CHAPTER 6

Troubleshooting Layer 3 Connectivity

This section explains how to troubleshoot Layer 3 routing problems. If a ping to a remote site fails, the cause could be in an interface or in the Layer 3 routing. The overall approach for troubleshooting a failed ping should be to troubleshoot the interface failures and interface connectivity first, then proceed to troubleshooting Layer 3 routing if necessary.

Note

For interface troubleshooting, perform the procedures listed in Chapter 2, “Verifying and Troubleshooting Interface Status” and Chapter 3, “Troubleshooting Interface Connectivity.”

This chapter contains the following topics:

- Using show and debug Commands, page 6-125
- Traffic Loss, page 6-128
- Packets Are Punted and Switched in Software, page 6-129
- Traceroute Fails, page 6-130
- Adding Routes Fails, page 6-131
- Continuous Tracebacks, page 6-133
- fib_mgr Does Not Come Up During LC Reload or After Multiple Process Restarts, page 6-134
- CEF Entries Out of Sync, page 6-135
- fib_mgr Crashes, page 6-136
- Tracebacks Appearing, page 6-136
- Traffic Loss Because of Changing encap on a Subinterface, page 6-137
- Traffic Loss during RSP Failover, page 6-138
- Troubleshooting Virtual Router Redundancy Protocol, page 6-138
- Additional Information On Routing Configuration Commands, page 6-142

Using show and debug Commands

SUMMARY STEPS

1. `show cef location node-id`
2. `show cef ipv4 {prefix/mask} location node-id`
Using show and debug Commands

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> show cef location node-id</td>
<td>View all IPv4 routes of Cisco Express Forwarding (CEF) on an LC.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show cef location 0/2/CPU0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> show cef ipv4 {prefix/mask} location node-id</td>
<td>View a prefix’s route on an LC.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show cef ipv4 192.168.1.1/32 location 0/2/CPU0</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> show bgp summary</td>
<td>View Border Gateway Protocol (BGP) neighbors without an inbound and outbound policy for each active address family.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Use this when there are many routes.</td>
</tr>
<tr>
<td><strong>Step 4</strong> show bgp [{ipv4</td>
<td>all} {unicast</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show bgp dampened-paths</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> show bgp flap-statistics [ip-address[/mask]]</td>
<td>View BGP flap statistics.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Use this for routes that have had dampening enabled.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>RP/0/RSP0/CPU0:router# show bgp flap-statistics</td>
<td>If you do not specify arguments or keywords, all routes for the address family are displayed.</td>
</tr>
<tr>
<td></td>
<td>If you enter an IP address without mask or prefix length, the longest matching prefix is displayed.</td>
</tr>
<tr>
<td>Step</td>
<td>Command or Action</td>
</tr>
<tr>
<td>------</td>
<td>-------------------</td>
</tr>
<tr>
<td>6</td>
<td>show arp [vrf vrf-name] [ip-address [location node-id]</td>
</tr>
<tr>
<td></td>
<td>hardware-address [location node-id]</td>
</tr>
<tr>
<td></td>
<td>traffic] [location node-id</td>
</tr>
<tr>
<td></td>
<td>interface-name]</td>
</tr>
<tr>
<td>7</td>
<td>show interface accounting [location]</td>
</tr>
<tr>
<td>8</td>
<td>show cef ipv4 {prefix/mask} hardware {ingress</td>
</tr>
<tr>
<td></td>
<td>egress} location node-id</td>
</tr>
<tr>
<td>9</td>
<td>show cef platform trace common [all</td>
</tr>
<tr>
<td></td>
<td>errors</td>
</tr>
<tr>
<td></td>
<td>events</td>
</tr>
<tr>
<td></td>
<td>info] [location node-id]</td>
</tr>
<tr>
<td>10</td>
<td>show cef vrf [vrfname] [prefix]</td>
</tr>
</tbody>
</table>

Example

```
RP/0/RSP0/CPU0:router# show cef vrf vrf1 192.168.1.2 hardware egress location 0/1/CPU0
192.168.1.2/32, version 0, internal 0x40800001 (ptr 0xaac1c468) [1], 0x0 (0xaab8c7b0), 0x0 (0x0)
Updated Oct  1 21:29:37.684
local adjacency 130.130.1.2
Prefix Len 32, traffic index 0, Adjacency-prefix, precedence routine (0)
via 130.130.1.2, GigabitEthernet0/1/0/0, 3 dependencies, weight 0, class 0 [flags 0x0]
path-idx 0
next hop 130.130.1.2
local adjacency

TBM Node Data:
Node (0x00000100):0 0x8d40047d 0x00000000 0xffffffff 0xffffffff
```
Traffic Loss

This section provides steps for troubleshooting traffic loss.

**Step 1** Check for packet loss by examining transmitted packets on the local router and the receive packets on the destination router.
Chapter 6  Troubleshooting Layer 3 Connectivity

Packets Are Punted and Switched in Software

Step 1  Verify that the hardware chains for the destination IP address are pointing to either of the following:
- COMPLETE adjacency—Valid outgoing path exists.
- PUNT adjacency—Hardware does not know how to send the packet out, it just punts (diverts) the packet to be switched in software. If the transmit adjacency is PUNT, this could be because ARP is not resolved yet.

Step 2  To show if an ARP entry exists for the destination IP, use the show arp location command:

```
RP/0/RSP0/CPU0:router# show arp location node-id
```

a. If an ARP entry does not exist or is incomplete, add a static ARP entry. Ensure that the Tx adjacency points to ‘COMPLETE’.

```
RP/0/RSP0/CPU0:router# show cef (ipv4) 192.168.1.1/32 hardware egress detail location 0/4/CPU0
```

b. If so, then it means the issue is that of ARP entry not getting updated. Troubleshooting should now focus on why the ARP entry is not getting added (this includes steps like show arp, show arp idb, show adjacency gig node-id detail location node-id, show arp trace, and so forth).

c. If the Tx adjacency still points to ‘PUNT’, it means ARP is adding the entry in its database, but fib_mgr fails to mark the adjacency as ‘COMPLETE’.

```
d. This could be a fib_mgr, ARP, or AIB problem. Delete and reconfigure the static ARP entry with AIB and CEF debugs on. The debugs show if ARP is adding the entry inside the AIB and if the AIB is informing fib_mgr.
```

Step 3  View the ARP information on the particular LC or RSP.

```
show arp location node-id
```

Step 4  View any PI code ltrace errors recorded.
Traceroute Fails

Use traceroute to verify the connectivity to a destination. When traceroute fails to a destination, use the following commands:

- `show cef {ipv4} {destination_ip}/(mask) hardware egress detail location node-id` — View the hardware data structures involved with the prefix.
- `show interface location {outgoing_interface} accounting` — View input and output packets from the outgoing interface.

Step 1
Check if the destination IP address has the proper transmit adjacency. See the ‘Tx Adjacency’ state (it should be ‘COMPLETE’).

```
RP/0/RSP0/CPU0:router# show cef ipv4 prefix hardware egress detail location node-id
```

Step 2
If the transmit adjacency is not complete, there is an issue. If it is pointing to ‘PUNT’, that means probably the mac-address corresponding to the destination IP has not been learned. Try adding a ‘static arp’ entry and see if transmit adjacency moves to ‘COMPLETE’. If the destination IP is advertised by a routing protocol such as OSPF, then the transmit adjacency should never show as ‘PUNT’.

If the transmit adjacency is shown as ‘DROP’, that means there is a static route to the destination IP explicitly pointing the route to a DROP.

If the transmit adjacency is shown as ‘COMPLETE’, it means there is no problem in the hardware chains that are set up. You should see the counters.

Step 3
See if the output packets are equal to the traceroute packets sent.

```
RP/0/RSP0/CPU0:router# show interface location outgoing_interface accounting
```

Workaround

Step 1
Use the `shut` command (followed by `commit`) and the `no shut` command (followed by `commit`) on the outgoing interface.

Step 2
Add a static ARP entry for the destination IP.

---

Step 3
Packets could be dropped in the fabric. To verify this, view the fabric counters.

---

Traceroute Fails

Use traceroute to verify the connectivity to a destination. When traceroute fails to a destination, use the following commands:

- `show cef {ipv4} {destination_ip}/(mask) hardware egress detail location node-id` — View the hardware data structures involved with the prefix.
- `show interface location {outgoing_interface} accounting` — View input and output packets from the outgoing interface.

Step 1
Check if the destination IP address has the proper transmit adjacency. See the ‘Tx Adjacency’ state (it should be ‘COMPLETE’).

```
RP/0/RSP0/CPU0:router# show cef ipv4 prefix hardware egress detail location node-id
```

Step 2
If the transmit adjacency is not complete, there is an issue. If it is pointing to ‘PUNT’, that means probably the mac-address corresponding to the destination IP has not been learned. Try adding a ‘static arp’ entry and see if transmit adjacency moves to ‘COMPLETE’. If the destination IP is advertised by a routing protocol such as OSPF, then the transmit adjacency should never show as ‘PUNT’.

If the transmit adjacency is shown as ‘DROP’, that means there is a static route to the destination IP explicitly pointing the route to a DROP.

If the transmit adjacency is shown as ‘COMPLETE’, it means there is no problem in the hardware chains that are set up. You should see the counters.

Step 3
See if the output packets are equal to the traceroute packets sent.

```
RP/0/RSP0/CPU0:router# show interface location outgoing_interface accounting
```

Workaround

Step 1
Use the `shut` command (followed by `commit`) and the `no shut` command (followed by `commit`) on the outgoing interface.

Step 2
Add a static ARP entry for the destination IP.
Adding Routes Fails

Perform the steps in this section to troubleshoot failures in adding routes. During Out Of Resource (OOR), the router does not accept additional routes until existing routes are deleted.

**Note**
The sample commands in this section are applicable to Ethernet LCs, not SIP-700 LCs.

**Step 1** Determine if any resources are experiencing problems. View the state of various data structures. Ideally the state should be GREEN. If it is either YELLOW or RED, it indicates an OOR condition.

```
RP/0/RSP0/CPU0# show cef resource location node-id

CEF resource availability summary state: GREEN
CEF will work normally
ipv4 shared memory resource: GREEN
ipv6 shared memory resource: GREEN
mpls shared memory resource: GREEN
common shared memory resource: GREEN
DATA_TYPE_TABLE_SET hardware resource: GREEN
DATA_TYPE_TABLE hardware resource: GREEN
DATA_TYPE_IDB hardware resource: GREEN
DATA_TYPE_IDB_EXT hardware resource: GREEN
DATA_TYPE_LEAF hardware resource: GREEN
DATA_TYPE_LOADINFO hardware resource: GREEN
DATA_TYPE_PATH_LIST hardware resource: GREEN
DATA_TYPE_NHINFO hardware resource: GREEN
DATA_TYPE_LABEL_INFO hardware resource: GREEN
DATA_TYPE_FRR_NHINFO hardware resource: GREEN
DATA_TYPE_ECD hardware resource: GREEN
DATA_TYPE_RECURSIVE_NH hardware resource: GREEN
DATA_TYPE_TUNNEL_ENDPOINT hardware resource: GREEN
DATA_TYPE_LOCAL_TUNNEL_intf hardware resource: GREEN
DATA_TYPE_ECD_TRACKER hardware resource: GREEN
DATA_TYPE_ECD_V2 hardware resource: GREEN
DATA_TYPEATTRIBUTE hardware resource: GREEN
DATA_TYPE_LSPA hardware resource: GREEN
DATA_TYPE_LDI_LW hardware resource: GREEN
DATA_TYPE_LDSH_ARRAY hardware resource: GREEN
DATA_TYPE_TE_TUN_INFO hardware resource: GREEN
DATA_TYPE_IDB_VRF_LCL_CEF hardware resource: GREEN
DATA_TYPE_TABLE_UNRESOLVED hardware resource: GREEN
DATA_TYPE_MOL hardware resource: GREEN
DATA_TYPE_MPI hardware resource: GREEN
DATA_TYPE_SUBS_INFO hardware resource: GREEN
DATA_TYPE_GRE_TUNNEL_INFO hardware resource: GREEN
```

**Step 2** Determine which hardware table is OOR. Compare ‘max entries’ and ‘used entries’ too see which of the data structures is using the entries close to the max limit.

```
RP/0/RSP0/CPU0# show cef platform resource location node-id

CEF resource availability summary state: GREEN
CEF will work normally
ipv4 shared memory resource: GREEN
ipv6 shared memory resource: GREEN
mpls shared memory resource: GREEN
common shared memory resource: GREEN
DATA_TYPE_TABLE_SET hardware resource: GREEN
DATA_TYPE_TABLE hardware resource: GREEN
DATA_TYPE_IDB hardware resource: GREEN
DATA_TYPE_IDB_EXT hardware resource: GREEN
DATA_TYPE_LEAF hardware resource: GREEN
DATA_TYPE_LOADINFO hardware resource: GREEN
DATA_TYPE_PATH_LIST hardware resource: GREEN
DATA_TYPE_NHINFO hardware resource: GREEN
DATA_TYPE_LABEL_INFO hardware resource: GREEN
DATA_TYPE_FRR_NHINFO hardware resource: GREEN
DATA_TYPE_ECD hardware resource: GREEN
DATA_TYPE_RECURSIVE_NH hardware resource: GREEN
DATA_TYPE_TUNNEL_ENDPOINT hardware resource: GREEN
DATA_TYPE_LOCAL_TUNNEL_intf hardware resource: GREEN
DATA_TYPE_ECD_TRACKER hardware resource: GREEN
DATA_TYPE_ECD_V2 hardware resource: GREEN
DATA_TYPEATTRIBUTE hardware resource: GREEN
DATA_TYPE_LSPA hardware resource: GREEN
DATA_TYPE_LDI_LW hardware resource: GREEN
DATA_TYPE_LDSH_ARRAY hardware resource: GREEN
DATA_TYPE_TE_TUN_INFO hardware resource: GREEN
DATA_TYPE_IDB_VRF_LCL_CEF hardware resource: GREEN
DATA_TYPE_TABLE_UNRESOLVED hardware resource: GREEN
DATA_TYPE_MOL hardware resource: GREEN
DATA_TYPE_MPI hardware resource: GREEN
DATA_TYPE_SUBS_INFO hardware resource: GREEN
DATA_TYPE_GRE_TUNNEL_INFO hardware resource: GREEN
```

**Note**

IPV4_LEAF_P usage is same on all NPs
Adding Routes Fails

NP: 0  struct 23: IPV4_LEAF_P  (maps to ucode stru = 54 in TopSearch1)
Used Entries: 298 Max Entries: 524288

IPV6_LEAF_P usage is same on all NPs
NP: 0  struct 24: IPV6_LEAF_P  (maps to ucode stru = 55 in TopSearch1)
Used Entries: 4 Max Entries: 131072

R_LDI usage is same on all NPs
NP: 0  struct 6: R_LDI           (maps to ucode stru = 11 in TopSearch1)
Used Entries: 8 Max Entries: 65536

NR_LDI usage is same on all NPs
NP: 0  struct 7: NR_LDI          (maps to ucode stru = 12 in TopSearch1)
Used Entries: 31 Max Entries: 524288

RPF_STRICT usage is same on all NPs
NP: 0  struct 9: RPF_STRICT      (maps to ucode stru = 15 in TopSearch1)
Used Entries: 0 Max Entries: 65536

NP: 0  struct 12: TX_ADJ          (maps to ucode stru = 18 in TopSearch1)
Used Entries: 21 Max Entries: 131072

NP: 1  struct 12: TX_ADJ          (maps to ucode stru = 18 in TopSearch1)
Used Entries: 21 Max Entries: 131072

NP: 2  struct 12: TX_ADJ          (maps to ucode stru = 18 in TopSearch1)
Used Entries: 18 Max Entries: 131072

NP: 3  struct 12: TX_ADJ          (maps to ucode stru = 18 in TopSearch1)
Used Entries: 18 Max Entries: 131072

RX_ADJ usage is same on all NPs
NP: 0  struct 13: RX_ADJ          (maps to ucode stru = 19 in TopSearch1)
Used Entries: 28 Max Entries: 32768

TE_NH_ADJ usage is same on all NPs
NP: 0  struct 14: TE_NH_ADJ       (maps to ucode stru = 20 in TopSearch1)
Used Entries: 6 Max Entries: 32768

L2VPN_LDI usage is same on all NPs
NP: 0  struct 16: L2VPN_LDI       (maps to ucode stru = 13 in TopSearch1)
Used Entries: 0 Max Entries: 32768

LABEL_UFIB usage is same on all NPs
NP: 0  struct 28: LABEL_UFIB      (maps to ucode stru = 1 in TopParse)
Used Entries: 4 Max Entries: 290000

Step 3  After determining which data structure is OOR, verify if it is expected or unexpected. Usually, for each LEAF (either IPv4), it requires four entries of NR_LDI structure. So if you find the NR_LDI structure going OOR, see if you have appropriate number of IP LEAFs to take this NR_LDI number to such a limit.

Step 4  If show cef resource location node-id shows the state in GREEN, it means that the problem is not caused by an OOR condition. The reason for not being able to add further routes is some thing else. Enable the following debugs to observe what is happening:

- RP/0/RSP0/CPU0:router# debug cef errors location node-id
- RP/0/RSP0/CPU0:router# debug cef (ipv4) error location node-id
- If you observe any tracebacks, decode the tracebacks by using SBT tool.
Continuous Tracebacks

When tracebacks appear continuously on the console (typically every 15 seconds), programming of the entry inside the hardware is not successful. This causes the software to try repeatedly after every 15 seconds. It is possible that the layer just above the hardware or the hardware itself is not up and running.

Step 1

View all platform ltrace common messages. Verify that both CPPs are in ACTIVE_SOLO state.

Example

```
show controllers pse qfp system state location 0/1/CPU0
```

CPP HA client processes registered (5 of 5)
- cpp_sp : Initialized
- cpp_cdm : Initialized
- cpp_driver1 : Initialized
- cpp_driver0 : Initialized
- cpp_cp : Initialized

------------------------------------------
CPP 0: dir=INGRESS Role: curr=ACTIVE_SOLO next=ACTIVE_SOLO <<< CPP 0 in ACTIVE_SOLO state
Client State: ENABLE
Image: /pkg/ucode/cpp/cpp-thor-ucode
Image: thor_ingress
HW: CPP10
Built by: eruan
Host: sjc-lds-447
Time: Tue Sep 28 15:04:57 2010
Component: cpp/dp asr41-9k-cgn/2
Load Cnt: 1 Last load: Oct 01, 2010 21:27:36.488431
Active Threads: 0-159
Stuck Threads: <NONE>
Fault Manager Flags:
- ignore_fault: FALSE
- ignore_stuck_thread: FALSE
- crashdump_in_progress: FALSE
------------------------------------------

CPP 1: dir=EGRESS Role: curr=ACTIVE_SOLO next=ACTIVE_SOLO <<< CPP 1 in ACTIVE_SOLO state
Client State: ENABLE
Image: /pkg/ucode/cpp/cpp-thor-ucode
Image: thor_egress
HW: CPP10
Built by: eruan
Host: sjc-lds-447
Time: Tue Sep 28 14:55:59 2010
Component: cpp/dp asr41-9k-cgn/2
Load Cnt: 1 Last load: Oct 01, 2010 21:27:36.500431
Active Threads: 0-159
Stuck Threads: <NONE>
Fault Manager Flags:
  ignore_fault:          FALSE
  ignore_stuck_thread:   FALSE
  crashdump_in_progress: FALSE

Step 2 View all platform ltrace protocol messages for IPv4 or IPv6.

    show cef platform trace {ipv4 | ipv6} all reverse location node-id

Step 3 Check that the NP provisioning layer (or PRM) is up. PRM is a layer just above hardware. If PRM is down, no entry is programmed in hardware, indicating that NP may have had a problem during initialization.

    show controllers NP summary

Step 4 View the NP driver logs to find out if there have been NP initialization errors. If there are NP initialization errors, it is likely an NP problem.

    show controllers NP drvlog location node-id

Step 5 Use the SBT tool to decode the tracebacks. From root of the workspace, use ./util/bin/sbt -p (process_name) -f (log_file).

Workaround

Step 1 Restart prm_server process.
Step 2 Reboot LC.

fib_mgr Does Not Come Up During LC Reload or After Multiple Process Restarts

Fib_mgr depends on underlying hardware. If the underlying process or hardware does not come up, it is likely that fib_mgr will not come up.

- **show controllers NP summary location node-id**—Check that the NP provisioning layer (or PRM) is up.
- **show controllers NP drvlog location node-id**—View the NP driver logs.
• **show cef platform trace common all reverse location node-id**—View platform ltrace common messages.

• **show cef platform trace common event reverse location node-id**—View platform ltrace common events.

• **show cef platform trace (ipv4 | ipv6 | mpls) error reverse location node-id**—View platform ltrace error messages recorded for protocols IPv4, IPv6, or MPLS.

• **show cef trace all reverse location node-id**—View all CEF ltrace messages.

**Step 1**

Use the **show controllers NP summary location** and **show controllers NP drvlog location** commands to determine if either the PRM or the underlying NP has a problem. If so, the fib_mgr will not come up. Troubleshoot at the PRM layer or NP layer.

**Step 2**

If both CPPs are in ACTIVE_SOLO state, the problem is likely a software bug. In this case, collect the core file and decode the tracebacks using the SBT tool. From root of the workspace, use ./util/bin/sbt -p (process_name) -f (log_file).

**Workaround**

**Step 1**

Restart the prm_server process.

**Step 2**

Reboot the LC.

**CEF Entries Out of Sync**

The **cef** entry on RSP may be pointing to the management interface and as a result the traffic originating from the router may go out instead of through the LC interface.

• **show controllers np drvlog location node-id**—Shows the PRM view of the Direct Table on the NP.

• **show tech-support cef**—Collects relevant platform independent traces.

• **show cef trace events reverse location node-id**—View platform independent cef ltrace events.

• **show cef trace errors reverse location node-id**—View platform independent cef ltrace errors.

• **show cef platform trace common event reverse location node-id**—View CEF platform common event traces.

• **show cef platform trace common error reverse location node-id**—View CEF platform common error traces.

**Step 1**

Look for a default route 0.0.0.0/0 configured to go out through the management interface.

**Step 2**

Look for a static ARP configured for the prefix in question. It is possible that ARP is installing two entries through both the management interface and also through the LC interface (because the prefix is reachable by both routes).
Step 3  If the above is not the case, use the `show arp` command to see if an ARP entry is advertising through the management interface. If this is the case, clear the ARP and verify the cef entries again.

Workaround

- Use the `shut` command (followed by `commit`) and the `no shut` command (followed by `commit`) on the management interface.
- Use the `clear arp-cache` command.
- Reboot the LC.

fib_mgr Crashes

- `show cef platform trace common all reverse location node-id`—View CEF platform common traces.
- `show cef platform trace common event reverse location node-id`—View CEF platform common event traces.
- `show cef platform trace common error reverse location node-id`—View CEF platform common error traces.
- `show cef platform trace {ipv4 | ipv6 | mpls} error reverse location node-id`—View CEF platform protocol traces for IPv4 or MPLS.

Step 1  If the trigger is a prm restart or crash, this is expected.
Step 2  If the underlying process (prm_server) is down or crashed, it is likely fib_mgr will not come up.
Step 3  Save the core file.
Step 4  Use the SBT to decode the tracebacks. From root of the workspace, use `./util/bin/sbt -p (process_name) -f (log_file)`.
Step 5  Save the console logs.

Workaround

Restart fib_mgr or reboot the LC.

Tracebacks Appearing

In this scenario, a few error tracebacks appear on the console because of some trigger (such as interface `shut/no shut`, or any other similar trigger).

- `show cef trace event location node-id`—View CEF traces for major events.
- `show cef trace errors location node-id`—View CEF traces for major errors.
- `show cef platform trace common errors location node-id`—View CEF platform traces for common errors across all protocols.
Traffic Loss Because of Changing encap on a Subinterface

When traffic is being forwarded through a Layer 3 subinterface and if the encapsulation is changed on that subinterface, it is sometimes observed that the traffic does not resume until after 15 seconds.

- show cef trace event reverse location node-id—View CEF trace messages for major events.
- show cef trace error reverse location node-id—View CEF trace messages for major errors.
- show cef platform trace common error location node-id—View CEF platform traces for common errors across all protocols.
- show cef platform trace (ipv4 | ipv6 | mpls) event location node-id—View CEF platform traces for major events in protocols IPv4, IPv6, or MPLS.
- show cef platform trace (ipv4 | ipv6 | mpls) error location node-id—View CEF platform traces for major errors in protocols IPv4, IPv6, or MPLS.
- show arp trace location node-id—Shows arp related traces.
- show arp-gmp trace location node-id—Shows arp-gmp related traces.
- show arp location node-id—View ARP-related information.

This type of traffic loss could happen typically when there is a static arp entry for the prefix which is experiencing traffic loss. For example, consider the following configuration:

```
interface GigabitEthernet0/4/0/39.2
ipv4 address 209.165.201.1 255.0.0.0
dot1q vlan 300
```

When encapsulation changes from `dot1q vlan 300` to `dot1q vlan 200` on the subinterface, fib_mgr deletes all prefixes corresponding to this interface and creates them again. It takes 15 seconds to add all prefixes; traffic does not get forwarded for that time. For example, there is an interface with address 192.0.2.0/8. There is a static ARP entry for 192.0.2.5.

```
RP/0/RSP0/CPU0:router# show run | inc arp
```
The delay is less likely to happen with regular adjacency (not the static ARP).

When VLAN color changes, the following occurs:

- Adjacency is deleted, and the adjacency route 192.0.2.5 is deleted.
- Connected route is deleted.
- Adjacency is added before the connected route is added. The FIB treats adding an adjacency without a covering connected route as an error, so the route 192.0.2.5 is placed in retry.
- Connected route 192.0.2.0/8 is added.
- Because the FIB retry timer is 15 seconds, the adjacency route 192.0.2.5 is added after 15 seconds.

**Workaround**

Remove the static ARP entry.

---

**Traffic Loss during RSP Failover**

Sometimes RSP switchover (keyword `failover` in CLI) causes traffic loss. This may mean the IGP over which the prefixes are learned is going down. The following assumes OSPF as the IGP.

- `show process failover`—Shows process details during failover.
- `debug ospf ha`—Enables OSPF HA related debugs.
- `debug ospf instance nsf`—View before failover and collect the debug log.
- `show process failover`—View before failover and collect the debug log.
- `show redundancy`—Provides status of the standby node after failover.

Check if the next hop router had a failover.

- If so, the OSPF will go down.
- If not, verify that `nsf cisco` is configured under OSPF.
  - If `nsf cisco` is configured, see if the next hop is reachable during failover.
  - If the next hop is not reachable, a link may be going down or having negotiation problems.
  - If the next hop is reachable, the problem is likely a software bug.

**Workaround**

Reload the router.

---

**Troubleshooting Virtual Router Redundancy Protocol**

*Virtual Router Redundancy Protocol* (VRRP) enables a group of routers to form a single virtual router. This section contains the following subsections:

- Using show and debug Commands, page 6-139
- VRRP Fails to Reach Active State, page 6-140
- Tracked Interface Failing, Router State Not Changed, page 6-140
- VRRP State Flapping, page 6-140
Using show and debug Commands

**SUMMARY STEPS**

1. `show vrrp [interface type interface-id] [brief]`
2. `show vrrp interface type interface-id detail`
3. `show vrrp [interface {type interface-id}] statistics [all]`
4. `show controllers type interface-id`
5. `debug vrrp [ all | edm | events | packets ]`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> <code>show vrrp [interface type interface-id] [brief]</code></td>
<td>View all VRRP groups status.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/0/CPU0:# show vrrp brief</td>
</tr>
<tr>
<td><strong>Step 2</strong> <code>show vrrp interface type interface-id detail</code></td>
<td>View detailed information of VRRP groups.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/0/CPU0:# show vrrp gigabitEthernet 0/1/0/1 detail</td>
</tr>
<tr>
<td><strong>Step 3</strong> <code>show vrrp [interface {type interface-id}] statistics [all]</code></td>
<td>View VRRP statistics.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/0/CPU0:# show vrrp statistics</td>
</tr>
<tr>
<td><strong>Step 4</strong> <code>show controllers type interface-id</code></td>
<td>View the VRRP group MAC addresses as part of unicast filter list.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/0/CPU0:# show controllers gigabitEthernet 0/3/0/9</td>
</tr>
<tr>
<td><strong>Step 5</strong> `debug vrrp [ all</td>
<td>edm</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>RP/0/0/CPU0:# debug vrrp packets tengige 0/3/0/9</td>
</tr>
</tbody>
</table>
Troubleshooting Virtual Router Redundancy Protocol

Chapter 6    Troubleshooting Layer 3 Connectivity

VRRP Fails to Reach Active State

Run the following command on both routers:

```
RP/0/RSP0/CPU0:router# show vrrp detail
```

Misconfiguration

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Ensure that the interface with VRRP configured is up.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Ensure that an IP address is configured, on the same subnet as the interface, and delay is configured.</td>
</tr>
</tbody>
</table>

```
RP/0/RSP0/CPU0:router# show vrrp detail
```

Higher Priority Router Already Active

Examine the output of the `show vrrp` command:

- If the Master address for VRRP shows an IP address instead of local, the router with that IP address is Active.
- If preemption is enabled, but the other router has higher priority, then it will remain in the Active state.

Operational priority may not match the configured priority. If interfaces are down, this negatively impacts operational priority.

Preemption is Disabled and Another Router Already Active

Examine the output of the `show vrrp` command. If preemption is disabled, and the router has higher priority, it will not take over unless preemption is enabled.

Tracked Interface Failing, Router State Not Changed

On both routers:

```
RP/0/RSP0/CPU0:router# show vrrp detail
```

If preemption is enabled and this router has higher operational priority than the other router, this router remains in the Active state. Configured priority or the decrement for tracked interfaces needs to be configured appropriately such that the state transition takes place. If the IP address is the same as the interface IP address, the router does not change to the Standby state.

VRRP State Flapping

On both routers:

```
Step 1  RP/0/RSP0/CPU0:router# show vrrp detail
```
**Step 2**  
RP/0/RSP0/CPU0:router# debug vrrp packets

Check timestamps to determine whether there is a delay in sending or receiving packets. Check the CPU usage to see if some process is hogging the system resources.

**Step 3**  
RP/0/RSP0/CPU0:router# show spp node-counters location interface-running-vrrp

---

**More Than One VRRP Router Active**

**Step 1**  
Verify that the same IP is configured on both ends.

RP/0/RSP0/CPU0:router# show vrrp detail

**Step 2**  
Check timestamps to determine whether there is a delay in sending or receiving packets. Check the CPU usage to see if a process is overusing resources.

**Step 3**  
Enter the debug command for VRRP packets on the peer.

RP/0/RSP0/CPU0:router# debug vrrp packets

Check for lines similar to:  
RP/0/RSP0/CPU0:Sep 8 14:16:39.217 : vrrp[357]: Gi0/5/0/0: VR1: Pkt: ADVER: IN: pri 100 src 192.0.2.11. This means advertisement packets are being received by VRRP. If these are absent, no packets are being received and VRRP becomes active.

Look for lines similar to:  
RP/0/RSP0/CPU0:Sep 8 14:18:47.876 : vrrp[357]: Gi0/5/0/0: VR1: Pkt: ADVER: Out: pri 100 src 192.0.2.11. This means the peer is sending VRRP packets.

**Step 4**  
Check the output of the show spp node-counters location interface-running-vrrp on both routers, and look for packet drops.

RP/0/RSP0/CPU0:router# show spp node-counters location interface-running-vrrp

---

**VRRP Active Router Not Forwarding Traffic**

On both routers:

**Step 1**  
Find the virtual MAC address for the group.

RP/0/RSP0/CPU0:router# show vrrp detail

**Step 2**  
RP/0/RSP0/CPU0:router# show ether-ctrl trace

**Step 3**  
Ensure that the virtual MAC address is in the unicast address filter list and verify the router is receiving traffic.

RP/0/RSP0/CPU0:router# show controllers type interface-running-vrrp
Traffic Loss or Unexpected VRRP State After Interface shut/no shut

In case of shut/no shut on a VRRP-enabled interface, the following has been observed:

- If preemption is enabled, recovery times are higher than switchover times. This means higher traffic loss has occurred when the interface is no shut.
- If preemption is disabled, some VRRP groups are preempted after no shut of an interface.

If you observe either of the above conditions after an interface no shut, perform the following steps on both routers.

Step 1  RP/0/RSP0/CPU0:router# show vrrp detail
Step 2  RP/0/RSP0/CPU0:router# show ether-ctrl trace
Step 3  RP/0/RSP0/CPU0:router# show controllers type interface-running-n
Step 4  RP/0/RSP0/CPU0:router# debug vrrp packets interface—For the interface on which no shut is being performed.
Step 5  Enter the no shut command.
Step 6  Observe the console logs and look for lines similar to:

RP/0/RSP0/CPU0:Sep 8 14:16:39.217 : vrrp[357]: Gi0/5/0/0: VR1: Pkt: ADVER: IN: pri 100
crc 192.0.2.11.

Note the time lag between the no shut and the first such message seen. For that amount of time, there is traffic loss between two routers.

Step 7  If there is no traffic flowing between two routers after a no shut event, check the STP configuration on the Cisco ASR 9000 Series Router. Lowering the fwd delay timer might help in reducing the traffic loss.

Step 8  For preemption disabled case, if the groups still preempt after reducing the fwd delay timer, repeat Step 1 through Step 4, and find the time period of traffic loss between the two routers. The preemption can be avoided by configuring the minimum delay to be higher than the time period of traffic loss. Minimum delay can be configured as follows:

RP/0/RSP0/CPU0:router(config)# router vrrp interface gigabitEthernet 0/2/0/10 vrrp delay
minimum 10 reload 5

Additional Information On Routing Configuration Commands

Use the following guides if you need to review routing configuration commands

- Cisco ASR 9000 Series Aggregation Services Router Routing Configuration Guide, Release 4.0
- Cisco ASR 9000 Series Aggregation Services Router Routing Command Reference, Release 4.0