down.
INVALID_IFIDX	Egress Interface if-index for this route is invalid.
SLA_STATE_DOWN	State given by IP SLA tracking is down.
HA_STANDBY	When the local Edge is a Standby, all routes synced from the active are marked as reachable for operational convenience.
LOCAL_MGMT	Management routes are always reachable.
LOOPBACK	Loopback IP address is always reachable.
SELF_ROUTE	Self IP routes are always reachable.
RECUR_UNRES	Recursive routes are marked as reachable so that recursive resolution can be done for operational convenience.
VPN_VIA_NAT	vpnViaNat routes are always reachable.
SLA_STATE_UP	State given by IP SLA tracking is up.
IF_RESOLVED	Egress interface is up and resolved.
PR_REACHABLE	In case of overlay routes, the remote peer, which is either Gateway or Edge, is reachable.

5.18 Flush Flows

What is the Purpose of This Test

To clear all current active sessions/specific flows in the device.

When Can You Run This Test

When specific traffic is not hitting the correct business policy or having issues with the specific flow.

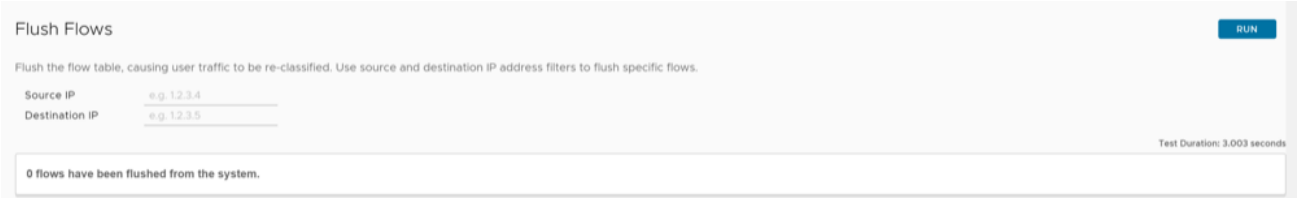
Run the **Flush Flows** test on the required Edge by providing the Source or Destination IP address or both. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Verify if flows flushed is equal to the flows seen. This would create a minor and temporary blip in network.

Following is an example of the test output:

Figure 5-24: Flush Flows Sample Output



5.19 Flush NAT

What is the Purpose of This Test

To remove all the existing NAT entries from the device.

When Can You Run This Test

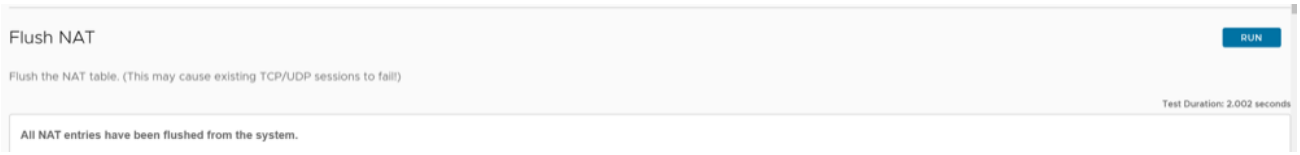
Run this test if NAT table is having improper NAT entry or if you have NAT related issues. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

All the NAT entries would be flushed and could cause an impact for current traffic.

Following is an example of the test output:

Figure 5-25: Flush NAT Sample Output



5.20 NAT Table Dump

What is the Purpose of This Test

To verify if proper NATing is happening on the device.

When Can You Run This Test

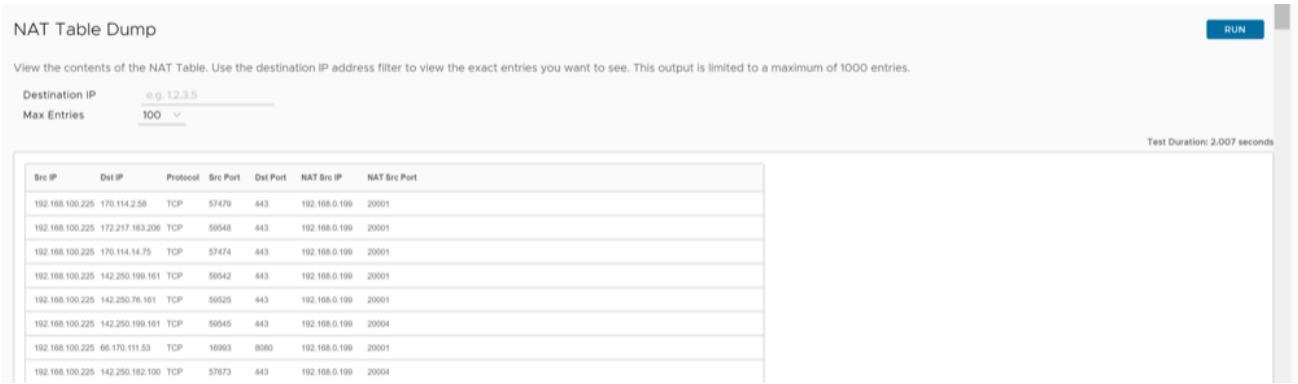
Run this test if there are NATing issues. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Use the Destination filter to view the exact entries in the NAT table. The output is limited to a maximum of 1000 entries.

Following is an example of the test output:

Figure 5-26: NAT Table Dump Sample Output



5.21 List Active Flows

What is the Purpose of This Test

Run this test to understand the behavior of current flows.

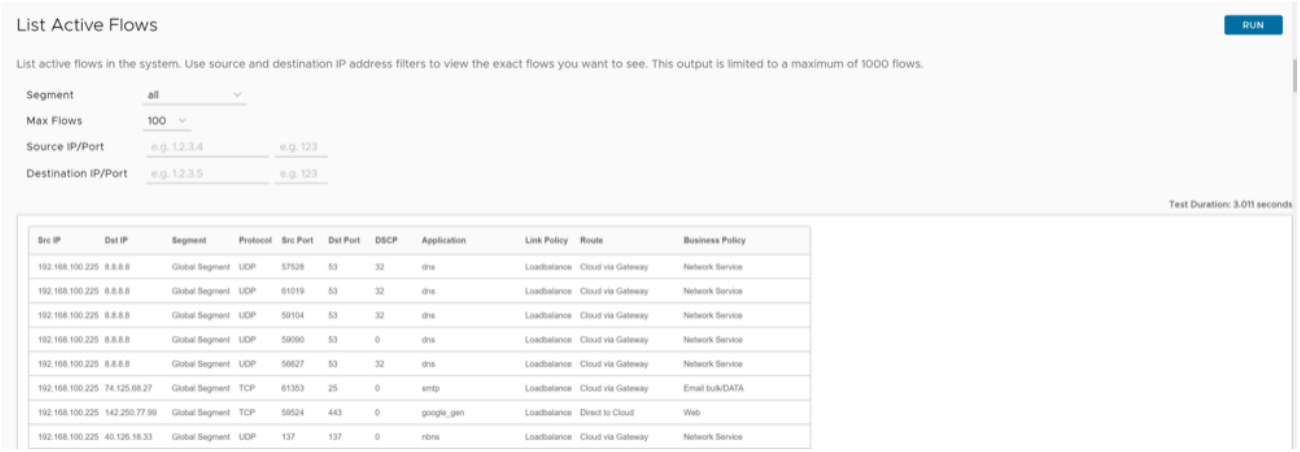
When Can You Run This Test

To analyze how an incoming traffic flow is sent out from VeloCloud SD-WAN. For specific flow issues, you could use Source IP/Port or Destination IP/Port as filters to view that specific traffic flow. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-27: List Active Flows Sample Output



1. Verify if the traffic is hitting the correct business policy.
2. Verify if the traffic LINK policy and Route are taken as configured in business policy.
3. Verify if the application map is able to identify the traffic with correct application signature.
4. Try using flush flows if configuration changes of business policy is done prior to this.

Note: The output is limited to a maximum of 1000 entries.

5.22 Show BFD/BFDv6 Peer Status

What is the Purpose of This Test

Run this test to determine BFDv6 status, with BFD peer. The status can be "UP" or "Down" or "Init".

When Can You Run This Test

- If BFD is failing to come up (or) is down.
- To check the "Local and Remote" timers.
- To check the "Diagnostics" code if BFD is down.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-28: Show BFD/BFDv6 Peer Status Sample Output

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

The Remote Diagnostics output displays the following information:

Table 5: Show BFD/BFDv6 Peer Status Field Descriptions

Field	Description
Peer	Specifies the IP address of BFD peer.
local-address	Specifies the local IPv4 or IPv6 address of the Edge interface in BFD.
ID	Specifies the ID of the BFD peer.
Remote ID	Specifies the remote ID of the BFD peer.
Status	Specifies the status of BFD peer. The status can be "UP" or "Down" or "Init".
Uptime	Specifies the time in seconds, for how long the BFD is UP or Down.
Diagnostics/Remote Diagnostics	Specifies a reason if BFD is down.
Local timers/Remote timers	Indicates the configured Receive and Transmission interval values in milliseconds (ms). Also, displays the Echo transmission interval value in milliseconds (ms) if the Echo Transmission mode is activated on peer. **Note:** The Echo Transmission mode is not supported on Edge.

5.23 Show BFD/BFDv6 Peer Counters

What is the Purpose of This Test

Run this test to determine the BFD packet counters (Sent/Received).

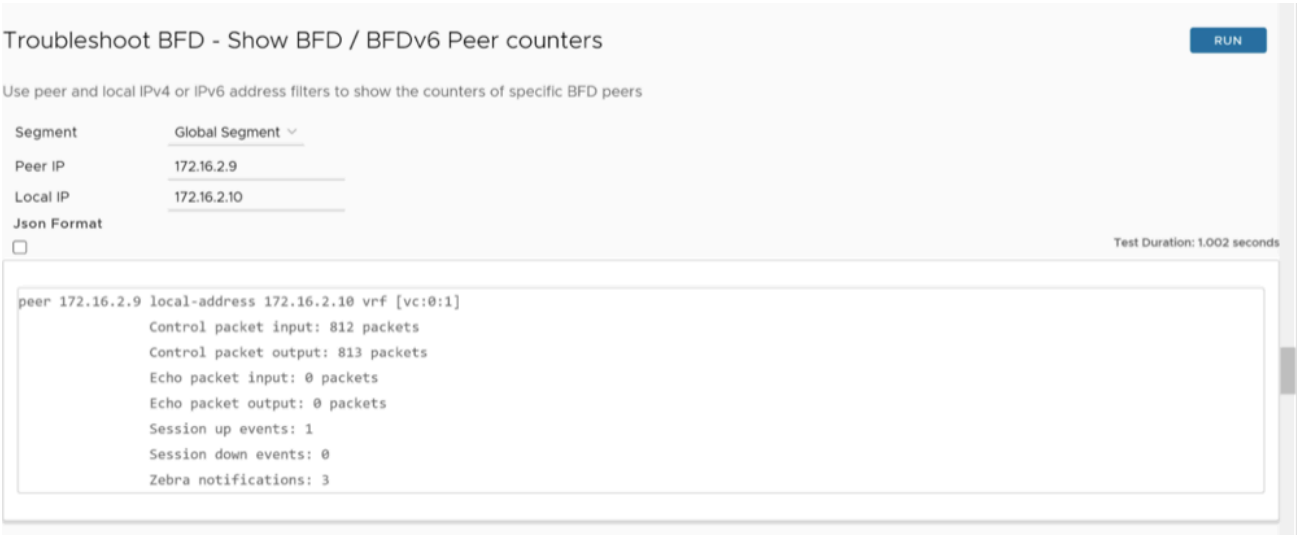
When Can You Run This Test

Run this test if BFD is failing to come Up (or) is Down. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-29: Show BFD/BFDv6 Peer Counters Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

The Remote Diagnostics output displays the following information:

- Count of BFD control Packets Sent/Received, if either only sent or only received counter is incrementing, it indicates a problem on non-incrementing side.
- Count of Echo packets Sent/Received. The Echo mode is not supported, so ideally should be deactivated on peer too.
- Count of Session Up/Down events, confirming how many UP/Down events are seen.
- Count of Zebra notifications.

5.24 Show BFD Settings

What is the Purpose of This Test

To determine the BFD IPv4 settings and neighbor status.

When Can You Run This Test

Run this test to determine the State of the BFD IPv4, along with interval values. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-30: Show BFD Settings Sample Output

5.25 Show BFDv6 Settings

What is the Purpose of This Test

To determine the BFD IPv6 settings and neighbor status.

When Can You Run This Test

Run this test to determine the State of the BFD IPv6, along with interval values. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-31: Show BFDv6 Settings Sample Output

5.26 List BGP Redistributed Routes

What is the Purpose of This Test

Run this test to list the IPv4 routes redistributed by the Edge to the BGP peers.

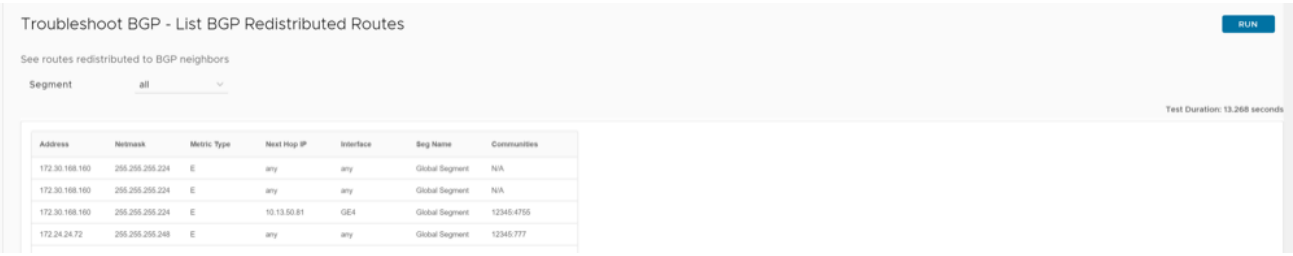
When Can You Run This Test

Run this test to check and confirm if a prefix is redistributed by the Edge to its BGP neighbors. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **List BGP Redistributed Routes** test on the required Edge. Following is an example of the test output:

Figure 5-32: List BGP Redistributed Routes Sample Output



The redistributed route, **Netmask** must be present in the **List BGP Redistributed** table. The redistributed route must be tagged with the expected **Metric Type**, **Next Hop IP** address, exit **Interface**, **Segment Name**, and **Community tag**.

5.27 List BGP Routes

What is the Purpose of This Test

Run this test to list the entire BGP routes. Use **IPv4** or **IPv6** prefix to filter specific BGP routes or leave the prefix empty to see all the routes.

When Can You Run This Test

Run this test to check and confirm:

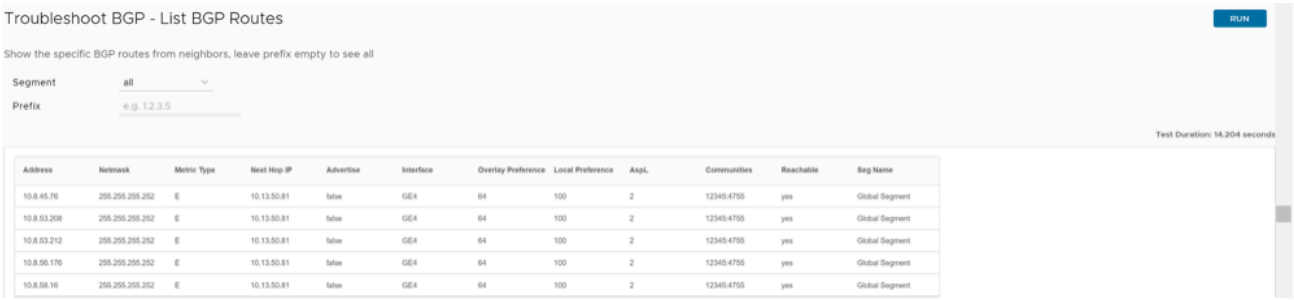
- If a prefix is received in the BGP Table.
- If a BGP prefix is allowed to redistribute into the SD-WAN overlay route table.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **List BGP Routes** test on the required Edge. Following is an example of the test output:

Figure 5-33: List BGP Routes Sample Output



Verify the following in the output:

- BGP routes received from BGP neighbors must be present in the table.
- The BGP route must have the advertise flag set to **True**, and the reachable flag set to **Yes**, to redistribute the received prefix into the overlay route table.
- The BGP route must be tagged with the expected **Metric Type**, **Next Hop IP** address, exit **Interface**, **Segment Name**, and **Community Tag**.

5.28 List Routes per Prefix

What is the Purpose of This Test

Run this test to list all the possible routes (Overlay and Underlay routes) available in the Edge for the specific destination prefix.

When Can You Run This Test

Run this test to check and confirm:

- If the Edge has received both Overlay and Underlay routes (if available) for the specific destination prefix.
- The availability of the primary and secondary route paths (if available) for the specific destination prefix.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **List Routes per Prefix** test on the required Edge. Following is an example of the test output.

Figure 5-34: List Routes per Prefix Sample Output

The **`List Routes per Prefix`** command will display all the possible route entries for the specific destination prefix with the associated local or overlay preferences and community tags.

Based on the route order displayed in the Orchestrator UI, you could check the availability of the primary and backup route paths for the specific destination prefix in each segment. i.e., 1st Route Entry = Best path (Primary path), 2nd Route Entry = Second Best Path, 3rd Route Entry = Third Best Path, and so on.

The **Route Type** identifies the origin of the specific route entry (Overlay or Underlay route).

The **Next Hop IP** and **Dest Name** identify the next-hop device IP (for Underlay route types) or the Edge name (for Overlay route types), which advertised the specific route entry.

5.29 Show BGP Neighbor Advertised Routes

What is the Purpose of This Test

Run this test to list the IPv4 routes advertised by the Edge to the specific BGP peer.

When Can You Run This Test

Following are the scenarios when you can run this test:

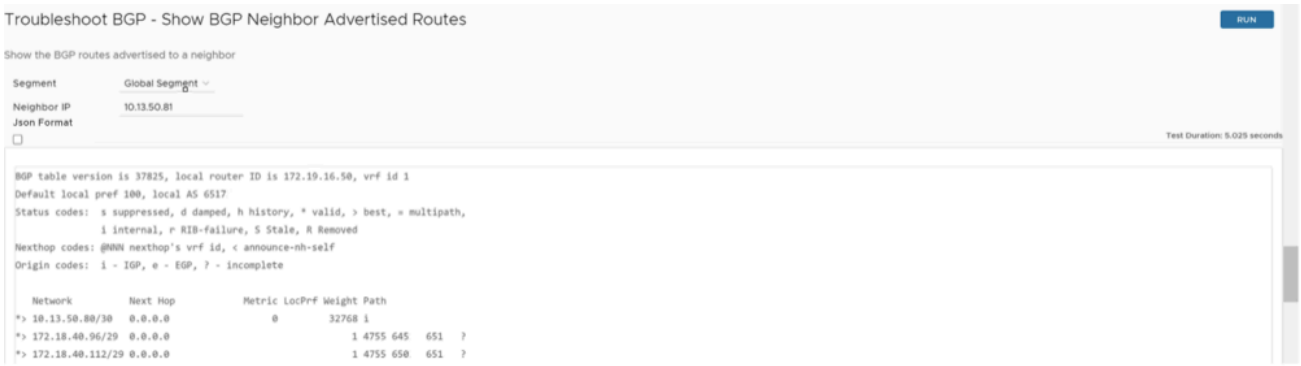
- To check and confirm whether the Edge advertises the IPv4 prefixes to the specific BGP peer.
- If a prefix is denied by the BGP neighbor outbound filter configuration in the Orchestrator, then the Edge does not advertise the denied prefix to the specific BGP neighbor.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGP Neighbor Advertised Routes** test on the required Edge. Following is an example of the test output:

Figure 5-35: Show BGP Neighbor Advertised Routes Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The Edge must list all the advertised prefixes in the table. If a prefix is missed, verify the BGP neighbor status and the outbound filter configuration in the Orchestrator.
- A valid route is marked with the sign "*" and the best route is marked with the sign ">".
- All the prefixes must be tagged respectively with **Metric** (only for self-originated routes), **Local Preference**, **Weight**, **Next Hop** IP Address, **AS-Path**, and **Route Origin** code attributes.

5.30 Show BGP Neighbor Learned Routes

What is the Purpose of This Test

Run this test to list the IPv4 prefixes accepted by the Edge from the BGP peer after applying the BGP neighbor inbound filters configured in the Orchestrator.

When Can You Run This Test

Following are the scenarios when you can run this test:

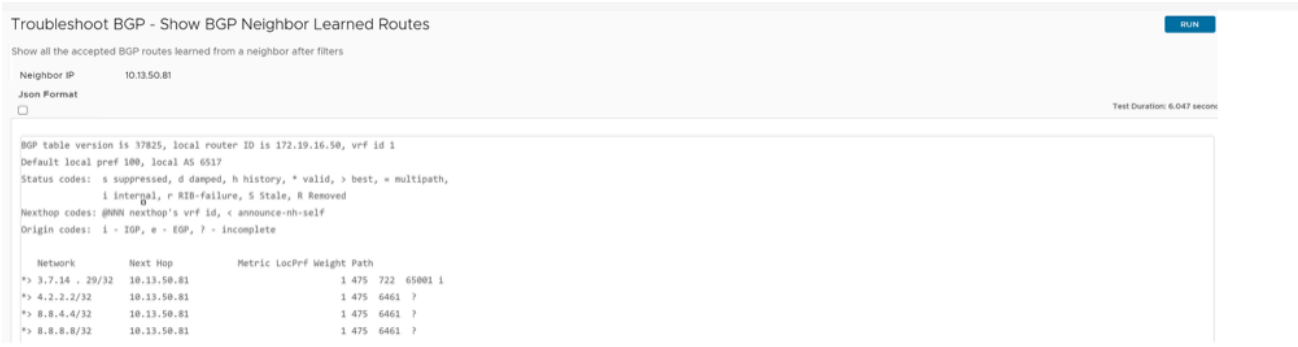
- To check and confirm if the Edge has accepted BGP routes learned from a neighbor.
- If a specific route is not present in the **Show BGP Neighbor Learned Routes** table, then we must verify the BGP neighbor inbound filter in the Orchestrator and run the diagnostic command **`Show BGP Neighbor Received Routes`** to list all the BGP routes advertised by the BGP Neighbor before applying the BGP inbound filters.

What to Check in the Test Output

Run the **Show BGP Neighbor Learned Routes** test on the required Edge. For instructions, see Run Remote Diagnostic Tests on Edges.

Following is an example of the test output:

Figure 5-36: Show BGP Neighbor Learned Routes Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The prefix learned and accepted by the Edge from its BGP neighbor must be present.
- A valid BGP route is marked with the sign "*" and the best route is marked with the sign ">".
- The output must display the total number of prefixes received and accepted by the Edge after applying the inbound filters.

5.31 Show BGP Neighbor Received Routes

What is the Purpose of This Test

Run this test to list the IPv4 prefixes accepted by the Edge from the BGP peer before applying the BGP neighbor inbound filters configured in the Orchestrator.

When Can You Run This Test

Run this test to check and confirm that all the prefixes are received by the Edge from a BGP neighbor. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGP Neighbor Received Routes** test on the required Edge. Following is an example of the test output:

Figure 5-37: Show BGP Neighbor Received Routes Sample Output

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The output must display the total number of prefixes received by the Edge.
- The output must list the number of BGP prefixes filtered by the inbound filters.
- The Edge must list all the received prefixes.
- If a BGP prefix is not present in the table, check the BGP neighborship status and the list of BGP prefixes advertised by the BGP neighbor device to the Edge.

5.32 Show BGP Neighbor Details

What is the Purpose of This Test

Run this test to check the BGP neighbor status and the BGP neighborship uptime.

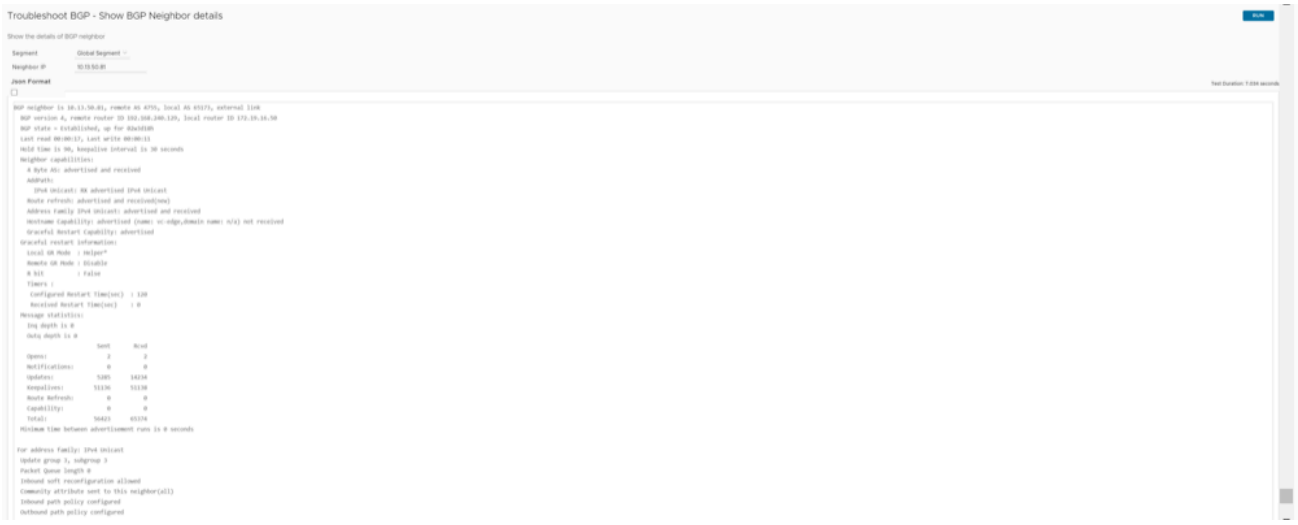
When Can You Run This Test

Run this test to check the details about a neighbor. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGP Neighbor Details** test on the required Edge. Following is an example of the test output:

Figure 5-38: Show BGP Neighbor Details Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

You can check the following parameters in the output:

- BGP neighborship status
- BGP neighborship uptime
- Local AS
- Remote AS
- Edge BGP Router ID
- Peer device BGP Router ID
- BGP Hold and Keep-alive Timers
- Neighbor capabilities

Verify the following in the output:

- Message statistics should display the number of BGP packets sent and received by the Edge.
- Check and confirm if any inbound/outbound filters are attached to the specific BGP neighbor and the list of prefixes accepted by the Edge from the BGP neighbor after applying the inbound filters.
- Verify the source IP address/port number and the destination IP address/port number used by the Edge to establish BGP neighborship. This can be done by looking into the local host/port and foreign host/port details.

5.33 Show BGP Routes per Prefix

What is the Purpose of This Test

Run this test to verify a particular route detail in BGP.

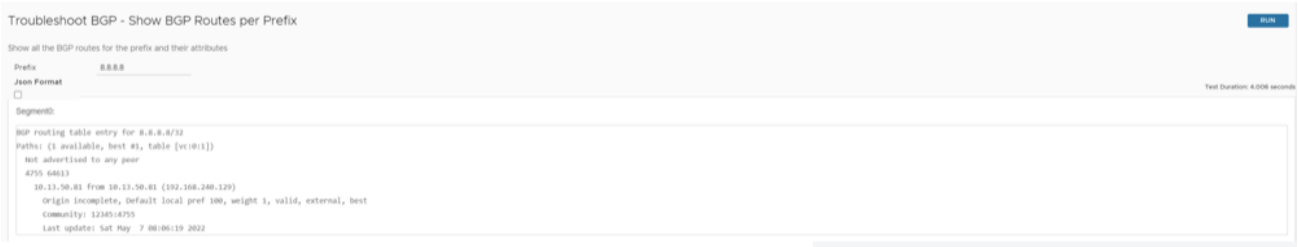
When Can You Run This Test

Run this test to verify how the best path route is selected using BGP attributes. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGP Routes per Prefix** test on the required Edge. Following is an example of the test output:

Figure 5-39: Show BGP Routes per Prefix Sample Output



The output provides the following details:

- Best path to the routes along with all other paths.
- BGP attributes to compare and check how best path is chosen.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

5.34 Show BGP Summary

What is the Purpose of This Test

This test displays the BGP neighbors associated with the device.

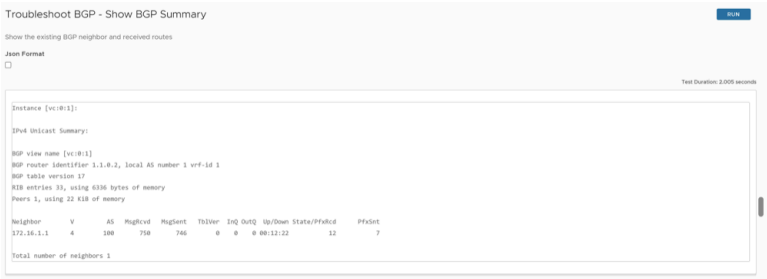
When Can You Run This Test

Run this test to understand if the BGP neighbor is up or not with neighbor details. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

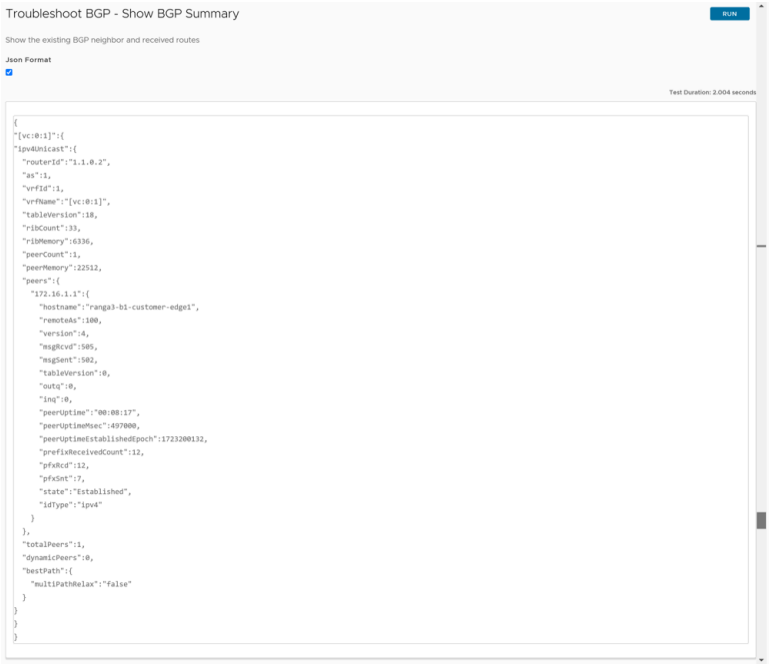
Run the **Show BGP Summary** test on the required Edge. Following is an example of the test output in string format:

Figure 5-40: Show BGP Summary Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Figure 5-41: Show BGP Summary Sample Output in JSON Format



Verify the following in the output:

- **Neighbor**: Provides IP of peer which we used to peer in configuration
- **V**: Indicates the version of BGP running in device.
- **AS**: Provides the remote peer AS path
- **MsgRcd**: Number of packets BGP has received from its peer
- **Msg Sent**: Number of packets BGP has sent to its peer
- **TblVer**: Provides the detail of current BGP table version changes if any
- **Inq**: Number of packets in queue to be received from its peer

- **outq**: Number of packets in out queue to be sent to its peer
- **State/Pfx rcd**: Indicates the State and Prefix received as described in the following table:

Table 6: Show BGP State and Descriptions

State	Description
Idle	BGP resources are initialized by the router. BGP inbound connection attempts are refused. BGP initiates a TCP connection to the peer.
Connect	BGP waits for the 3WHS to complete. If successful, the OPEN message is sent to the peer and BGP moves to the OpenSent state. If unsuccessful, BGP continues to the Active state. However, if the ConnectRetry expires, BGP remains in this state, with the timer being reset and a new 3WHS being initiated.
Active	The ConnectRetry timer is reset, and BGP returns to the Connect state.
OpenSent	BGP waits for an OPEN message from its peer. Once received, BGP moves to the OpenConfirm state.
OpenConfirm	BGP waits for a keepalive message from its peer. If the message is received before the timeout expires, BGP moves to the Established state. Otherwise, BGP transitions to Idle.
Established	Both peers exchange UPDATE messages. If there is an error within any of the UPDATE messages, the BGP peer sends a NOTIFICATION message and enters the Idle state.

Note: Any other state with a number represents the number of subnets received by the BGP from its peer.

5.35 Show BGP Table

What is the Purpose of This Test

To view complete BGP table.

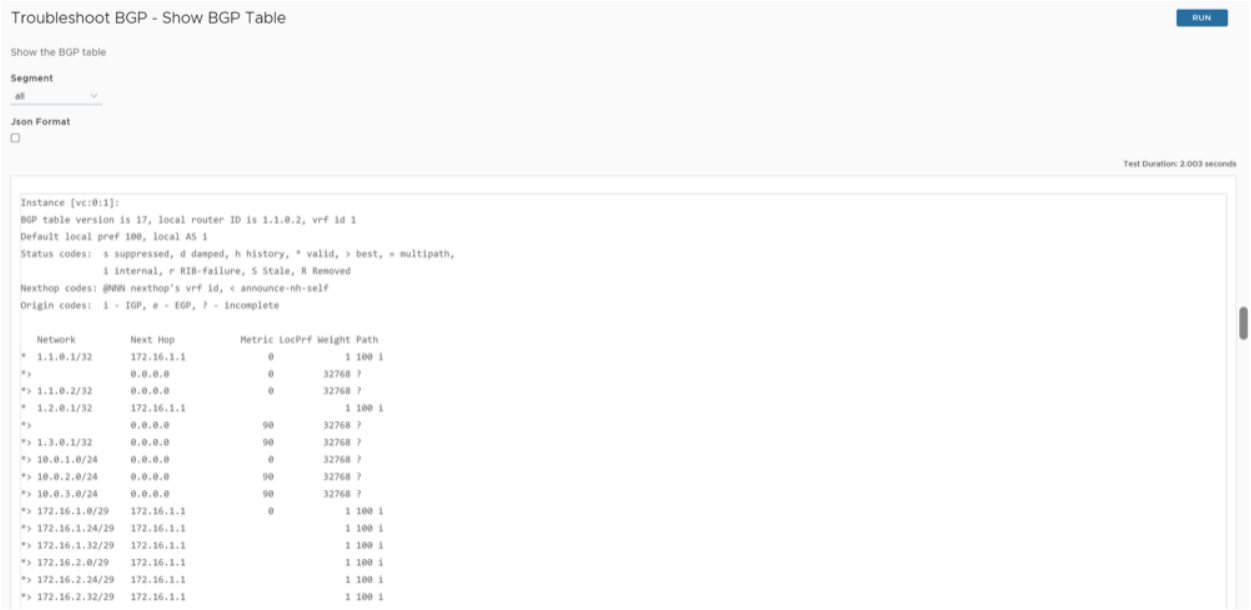
When Can You Run This Test

To verify whether the list of subnets and the next hop are as expected. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGP Table** test on the required Edge. Following is an example of the test output:

Figure 5-42: Show BGP Table Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- **Network**: *> is the best path to IP prefix. If there is no ">" before the prefix, then it is not injected to the forwarding table.
- **Nexthop**: Provide the next hop IP to which IP subnet must be sent.
- **Metric**: BGP metric is displayed.
- **Local pref**: BGP local preference, if present for a route, is displayed.
- **Weight**: BGP weight attribute of the route is displayed.
- **Path**: BGP AS path of the route is displayed.

5.36 Show BGPv6 Neighbor Advertised Routes

What is the Purpose of This Test

Run this test to list the IPv6 routes advertised by the Edge to the specific BGPv6 neighbor.

When Can You Run This Test

Following are the scenarios when you can run this test:

- To check and confirm whether the Edge advertises the IPv6 prefixes to the specific BGPv6 neighbor.

- If a prefix is denied by the BGPv6 neighbor outbound filter configuration in the Orchestrator, then the edge does not advertise the denied prefix to the specific BGPv6 neighbor.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGPv6 Neighbor Advertised Routes** test on the required Edge. Following is an example of the test output:

Figure 5-43: Show BGPv6 Neighbor Advertised Routes Sample Output

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The Edge must list all the advertised prefixes in the table. If a prefix is missed, verify the BGPv6 neighbor status and the outbound filter configuration in the Orchestrator.
- A valid route is marked with the sign "*" and the best route is marked with the sign ">".
- All the prefixes must be tagged with **Metric** (only for self-originated routes), **Local Preference**, **Weight**, **Next Hop** IP Address, **AS-Path**, and **Route Origin** code attributes.

5.37 Show BGPv6 Neighbor Learned Routes

What is the Purpose of This Test

Run this test to list the IPv6 prefixes accepted by the Edge from the BGP neighbor after applying the BGP neighbor inbound filters configured in the Orchestrator.

When Can You Run This Test

Following are the scenarios when you can run this test:

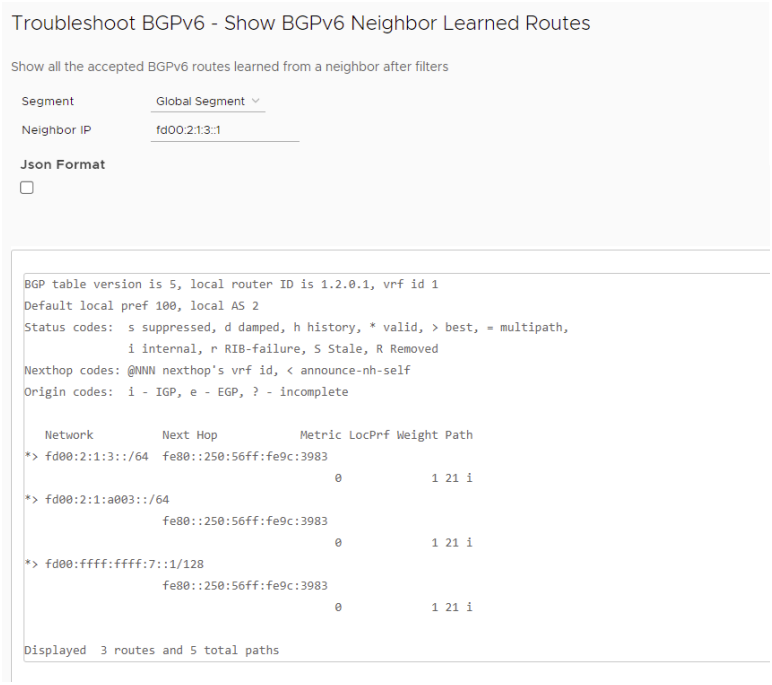
- To check and confirm if the Edge has accepted BGP routes learned from a neighbor.
- If a specific route is not present in the **Show BGPv6 Neighbor Learned Routes** table, then we must verify the BGPv6 neighbor inbound filter in the Orchestrator and run the diagnostic command **`Show BGPv6 Neighbor Received Routes`** to list all the BGP routes advertised by the BGP Neighbor before applying the BGP inbound filters.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGPv6 Neighbor Learned Routes** test on the required Edge. Following is an example of the test output:

Figure 5-44: Show BGPv6 Neighbor Learned Routes Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The prefix learned and accepted by the Edge from its BGP neighbor must be present.
- A valid BGP route is marked with the sign "*" and the best route is marked with the sign ">".
- The prefix accepted by the Edge must hold the correct next-hop IP address, Metric, Local preference, Weight, and AS-path.
- The output must display the total number of prefixes received and accepted by the Edge after applying the inbound filters.

5.38 Show BGPv6 Neighbor Received Routes

What is the Purpose of This Test

Run this test to list the IPv6 prefixes received by the Edge from the BGPv6 neighbor before applying the BGPv6 neighbor inbound filters configured in the Orchestrator.

When Can You Run This Test

Run this test to check and confirm that all the prefixes are received by the Edge from a BGPv6 neighbor. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGPv6 Neighbor Received Routes** test on the required Edge. Following is an example of the test output:

Figure 5-45: Show BGPv6 Neighbor Received Routes Sample Output

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The output must display the total number of prefixes received by the Edge.
- The output must list the number of BGPv6 prefixes filtered by the inbound filters.
- The Edge must list all the received prefixes.
- If a BGPv6 prefix is not present in the table, check the BGPv6 neighborship status and also check the list of BGPv6 prefixes advertised by the peer BGP on the neighbor device.

5.39 Show BGPv6 Neighbor Details

What is the Purpose of This Test

Run this test to check the BGPv6 neighbor status and the BGPv6 neighborship uptime.

When Can You Run This Test

To confirm the current status of the BGPv6 neighborship and list the BGPv6 neighborship attributes. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

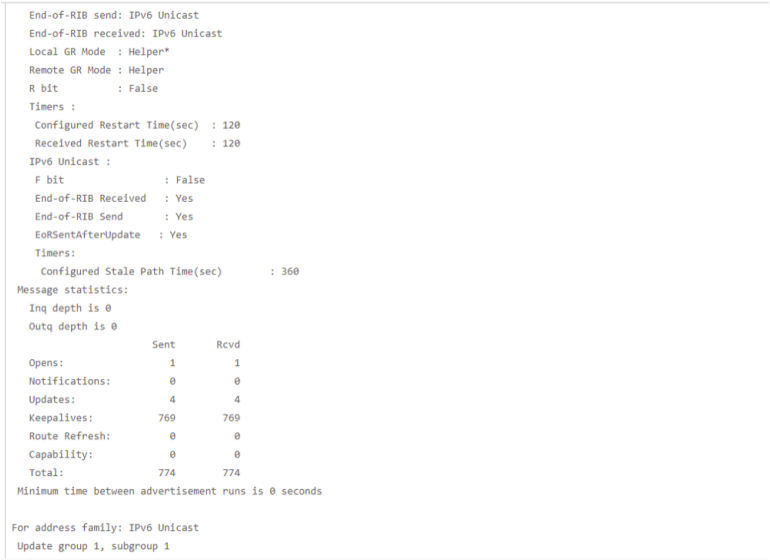
Run the **Show BGPv6 Neighbor Details** test on the required Edge.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Following is an example of the test output:

Figure 5-46: Show BGPv6 Neighbor Details Sample Output

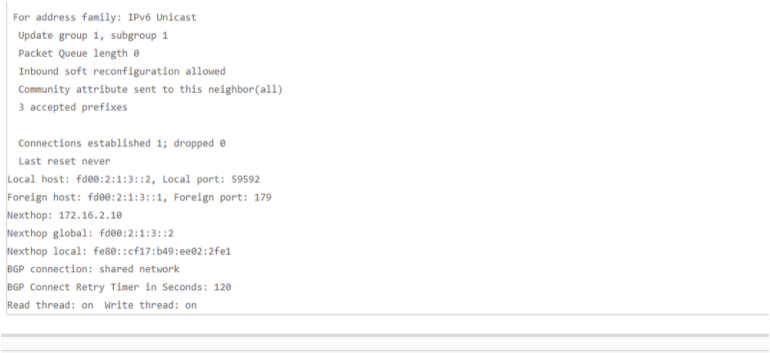
Figure 5-47: Show BGPv6 Neighbor Details Sample Output



```
   End-of-RIB send: IPv6 Unicast
   End-of-RIB received: IPv6 Unicast
   Local GR Mode  : Helper*
   Remote GR Mode : Helper
   R bit          : False
   Timers :
    Configured Restart Time(sec)  : 120
    Received Restart Time(sec)    : 120
   IPv6 Unicast :
    F bit                : False
    End-of-RIB Received  : Yes
    End-of-RIB Send      : Yes
    EoRSentAfterUpdate   : Yes
    Timers:
     Configured Stale Path Time(sec)       : 360
 Message statistics:
   Inq depth is 0
   Outq depth is 0
                        Sent       Rcvd
   Opens:                  1          1
   Notifications:          0          0
   Updates:                4          4
   Keepalives:           769        769
   Route Refresh:          0          0
   Capability:             0          0
   Total:                774        774
 Minimum time between advertisement runs is 0 seconds

For address family: IPv6 Unicast
 Update group 1, subgroup 1
```

Figure 5-48: Show BGPv6 Neighbor Details Sample Output



```
 For address family: IPv6 Unicast
  Update group 1, subgroup 1
  Packet Queue length 0
  Inbound soft reconfiguration allowed
  Community attribute sent to this neighbor(all)
  3 accepted prefixes

  Connections established 1; dropped 0
  Last reset never
Local host: fd00:2:1:3::2, Local port: 59592
Foreign host: fd00:2:1:3::1, Foreign port: 179
Nexthop: 172.16.2.10
Nexthop global: fd00:2:1:3::2
Nexthop local: fe80::cf17:b49:ee02:2fe1
BGP connection: shared network
BGP Connect Retry Timer in Seconds: 120
Read thread: on  Write thread: on
```

Verify the following in the output:

- Check the following parameters in the **BGPv6 neighbor table**:
  - BGPv6 neighborship status
  - BGPv6 neighborship uptime
  - Local AS
  - Remote AS
  - Edge BGP Router ID
  - Peer device BGP Router ID
  - BGP Hold and Keep-alive Timers
  - Neighbor capabilities
- The number of BGP packets (i.e., Open, Notifications, Updates, Keepalives, Route Refresh, and Capability) sent and received by the Edge.

- Any inbound or outbound filters attached to the specific BGP neighbor.
- List of prefixes accepted by the Edge from the BGP neighbor after applying the inbound filters.
- The source IP address and port number and the destination IP address/port number used by the Edge to establish BGPv6 neighborship.

5.40 Show BGPv6 Routes per Prefix

What is the Purpose of This Test

Run this test to list all the possible Overlay and Underlay routes available in the Edge for the specific destination prefix.

When Can You Run This Test

Following are the scenarios when you can run this test:

- To check and confirm if the Edge is received for both Overlay and Underlay routes (if available) for the specific destination prefix.
- To check and confirm the availability of the primary and secondary route paths (if available) for the specific destination prefix.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show BGPv6 Routes per Prefix** test on the required Edge. Following is an example of the test output:

Figure 5-49: Show BGPv6 Routes per Prefix Sample Output

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The output must display all the possible route entries for the specific destination prefix with the associated local or overlay preferences and community tags.
- Based on the route order displayed in the Orchestrator, check the availability of the primary and backup route paths for the specific destination prefix in each segment.

- The **Route Type** flag must help us to identify the origin of the specific route entry (i.e., Overlay or Underlay route).
- The **Next Hop IP** and **Dest Name** flags must help us identify the next-hop device IP (for Underlay route types) or the Edge name (for Overlay route types), which advertised the specific route entry.

5.41 Show BGPv6 Summary

What is the Purpose of This Test

Run this test to view the BGPv6 neighbors associated with the device.

When Can You Run This Test

Run this test to understand if the BGPv6 neighbor is up or not with neighbor details. For instructions, see Run Remote Diagnostic Tests on Edges.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

What to Check in the Test Output

Run the **Show BGPv6 Summary** test on the required Edge. Following is an example of the test output:

Figure 5-50: Show BGPv6 Summary Sample Output

5.42 Show BGPv6 Table

What is the Purpose of This Test

Run this test to view the completed BGPv6 table.

When Can You Run This Test

Run this test to verify the list of subnets and the next hop. For instructions, see Run Remote Diagnostic Tests on Edges.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

What to Check in the Test Output

Run the **Show BGPv6 Table** test on the required Edge. Following is an example of the test output:

Figure 5-51: Show BGPv6 Table Sample Output

5.43 List OSPF Redistributed Routes

What is the Purpose of This Test

Run this test on all or selected segments to confirm if the Overlay routes and Underlay routes (Dynamic/ Static/ Connected) are redistributed to the Edge with correct route metric.

When Can You Run This Test

Run this test in the following scenarios:

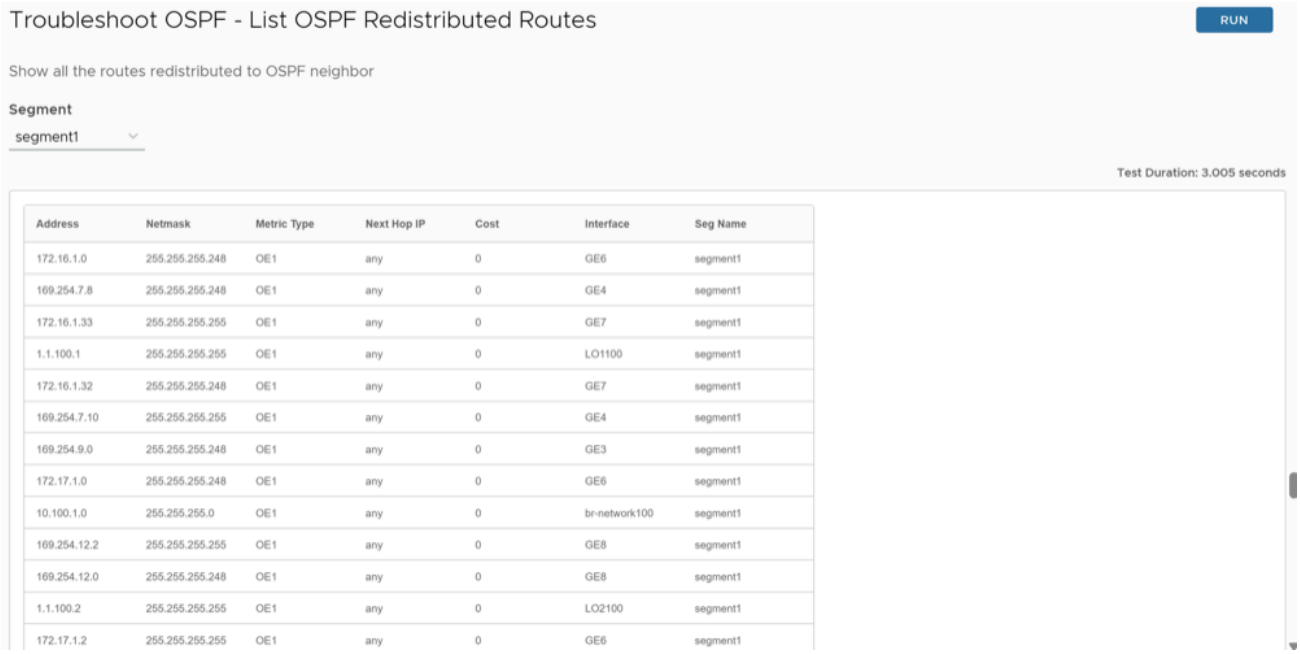
- There is an issue in route redistribution.
- LAN is not able to communicate with specific subnets over overlay VPN.
- OSPF is configured.
- LAN router is not able to see the routes in its routing table that is received from overlay.

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **List OSPF Redistributed Routes** test on the required Edge. Following is an example of the test output:

Figure 5-52: List OSPF Redistributed Routes Sample Output



Verify the following in the output:

- The route must be present in the OSPF redistribute table.
- Lowest cost is preferred if it has multiple routes. If the same route with cost 0 and 1 is learned, then the route with cost 0 is preferred.
- Verify the route metric. The following is the order of route type preference:
  - External routes type 1, O E1
  - External routes type 2, O E2

5.44 List OSPF Routes

What is the Purpose of This Test

Run this test on all or selected segments to show the routes received from peer LAN router.

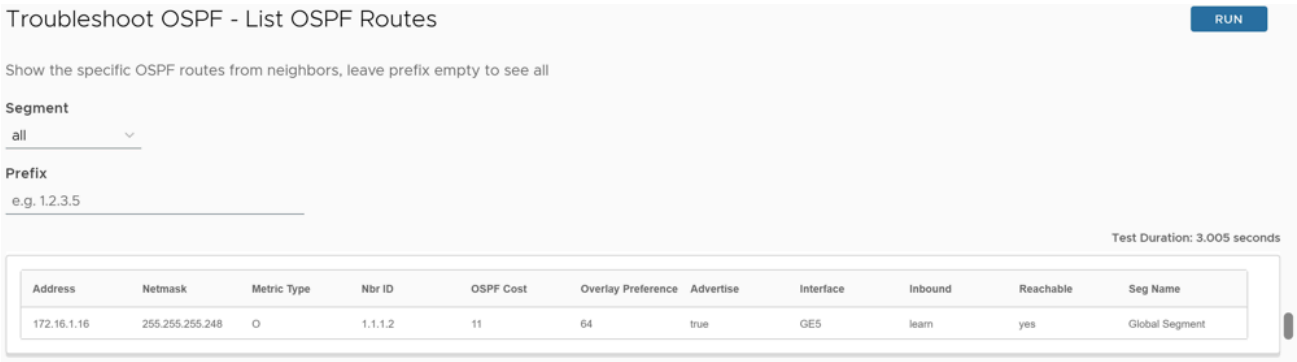
When Can You Run This Test

You can run this test if the peer Edges are not able to receive OSR or DOSR routes from this Edge. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **List OSPF Routes** test on the required Edge. Following is an example of the test output:

Figure 5-53: List OSPF Routes Sample Output



Verify the following in the output:

- Verify the list of routes received from OSPF neighbors.
- Verify the cost attached to it. Lowest cost is preferred if it has multiple routes and the same would be advertised to neighbor Edges.
- Verify the route metric. The following is the order of route type preference:
  - intra-area routes, O
  - inter-area routes, O IA
  - external routes type 1, O E1
  - external routes type 2, O E2

5.45 Show OSPF Database

What is the Purpose of This Test

Run this test on all or selected segments to verify if the OSPF database is equivalent to show ip ospf database.

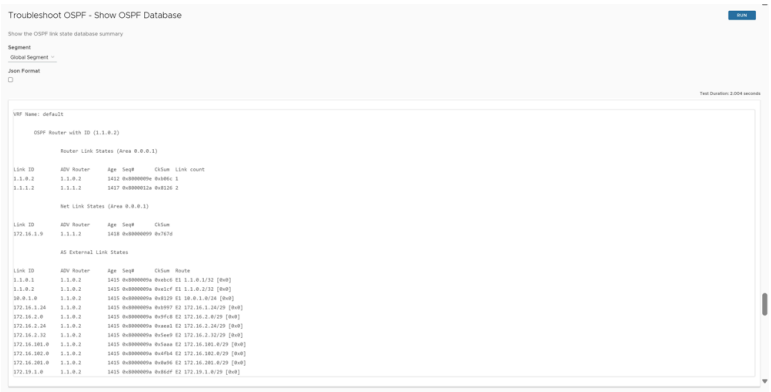
When Can You Run This Test

You can run this test when the OSPF database contains all LSAs that describe the network topology. The OSPF database output displays the content of the LSDB and verifies information about specific LSAs. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show OSPF Database** test on the required Edge. Following is an example of the test output:

Figure 5-54: Show OSPF Database Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- Verify the subnet and its advertising router if there is any asymmetric routing issue or if a different path is taken than expected.
- Verify age if there are route flaps.
- Verify if Peer database and Edge database match, in case of any issues with DB of OSPF.

5.46 Show OSPF Database for E1 Self-Originate Routes

What is the Purpose of This Test

This test provides the list of E1 self-originated routes.

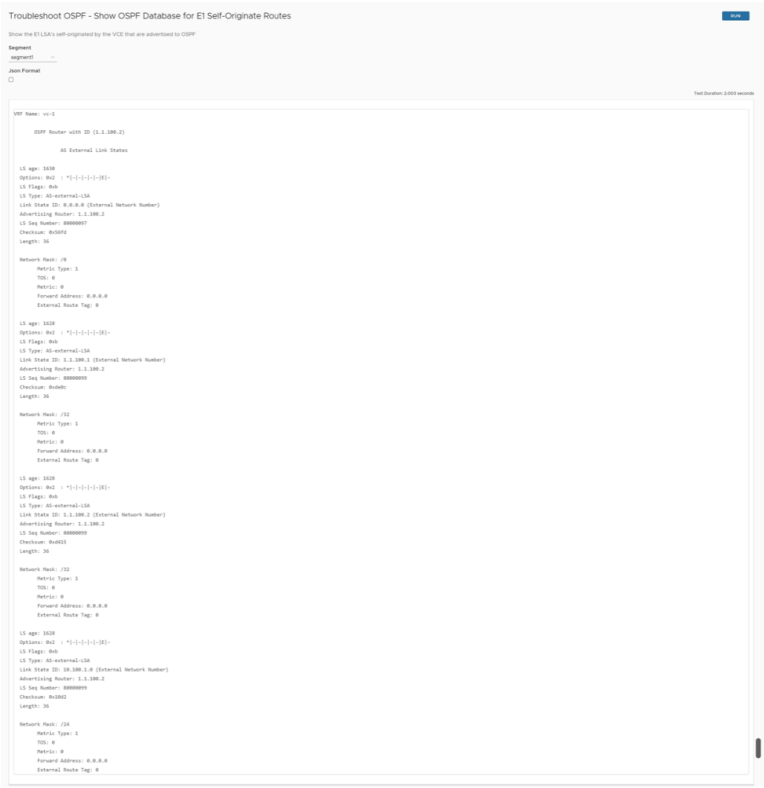
When Can You Run This Test

Run this test on all or selected segments to troubleshoot routing issues on E1 type self-originated routes. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show OSPF Database for E1 Self-Originate Routes** test on the required Edge. Following is an example of the test output:

Figure 5-55: Show OSPF Database for E1 Self-Originate Routes Sample Output



Select the Json Format check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- The LSA type is set to external LSA.
- The Advertising router should be the router which is originating this route.
- For LS age, check if there are any advertisement issues.

5.47 Show OSPF Neighbors

What is the Purpose of This Test

This test indicates if there are issues with OSPF neighbors.

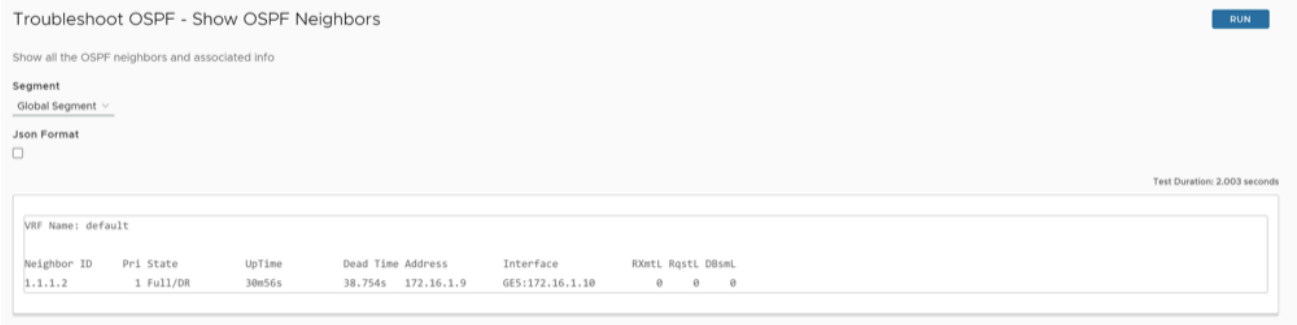
When Can You Run This Test

Run this test on all or selected segments when there is an OSPF neighborship formation issue. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show OSPF Neighbors** test on the required Edge. Following is an example of the test output:

Figure 5-56: Show OSPF Neighbors Sample Output



Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Verify the following in the output:

- Verify if the state of neighbor is Full/(DR or BDR). If this state is not seen, then there is issue with OSPF neighbors' convergence.
- Down state, Attempt/Init state would be seen if you have sent/received OSPF hello packet. If the neighbor is in this state for too long, verify the Area ID, Neighbor ID, Router ID, and OSPF configuration.
- If it is stuck between 2-way state or Ext start state, verify the MTU issues.
- *RxmtL* is LSA retransmission list. This value is used when retransmitting Database Description and Link State Request packets. The default value is 5 seconds. If you see any values, then there is DBD packet exchange issue in transit. It should not be too high ideally. Verify transit path.
- *DBsmL* is database summary list for the neighbor.
- *Rqstl* is Link State Request List.

5.48 Show OSPF Route Table

What is the Purpose of This Test

Run this test on all or selected segments to verify routes and their view on OSPF protocol.

When Can You Run This Test

Run this test to verify how routes are learned on OSPF table. For instructions, see Run Remote Diagnostic Tests on Edges.

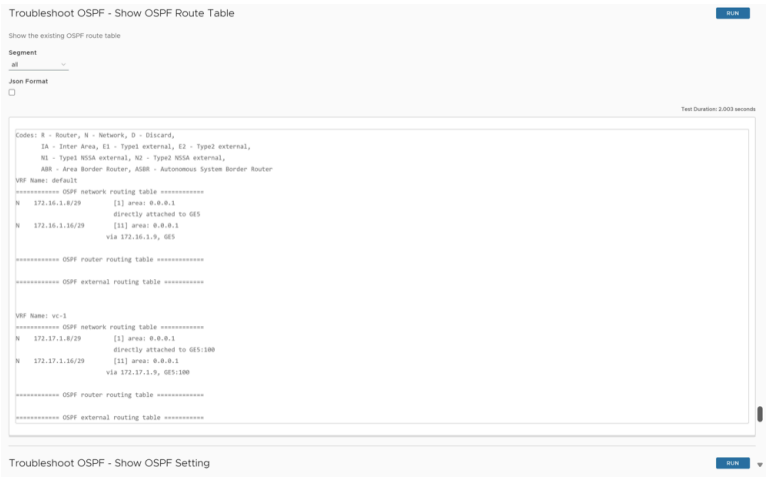
What to Check in the Test Output

Run the **Show OSPF Route Table** test on the required Edge.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Following is an example of the test output:

Figure 5-57: Show OSPF Route Table Sample Output



5.49 Show OSPF Setting

What is the Purpose of This Test

Run this test to verify if the OSPF configuration is pushed properly from Edge to Orchestrator.

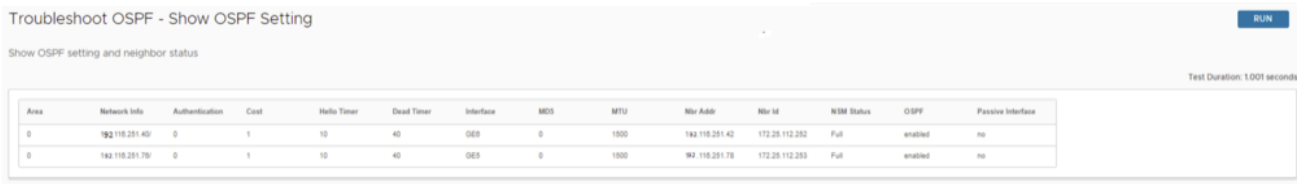
When Can You Run This Test

Run this test to verify Edge OSPF configuration. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show OSPF Setting** test on the required Edge. Following is an example of the test output:

Figure 5-58: Show OSPF Setting Sample Output



Verify the following in the output:

- Verify if the Edge has got the configuration like area ID, network, authentication cost, and hello timer pushed from the Orchestrator if you have customer OSPF settings.

5.50 List OSPFv3 Redistributed Routes

What is the Purpose of This Test

Run this test on all or selected segments to view all the routes redistributed to the OSPFv3 neighbor.

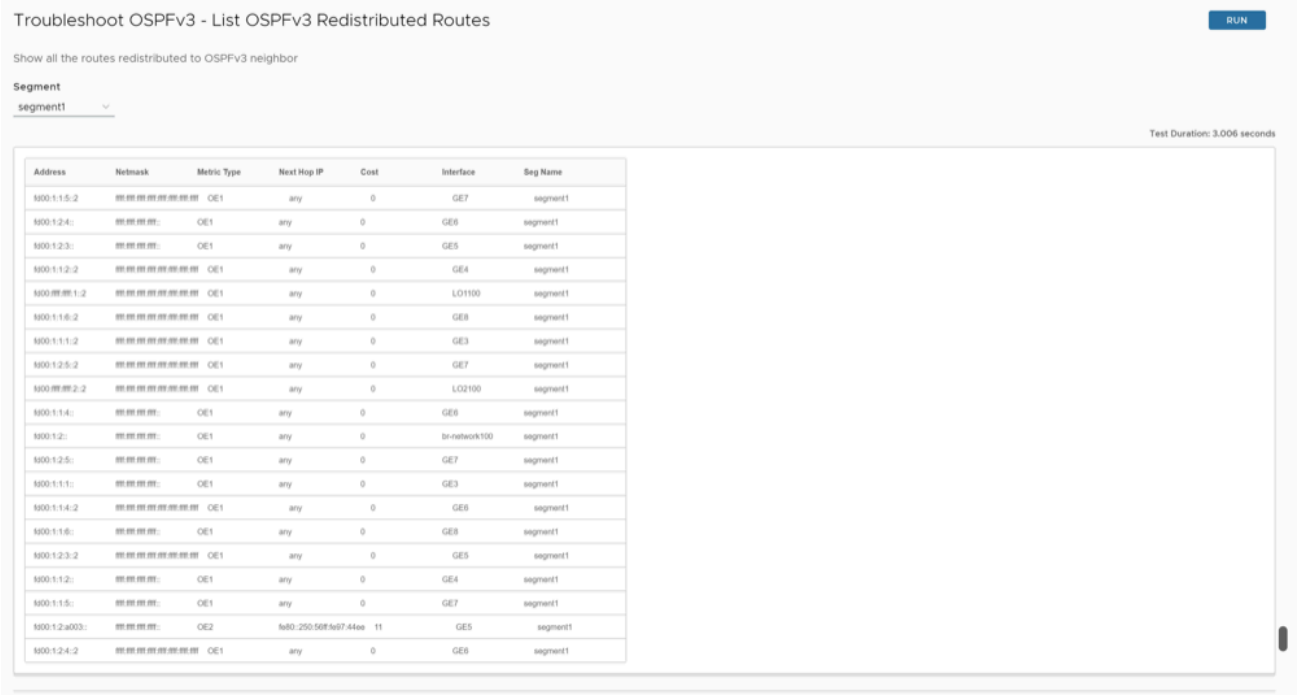
When Can You Run This Test

Run this test to verify OSPFv3 redistributed routes. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **List OSPFv3 Redistributed Routes** test on the required Edge. Following is an example of the test output:

Figure 5-59: List OSPFv3 Redistributed Routes Sample Output



5.51 List OSPFv3 Routes

What is the Purpose of This Test

Run this test on all or selected segments to view the OSPFv3 routes learned from OSPFv3 neighbors for the specified Prefix. Displays all the OSPFv3 routes from the neighbors if the Prefix is not specified. This test displays OSPFv3 routes with actions such as an inbound filter with Overlay Flow Control from Orchestrator applied.

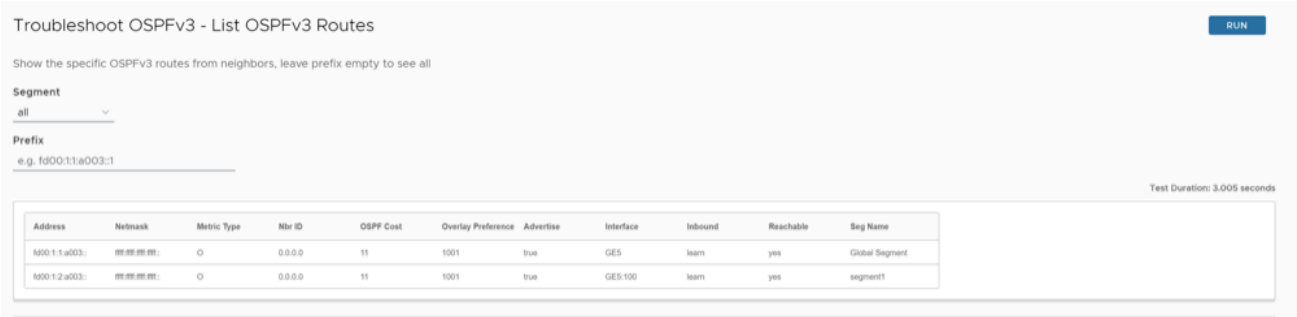
When Can You Run This Test

Run this test to verify OSPFv3 Routes. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **List OSPFv3 Routes** test on the required Edge. Following is an example of the test output:

Figure 5-60: List OSPFv3 Routes Sample Output



5.52 Show OSPFv3 Database

What is the Purpose of This Test

Run this test on all or selected segments to view the OSPFv3 link state database summary.

When Can You Run This Test

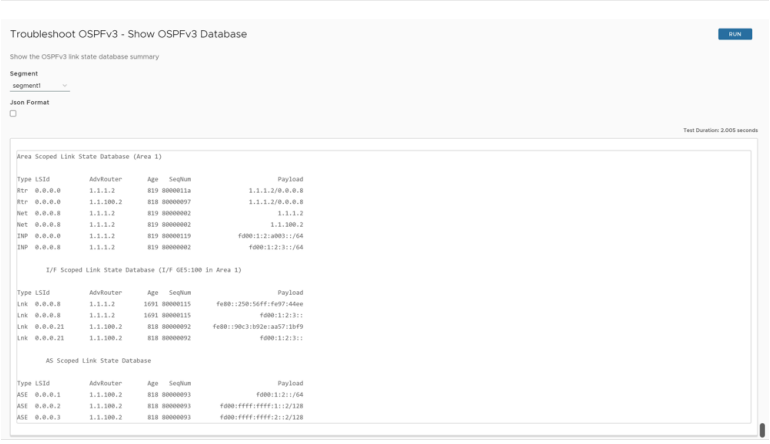
Run this test to verify the OSPFv3 link state database. For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Following is an example of the test output:

Figure 5-61: Show OSPFv3 Database Sample Output



5.53 Show OSPFv3 Database for E1 Self-Originate Routes

What is the Purpose of This Test

Run this test on all or selected segments to view the E1 LSA's self-originated routes that are advertised to OSPFv3 router by the Edge.

When Can You Run This Test

For instructions, see Run Remote Diagnostic Tests on Edges.

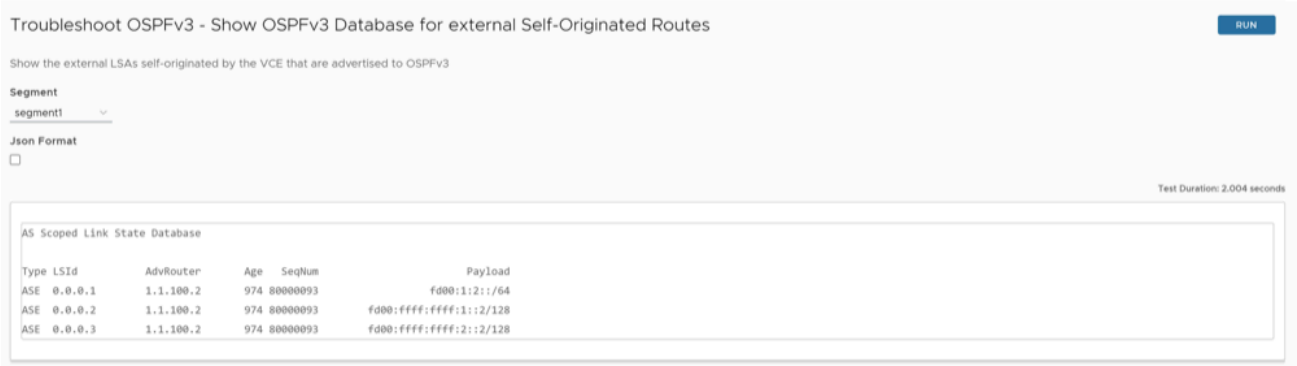
What to Check in the Test Output

Run the E1 LSA's self-originated routes that are advertised to OSPFv3 router by the Edge.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Following is an example of the test output:

Figure 5-62: Show OSPFv3 Database for E1 Self-Originate Routes Sample Output



5.54 Show OSPFv3 Neighbors

What is the Purpose of This Test

Run this test on all or selected segments to view all the OSPFv3 neighbors and associated information.

When Can You Run This Test

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the **Show OSPFv3 Neighbors** test on the required Edge.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Following is an example of the test output:

Figure 5-63: Show OSPFv3 Neighbors Sample Output

5.55 Show OSPFv3 Route Table

What is the Purpose of This Test

Run this test on all or selected segments to view the existing OSPFv3 route table, which displays OSPFv3 information from both learned and redistributed routes.

When Can You Run This Test

Run this test to view the existing OSPFv3 route table. For instructions, see Run Remote Diagnostic Tests on Edges.

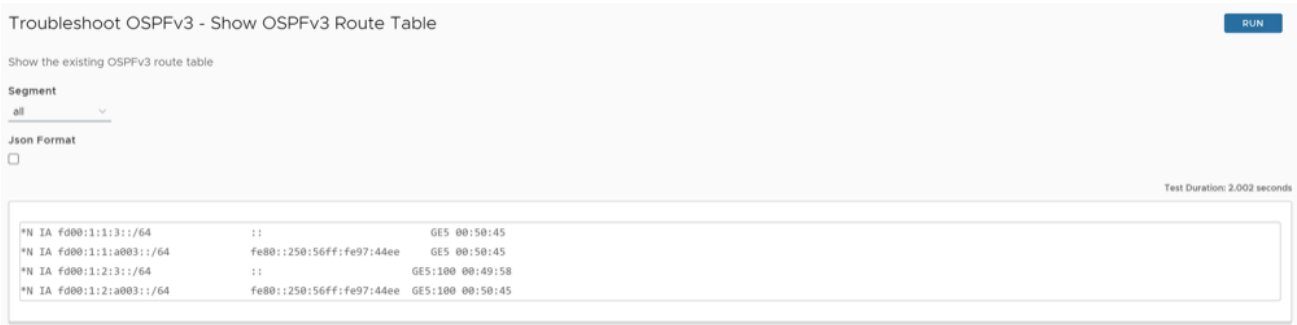
What to Check in the Test Output

Run the **Show OSPFv3 Route Table** test on the required Edge.

Select the **Json Format** check box if you want the test output in JSON format for automation validation.

Following is an example of the test output:

Figure 5-64: Show OSPFv3 Route Table Sample Output



5.56 Show OSPFv3 Setting

What is the Purpose of This Test

Run this test to view the OSPFv3 setting and neighbor status.

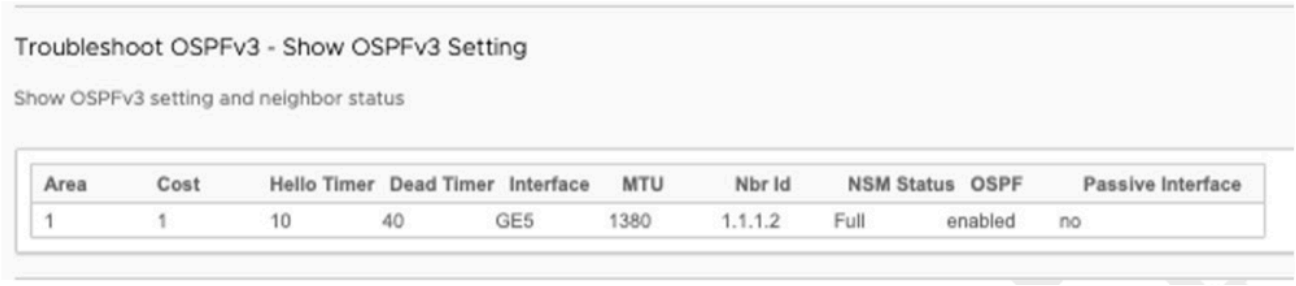
When Can You Run This Test

For instructions, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Run the OSPFv3 setting and neighbor status on the required Edge. Following is an example of the test output:

Figure 5-65: Show OSPFv3 Setting Sample Output



5.57 Dump Context Logging Information

What is the Purpose of This Test

Context logging is per Linux thread logging infrastructure. This test lists the threads which use context logging.

Note: Context Logging is only integrated for the Routing feature in 5.1.0.0 release.

When Can You Run This Test

Run this test to dump the threads which used context logging. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

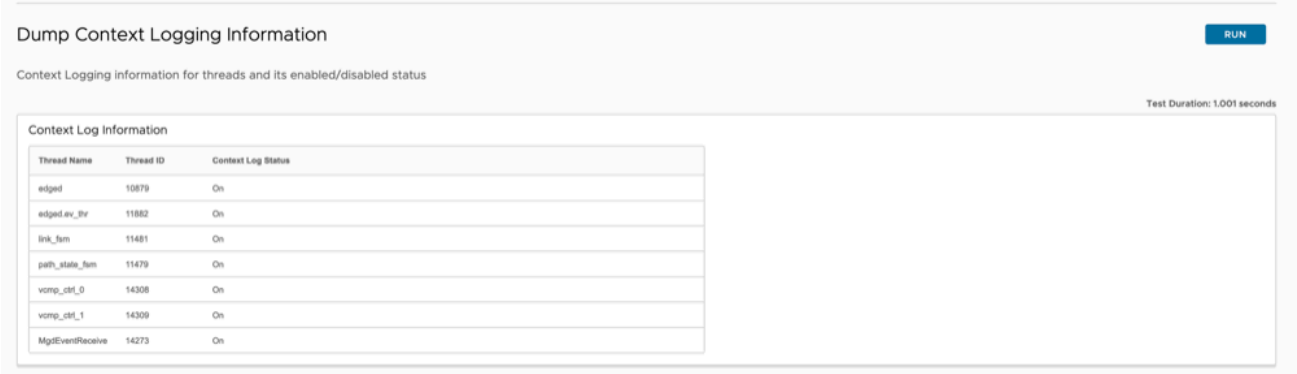
What to Check in the Test Output

The test will dump the Thread Name, Thread ID, and Context Log Status (On or Off) for the thread which uses Context Logging.

- Context Log Status 'On' means context logging is activated for the given thread.
- Context Log Status 'Off' means context logging is deactivated for the given thread.

Following is an example of the test output:

Figure 5-66: Dump Context Logging Information Sample Output



5.58 Enable or Disable Context Logging

What is the Purpose of This Test

Context logging is per Linux thread logging infrastructure. This can be used to activate or deactivate context logging.

Note: Context Logging is only integrated for the Routing feature in 5.1.0.0 release.

When Can You Run This Test

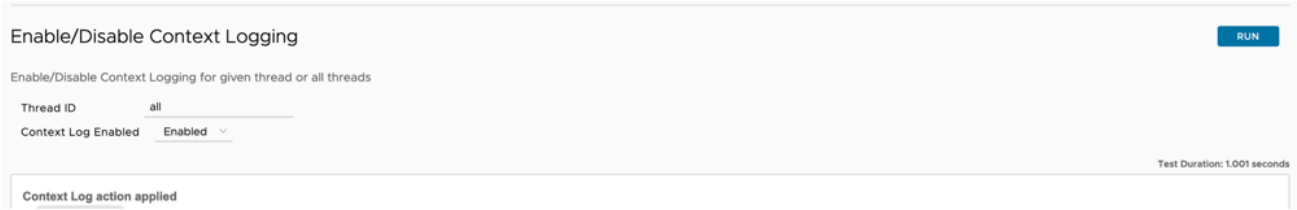
Run this test to activate or deactivate context logging for specific thread or all threads. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Specify a thread ID if you want to activate or deactivate context logging for a specific thread, or else specify the **All** option. Once the test is run, the action will be applied. You can validate the changes using the "Dump Context Logging Information" test command.

Following is an example of the test output:

Figure 5-67: Enable or Disable Context Logging Sample Output



5.59 Gateway

What is the Purpose of This Test

To change the traffic path of cloud (Internet) traffic.

When Can You Run This Test

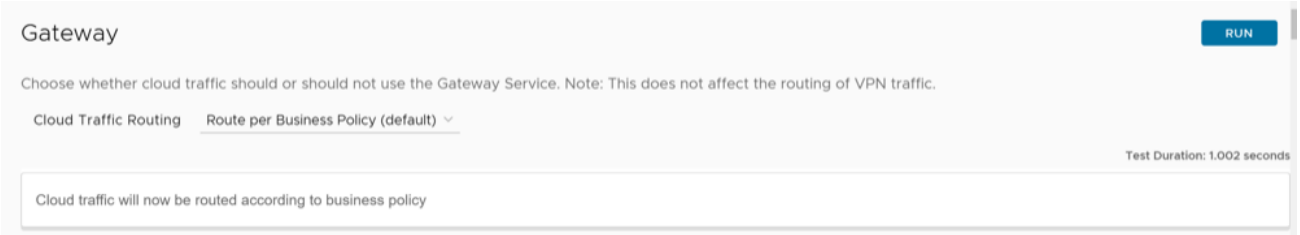
Run this test if you want to change the path of data traffic either to use Business policy, or to use Gateway Path (Multi-Path), or to go direct on WAN Interface.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-68: Gateway Sample Output



5.60 HA Info

What is the Purpose of This Test

To know more about the HA device, IP, Link, Status, and connection information.

When Can You Run This Test

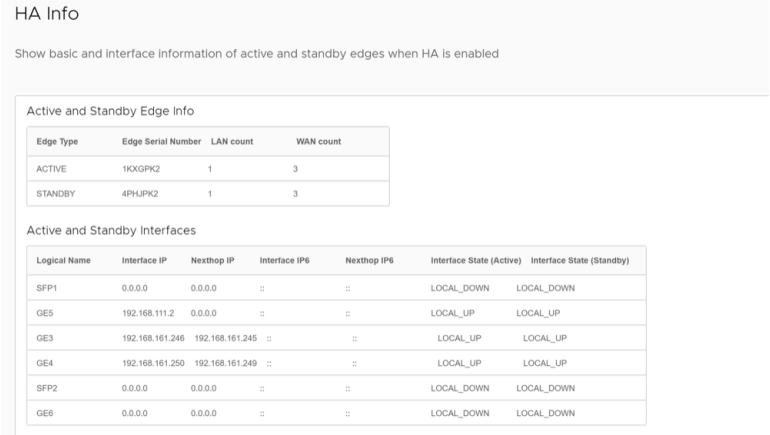
Run this test to view the basic and interface information of active and standby Edges when HA is activated.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

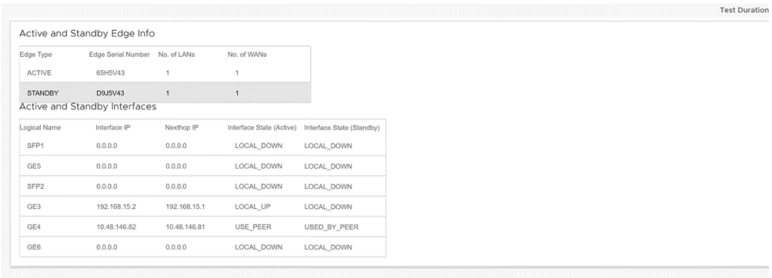
Following is an example of the **Standard HA** test output:

Figure 5-69: Standard HA Info Sample Output



Following is an example of the **Enhanced HA** test output:

Figure 5-70: Enhanced HA Info Sample Output



5.61 List Clients

What is the Purpose of This Test

Run this test to display DHCP lease expiry and interfaces through which client is connected to Edge.

When Can You Run This Test

To verify if the clients are getting the correct DHCP address. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-71: List Clients Sample Output

1. Verify if the LAN client IP and MAC are seen on this for DHCP. If it is not listed, try running the same by restarting the DHCP service.
2. Use this test to verify bogus devices in LAN device.

5.62 List Paths

What is the Purpose of This Test

To check the active tunnel paths between local WAN links and each peer, and gateway status.

When Can You Run This Test

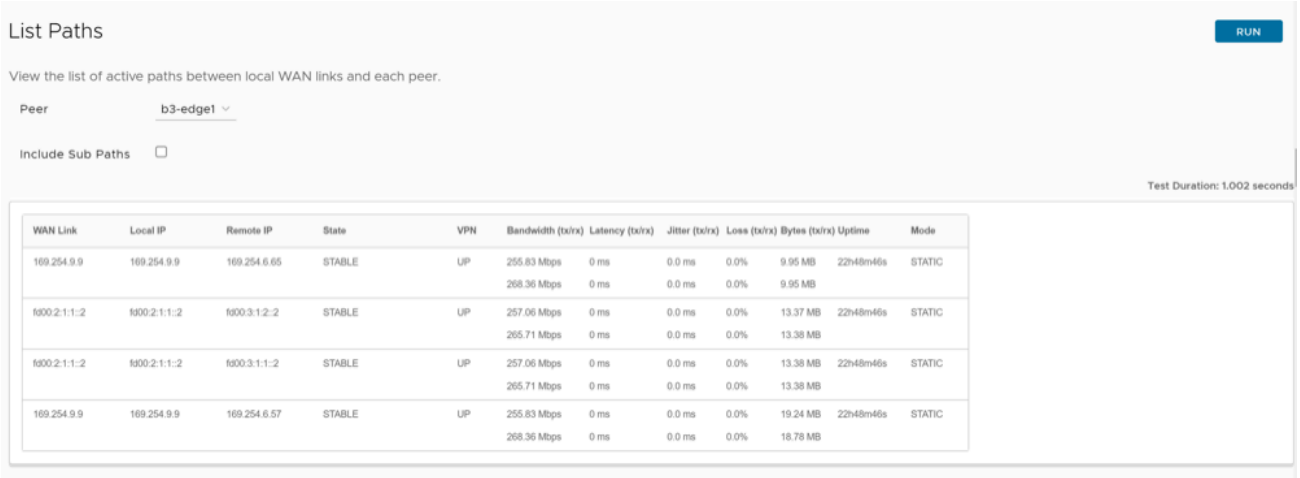
Whenever you face site-to-site communication issues or Gateway traffic issues, run this test to check if the tunnel is UP for destination.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-72: List Paths Sample Output



The Remote Diagnostics output displays the following information:

Table 7: List Paths Field Descriptions

Field	Description
WAN Link	Specifies the WAN link IP used to form tunnel.
Local IP	Specifies the Physical Interface IP for the WAN link.
Remote IP	Specifies the Peer IP to which tunnel is formed.
State	Specifies the tunnel state. The state can be any of the following: • Stable- Tunnel is up and stable. • Unstable- Tunnel is having packet loss, jitter, or latency. You can check the outputs to find what is the cause for the unstable tunnel. • BW_Measurement- In this state both peers negotiate tunnel between them. • Quiet- No traffic is seen across the tunnel. Anything other than "Stable" state confirms that the issue with WAN link.
VPN	Specifies the VPN state and it should be UP for good tunnel.
Bandwidth (tx/rx)	Indicates the bandwidth of the tunnel.
Latency (tx/rx)	Indicates the latency of the tunnel.
Jitter (tx/rx)	Indicates the jitterness of the tunnel.
Loss (tx/rx)	Indicates the packet loss of the tunnel.
Bytes (tx/rx)	Indicates the bytes of the tunnel in MB.
Uptime	Indicates the duration of tunnel uptime.
Mode	Specifies the mode of the tunnel: • Static- Static tunnels are UP always. • Dynamic- Dynamic tunnels are usually seen for b2b tunnel and it will be formed on requirement basis.

5.63 MIBs for Edge

What is the Purpose of This Test

To gather the VeloCloud SD-WAN and Edge MIB's supported by the specific Edge.

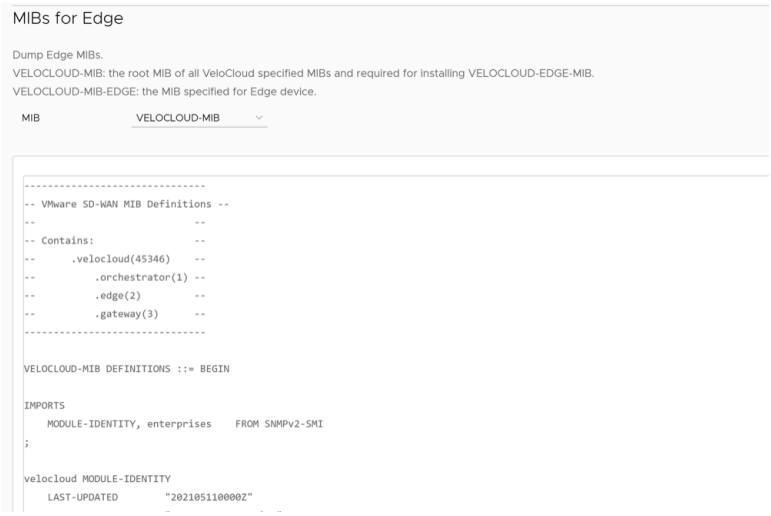
When Can You Run MIBs for Edge Test

Select the **MIBs for Edge** button to download MIB dump file. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-73: MIBs for Edge Sample Output



Copy the entire output to a notepad and save it as a `.mib` file which could further be used on SNMP server to poll the Edge.

5.64 NTP Dump

What is the Purpose of This Test

Run this test to verify the date, time, and NTP server details used by the Edge.

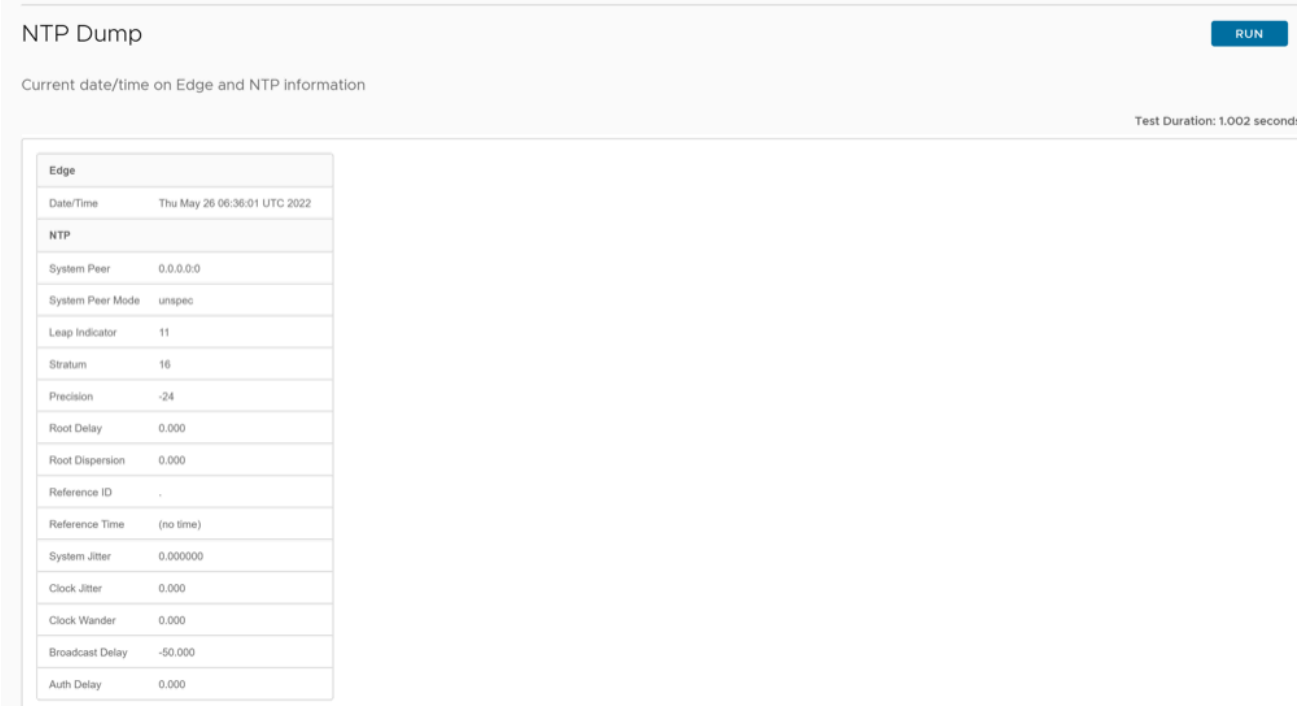
When Can You Run This Test

Run this test if you have an issue with date and time for logs and system time. The time is denoted in UTC time format. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-74: NTP Dump Sample Output



In the output, check the following:

1. Verify if the date and time is correct in the UTC time format.
2. Verify if the system peer is as mentioned in NTP configuration.
3. The Leap Indicator indicates if an impending leap second is to be inserted or deleted in the last minute of the current day. Therefore, if the leap second occurs, the NTP client that is running is one second faster or faster than the seconds mentioned in leap indicator than the actual time.
4. Stratum is the distance between Edge and NTP server. Lesser the Stratum number closer the server is. If the Stratum number is above 16 then it is considered to be unsynchronized.
5. If clock is unsynchronized, change the NTP server or PCAP's to determine if the response is received by the Edge for NTP sync request.

5.65 Reset USB Modem

What is the Purpose of This Test

Run this test to reset the USB and CELL interface.

When Can You Run This Test

Run this test when USB is not detected by the device or if the USB has issues with forming tunnels. For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-75: Reset USB Modem Sample Output

Verify the following:

- **1.** Check if the Restart command has been issued.

  2. Navigate to **Monitor** > **Edges** > **Device** configuration page for checking USB carrier and tunnel information for the respective port.

5.66 System Information

What is the Purpose of This Test

The **`System Information`** command is used to view the system load and WAN stability statistics of the Edge.

When Can You Run This Test

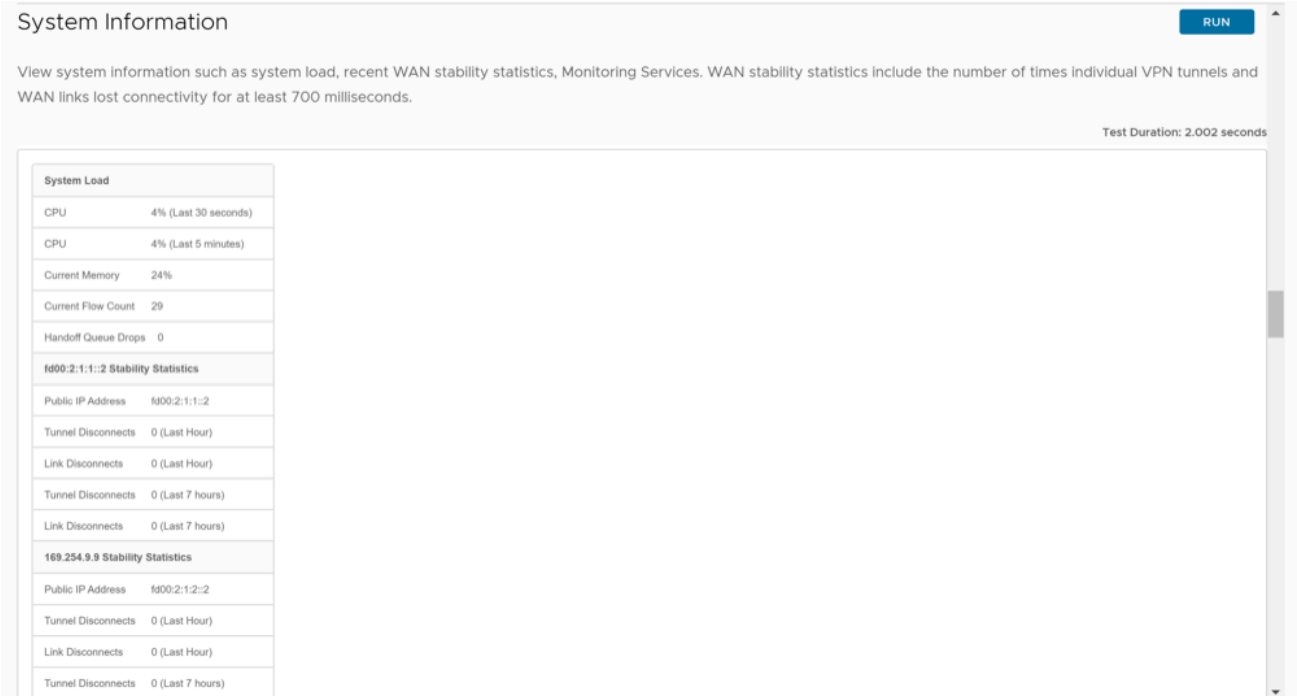
Run this test to view system information such as system load, recent WAN stability statistics, monitoring services details, and so on. The tunnel disconnects do not include the count of direct IPsec connections. WAN stability statistics include the number of times individual VPN tunnels and WAN links lost connectivity for at least 700 milliseconds.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-76: System Information Sample Output



5.67 Route Table Dump

What is the Purpose of This Test

The **`Route Table Dump`** command lists the complete Routing table in IPv4.

When Can You Run This Test

Run this test to verify the Route in the FIB table of IPv4. You can run the test by specifying any of the following options:

- **Segment**- Select the segment for which routes must be displayed. Select "all" for all segments.
- **Prefix**- Specify a particular prefix for which routes must be displayed.
- **Routes**- Select any of the following options from the drop-down menu:
  - **all**- Display all the routes for every prefix.
  - **preferred**- Display the most preferred route alone for every prefix (this is the route being used for data forwarding).

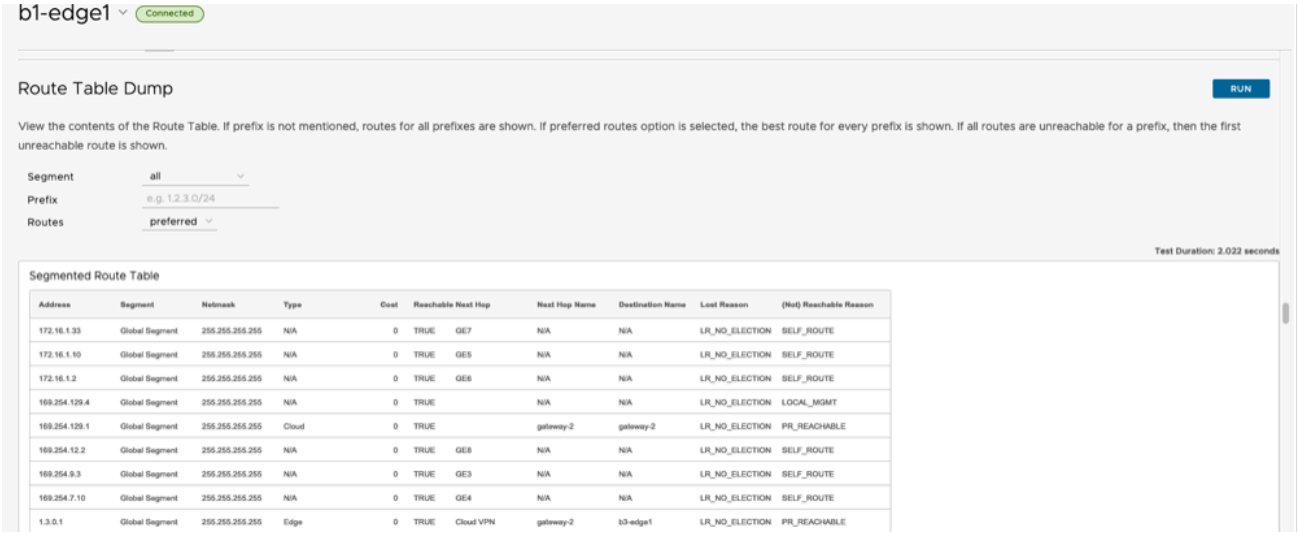
For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

Note: The **`Route Table Dump`** command output has a limit of 16000 routes.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-77: Route Table Dump Sample Output



The Remote Diagnostics output displays the following information:

Table 8: Route Table Dump Field Descriptions

Field	Description
Address	Specifies the IPv4 routes available in the table.
Segment	Specifies the segment in which the routes are available and handled by the Edge.
Netmask	Specifies the range of addresses in IPv4.
Type	Specifies the route type, such as Cloud, Edge2Edge, any (Underlay or Connected), and so on.
Cost	Specifies the route cost or metric used in selection of route criteria.
Preference	Specifies the route preference.
Order	Specifies the order of the route.
Reachable	Specifies the status of the route whether it is True for Reachable or False for Not Reachable.
Next Hop	Indicates the local exit interface in case of local routes. In case of overlay/remote routes, it indicates the type of next hop. For example, "Cloud gateway" in case of cloud routes, "Cloud VPN" in case of data center, or "edge to edge" routes etc,
Next Hop Name	Specifies the name of the next hop device.
Destination Name	Specifies the name of the destination device.
Lost Reason	Specifies the code for the reason why a route loses the routing preference calculation logic to the next preferred route, on both Edges and Gateways.
(Not) Reachable Reason	Specifies the reason for the route being reachable or not reachable.

Note: An unresolved route, learned over multi-hop BGP, might point to an intermediate interface.

The following table lists the reason codes for an Edge and the corresponding descriptions:

Table 9: Reason Code and Descriptions for Edges

Reason Code	Description
PR_UNREACHABLE	In case of overlay routes, the remote peer, which is either Gateway or Edge, is not reachable.
IF_DOWN	Egress Interface is down.
INVALID_IFIDX	Egress Interface if-index for this route is invalid.
SLA_STATE_DOWN	State given by IP SLA tracking is down.
HA_STANDBY	When the local Edge is a Standby, all routes synced from the active are marked as reachable for operational convenience.
LOCAL_MGMT	Management routes are always reachable.
LOOPBACK	Loopback IP address is always reachable.
SELF_ROUTE	Self IP routes are always reachable.
RECUR_UNRES	Recursive routes are marked as reachable so that recursive resolution can be done for operational convenience.
VPN_VIA_NAT	vpnViaNat routes are always reachable.
SLA_STATE_UP	State given by IP SLA tracking is up.
IF_RESOLVED	Egress interface is up and resolved.
PR_REACHABLE	In case of overlay routes, the remote peer, which is either Gateway or Edge, is reachable.
LR_NO_ELECTION	Best route.
LR_NP_SWAN_VS_VELO	Predecessor is selected because it is a non-preferred static WAN route (route configured with preferred flag set to false) when compared to the current route which is a via VeloCloud route.
LR_NP_SWAN_VS_DEFRT	Predecessor is selected because it is a non-preferred static WAN route when compared to the current route which is default route.
LR_NP_ROUTE_TYPE	Predecessor is selected because its route type is better when compared to the current route. Also, one of the routes being compared is a non-preferred route in this case.
LR_BGP_LOCAL_PREF	Both routes are learned using BGP. The predecessor is selected because it has a higher local preference than the current route.
LR_BGP_ASPATH_LEN	Both routes are learned using BGP. Predecessor is selected because it has a lower AS path value than the current route.
LR_BGP_METRIC	Both routes are learned using BGP. Predecessor is selected because it has a lower metric value than the current route.
LR_EXT_OSPF_INTER	Predecessor is selected because it is a route learned from OSPF with an inter or intra area metric when compared to the current route which is learned from BGP.
LR_EXT_BGP_RT	Predecessor is selected because it is a route learned from BGP when compared to the current route which is a route learn from OSPF with metric type OE1 or OE2.
LR_EXT_METRIC_TYPE	Both routes are OSPF routes. The predecessor is selected because it has a better metric type than the current route. Order of preference for OSPF metric types: OSPF_TYPE_INTRA, OSPF_TYPE_INTER, OSPF_TYPE_OE1, OSPF_TYPE_OE2.
LR_EXT_METRIC_VAL	Both routes are OSPF routes. The predecessor is selected because it has a lesser metric than the current route.
LR_EXT_NH_IP	Both routes are OSPF ECMP routes. The current route is lost to the predecessor since it was learned later.
LR_PG_BGP_ORDER	Both are remote BGP routes with same BGP parameters. The current route is selected because it is a Partner Gateway (PG) route and has a lesser "order" value when compared to the current route.
LR_NON_PG_BGP_ORDER	Both are remote BGP routes with same BGP parameters. The current route is selected because it is a non-PG route and has a lesser "order" value when compared to the current route.

Reason Code	Description
LR_EXT_ORDER	Both are remote OSPF routes with same metric. The predecessor is selected because it has a lesser order value than the current route.
LR_PREFERENCE	Both are either BGP or OSPF routes. The predecessor is selected because it has a lesser preference value than the current route.
LR_DCE_NSD_STATIC_PREF DCE- Data center, NSD- Non-SDWAN site	Both are local NSD static routes. The predecessor is selected because it is a preferred route (preferred flag set to true) when compared to the current which is non-preferred.
LR_DCE_NSD_STATIC_METRIC	Both are NSD static routes. The predecessor is selected because it has a lesser metric value than the current route.
LR_DCE_NON_REMOTE	Both are NSD static routes. The predecessor is selected because it is a local route (non-remote), and the current route is a remote route.
LR_DCE_NSD_STATIC_ REMOTE_ORDER	Both are remote NSD static routes. The predecessor is selected because it has a lesser order value when compared to the current route.
LR_DCE_DC_DIRECT	Both are NSD static routes. The predecessor is selected because its DC_DIRECT flag is set, and the current route does not have this flag set. This is the route with "n - nonVelocloud" flag set in the debug.py --routes output. These are routes learned from an NVS from Edge.
LR_DCE_LOGICAL_ID	Both are NSD static routes. The predecessor is selected because it has a better logical ID than the current route.
LR_NETMASK	The predecessor is selected because it has a higher netmask than the current. This will not hit since the netmask is different, it is a separate network/route entry of its own.
LR_NETADDR	The predecessor is selected because it has a higher network address than the current. This will not hit since the network address is different, it is a separate network/route entry of its own.
LR_CONN_FLAG	The predecessor is selected because it is a connected route, and the current route is not a connected route.
LR_SELF_FLAG	The predecessor is selected because it is a self route, and the current route is not a self-route.
LR_SLAN_FLAG	The predecessor is selected because it is a static LAN route, and the current route is not a static LAN route.
LR_SWAN_FLAG	The predecessor is selected because it is a static WAN route, and the current route is not a static WAN route.
LR_NSD_STATIC_LOCAL	The predecessor is selected because it is a local NSD static route, and the current route is an NSD BGP route.
LR_NSD_BGP_VS_NON_ PREF_STATIC	The predecessor is selected because it is a NDS BGP route, and the current route is a local NSD static non-preferred route.
LR_NSD_STATIC_PREF_ VS_NSD_STATIC	The predecessor is selected because it is an NSD static preferred route, and the current route is not an NSD static route.
LR_CONN_STATIC_VS_NSD_BGP	The predecessor is selected because it is a remote connected/static route, and the current route is an NSD BGP route.
LR_OPG_SECURE_STATIC	The predecessor is selected because it is a PG secure static route, and the current is not.
LR_ROUTED_VS_VELO	The predecessor is selected because it is a route learned from routing protocols when compared the current route which is a "v - ViaVeloCloud" route.
LR_INTF_DEF_VS_ROUTED	The predecessor is selected because it is an interface default cloud route when compared to the current route which is a route learned using routing protocols (local or remote).
LR_ROUTE_TYPE	The predecessor is selected because it has a better route than the current.
LR_E2DC_REMOTE	The predecessor is selected because it is a, Edge2DC route, and it is a local route and the current route is a remote route.
LR_CONNECTED_LAN	Both are connected routes. The predecessor is selected because it is a connected LAN route, and the current route is not a connected LAN route.

Reason Code	Description
LR_VELO_REMOTE_FLAG	Both are cloud routes. The predecessor is selected because it is a remote route when compared to the current which is a local cloud route.
LR_VELO_EdgeD_ROUTED	Both are cloud routes. The predecessor is selected because it is a route learned via routing protocol, and the current route is not learned via routing protocol.
LR_VELO_PG_ROUTE	Both are cloud routes. The predecessor is selected because it is a PG route, and the current route is not a PG route.
LR_VIA_VELO_ROUTE	Both are cloud routes. The predecessor is selected because it is a via VeloCloud route, and the current is not a via-velocloud route.
LR_REMOTE_NON_ROUTED	Both are remote (overlay) routes. The predecessor is selected because it is a route not learned via routing protocol (static/connected), and the current route is a route learned via routing protocol.
LR_REMOTE_DCE_FLAG	Both are remote (overlay) routes. The predecessor is selected because it is a data center Edge route ("D - DCE" is set in the debug.py --routes output), and the current is not a data center Edge route.
LR_METRIC	The predecessor is selected because it has a lesser metric than the current route.
LR_ORDER	The predecessor is selected because it has a lesser order than the current route.
LR_LOGICAL_ID	The predecessor is selected because it has a better logical ID than the current route.
LR_EXT_BGP_VIA_PRIMGW	Both are BGP routes. The predecessor is selected because it is an NSD BGP route learned from the primary NSD Gateway. The current route might have been learned from the redundant NDS Gateway.

The following table lists the reason codes for a Gateway and the corresponding descriptions:

Table 10: Reason Code and Descriptions for Gateways

Reason Code	Description
LR_NO_ELECTION	Best route.
LR_NVS_STATIC_PREF	The predecessor is selected because it is an NVS static route, and the current route is not.
LR_EXT_BGP_VS_OSPF	Predecessor is selected because it is a BGP route, and the current route is an OSPF route with metric type OE1/OE2.
LR_EXT_BGP_ROUTE	Both are cloud routes. The predecessor is selected because it is a BGP learned cloud route, and the current route is not (it is static).
LR_CLOUD_ROUTE_VS_ANY	The predecessor is selected because it is an Edge2Edge or Edge2Datacenter route, and the current route is a cloud static route. Edge2Edge/Edge2Datacenter > Cloud static.
LR_BGP_LOCAL_PREF	Both are either Edge2Edge or Edge2Datacenter routes learned via BGP. The predecessor is selected because it has a greater local preference value than that of the current route.
LR_BGP_ASPATH_LEN	Both are either Edge2Edge or Edge2Datacenter routes learned via BGP. The predecessor is selected because it has a lesser AS path value than that of the current route.
LR_BGP_METRIC	Both are either Edge2Edge or Edge2Datacenter routes learned via BGP. The predecessor is selected because it has a lesser metric value than that of the current route.
LR_DCE_NSD_STATIC_PREF	Both are Edge2Datacenter routes. Predecessor is selected because it is an NSD static route, and the current route is not.
LR_DCE_NSD_STATIC_METRIC	Both are Edge2Datacenter static routes. Predecessor is selected because it has lesser metric value than that of the current route.
LR_DCE_NSD_STATIC_ GW_NON_REMOTE	Both are Edge2Datacenter static routes. Predecessor is selected because it is a local route, and the current is a remote route.
LR_DCE_LOGICAL_ID	Both are Edge2Datacenter static routes. Predecessor is selected because it has better logical ID than that of the current route.
LR_E2DC_METRIC	Both are Edge2Datacenter routes. The predecessor is selected because its metric is lesser than that of the current route.
LR_DC_IPADDR	Both are Edge2Datacenter routes. The predecessor is selected because its data center IP address is lesser than that of the current route.
LR_E2DC_NETADDR	Both are Edge2Datacenter routes. The predecessor is selected because its network address is lesser than the current.
LR_E2E_PREFERENCE	Both are Edge2Edge routes. The predecessor is selected because its preference value is lesser than the current route.
LR_E2E_METRIC	Both are Edge2Edge routes. The predecessor is selected because its metric value is lesser than the current route.
LR_E2E_LOGICAL_ID	Both are Edge2Edge routes. The predecessor is selected because it has better logical ID than the current route.
LR_E2E_NETADDR	Both are Edge2Edge routes. The predecessor is selected because its network address is lesser than the current.
LR_OPG_SECURE_STATIC	The predecessor is selected because it is a PG secure static route, and the current route is not a PG secure static.
LR_ROUTE_TYPE	The predecessor is selected because it has a better route type than the current route.
LR_NETMASK	The predecessor is selected because it has a higher netmask than the current.
LR_METRIC	The predecessor is selected because it has a lesser metric value than the current route.
LR_PREFERENCE	Both are routes learned from routing protocols. The predecessor is selected because it has a lesser preference value than the current route.
LR_NETADDR	The predecessor is selected because its network address is lesser than that of the current route.
LR_LOGICAL_ID	The predecessor is selected because its logical ID is better than the current route.

5.68 VPN Test

What is the Purpose of This Test

The **`VPN Test`** command is used to view the VPN Branch to Branch Connectivity between the Edges.

When Can You Run This Test

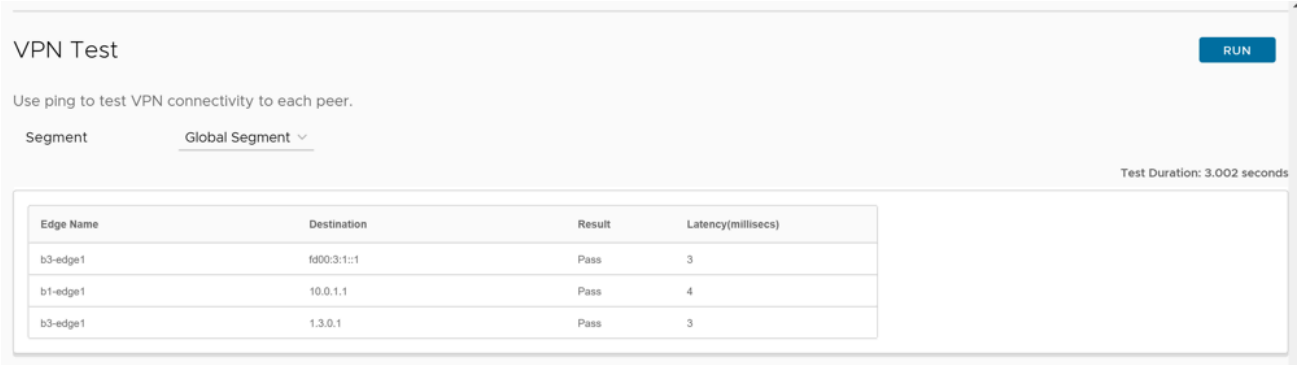
Run this test if the data traffic between the Branches or between Hub and Spoke are not bi-directional.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-78: VPN Test Sample Output



This Remote Diagnostics test will perform the VPN test by confirming the tunnel status between the Edges, and the output displays the following information:

Table 11: VPN Test Field Descriptions

Field	Description
Edge Name	Specifies the name of the peer Edge.
Destination	Specifies the logical ID of the destination device.
Result	The result displays one of the following: • Pass- Tunnels and traffic between the Edges are up and running. • Fail- Tunnels and traffic between the Edges are down.
Latency (milliseconds)	Specifies the latency in milli-seconds.

5.69 WAN Link Bandwidth Test

What is the Purpose of This Test

The **`WAN Link Bandwidth Test`** command is used to measure the Bandwidth on the specific link.

When Can You Run This Test

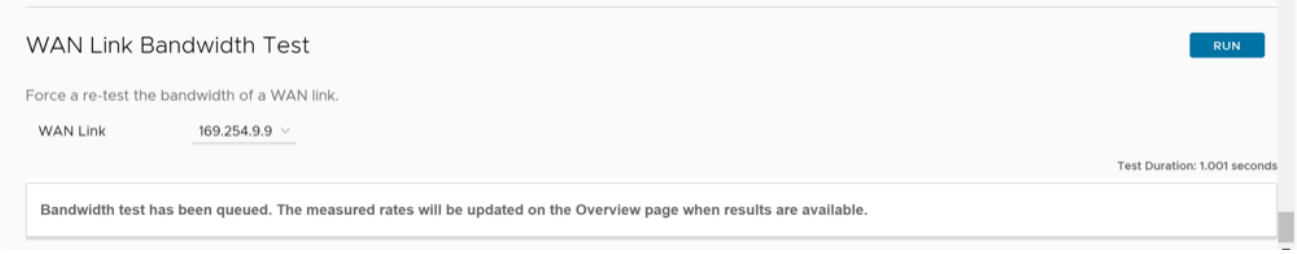
Run this test if you want to verify the bandwidth of the WAN Link. If the link measured wrong or inaccurate bandwidth, then you can perform this test to measure the bandwidth again.

For instructions on how to run a remote diagnostic test on Edges, see Run Remote Diagnostic Tests on Edges.

What to Check in the Test Output

Following is an example of the test output:

Figure 5-79: WAN Link Bandwidth Test Sample Output



Chapter 6

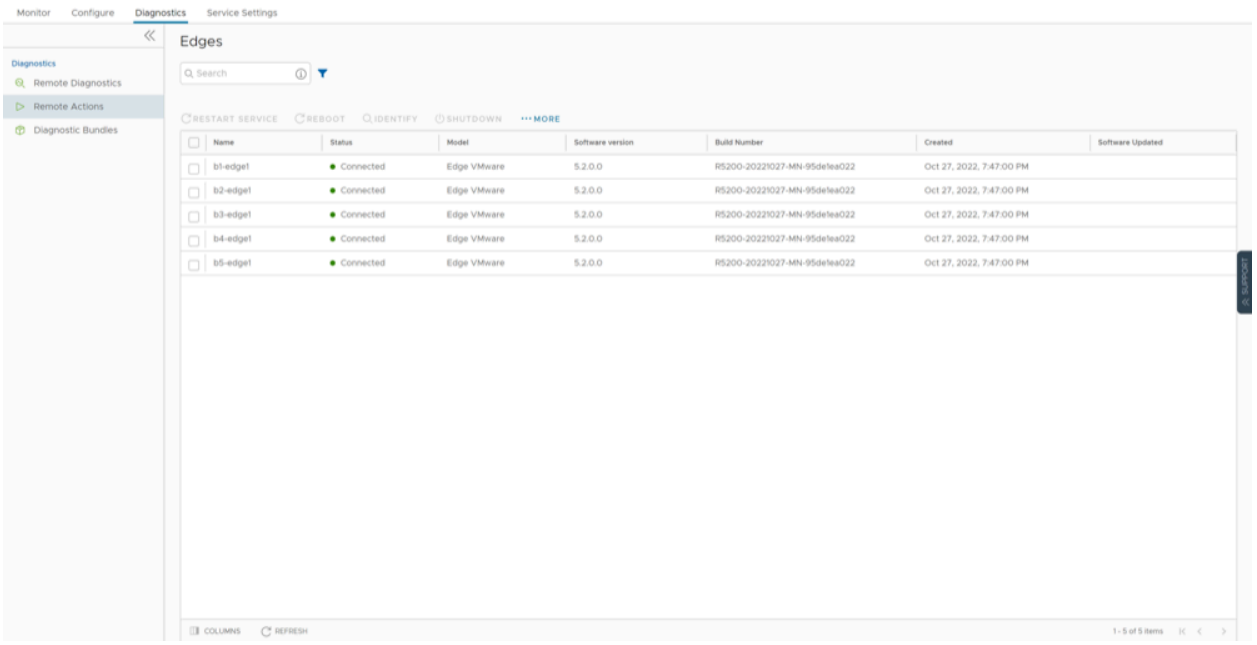
Remote Actions

You can perform actions like Restarting services, Rebooting, or deactivating an Edge remotely, from the Enterprise portal.

Note: You can perform the remote actions only on Edge that are in **Connected** state.

1. In the **SD-WAN** service of the **Enterprise** portal, you can perform remote actions from the **Diagnostics** > **Remote Actions** > **Edges** navigation path.

   Figure 6-1: Remote Actions



2. Select an Edge and perform any of the following remote actions:

   Table 12: Remote Actions- Options and Descriptions

Action	Description
Restart Service	Restarts the SD-WAN services on the selected Edge.
Reboot	Reboots the selected Edge.
Identify	Randomly flashlights on the selected Edge to identify the device.
Shutdown	Powers off the selected Edge. To restore the Edge, you must remove the power cable, and then plug it back into the Edge.
Deactivate	Resets the device configuration to its factory default state.
Force HA Failover	Forces HA Failover. This option is available only when the Edge is configured with High Availability and the state is HA ready.

3. You can also perform the remote actions for an Edge using the **Shortcuts** option available in the **Configure** > **Edges** or **Monitor** > **Edges** pages.
4. Select the **Shortcuts** > **Remote Actions** and perform any of the actions listed in the above table.

Note: The actions may take up to a minute to run on the device.

Chapter 7

Diagnostic Bundles for Edges

Diagnostic bundles allow Operator users to collect all the configuration files and log files into a consolidated Zipped file. The data available in the diagnostic bundles can be used for debugging purposes.

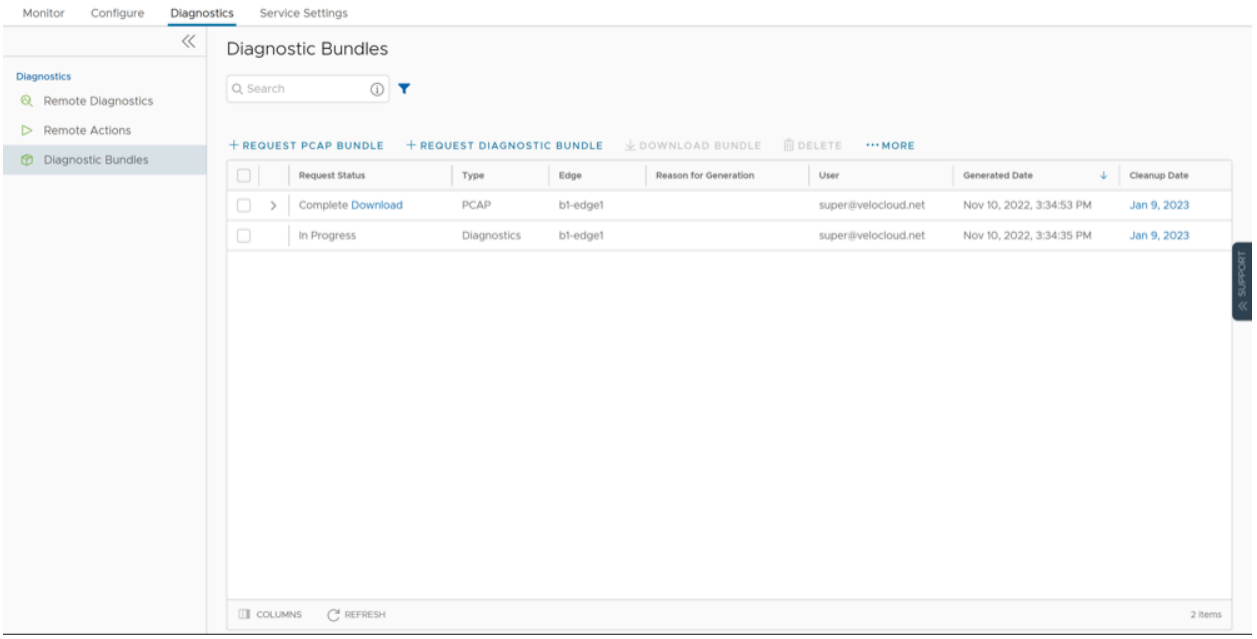
To generate and download Diagnostic Bundles:

1. In the **SD-WAN** service of the **Enterprise** portal, select the **Diagnostics** tab.
2. Select **Diagnostic Bundles** to request the following bundles:
   - **Request PCAP Bundle** – The Packet Capture bundle is a collection of the packet data of the network. Operators, Standard Admins and Customer Support can request PCAP bundles. For additional information, see Request Packet Capture Bundle for Edges.
   - **Request Diagnostic Bundle** – The Diagnostic bundle is a collection of all the configuration and logs from a specific Edge. Only Operators can request Diagnostic bundles. For additional information, see Request Diagnostic Bundle for Edges.

> **Note:** The **Request Diagnostic Bundle** option is available only for an Operator user. If you are a Partner user or an Enterprise user, you can request for a PCAP Bundle.

The generated bundles are displayed in the **Diagnostic Bundles** window.

Figure 7-1: Diagnostic Bundles



3. To download the details of generated bundles, select **More** > **Download CSV**. The details are downloaded in a CSV file.

7.1 Request Packet Capture Bundle for Edges

The Packet Capture bundle collects the packets data of a network. These files are used in analyzing the network characteristics. You can use the data for debugging an Edge device.

To generate a PCAP bundle:

1. In the **SD-WAN** service of the **Enterprise** portal, select the **Diagnostics** tab.
2. Select **Diagnostic Bundles** > **Request PCAP Bundle**.
3. In the **Request PCAP Bundle** window that appears, configure the following:

Figure 7-2: Request PCAP Bundle

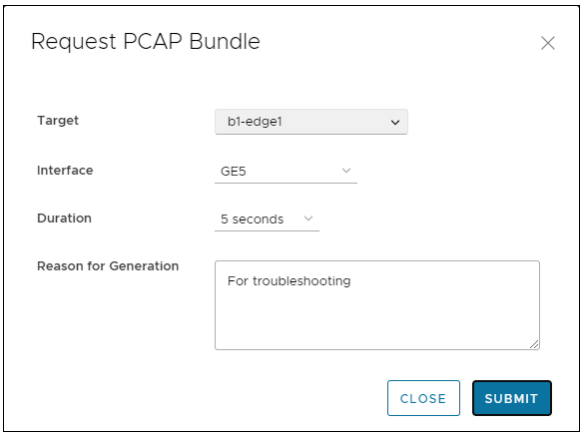


Table 13: Request PCAP Bundle- Options and Descriptions

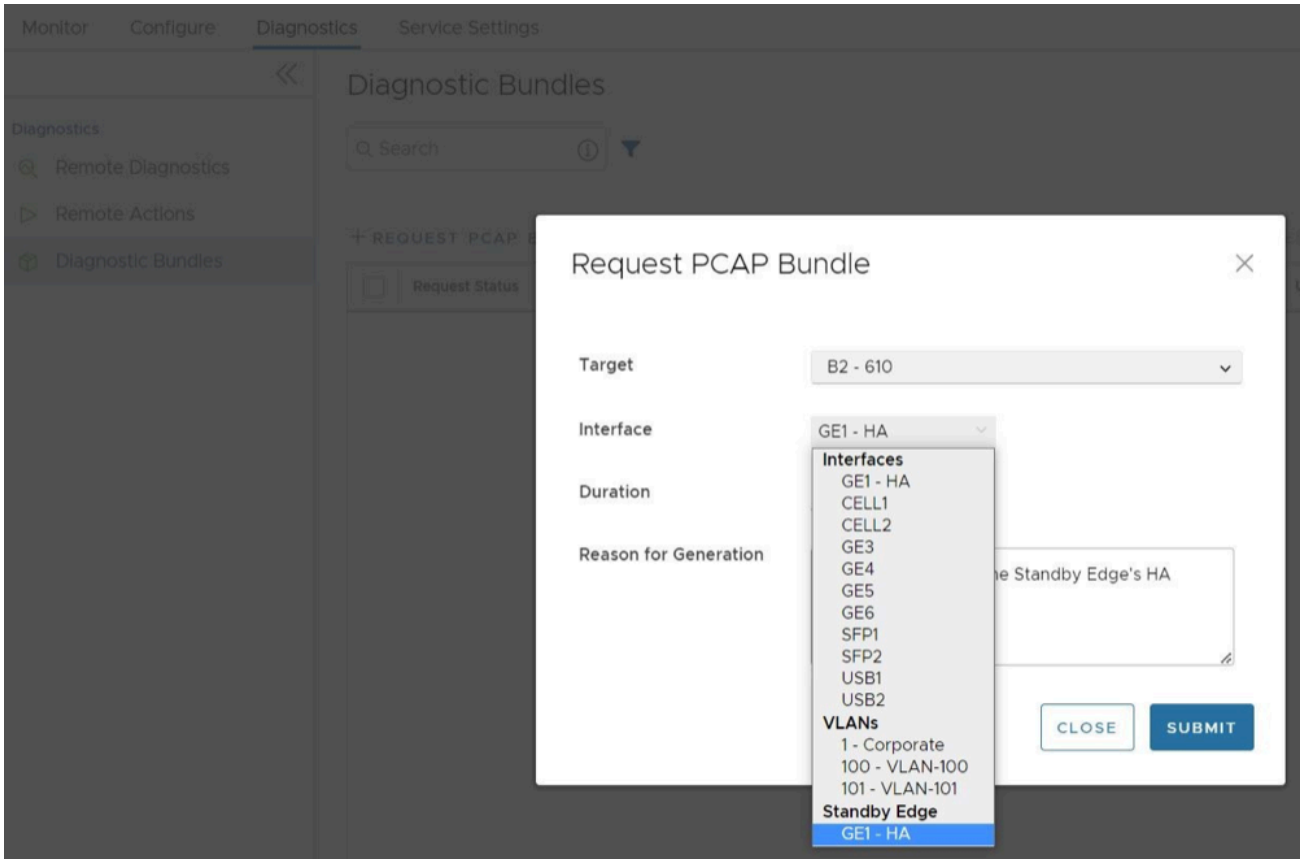
Option	Description
Target	Choose the target Edge from the drop-down list. The packets are collected from the selected Edge.
Interface	Choose an Interface or a VLAN from the drop-down list. The packets are collected on the selected Interface.
Duration	Choose the time in seconds. The packets are collected for the selected duration.
Reason for Generation	Optionally, you can enter your reason for generating the bundle.

The window displays the details of the bundle being generated, along with the status.

Packet Capture for Edges configured for High Availability

In Release 5.2.0 and later, a user can request a packet capture for the Standby Edge's HA interface, the interface that connects the Standby Edge to the Active Edge. This option appears at the bottom of the menu and reads: **Standby Edge**, and then lists the HA interface.

Figure 7-3: Request PCAP Bundle for HA Edge



7.2 Request Diagnostic Bundle for Edges

A Diagnostic bundle is a collection of configuration files, logs, and related events from a specific Edge.

To generate a Diagnostic bundle:

1. In the **SD-WAN** service of the **Enterprise** portal, select the **Diagnostics** tab.
2. Select **Diagnostic Bundles** > **Request Diagnostic Bundle**.

3. In the **Request Diagnostic Bundle** window, configure the following:

Figure 7-4: Request Diagnostic Bundle

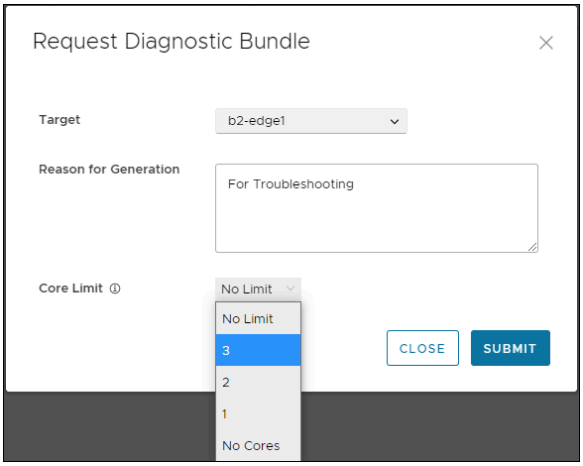


Table 14: Request Diagnostic Bundle- Options and Descriptions

Option	Description
Target	Select the target Edge from the drop-down list. The data is collected from the selected Edge.
Reason for Generation	Optionally, you can enter your reason for generating the bundle.
Core Limit	Select a Core Limit value from the drop-down, which is used to reduce the size of the uploaded bundle when the Internet connectivity is experiencing issues.

The **Diagnostic Bundles** window displays the details of the bundle being generated, along with the status.

4. To download the generated Diagnostic bundles:
   a. In the **Diagnostic Bundles** window, select the **Complete** link or select the bundle and select **Download Bundle**.
      The bundle is downloaded as a ZIP file.
   b. For troubleshooting purpose, you can send the downloaded bundle to a Arista Support representative for debugging the data.
5. The completed bundles get deleted automatically on the date displayed in the **Cleanup Date** column.
   a. To change the cleanup date, select the link to the cleanup date or choose the bundle and select **More** > **Update Cleanup Date**.
   b. In the **Update Cleanup Date** window, choose the date on which the selected bundle should be deleted.
   c. If you want to retain the bundle, select the **Keep Forever** option, so that the bundle does not get deleted automatically.
   d. To delete a bundle manually, select the bundle and select **Delete**.

Chapter 8

Diagnostic Bundles for Gateways

Run diagnostics for Gateways to collect diagnostic bundles and packet capture files for troubleshooting purpose.

- Request Diagnostic Bundles for Gateways
- Request Packet Capture Bundle for Gateways

8.1 Request Diagnostic Bundles for Gateways

Diagnostic bundles allow users to collect all the configuration files and log files from a specific VeloCloud Gateway into a consolidated zipped file. The data available in the diagnostic bundles can be used for troubleshooting the SD-WAN Gateways.

As an Operator Superuser and Operator Admin user, you can create, manage, download, and delete diagnostic bundles for Gateways created by both Operator and Partner users.

Note: Operator Business Specialist user and Operator IT support users can only view the generated Diagnostic bundles and download the CSV file.

Partner Super user and Admin with Gateway management access activated can create, manage, and delete diagnostic bundles only for Gateway created by a Partner or a Partner managed Gateway created by your Operator. The Partner IT support users can only view the generated Diagnostic bundles and download the CSV file.

Note:

- The Diagnostic bundles feature is not supported for Partner Business Specialist user.
- The **Request Diagnostic Bundle** and **Download Bundle** options are available only for Partners with Gateway management access activated. If the Gateway management access is deactivated for a Partner, then the Partner can only view the generated Diagnostic bundles and download only the CSV file but cannot request a new Diagnostic bundle or download the generated bundle. To request Gateway Management access, Partners should contact the Operator Superuser.

To generate a new Diagnostic bundle:

1. In the **Operator** portal, select the **Gateway Management** tab and select **Diagnostic Bundles** in the left navigation pane.

   The **Diagnostic Bundles** page appears with the existing diagnostic bundles.
2. To generate a new Diagnostic bundle, select **Request Diagnostic Bundle**.

3. In the **Request Diagnostic Bundle** dialog, configure the following details and select **Submit**.

Figure 8-1: Request Diagnostic Bundle

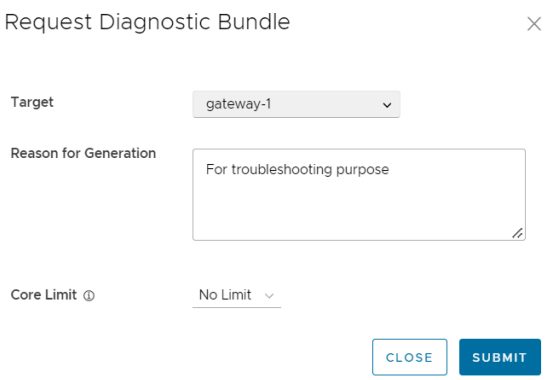


Table 15: Request Diagnostic Bundle- Options and Descriptions

Option	Description
Target	Select the target Gateway from the drop-down list. The data is collected from the selected Gateway.
Reason for Generation	Optionally, you can enter your reason for generating the bundle.
Core Limit	Select a Core Limit value from the drop-down, which is used to reduce the size of the uploaded bundle when the Internet connectivity is experiencing issues.

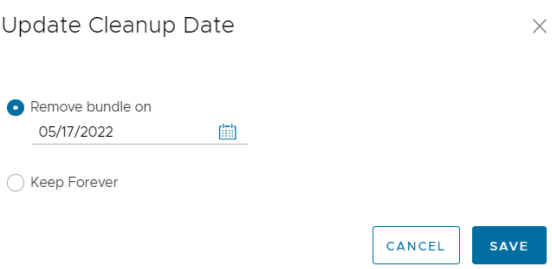
The **Diagnostic Bundles** page displays the details of the bundle being generated, along with the status.

4. To search a specific diagnostic bundle, enter a relevant search text in the **Search** box. For advanced search, select the **Filter** icon next to the **Search** box to filter the results by specific criteria.
5. You can download the generated Diagnostic bundles to troubleshoot an Edge.
   a. To download a generated bundle, select the link next to **Complete** in the **Request Status** column or select the bundle and select **Download Bundle**.

      The bundle is downloaded as a ZIP file.

   b. You can send the downloaded bundle to a Arista Support representative for debugging the data.
6. The completed bundles get deleted automatically on the date displayed in the **Cleanup Date** column.
   a. You can select the link to the **Cleanup Date** or choose the bundle and select **More** > **Update Cleanup Date** to modify the Date.

      Figure 8-2: Update Cleanup Date



   b. In the **Update Cleanup Date** dialog, choose the date on which the selected Bundle would be deleted.

c. If you want to retain the Bundle, select the **Keep Forever** option, so that the Bundle does not get deleted automatically.

d. To delete a bundle manually, select the bundle and select **Delete**.

8.2 Request Packet Capture Bundle for Gateways

The Packet Capture bundle collects the packets data of a network. These files are used in analyzing the network characteristics. You can use the data for debugging the network traffic and determining network status.

To generate a PCAP bundle:

1. In the **Operator** portal, select the **Gateway Management** tab and select **Diagnostic Bundles** in the left navigation pane.

   The **Diagnostic Bundles** page appears with the existing diagnostic bundles.
2. To generate a new PCAP bundle, select **Request PCAP Bundle**.
3. In the **Request PCAP Bundle** dialog, configure the following details and select **Generate**.

Figure 8-3: Request PCAP Bundle

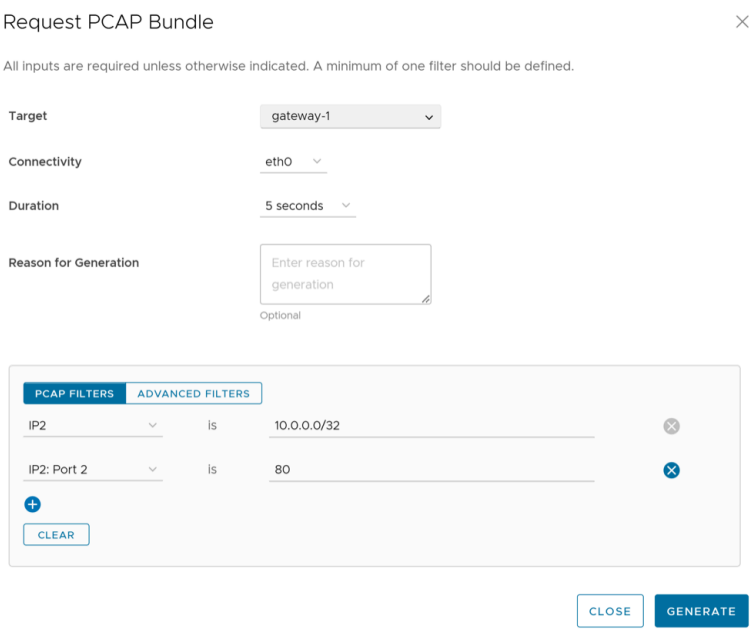


Table 16: Request PCAP Bundle- Options and Descriptions

Option	Description
Target	Choose the target Gateway from the drop-down list. The packets are collected from the selected Gateway.
Connectivity	Choose an Interface or an Edge ID from the drop-down list. The packets are collected on the selected Interface or Edge associated to the Gateway.
Duration	Choose the time in seconds. The packets are collected for the selected duration. The default value is 5 seconds.
Reason for Generation	Optionally, you can enter your reason for generating the bundle.
PCAP Filters	You can define PCAP filters by which you want to control the PCAP data to be generated by choosing the following options: • **IP1**- Enter an IPv4 address, or IPv6 address, or Subnet mask. • **IP2**- Enter an IPv4 address, or IPv6 address, or Subnet mask. • **IP1:Port1**- Enter a Port ID associated with IP1. • **IP2:Port2**- Enter a Port ID associated with IP2. • **Protocol**- Select a protocol from the list. **Note:** If you choose to use the PCAP filtering capability then you must define at least one filter.
Advanced Filters	You can define free form filters by which you want to control the PCAP data to be generated.

The **Diagnostic Bundles** page displays the details of the PCAP bundle being generated, along with the status.

4. To download a generated bundle, select the link next to **Complete** in the **Request Status** column or select the bundle and select **Download Bundle**.

   The bundle is downloaded as a ZIP file.

5. The completed bundles get deleted automatically on the date displayed in the **Cleanup Date** column.

   a. You can select the link to the **Cleanup Date** or choose the bundle and select **More** > **Update Cleanup Date** to modify the Date.

   b. To delete a bundle manually, select the bundle and select **Delete**.

Chapter 9

References

9.1 Related Documents

The following documentation is available for ***Arista VeloCloud SD-WAN***:

- *Arista VeloCloud SD-WAN Operator Guide*
- *Arista VeloCloud SD-WAN Administration Guide*
- *Arista VeloCloud SD-WAN Orchestrator Deployment and Monitoring Guide*
- *Arista VeloCloud SD-WAN Gateway Monitoring Guide*
- *Arista VeloCloud SD-WAN Partner Guide*
- *Arista VeloCloud SASE Global Settings Guide*
- *Arista VeloCloud SD-WAN Design Guide for Enhanced Firewall Services*
- *Arista VeloCloud SD-WAN API*
- *Arista VeloCloud Portal API*