ERICSSON

Troubleshooting L3VPNs

SmartEdge OS Software

FAULT TRACING DIRECT.

9/154 51-CRA 119 1170/1-V2 Uen B

Contents

1 Overview

A typical BGP/MPLS VPN topology consists of multiple customer sites connected to a service-provider network. Customer edge (CE) routers provide customer access to the service-provider network over a data link to one or more provider edge (PE) routers. The CE routers establish an adjacency with their directly connected PE routers, and the CE routers advertise IPv4 routes to the PE router. The CE routers also learn IPv4 routes from their PE routers. These IPv4 routes only become VPNv4 routes once they enter the provider backbone.

In the SmartEdge® implementation, PE routers maintain a separate VPN context for each private network. Connections to CE routers are bound to the appropriate context. Access to the service provider core is through the local context in each PE router. Because the VPN runs from private VPN context to private VPN context, the customer can have visibility into the entire network, including the private context inside the SmartEdge router, without having any visibility into the public space or to other private contexts.

PE routers can be directly connected, or can be connected through provider (P) routers. P routers have no visibility into private networks; they simply provide connectivity from one PE router to another.

PE routers advertise VPN routes learned from CE routers across the service provider core by using Interior Border Gateway Protocol (iBGP). All iBGP features, including route reflectors, are available to ensure scalable iBGP connectivity across the service provider core. The PE routers use Label Distribution Protocol (LDP) or Resource Reservation Protocol (RSVP) to build label-switched paths (LSPs); the PE routers function as edge label-switching routers (LSRs), and each private network has its own set of LSPs. Multiprotocol Label Switching (MPLS) is then used to forward VPN data traffic across the provider's backbone.

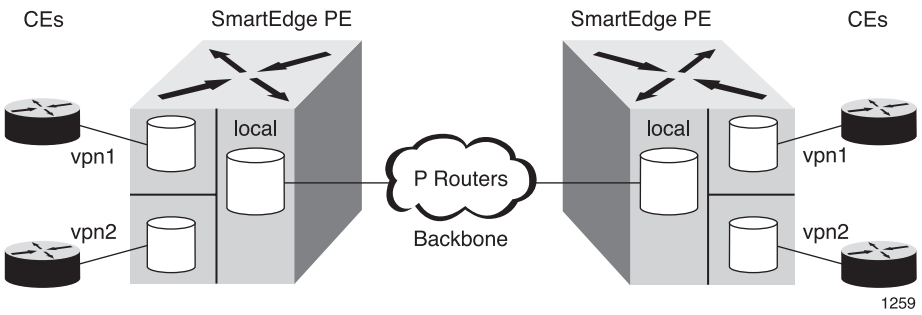

Figure 1 BGP/MPLS VPN Topology

An MPLS/BGP VPN has several components that must be operational for the VPN to function:

1. The provider network routers—–PE and P routers—must run either OSPF or IS-IS to support LDP or RSVP. The link-state routing protocol discovers the paths from PE router-to-PE router, which is used by LDP, a signaling protocol to build LSPs

2. PE routers configured as iBGP peers.

3. Routes from the private networks are transported by the provider network, and are associated with a Forwarding Equivalence Class (FEC). BGP then assigns a next-hop and an additional VPN label to the FEC.

   For every IP prefix in the local VPN, BGP notifies the remote VPN sites the label to attach to traffic destined to that prefix. When that traffic arrives from the remote end, the PE sends it to the next-hop given by the nexthop-label mapping.

   LSPs are then built using LDP or RSVP; the PEs function as edge LSRs, with MPLS providing the label-switching intelligence to transport VPN data across the provider backbone. For information about RSVP and LDP, see the *Configuring MPLS* document.

1.1 Troubleshooting L3VPNs

Use the following procedure as a guide to troubleshoot L3VPNs.

1 Verify the Connectivity in the PE Network

2 Verify the BGP Neighborship

3 Verify the PE VPN Configuration and CE Routes

4 Verify the BGP Routing Table in the VPN PE Context

5 Verify BGP VPN Routes in Context Local

6 Verify the Label Exchange between VPN Contexts

Note: If think you have a transient issue, start troubleshooting from the bottom layer.

To do this, check the following:

1 Ports

2 IGP routes and route up time.

3 Sessions (for example, LDP or BGP)

4 Bindings (for example, LDP bindings or VPN labels)

5 Statistics

2 Step 1: Verify the Connectivity in the PE Network

Use figure 2 as a guide to troubleshoot an end-to-end L3VPN setup.

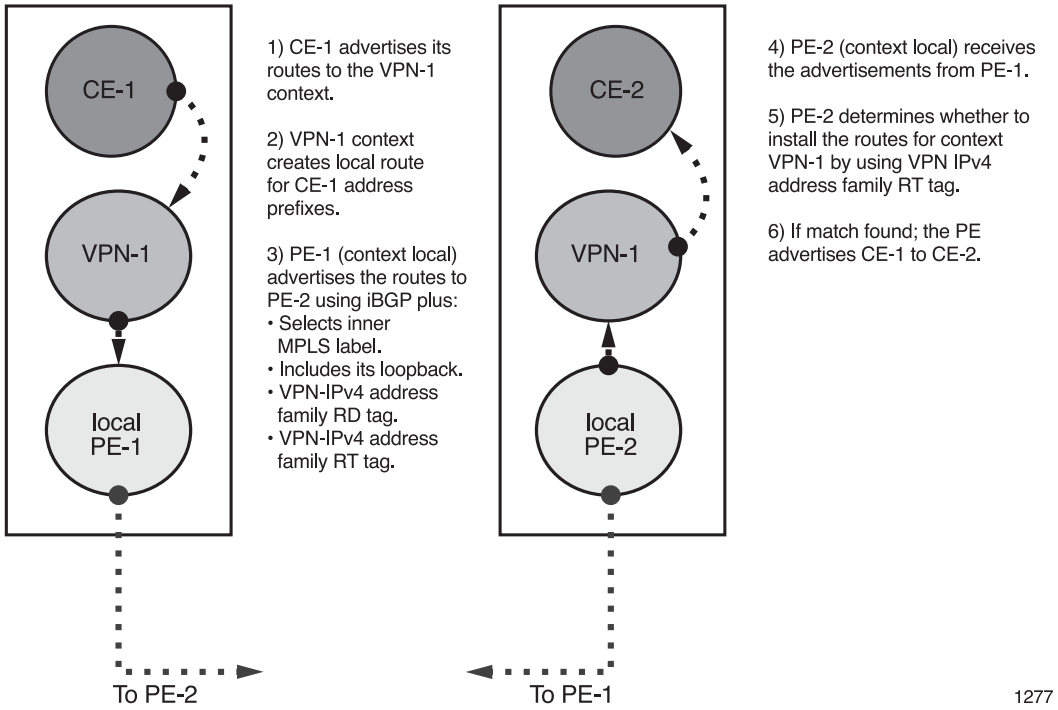

Figure 2 End-to-End L3VPN Setup Flowchart

For information about how to configuring L3VPNs, see the *Configuring BGP/MPLS VPN* document.

For information about RD and RT, see the *Configuring MPLS* document.

This document uses the sample topology in Figure 3 as a basis for troubleshooting routing and the label-switched path.

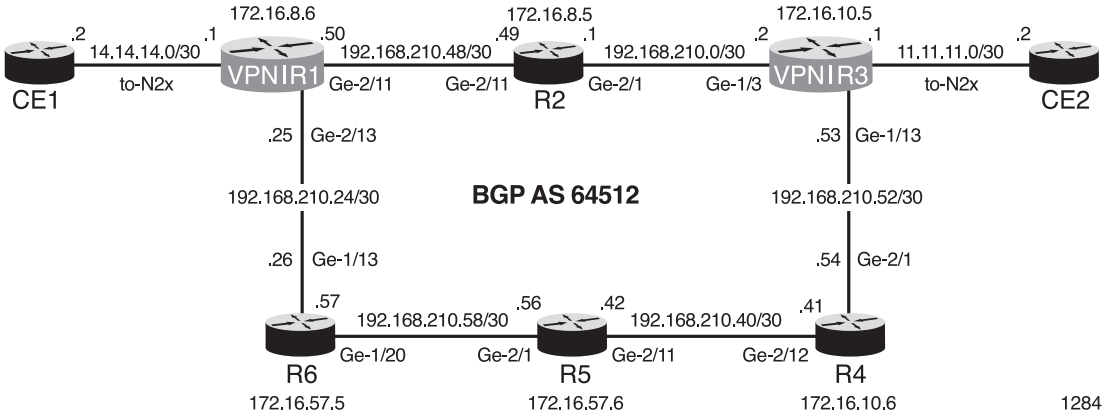

Figure 3 L3VPN Sample Topology

Use the following procedure to verify that you can provide L3VPN services to your CEs.

- Step A: Check the IP Connectivity
- Step B: Check the MPLS Connectivity
- Step C: Check the BGP Connectivity

2.1 Step A: Check IP Connectivity Among PE Routers

MPLS depends on the IP layer. When there is a problem in the IP layer, MPLS will not work. In the local context of each PE, check the IP connectivity among PEs:

1 Check the Interfaces

2 Ping the PE Peers Loopback Address and Check Connectivity

2.1.1 Check the Interfaces

Use the **`show ip interface brief`** command to make sure the interfaces are up. This command displays information about all interfaces, associated addresses, states, and bindings, including the interface bound to the Ethernet management port on the controller card.

For detailed information about each field displayed, see the *Command List*.

The following example displays output from the **`show ip interface brief`** command. Interfaces **`lo1`**, **`ge-2/11`**, and**`ge-2/13`** are functioning correctly.

```
[local]R1#show ip interface brief
Sat Mar 20 00:38:00 2010
Name              Address                  MTU   State    Bindings
ge-2/11           192.168.210.50/30        1500  Up       ethernet 2/11
ge-2/13           192.168.210.25/30        1500  Up       ethernet 2/13
lo1               172.16.8.6/32            1500  Up       (Loopback)


[local]R1#ping 192.168.210.49       <==  Ping router R2 physical interface
PING 192.168.210.49 (192.168.210.49): source 192.168.210.50, 36 data bytes,
timeout is 1 second
!!!!!                               <==  !!!!! Indicates a successful ping

----192.168.210.49 PING Statistics----
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.667/1.820/1.905/0.094 ms
```

Note: If the ping fails, check the status of the interface and check for configuration mismatches. For example, an ACL might prevent connectivity.

2.1.2 Ping and Traceroute the PE Peer Loopback Address

On the PE routers, use the **ping** and **traceroute** command to check IP connectivity (the route from IGP) with your PE peer's loopback address.

Note: Make sure the source address is the same one used by BGP and LDP. Ping from both ends.

```
[local]R1#ping 172.16.10.5 source 172.16.8.6 <== Ping router R3
PING 172.16.10.5 (172.16.10.5): source 172.16.8.6, 36 data bytes,
timeout is 1 second
!!!!!                                    <==  !!!!! Indicates a successful ping

----172.16.10.5 PING Statistics----
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.881/1.385/1.831/0.462 ms

[local]R1#traceroute 172.16.10.5       <==  Traceroute router R3
se_traceroute to 172.16.10.5 (172.16.10.5), 30 hops max, 40 byte packets
 1  192.168.210.49 (192.168.210.49)   2.969 ms  2.877 ms  2.715 ms
 2  172.16.10.5 (172.16.10.5)   3.796 ms  2.961 ms  3.904 ms        <==  Successful traceroute
```

Note: When the ping fails, check the underlying IGP (OSPF or ISIS), MPLS and label protocol (LDP or LDP with RSVP). For information about troubleshooting MPLS, see *Troubleshooting MPLS*.

2.2 Step B. Check MPLS Connectivity

On the PE routers, in the local context, validate MPLS connectivity. Can you ping MPLS from the local PE loopback interface by using the **ping mpls [ldp | rsvp]** ***ip-address*** **source** ***ip-address***, from its local context, to the remote PE loopback address? For information about how to check MPLS connectivity, see the *Troubleshooting MPLS* document.

2.3 Step C. Check BGP Connectivity in the Local Context

A BGP session is a TCP connection in which routing information is exchanged using the BGP protocol. If you have problems with BGP connectivity, check your TCP connection.

Use the **`show tcp`** command on the PE router to verify that you can establish a TCP connection (port 179) to the iBGP neighbors that you want to communicate with. If you cannot establish a TCP connection:

- Check the ports and source and destination addresses. Do they match the configuration?
- Check if there a firewall or access list blocking the port. If so, test the port by using telnet to contact the remote IP address.

The following example shows that router `R1` has successfully established a TCP connection with its BGP neighbor, router `R3`.

```
[local]R1#show tcp

Active Internet connections
PCB      Recv-Q Send-Q Local Address          Foreign Address        State
18d3f0d0      0      0 172.16.8.6.179        172.16.10.5.56526      ESTABLISHED
<== R1 established a connection with BGP neighbor, router R3


 b2e4a28      0      0 10.12.225.6.23        155.53.112.47.53776    ESTABLISHED
 18d3f008     0      0 172.16.8.6.54744      172.16.57.6.179        ESTABLISHED
 b2e4e10      0      0 172.16.8.6.65260      172.16.10.6.179        ESTABLISHED
 b2e4ed8      0      0 172.16.8.6.49761      172.16.8.5.179         ESTABLISHED
 b2e4d48      0      0 172.16.8.6.57376      172.16.57.5.179        ESTABLISHED
 b2e4af0      0      0 127.0.2.5.7500        *.*                    LISTEN
 b2e4960      0      0 127.0.2.5.52354       127.0.2.3.6667         ESTABLISHED
 b2e4898      0      0 127.0.2.5.51399       *.*                    LISTEN
 b2e4708      0      0 127.0.2.5.65106       127.0.2.3.6667         ESTABLISHED
 b2e4640      0      0 127.0.2.5.51346       127.0.2.3.6666         ESTABLISHED
 b2e4578      0      0 127.0.2.5.52537       127.0.2.3.6666         ESTABLISHED

IP Path MTU discovery is disabled
TCP keep-alive idle = 14400
TCP keep-alive interval = 150
TCP keep-alive count = 8
TCP in persist will die if > 600 seconds and free sys mem < 50MB
```

3 Step 2: Verify the BGP Neighborship

Use Table 1 as a guide to verify BGP neighborship.

Table 1 Tasks to Troubleshoot BGP Neighborship

Task	Command	Notes	Checked?
Step A: Verify the BGP Configuration in the Local Context	**`show configuration bgp`**	All commands are issued from the local context.	
Step B: Display a Summary of BGP Neighbors	**`show bgp summary`**		
Step C: Check the State of BGP Neighbors	**`show bgp neighbor summary`**		
Step D: Check the State of a Specific Neighbor	**`show bgp neighbor`** ***`address`***		
Step E: Display the BGP Neighbor Reset Information	**`show bgp reset-log`**		
Step F: Debug BGP	For a list of **`debug bgp`** commands, see Section 3.6 on page 17.		

3.1 Step A: Verify the BGP Configuration in the Local Context

Use the **`show configuration bgp`** command to verify the BGP configuration in the local context.

Check the following:

- AS number
- Peer IP address

```
[local]R1#show configuration bgp
Building configuration...

Current configuration:
context local
context local
!
 router bgp 64512                    <== AS number
  router-id 172.16.8.6
 timers keepalive 5 holdtime 15
  maximum restart-time 180
  maximum retain-time 300
  address-family ipv4 unicast
   redistribute connected
!
  peer-group PBN_INT internal
   update-source lo1
   address-family ipv4 unicast
   address-family ipv4 vpn
!
  neighbor 172.16.8.5 internal
   peer-group PBN_INT
!
  neighbor 172.16.10.5 internal      <== Peer IP address (router R3)
   peer-group PBN_INT
!
  neighbor 172.16.10.6 internal
   peer-group PBN_INT
!
  neighbor 172.16.57.5 internal
   peer-group PBN_INT
!
  neighbor 172.16.57.6 internal
   peer-group PBN_INT
!
! ** End Context **
!
```

3.2 Step B: Display a Summary of BGP Neighbors

Use the **`show bgp summary`** command to display BGP neighbor summary information.

The following example shows two neighbors that have established a connection. If active, the `PfxRcvd/Sent` field can have states that are active, connect, or Idle. If everything works correctly, you should see something similar to the following:

- The BGP output shows that BGP neighbors have established neighborship with router `R1`.
- The `Up/Down` column shows the length of time that BGP neighborship was established with router `R1`.
- If active, the `PfxRcvd/Sent` column shows the number of iBGP routes (`unicast` and `vpn`) exchanged between each BGP peer with router `R1`.

  When the this column has a state of `connect` or `idle`, the BGP session has not been established.

In the following example:

- Router `R1`, with the IP address, `172.16.8.6`, sent 6 routes to router `R3`, which has IP address `172.16.10.5`.
- Router `R1` received 5 routes from router `R3`, which has IP address `172.16.10.5`.

```
[local]R1#show bgp summary
Address Family: ipv4 unicast
BGP router identifier: 172.16.8.6, local AS number: 64512
BGP route table version: 48, RIB table version: 48, deleted vers: 48
Neighbors Configured: 5, Established: 5
Sourced paths: redistributed: 6, networked: 0, aggregated: 0
Dampening: Disabled
Flap-statistics: Disabled

Entry Type      Count          Memory
Network            21            1828
Path               33            1056

Neighbor              AS  MsgRcvd  MsgSent    TblVer  InQ OutQ Rst  Up/Down  PfxRcvd/Sent
172.16.8.5         64512      265      297        48    0    0   0 00:24:59      6       6
172.16.10.5        64512      102       86        48    0    0   1 00:06:40      5       6  <== Router R3
172.16.10.6        64512      367      304        48    0    0   0 00:25:32      5       6
172.16.57.5        64512      858      955        48    0    0   0 01:21:18      6       6
172.16.57.6        64512      291      295        48    0    0   0 00:24:47      5       6
```

Recommended Action: When you find an issue, and you have already checked IP and TCP connectivity, use the **`debug bgp event`** command to isolate the fault.

3.3 Step C: Check the State of BGP Neighbors

Use the **`show bgp neighbor summary`** command to verify the state of your BGP neighbors.

Table 2 BGP Neighbor State Checklist

Task	Checked?
Are all your BGP neighbors coming up— is their state Established? When the neighbor is not in Established state, ping the neighbor loopback address to check IP connectivity.	
Is the neighbor loopback address used as a peering address and that the own loopback address is used as the source for BGP updates?	
Do the `CapSent` and `CapRecv` fields both contain `4bytesAS`, `vpn`, and `restart`?	

The following example displays BGP neighbor summary information from router `R1`. Here, your peers have successfully established BGP sessions, such as with router `R3` with IP address `172.16.10.5`.

```
[local]R1#show bgp neighbor summary
BGP router identifier: 172.16.8.6, local AS number: 64512
Neighbors Configured: 5, Established: 5

Neighbor                 AS MsgRcvd MsgSent  InQ OutQ Rst  Up/Down State
172.16.8.5            64512     259     291    0    0   0 00:24:30 Established  <== Router R2
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS restart unicast vpn
  unicast  : rcvd: 6 imported: 0 active: 3 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 4 imported: 4 active: 6 history: 0 dampened: 0 sent: 7

172.16.10.5           64512      94      81    0    0   1 00:06:11 Established  <== Router R3
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS restart unicast vpn
  unicast  : rcvd: 5 imported: 0 active: 3 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 1 imported: 1 active: 2 history: 0 dampened: 0 sent: 7

172.16.10.6           64512     360     298    0    0   0 00:25:03 Established  <== Router R4
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS restart unicast vpn
  unicast  : rcvd: 5 imported: 0 active: 2 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 1 imported: 1 active: 2 history: 0 dampened: 0 sent: 7

172.16.57.5           64512     852     949    0    0   0 01:20:49 Established  <== Router R6
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS restart unicast vpn
  unicast  : rcvd: 6 imported: 0 active: 4 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 2 imported: 2 active: 4 history: 0 dampened: 0 sent: 7

172.16.57.6           64512     285     289    0    0   0 00:24:18 Established  <== Router R5
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS restart unicast vpn
  unicast  : rcvd: 5 imported: 0 active: 3 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 1 imported: 1 active: 1 history: 0 dampened: 0 sent: 7
```

3.4 Step D: Check the State of a Specific BGP Neighbor

Use the **`show bgp neighbor`** command to check the state of a specific BGP neighbor.

The following example displays information for BGP neighbor router `R2` with IP address `172.16.8.5`, which has an expired hold timer.

```
[local]R1#show bgp neighbor 172.16.18.5                          <== Router R2
BGP neighbor: 172.16.8.5, remote AS: 64512, internal link
  Version: 4, router identifier: 172.16.8.5
  Peer Group member: PBN_INT
  State: Established for 00:01:27
  Last read 00:00:00, last send 00:00:00
  Hold time: configured 15, negotiated 5
  Keepalive time: configured 5, negotiated 1
  Local restart timer 180 sec, stale route retain timer 300 sec
  Received restart timer 180 sec, flag 0x0
  Minimum time between advertisement runs: 5 secs
  Source IP address used from interface: lo1
  Source (local) IP address: 172.16.8.6
  Received messages: 93 (2013 bytes), notifications: 2, in queue: 0
  Sent messages: 98 (2542 bytes), notifications: 2, out queue: 0
  Reset count: 5, last reset time: 00:01:36, reset reason: Notification recv (hold time expired)

                                                                 <== Check port and bindings

 CapSent: refresh, 4byteAS, unicast, vpn, restart
  CapRcvd: refresh, 4byteAS, unicast, vpn
           restart (time 180, flags 0x0, unicast, vpn)

  Address family: ipv4 unicast
    Peer Group member: PBN_INT
    BGP table version: 67, neighbor version: 67
    Routes: rcvd 6, imported 0, active 3, history 0, dampend 0, sent 6
```

Note: When the hold timer expires, check the port status and bindings by using the **`show port`** and **`show bindings`** commands.

3.5 Step E: Display the BGP Neighbor Reset Information

Use the **show bgp reset-log** command to display a summary of reset reasons for your BGP neighbors. The following table lists BGP neighbor reset reasons:

Table 3 BGP Reset Reasons

BGP Error Message	Description
Notification received (ceased: peer unconfigured)	The BGP neighbor is not configured to accept BGP messages . **Recommended Action**: Check the configuration by using the **show bgp configuration** command.
Notification sent (hold time expired)	The router missed 3 keep alive messages within 180 seconds and tears down the BGP session. **Recommended Action**: Check the port and bindings by using the **show port** and **show bindings** commands.
Remote/TCP close	The BGP neighborship was lost due to TCP failure.
User action	The user manually restarted the BGP neighbor.

In the following example, the **show bgp reset-log** output shows that the BGP neighborship was lost due to a TCP failure and the hold timer has expired.

```
[local]R1#show bgp neighbor 172.16.8.5 reset-log
Dump neighbor reset logs for ??? (6 total entries):
Neighbor   Context  StartTime       EndTime         Count Reason
???        40080001 Mar 22 22:50:50 Mar 22 22:50:50     1 Notification sent (hold time expired)
???        40080001 Mar 23 00:29:32 Mar 23 00:29:32     1 Remote/TCP close
???        40080001 Mar 23 00:34:07 Mar 23 00:34:07     1 Notification sent (hold time expired)
???        40080001 Mar 23 00:46:26 Mar 23 00:46:26     1 Notification recv (hold time expired)
???        40080001 Mar 23 01:45:41 Mar 23 01:45:41     1 Notification sent (hold time expired)
???        40080001 Mar 23 01:58:01 Mar 23 01:58:01     1 Notification recv (hold time expired)
```

Note: When the hold timer expires, check the port status and bindings by using the **show port** and **show bindings** commands.

When the neighboring BGP peers have not established a peering relationship, run the **show bgp notification** command to determine why the BGP peers are down. In the following example, the notification indicates that the hold timer has expired.

```
[local]#show bgp notification
Dump notification messaged logged:
Nov 9 00:36:03 notification msg received (nbr 192.168.3.7, 21 bytes,
repeated 0 times, code 4/0 (hold time expired) -                    <== Check port status and bindings.
ffff ffff ffff ffff ffff ffff ffff ffff 0015 0304 00
Nov 9 00:36:23 notification msg received (nbr 192.168.41.7, 21 bytes,
repeated 0 times, code 4/0 (hold time expired) -
ffff ffff ffff ffff ffff ffff ffff ffff 0015 0304 00
```

3.6 Step F: Debug BGP

Use the **debug** commands in Table 4 to debug BGP. Most **debug** commands, such as **debug bgp rib**, only show results when the corresponding event is triggered again—for example, a download or next-hop query.

Table 4 BGP Debug Commands

Command	Description
debug bgp event	Generates debug messages for BGP events.
debug bgp listen	Generates debug messages for incoming connections
debug bgp neighbor *address* message	Generates debug messages for BGP neighbors.
debug bgp neighbor *address* update	Generates debug messages for BGP updates.
debug bgp neighbor *address* session-state	Generates debug messages for BGP sessions.

Caution!

Risk of performance loss. Generating debug messages can severely affect system performance. Exercise caution when generating debug messages on a production system.

3.6.1 Debug BGP Events

The following example on router R1 shows an AS byte length configuration that does not match its BGP peer, router R2 with IP address 172.16.8.5.

```
[local]R1#debug bgp event
Jan 30 18:47:17: [0001]: %BGP-6-INFO: 172.16.8.5 rcv NOTIFICATION:
2/2 (open: bad peer AS) with 4 byte data  <==The AS byte length does not match its BGP peer, router R2.

Jan 30 18:47:17: [0001]: %BGP-7-EVENT: 172.16.8.5 completed reset
```

Recommended Action: Make sure the local and remote BGP peer have the same AS byte length.

3.6.2 Debug BGP Neighbor Messages

The following example shows successful keepalive debug messages for BGP neighbor router R2 with IP address 172.16.8.5.

```
[local]R1#debug bgp neighbor 172.16.8.5 message    <== Debug BGP neighbor (router R2)
[local]R1#terminal monitor
[local]R11#Nov 18 17:33:00: [0001]: %BGP-7-MESSAGE: 172.16.8.5 rcv KEEPALIVE
Nov 18 17:33:02: [0001]: %BGP-7-MESSAGE: 172.16.8.5 sent KEEPALIVE
```

3.6.3 Debug BGP Update Messages

The following example shows update messages for BGP neighbor router `R2` with IP address `172.16.8.5`.

```
[local]R1#debug bgp neighbor 172.16.8.5 update
[local]R1#terminal monitor
[local]R11#Nov 18 17:33:00: [0001]: %BGP-7-MESSAGE: 172.16.8.5 rcv KEEPALIVE
Nov 18 17:33:02: [0001]: %BGP-7-MESSAGE: 172.16.8.5 sent KEEPALIVE
```

Note: If you do not receive a keepalive message from the BGP neighbor, ping the neighbor.

3.6.4 Debug the BGP Session

Use Table 5 as a guide for troubleshooting BGP sessions.

Table 5 BGP Sequence

State	Recommended Action
1. Idle state	Ping the BGP neighbor to check connectivity.
2. Connect	Check TCP (port connection between peers); ping neighbors.
3. OpenSent	Check the BGP level. • The BGP neighbor might not be configured for your IP address, or the peer has an incorrect IP address. • You may have configured your neighbor's loopback address correctly, but you are sending BGP messages are on the interface address and not on the configured loopback address.
4. OpenConfirmed	Not Applicable
5. Established	Not Applicable

The following example displays a BGP session that has not established a connected route.

```
[local]R1# debug bgp neighbor 172.16.8.5 session-state
[local]R1#terminal monitor
Jan 30 18:21:30: [0001]: %BGP-7-SESSION: 172.16.8.5 active open: start
Jan 30 18:21:30: [0001]: %BGP-7-SESSION: 172.16.8.5 active open: no connected route   <== Check configuration
```

Recommended Action: When you receive a `no connected route` message:

1. Check the local and iBGP neighbor configuration by issuing the **`show configuration bgp`** command.

2. Verify that the interfaces are up by issuing the **`show ip interfaces`** command.

3. Ping the BGP neighbor by using the **`ping`** command. Specify the BGP router ID as the source address.

4 Step 3: Verify the PE VPN Configuration and CE Routes

Use figure 4 as a guide to troubleshooting the packet exchange among CE peers.

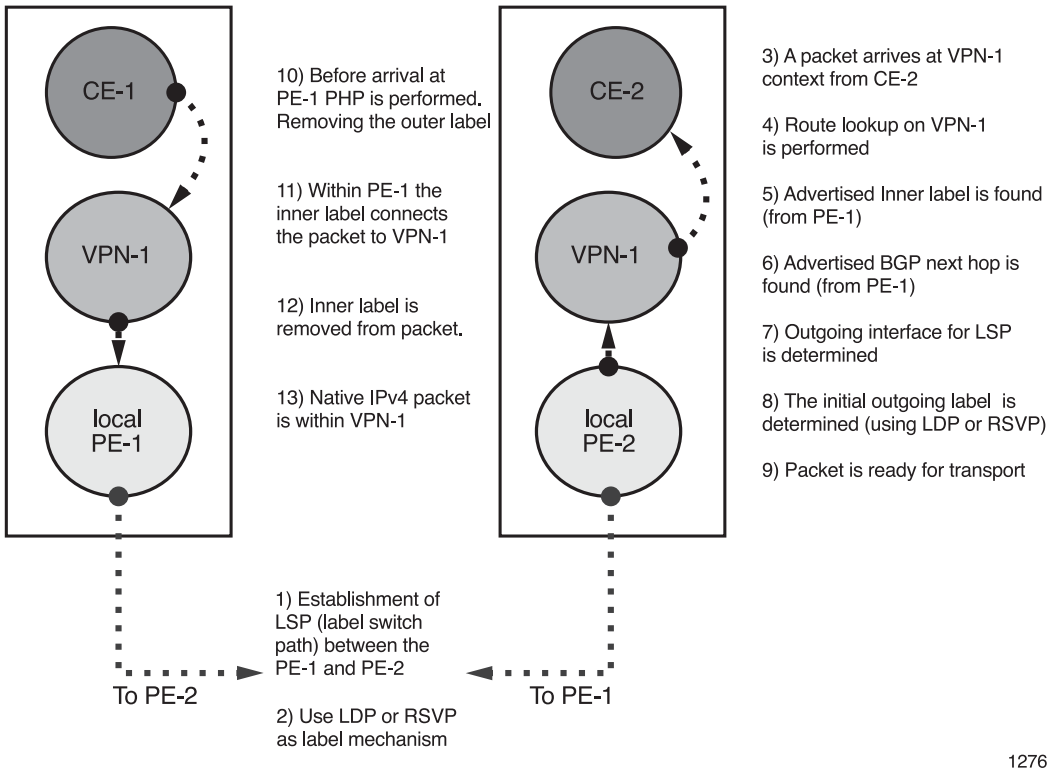

Figure 4 Packet Exchange between CE Peers Flowchart

Use the following procedure to verify PE VPN configuration and CE routes.

a Verify Context

b Verify Configuration

c Verify IP Routing Table in VPN Context

d Verify BGP Routing Table in the VPN Context

4.1 Step A: Verify the Context

Use the **`show context all`** and **`show configuration context`** commands to display and verify all the BGP VPNs configured on your system. In this example, VPN `172.16.8.6:140` displays in the **`show context all`** output and correctly matches the configuration on router `1` VPN context `vpn1-R1`.

```
[local]R1#show context all
Context Name               Context ID        VPN-RD              Description
-------------------------------------------------------------------------------
local                      0x40080001
vpn1-R1                    0x4008000c        172.16.8.6:140      <== VPN route on context CE1
```

4.2 Step B: Verify the VPN Configuration on the PE Routers

Use the **show configuration context *name*** command to verify the configuration on the PE routers.

In the following configuration examples, you verify that the correct export target, `64512:140`, from `R1` in VPN context `vpn1-R1`, is being imported into the R3 VPN context `vpn1-R3`.

The configuration on router `R1` VPN context is as follows:

```
[local]R1#show configuration context vpn1-R1
Building configuration...

Current configuration:
!
context vpn1-R1 vpn-rd 172.16.8.6:140          <== VPN 172.16.8.6:140 present on context vpn1-R1.
!                                              <==  in the show context all command

no ip domain-lookup
!
 interface to-CE1                              <== The interface that connects to CE1.
  ip address 14.14.14.1/30
 no logging console
 !
 router bgp vpn
  address-family ipv4 unicast
   export route-target 64512:140
   import route-target 64512:140
   redistribute connected
!
```

The configuration on router R3 VPN context is as follows:

```
[local]R3#show configuration context vpn1-R3
Building configuration...

Current configuration:
!
context vpn1-R3 vpn-rd 172.16.10.5:140
!
 no ip domain-lookup
!
 interface to-CE2                          <== The interface that connects to CE2.
  ip address 11.11.11.1/30
 no logging console
!
 router bgp vpn
  address-family ipv4 unicast
   export route-target 64512:140
   import route-target 64512:140           <== Import RT correctly matches export RT from R1.
   redistribute connected
!
```

4.3 Step C: Verify the IP Routing Table in the PE VPN Context

To verify the IP Routing Table in the PE VPN context:

1. Ping the Remote CE from the PE VPN.
2. Check the Remote CE Routes in the PE VPN.
3. Verify that the iBGP Routes are Installed in the RIB.

4.3.1 Step 1: Ping the Remote CE Router from the PE VPN

Use the **ping** command to verify connectivity with the remote CE router.

On each PE router, in the VPN context, use the **ping** command to verify connectivity to each remote CE router.

```
[vpn1-R1]R1#ping 11.11.11.2
PING 11.11.11.2 (11.11.11.2): source 14.14.14.1, 36 data bytes,     <== Ping CE2.
timeout is 1 second
!!!!!                                                               <==!!!!! indicates a successful ping

----11.11.11.2 PING Statistics----
5 packets transmitted, 5 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 1.804/2.324/3.404/0.679 ms
```

Note: If the ping is unsuccessful, use the **traceroute** command and verify the inner and outer MPLS labels.

4.3.2 Step 2: Check the Remote CE Routes in the PE VPN

Use the **show ip route** command to display the CE route entries (for example, router CE1) on the PE VPN context (for example, router R1 VPN context).

The IP routing table on the router R1 VPN context is as follows:

```
[vpn1-R1]R1#show ip route
Codes: C - connected, S - static, S dv - dvsr, R - RIP, e B - EBGP, i B - IBGP
       O   - OSPF, O3  - OSPFv3, IA - OSPF(v3) inter-area,
       N1  - OSPF(v3) NSSA external type 1, N2  - OSPF(v3) NSSA external type 2
       E1  - OSPF(v3) external type 1, E2  - OSPF(v3) external type 2
       i   - IS-IS, L1 - IS-IS level-1,  L2  - IS-IS level-2, N - NAT
       IPH - IP Host, SUB A - Subscriber address, SUB S - Subscriber static
       MIP F - Mobile-IP Foreign Agent, MIP H - Mobile-IP Home Agent
       A - Derived Default, MH - Media Nexthop
       >   - Active Route, * - LSP

Type    Network             Next Hop        Dist  Metric    UpTime  Interface
> i B   11.11.11.0/30       172.16.10.5      200       0  00:19:23  <== Remote VPN learned routes from CE2
> i B   12.12.12.0/30       172.16.10.6      200       0  00:19:23
> i B   13.13.13.0/30       172.16.57.5      200       0  00:19:23
> C     14.14.14.0/30                          0       0  00:00:08  to-n2x <== Local routes from CE1
> i B   172.16.8.5/32       172.16.8.5       200       0  00:19:23
> i B   192.168.6.16/30     172.16.57.5      200       0  00:19:23
> i B   192.168.8.52/30     172.16.8.5       200       0  00:19:23
> i B   192.168.8.68/30     172.16.8.5       200       0  00:19:23
> i B   192.168.8.72/29     172.16.8.5       200       0  00:19:23


[vpn1-R1]R1#show ip route 11.11.11.2
    Longest match Routing entry for 11.11.11.2/32 is 11.11.11.0/30 , version 22
    Route Uptime 01:52:41
    Paths: total 1, best path count 1
        Route redistributed to ospf 255

    Route has been downloaded to following slots
     02/0

    Path information :

      Active path :
      Known via bgp 64512, type-Internal BGP, distance 200, metric 0,
      Tag 0, Next-hop 172.16.10.5, NH-ID 0x31B00004
      Label 589824                                                  <== This is the internal label.
```

If the remote CE routes are not in the IP routing table:

1 Check that the remote PE VPN is redistributing the CE network into the BGP by using the **show configuration** command.

2 Check that the correct route targets are being imported.

3 Check that the VPN routes are correctly advertised in the local context of the remote PE router. For example, see Section 6.4 on page 33.

4 Check that the CE route is in the BGP routing table. For an example, see Section 5.1 on page 27.

5 If the CE route is not in the BGP routing table, check the output of the **show ip route registered next-hop** command. In this example, check that router R3 is registered.

Note: If you cannot resolve this issue after performing this procedure, contact your local technical support representative.

4.3.3 Step 3: Verify that the iBGP Routes Are Installed in the RIB

BGP routes are installed in the BGP speaker RIB (routing table). To verify that the iBGP routes are installed in the RIB, in the VPN context on the PE router, run the **`show bgp route ipv4 unicast`** command. If the IBGP routers are not present, issue the **`debug bgp rib`** command to check for the prefix that is not present in the IP routing table.

Caution!

Risk of performance loss. Debug messages can severely affect system performance. Exercise caution before enabling the generation of any debug messages on a production system.

BGP does not install a VPN route in the RIB when the next hop is not on the LSP; this indicates that LDP is not working correctly. Use the **`show bgp route`** ***`address`*** command to check if the next hop is not on the LSP.

```
[vpn1-R1]R1#show bgp route ipv4 unicast
Address Family: ipv4 unicast
BGP table version is 152, local router ID is 172.16.8.6
Status codes: d damped, h history, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete


VPN RD: 172.16.8.6:140
   Network            Next Hop         Metric  LocPrf  Weight Path
>i 11.11.11.0/30      172.16.10.5           0     100     100 ?
>i 12.12.12.0/30      172.16.10.6           0     100     100 ?
>i 13.13.13.0/30      172.16.57.5           0     100     100 ?
>  14.14.14.0/30      0.0.0.0               0     100   32768 ?

 <== The locally connected route is present in the RIB.

>i 100.100.100.100/32 172.16.10.5           0     100     100 ?
>  111.111.111.111/32 0.0.0.0               0     100   32768 ?
>i 172.16.8.5/32      172.16.8.5            0     100     100 ?
>i 192.168.6.16/30    172.16.57.5           0     100     100 ?
 i                    172.16.57.6           0     100     100 ?
>i 192.168.8.52/30    172.16.8.5            0     100     100 ?
>i 192.168.8.68/30    172.16.8.5            0     100     100 ?
>i 192.168.8.72/29    172.16.8.5            0     100     100 ?
```

5 Step 4: Verify the BGP Routing Table in the VPN PE Context

Verify that the prefix is in the BGP routing table.

Use the following procedure to verify the BGP routing table in the VPN PE context:

a Verify the BGP Route

b Verify a Specific BGP Route

5.1 Step A: Verify the BGP Route

Use the **`show bgp route`** command to verify the BGP route on the PE VPN context.

In the following example, in the BGP route highlighted in **bold**, **`>i`** indicates that there is a best path available for BGP to reach network `11.11.11.0/30`.

```
[vpn1_R1]R1#show bgp route
Address Family: ipv4 unicast
BGP table version is 99, local router ID is 172.16.8.6
Status codes: d damped, h history, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete


VPN RD: 172.16.8.6:140
   Network          Next Hop      Metric  LocPrf  Weight Path
>i 11.11.11.0/30    172.16.10.5       0     100     100 ?

<== ">i" Indicates that there is a best path available for BGP to reach network 11.11.11.0/30

>i 12.12.12.0/30    172.16.10.6       0     100     100 ?
>i 13.13.13.0/30    172.16.57.5       0     100     100 ?
>  14.14.14.0/30    0.0.0.0           0     100   32768 ?
>i 172.16.8.5/32    172.16.8.5        0     100     100 ?
>i 192.168.6.16/30  172.16.57.5       0     100     100 ?
 i                  172.16.57.6       0     100     100 ?
>  192.168.8.52/30  0.0.0.0           0     100   32768 ?
 i                  172.16.8.5        0     100     100 ?
>i 192.168.8.68/30  172.16.8.5        0     100     100 ?
>  192.168.8.72/29  0.0.0.0           0     100   32768 ?
 i                  172.16.8.5        0     100     100 ?
```

Note: When the prefix is not present in the BGP routing table, check that BGP is importing and exporting the route target by issuing the **`show configuration`** command. Make sure that the **`redistribution-connected`** command is enabled.

5.2 Step B: Verify a Specific BGP Route

Use the **show bgp route *address*** command to verify a specific BGP route on the PE VPN context.

```
[vpn1_R1]R1#show bgp route 11.11.11.0/30
BGP ipv4 unicast routing table entry: 11.11.11.0/30, version 64
Paths: total 1, best path count 1, best peer 172.16.10.5
Not advertised to any peer in this context

Local
  Imported from RD: 172.16.10.5:140        <== Verify that this is the RD configured in router R3 VPN.

  Nexthop 172.16.10.5 (751), peer 172.16.10.5 (172.16.10.5), AS 64512
  Origin incomplete, localpref 100, med 0, weight 100, internal, best

  Extended community: RT:64512:140         <== Verify that this RT was imported into R1 VPN
  Received label: 589824, allocated label: nolabel
```

Note: When the field that contains value `(751)` instead has a value that is `inaccessible`, the BGP route is not in the IGP table and, as a result, there is no best path. If the value is `inaccessible`, run the following commands, capture the results, and send it to your local technical support representative.

1. **debug bgp rib**

2. **show ip route registered next-hop** command and check if the `next-hop magic` field is not zero (0).

6 Step 5: Verify the BGP VPN Routes in the Local Context

Use the following procedure to verify the BGP VPN routes in the local context:

a Verify that all the VPN routes are Present on the Local Context

b Verify BGP Neighbors

c Verify the Received VPN Routes from the PE VPN (R3)

d Verify the Advertised VPN Routes to the PE VPN (R3)

6.1 Step A: Verify that All VPN Routes Are Present in the Local Context

Use the **`show bgp route ipv4 vpn`** command to verify all the VPN routes on the local router in the PE local context.

In the following example, verify that all the VPN routes are present on local router `R1` in the local context.

```
[local]R1#show bgp route ipv4 vpn
Address Family: ipv4 vpn
BGP table version is 71, local router ID is 172.16.8.6
Status codes: d damped, h history, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete


VPN RD: 64512:10
   Network            Next Hop                Metric  LocPrf  Weight Path
>  11.11.11.11/32     0.0.0.0                      0     100   32768 ?

VPN RD: 64512:100
   Network            Next Hop                Metric  LocPrf  Weight Path
>  10.10.10.10/32     0.0.0.0                      0     100   32768 ?

VPN RD: 172.16.8.5:140
   Network            Next Hop                Metric  LocPrf  Weight Path
>i 172.16.8.5/32      172.16.8.5                   0     100     100 ?
>i 192.168.8.52/30    172.16.8.5                   0     100     100 ?
>i 192.168.8.68/30    172.16.8.5                   0     100     100 ?
>i 192.168.8.72/29    172.16.8.5                   0     100     100 ?

VPN RD: 172.16.8.6:100
   Network            Next Hop                Metric  LocPrf  Weight Path
>  192.168.211.4/30   0.0.0.0                      0     100   32768 ?

VPN RD: 172.16.8.6:110
   Network            Next Hop                Metric  LocPrf  Weight Path
>  10.253.6.208/30    0.0.0.0                      0     100   32768 ?

VPN RD: 172.16.8.6:120
   Network            Next Hop                Metric  LocPrf  Weight Path
>  192.168.8.56/30    0.0.0.0                      0     100   32768 ?

VPN RD: 172.16.8.6:130
   Network            Next Hop                Metric  LocPrf  Weight Path
>  192.168.211.0/30   0.0.0.0                      0     100   32768 ?

VPN RD: 172.16.8.6:140
   Network            Next Hop                Metric  LocPrf  Weight Path
>i 11.11.11.0/30      172.16.10.5                  0     100     100 ?

<== Prefix 11.11.11.0/30 is a VPN route 172.16.10.5 is router R3.

>i 12.12.12.0/30      172.16.10.6                  0     100     100 ?
>i 13.13.13.0/30      172.16.57.5                  0     100     100 ?
>  14.14.14.0/30      0.0.0.0                      0     100   32768 ?
>i 172.16.8.5/32      172.16.8.5                   0     100     100 ?
>i 192.168.6.16/30    172.16.57.5                  0     100     100 ?
 i                    172.16.57.6                  0     100     100 ?
>  192.168.8.52/30    0.0.0.0                      0     100   32768 ?
 i                    172.16.8.5                   0     100     100 ?
>i 192.168.8.68/30    172.16.8.5                   0     100     100 ?
>  192.168.8.72/29    0.0.0.0                      0     100   32768 ?
 i                    172.16.8.5                   0     100     100 ?
```

6.2 Step B: Verify BGP Neighbors

Use the **`show bgp neighbor summary`** command to verify BGP neighbors.

In the following example, router `R1`, has received 1 VPN route from `R3` and advertised 7 VPN routes to router `R3`.

```
[local]R1#show bgp neighbor summary
BGP router identifier: 172.16.8.6, local AS number: 64512
Neighbors Configured: 5, Established: 5

Neighbor                 AS MsgRcvd MsgSent  InQ OutQ Rst  Up/Down State
172.16.8.5            64512      87      89    0    0   0 00:07:02 Established
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS unicast vpn restart
  unicast  : rcvd: 6 imported: 0 active: 3 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 4 imported: 4 active: 6 history: 0 dampened: 0 sent: 7

172.16.10.5            64512      76      87    0    0   0 00:06:45 Established
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS unicast vpn restart
  unicast  : rcvd: 5 imported: 0 active: 3 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 1 imported: 1 active: 2 history: 0 dampened: 0 sent: 7

<== 172.16.10.5  indicates that R1 has received 1 VPN route from router R3
and advertised 7 VPN routes to router R3.

  172.16.10.6            64512      77      87    0    0   0 00:06:45 Established
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS unicast vpn restart
  unicast  : rcvd: 5 imported: 0 active: 2 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 1 imported: 1 active: 2 history: 0 dampened: 0 sent: 7

172.16.57.5            64512      71      85    0    0   0 00:06:32 Established
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS unicast vpn restart
  unicast  : rcvd: 6 imported: 0 active: 4 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 2 imported: 2 active: 4 history: 0 dampened: 0 sent: 7

172.16.57.6            64512      79      87    0    0   0 00:06:46 Established
  CapSent  : refresh 4byteAS unicast vpn restart
  CapRcvd  : refresh 4byteAS unicast vpn restart
  unicast  : rcvd: 5 imported: 0 active: 3 history: 0 dampened: 0 sent: 6
  vpn      : rcvd: 1 imported: 1 active: 1 history: 0 dampened: 0 sent: 7
```

6.3 Step C: Verify the Received VPN Routes from the PE VPN (R3)

Use the **show bgp route ipv4 vpn neighbor *address* received** command to verify received VPN routes from the PE VPN—in this example, router R3 VPN.

```
[local]R1#show bgp route ipv4 vpn neighbor 172.16.10.5 received
Address Family: ipv4 vpn
BGP table version is 62, local router ID is 172.16.8.6
Status codes: d damped, h history, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete


VPN RD: 172.16.10.5:140
   Network            Next Hop               Metric  LocPrf  Weight Path
>i 11.11.11.0/30      172.16.10.5                 0     100     100 ?
<=== The above output shows the CE2 network received as a VPN route on router R1.
```

Note: If the route is not present, verify the advertised routes on the appropriate router. For an example, see Section 6.4 on page 33.

6.4 Step D: Verify the Advertised VPN Routes to the PE VPN (R3)

Use the **show bgp route ipv4 vpn neighbor *address* advertised** command to verify the advertised VPN routes to the PE VPN—in this example, router R3 VPN.

```
[local]R1#show bgp route ipv4 vpn neighbor 172.16.10.5 advertised
Address Family: ipv4 vpn
BGP table version is 62, local router ID is 172.16.8.6
Status codes: d damped, h history, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete


VPN RD: 172.16.8.6:100
   Network            Next Hop               Metric  LocPrf  Weight Path
>  192.168.211.4/30   0.0.0.0                     0     100   32768 ?

VPN RD: 172.16.8.6:110
   Network            Next Hop               Metric  LocPrf  Weight Path
>  10.253.6.208/30    0.0.0.0                     0     100   32768 ?

VPN RD: 172.16.8.6:120
   Network            Next Hop               Metric  LocPrf  Weight Path
>  192.168.8.56/30    0.0.0.0                     0     100   32768 ?

VPN RD: 172.16.8.6:130
   Network            Next Hop               Metric  LocPrf  Weight Path
>  192.168.211.0/30   0.0.0.0                     0     100   32768 ?

VPN RD: 172.16.8.6:140
   Network            Next Hop               Metric  LocPrf  Weight Path
>  14.14.14.0/30      0.0.0.0                     0     100   32768 ?

<== The above output shows the local CE1 network being advertised as a VPN route to router R3.

>  192.168.8.52/30    0.0.0.0                     0     100   32768 ?
>  192.168.8.72/29    0.0.0.0                     0     100   32768 ?
```

7 Step 6: Verify the Label Exchange between the VPN Contexts

Use the **`show bgp route labels`** command to verify BGP route label exchange between the VPN contexts.

The following example shows that network `11.11.11.0/30` can be reached by using the received label `589824`. Network `11.11.11.0/30` is advertised by allocating label `589824`.

For information about how to verify labels using the **`show mpls label-mapping`** command, see the *Troubleshooting MPLS* document.

```
[vpn1_R1]R1#show bgp route labels

VPN RD: 172.16.8.6:140
Network            Next Hop            Rcv Label      Alloc Label
11.11.11.0/30      172.16.10.5            589824         nolabel   <== Network 11.11.11.0/30 can be reached
12.12.12.0/30      172.16.10.6            589824         nolabel       by using the received label 589824
13.13.13.0/30      172.16.57.5            589824         nolabel
14.14.14.0/30      0.0.0.0                nolabel        589824
172.16.8.5/32      172.16.8.5             589824         nolabel
192.168.6.16/30    172.16.57.5            589824         nolabel
                   172.16.57.6            589824         nolabel
192.168.8.52/30    0.0.0.0                nolabel        589824
                   172.16.8.5             589824         nolabel
192.168.8.68/30    172.16.8.5             589824         nolabel
192.168.8.72/29    0.0.0.0                nolabel        589824
                   172.16.8.5             589824         nolabel

[vpn1_R3]R3#show bgp route labels

VPN RD: 172.16.10.5:140
Network            Next Hop            Rcv Label      Alloc Label
11.11.11.0/30      0.0.0.0                nolabel        589824    <== Network 11.11.11.0/30 is advertised
12.12.12.0/30      172.16.10.6            589824         nolabel       by allocating label 589824
13.13.13.0/30      172.16.57.5            589824         nolabel
14.14.14.0/30      172.16.8.6             589824         nolabel
172.16.8.5/32      172.16.8.5             589824         nolabel
192.168.6.16/30    172.16.57.5            589824         nolabel
                   172.16.57.6            589824         nolabel
192.168.8.52/30    172.16.8.5             589824         nolabel
                   172.16.8.6             589824         nolabel
192.168.8.68/30    172.16.8.5             589824         nolabel
192.168.8.72/29    172.16.8.5             589824         nolabel
                   172.16.8.6             589824         nolabel
```

Reference List

[1] *Configuring BGP/MPLS VPN*

[2] *Configuring MPLS*

[3] *Troubleshooting MPLS*

[4] *General Troubleshooting Guide*

[5] *Data Collection Guideline for the SmartEdge Router*