

# Vérification des compteurs matériels BFD sur les cartes de ligne DFC pour les périphériques 7600

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Informations générales](#)

[Topologie](#)

[Méthodologie de dépannage](#)

## Introduction

Ce document décrit comment vérifier les compteurs matériels BFD (Bidirectional Forwarding Detection) sur les cartes de ligne DFC (Distributed Forwarding Card) pour 7600 périphériques.

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Configuration et fonctionnalités des routeurs de la gamme 7600
- Configuration des modules de carte de ligne DFC

### Components Used

Les informations de ce document sont basées sur la version 15.3 de l'IOS 7600.

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.

## Informations générales

BFD est un protocole réseau conçu pour détecter les pannes de communication en moins d'une seconde dans tout type de chemin entre les systèmes (liaisons physiques directes, circuits virtuels, tunnels, LSP MPLS, etc.).

DFC signifie DFC, la principale différence entre les cartes de ligne DFC et CFC est que les cartes de ligne DFC ont une carte fille intégrée avec un moteur L2 et un moteur L3/4, ce qui décharge les recherches de transfert que sur les cartes de ligne CFC serait envoyé au superviseur pour être fait localement sur la carte de ligne.

Les paquets BFD sont traités dans le matériel de telle manière qu'ils n'affectent pas le CPU, ce qui signifie que dans les cartes DFC, ces paquets sont toujours reçus et transférés sans quitter la carte de ligne.

## Topologie

R1 (Te3/21)—R2

## Méthodologie de dépannage

Vous pouvez voir que R1 n'active pas la contiguïté BFD avec son voisin sur Tengig3/21.

Vérifiez les détails du voisin :

```
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 β-----Notice received packets are not incrementing

Tx Count: 9456 β----- 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

Vous pouvez également vérifier la même commande pour le matériel qui donne la même sortie, RX n'est pas reçu.

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
```

Après cela, vous pouvez vérifier les compteurs directement sur la carte de ligne.

Pour cela, vous avez besoin de la valeur LD (Local Discriminator) dans la sortie `show bfd neighbors details`, pour ce cas la valeur LD est 1.

LD, cette valeur est utilisée pour identifier de manière unique cette session et elle doit être unique et non nulle, pour toutes les sessions BFD de ce périphérique.

Vous **affichez le module** et voyez que la carte de ligne 3 est DFC.

Vous joignez la carte de ligne où vous voulez vérifier les valeurs BFD, dans ce cas, c'est la carte de ligne 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) B----- 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)

À ce stade, vous devez effectuer un dépannage supplémentaire et une capture SPAN est recommandée sur le périphérique voisin pour voir si ce périphérique envoie réellement des paquets.