

Verify BFD Hardware Counters on DFC Linecards for 7600 Devices

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Background Information](#)

[Topology](#)

[Troubleshooting Methodology](#)

Introduction

This document describes how to verify Bidirectional Forwarding Detection (BFD) hardware counters on Distributed Forwarding Card (DFC) linecards for 7600 devices.

Prerequisites

Requirements

Cisco recommends that you have basic knowledge of these topics:

- 7600 series routers configuration and features
- DFC linecard modules configuration

Components Used

The information in this document is based on 7600 IOS version 15.3.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Background Information

BFD is a network protocol designed to detect subsecond failures in communication in any kind of path between systems (direct physical links, virtual circuits, tunnels, MPLS LSPs, etc).

DFC stands for DFC, the main difference between DFC and CFC linecards is that DFC linecards have an embedded daughter card with an L2 engine and an L3/4 engine, this offloads forwarding lookups that on CFC linecards would be sent to the Supervisor to be done locally on the linecard.

BFD packets are treated in hardware that way they don't impact the CPU, this means that in DFC

cards this packets are always received and forwarded without leaving the linecard.

Topology

R1(Te3/21)-----R2

Troubleshooting Methodology

You can see that R1 does not bring up the BFD Adjacency with his neighbor on Tengig3/21.

Check the neighbor details:

```
R1# sh bfd nei det
```

```
IPv4 Sessions
```

NeighAddr	LD/RD	RH/RS	State	Int
172.31.11.34	1/0	Down	Down	Te3/21

```
Session Host: Hardware
```

```
OurAddr: 172.31.11.33
```

```
Handle: 1
```

```
Local Diag: 1, Demand mode: 0, Poll bit: 0
```

```
MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 5
```

```
Received MinRxInt: 200000, Received Multiplier: 5
```

```
Holddown (hits): 0(0), Hello (hits): 1000(0)
```

```
Rx Count: 37  Notice received packets are too low
```

```
Tx Count: 9401
```

```
Elapsed time watermarks: 0 0 (last: 0)
```

```
Registered protocols: ISIS CEF
```

```
Downtime: 02:36:34
```

```
Last packet: Version: 1 - Diagnostic: 0
```

```
State bit: Up - Demand bit: 0
```

```
Poll bit: 0 - Final bit: 0
```

```
C bit: 1
```

Multiplier: 5 - Length: 24
My Discr.: 77 - Your Discr.: 1
Min tx interval: 200000 - Min rx interval: 200000
Min Echo interval: 0

R1# **sh bfd nei det**

IPv4 Sessions

NeighAddr	LD/RD	RH/RS	State	Int
172.31.11.34	1/0	Down	Down	Te3/21

Session Host: Hardware

OurAddr: 172.31.11.33

Handle: 1

Local Diag: 1, Demand mode: 0, Poll bit: 0

MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 5

Received MinRxInt: 200000, Received Multiplier: 5

Holddown (hits): 0(0), Hello (hits): 1000(0)

Rx Count: 37 B-----Notice received packets are not incrementing

Tx Count: 9456 B----- Transmit packets are incrementing

Elapsed time watermarks: 0 0 (last: 0)

Registered protocols: ISIS CEF

Downtime: 02:36:34

Last packet: Version: 1 - Diagnostic: 0

State bit: Up - Demand bit: 0

Poll bit: 0 - Final bit: 0

C bit: 1

Multiplier: 5 - Length: 24

My Discr.: 77 - Your Discr.: 1

Min tx interval: 200000 - Min rx interval: 200000

Min Echo interval: 0

You can also check the same command for hardware which gives the same output, RX is not received.

R1#show bfd neighbors hardware details

IPv4 Sessions

NeighAddr	LD/RD	RH/RS	State	Int
172.31.11.34	1/0	Down	Down	Te3/21

Session Host: Hardware

OurAddr: 172.31.11.33

Handle: 1

Local Diag: 1, Demand mode: 0, Poll bit: 0

MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 5

Received MinRxInt: 200000, Received Multiplier: 5

Holddown (hits): 0(0), Hello (hits): 1000(0)

Rx Count: 37

Tx Count: 19337

Elapsed time watermarks: 0 0 (last: 0)

Registered protocols: ISIS CEF

Downtime: 05:22:16

Last packet: Version: 1 - Diagnostic: 0

State bit: Up - Demand bit: 0

Poll bit: 0 - Final bit: 0

C bit: 1

Multiplier: 5 - Length: 24

My Discr.: 77 - Your Discr.: 1

Min tx interval: 200000 - Min rx interval: 200000

Min Echo interval: 0

R1#show bfd neighbors hardware details

IPv4 Sessions

NeighAddr	LD/RD	RH/RS	State	Int
172.31.11.34	1/0	Down	Down	Te3/21

Session Host: Hardware

OurAddr: 172.31.11.33

```
Handle: 1

Local Diag: 1, Demand mode: 0, Poll bit: 0

MinTxInt: 1000000, MinRxInt: 1000000, Multiplier: 5

Received MinRxInt: 200000, Received Multiplier: 5

Holddown (hits): 0(0), Hello (hits): 1000(0)

Rx Count: 37

Tx Count: 19348

Elapsed time watermarks: 0 0 (last: 0)

Registered protocols: ISIS CEF

Downtime: 05:22:28

Last packet: Version: 1                - Diagnostic: 0

                State bit: Up          - Demand bit: 0

                Poll bit: 0            - Final bit: 0

                C bit: 1

                Multiplier: 5          - Length: 24

                My Discr.: 77          - Your Discr.: 1

                Min tx interval: 200000 - Min rx interval: 200000
```

After this, you can proceed to check counters directly on the linecard.

For this you need the Local Discriminator (LD) value on the show bfd neighbors details output, for this case LD value is 1.

LD, this value is used to uniquely identify this session and it must be unique and non-zero, for all BFD sessions in this device.

You do **show module** and see that Linecard 3 is DFC.

You attach the linecard where you want to check the BFD values, in this case it's linecard 3.

```
R1# attach 3
```

```
R1-dfc3# show platform npc bfd ld 1
```

```
bfd_pak_big 0
```

```
bfd_pak_authenticated 0
```

```
bfd_x40g_xlifid_ifnum0 0
```

```
bfd_wd_hash_table_retry_count 0
```

bfd_ld_hash_table_retry_count 0

x40g_sso_differ_ld_count 0

Current normal_event_qsize 0 and 0 paks crossed the limit.

***BFD Session info for ld(1) avlnode ld (1) ***

ifnum(25), slotunit(21), txtimer(1000000) detect_timer(0)

p bit(0), f bit(0), srcip(172.31.11.33) dstip(172.31.11.34)

wdog cnterid(65664) tags inner(0) outer(0) tx sess info(0x19F4B7E0)

ADJ registered(0x1) tag_count(0) tx sessid(830)

dmac(dccc.eeee.aaaa), smac(5033.eeeee.8888), rx statid(508546), tx statid(508545)

RX pkt count(5838365), TX pkt count (5208864) β ----- Here
you can see the counters for the RX and TX

IPV6 SA(::), IPV6 DA(::), no_adj_retry_tx (0)

R1# **show platform npc bfd ld 1**

bfd_pak_big 0

bfd_pak_authenticated 0

bfd_x40g_xlifid_ifnum0 0

bfd_wd_hash_table_retry_count 0

bfd_ld_hash_table_retry_count 0

x40g_sso_differ_ld_count 0

Current normal_event_qsize 0 and 0 paks crossed the limit.

***BFD Session info for ld(1) avlnode ld (1) ***

ifnum(25), slotunit(21), txtimer(1000000) detect_timer(0)

p bit(0), f bit(0), srcip(172.31.11.33) dstip(172.31.11.34)

wdog cnterid(65664) tags inner(0) outer(0) tx sess info(0x19F4B7E0)

ADJ registered(0x1) tag_count(0) tx sessid(830)

dmac(dccc.eeee.aaaa), smac(5033.eeeee.8888), rx statid(508546), tx statid(508545)

RX pkt count(5838365), TX pkt count (5208864) ƒ----- RX is not increasing

IPV6 SA(::), IPV6 DA(::), no_adj_retry_tx (0)

At this point, further you troubleshoot and a SPAN capture is recommended on the neighboring device to see if that device actually sends packets.