

# 瞭解 Nexus 交換器上的循環冗餘檢查錯誤

## 目錄

---

### [簡介](#)

### [必要條件](#)

#### [需求](#)

#### [採用元件](#)

### [背景資訊](#)

#### [適用硬體](#)

### [CRC 定義](#)

### [CRC 錯誤定義](#)

### [常見 CRC 錯誤症狀](#)

#### [Windows 主機接收的錯誤數](#)

#### [Linux 主機的 RX 錯誤數](#)

#### [網路裝置的 CRC 錯誤數](#)

##### [儲存轉送網路裝置的輸入錯誤數](#)

##### [直通網路裝置的輸入和輸出錯誤數](#)

### [追蹤和隔離 CRC 錯誤](#)

### [CRC 錯誤的根本原因](#)

### [解決 CRC 錯誤](#)

### [相關資訊](#)

---

## 簡介

本文檔介紹在介面計數器上觀察到的循環冗餘檢查(CRC)錯誤以及Cisco Nexus交換機的統計資訊。

## 必要條件

### 需求

思科建議您先瞭解乙太網路交換和 Cisco NX-OS 命令列介面 (CLI) 的基礎知識。如需詳細資訊，請參閱以下其中一份適用的文件：

- [Cisco Nexus 9000 NX-OS 基礎組態指南，10.2\(x\) 版](#)
- [Cisco Nexus 9000 系列 NX-OS 基礎組態指南，9.3\(x\) 版](#)
- [Cisco Nexus 9000 系列 NX-OS 基礎組態指南，9.2\(x\) 版](#)
- [Cisco Nexus 9000 系列 NX-OS 基礎組態指南，7.x 版](#)
  
- [疑難排解乙太網路](#)

### 採用元件

本文中的資訊係根據以下軟體和硬體版本：

- 從NX-OS軟體版本9.3(8)開始的Nexus 9000系列交換機
- 從NX-OS軟體版本9.3(8)開始的Nexus 3000系列交換機

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

## 背景資訊

本文檔詳細介紹在Cisco Nexus系列交換機上的介面計數器上觀察到的循環冗餘檢查(CRC)錯誤。本文檔介紹什麼是CRC，如何在乙太網幀的幀校驗序列(FCS)欄位中使用CRC，CRC錯誤在Nexus交換機上如何顯示，以及CRC錯誤在儲存轉發交換中的互動方式。本文還介紹了直通交換方案、CRC錯誤最可能的根本原因，以及如何排除和解決CRC錯誤。

### 適用硬體

本文件的這項資訊適用於所有 Cisco Nexus 系列交換器。本文件中的部分資訊亦可適用於其他思科路由及交換平台，例如 Cisco Catalyst 路由器與交換器。

## CRC 定義

CRC 是一種錯誤偵測機制，常用於電腦和儲存網路，可識別傳輸期間變更或損毀的資料。當連接至網路的裝置需要傳輸資料時，該裝置會針對資料，根據循環代碼執行計算演算法，以產生固定長度的數字。此固定長度數字稱為 CRC 值，但該值通常簡單俗稱為 CRC。此 CRC 值會附加於該資料，並透過網路傳輸至其他裝置。此遠端裝置對資料運行相同的循環代碼演算法，並將結果值與附加在資料上的CRC進行比較。如果兩個值都匹配，則遠端裝置假定資料已在網路上傳輸而沒有損壞。如果值不匹配，則遠端裝置會認為資料在網路傳輸過程中已損壞。該資料無法受信賴且會遭捨棄。

CRC 係用於透過多項電腦網路技術（例如，有線類型和無線類型的乙太網路、權杖環、非同步傳輸模式 (ATM) 及訊框中繼）進行錯誤偵測。乙太網幀在插入32位CRC值的幀末尾（緊接在幀的有效負載之後）有一個32位Frame Check Sequence (FCS)欄位。

舉例來說，假設一個情境，其中有兩台稱為 Host-A 和 Host-B 的主機，會透過其網路介面卡 (NIC) 直接與彼此連線。Host-A 需要透過網路，將句子「This is an example」傳送至 Host-B。Host-A 會使用「This is an example」的裝載，製作目的地為 Host-B 的乙太網路訊框，並計算出該訊框的CRC值為 16 進位值 0xABCD。Host-A將CRC值0xABCD插入乙太網幀的FCS欄位中，然後將乙太網幀從Host-A NIC傳輸到Host-B。

當主機B收到此幀時，它可以與主機A完全相同的演算法來計算幀的CRC值。主機B計算幀的CRC值為十六進位制值0xABCD，該值向主機B表明乙太網幀在傳輸到主機B時未損壞。

## CRC 錯誤定義

當裝置（網路裝置或連線至網路的主機）接收到乙太網路訊框，但該訊框之 FCS 欄位中的 CRC 值，與該裝置針對該訊框計算出的 CRC 值不相符時，就會發生 CRC 錯誤。

此概念可透過舉例有效示範。假設一個情境，其中有兩台稱為 Host-A 和 Host-B 的主機，會透過其網路介面卡 (NIC) 直接與彼此連線。Host-A 需要透過網路，將句子「This is an example」傳送至 Host-B。Host-A 會使用「This is an example」的裝載，製作目的地為 Host-B 的乙太網路訊框，並計算出該訊框的 CRC 值為 16 進位值 0xABCD。Host-A 將 CRC 值 0xABCD 插入乙太網幀的 FCS 欄位中，然後將乙太網幀從 Host-A NIC 傳輸到 Host-B。

但是，將 Host-A 連線至 Host-B 的實體媒體的損壞，將會損毀該訊框的內容，致使該訊框內的句子變更為「This was an example」，而非所需之「This is an example」的裝載。

當主機 B 收到此幀時，它可以計算幀的 CRC 值，並在計算中包括損壞的負載。主機 B 計算幀的 CRC 值為 0xDEAD 的十六進位制值，這與乙太網幀的 FCS 欄位中的 0xABCD CRC 值不同。此 CRC 值間的差異，即告訴 Host-B 表示乙太網路訊框已損毀，而該訊框已傳輸至 Host-B。因此，主機 B 無法信任此乙太網幀的內容，因此可以丟棄該幀。主機 B 通常也會在其網路介面卡 (NIC) 上增加某些錯誤計數器，例如「input errors」、「CRC errors」或「RX errors」計數器。

## 常見 CRC 錯誤症狀

CRC 錯誤通常會以下列兩種方式之一顯示：

1. 連接至網路之裝置介面上的錯誤計數器的數字增加或以非零方式顯示。
2. 由於網路連線裝置丟棄損壞的幀，導致透過網路的流量丟包/幀丟失。

這些錯誤的表現方式會略有不同，具體取決於您當時所使用的裝置。以下小節將會詳細介紹每種類型的裝置。

## Windows 主機接收的錯誤數

Window 主機的 CRC 錯誤數通常會顯示為非零之「接收的錯誤數」計數器（顯示於「命令提示字元」之 netstat -e 命令的輸出內容）。Windows 主機之「命令提示字元」的非零之「接收的錯誤數」計數器範例如下：

```
<#root>
```

```
>
```

```
netstat -e
```

```
Interface Statistics
```

```
Received
```

	Sent	
Bytes	1116139893	3374201234
Unicast packets	101276400	49751195
Non-unicast packets	0	0
Discards	0	0

```
Errors                47294
0
Unknown protocols    0
```

NIC 與其對應的驅動程式必須支援 NIC 接收之 CRC 錯誤數的核算功能，使 netstat -e 命令報告之「接收的錯誤數」得以準確無誤。大部分的現代 NIC 與其對應的驅動程式支援 NIC 接收之 CRC 錯誤數的準確核算功能。

## Linux 主機的 RX 錯誤數

Linux 主機的 CRC 錯誤數通常會顯示為非零之「RX 錯誤數」計數器（顯示於 ifconfig 命令的輸出內容）。Linux 主機之非零 RX 錯誤數的範例如下：

```
<#root>
$
ifconfig eth0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.0.2.10 netmask 255.255.255.128 broadcast 192.0.2.255
    inet6 fe80::10 prefixlen 64 scopeid 0x20<link>
    ether 08:62:66:be:48:9b txqueuelen 1000 (Ethernet)
    RX packets 591511682 bytes 214790684016 (200.0 GiB)

RX errors 478920

    dropped 0 overruns 0 frame 0
    TX packets 85495109 bytes 288004112030 (268.2 GiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Linux 主機的 CRC 錯誤數通常會顯示為非零之「RX 錯誤數」計數器（顯示於 ip -s link show 命令的輸出內容）。Linux 主機之非零 RX 錯誤數的範例如下：

```
<#root>
$
ip -s link show eth0

2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1
    link/ether 08:62:66:84:8f:6d brd ff:ff:ff:ff:ff:ff
    RX: bytes  packets  errors  dropped overrun mcast
    32246366102 444908978 478920    647    0    419445867
    TX: bytes  packets  errors  dropped carrier collsns
    3352693923 30185715 0        0        0        0
    altname enp11s0
```

NIC 與其對應的驅動程式必須支援 NIC 接收之 CRC 錯誤數的核算功能，使 ifconfig 或 ip -s link show 命令報告之「RX 錯誤數」得以準確無誤。大部分的現代 NIC 與其對應的驅動程式支援 NIC 接收之 CRC 錯誤數的準確核算功能。

## 網路裝置的 CRC 錯誤數

網路裝置在兩種轉發模式中運行：

- 儲存轉發模式
- 直通轉發模式

網路裝置處理收到的CRC錯誤的方式會因轉發模式而異。此處的子部分介紹了每種轉發模式的特定行為。

### 儲存轉送網路裝置的輸入錯誤數

當以儲存轉發模式運行的網路裝置收到幀時，網路裝置可以在驗證幀的CRC值之前緩衝整個幀（「儲存」），對幀做出轉發決策，並將幀從介面傳送出去（「轉發」）。因此，當以儲存轉發模式運行的網路裝置在特定介面上收到具有錯誤CRC值的損壞幀時，它可能會丟棄該幀並在介面上增加「輸入錯誤」計數器。

換句話說，損毀的乙太網路訊框不會透過以儲存轉送模式運作的網路裝置轉送；此類訊框會在輸入時遭捨棄。

Cisco Nexus 7000 系列交換器和 7700 系列交換器會以儲存轉送模式運作。Nexus 7000 或 7700 系列交換器的非零「輸入錯誤數」計數器和非零「CRC/FCS 數」計數器的範例如下：

```
<#root>
```

```
switch#
```

```
show interface
```

```
<snip>
```

```
Ethernet1/1 is up
```

```
  RX
```

```
    241052345 unicast packets  5236252 multicast packets  5 broadcast packets
```

```
    245794858 input packets  17901276787 bytes
```

```
    0 jumbo packets  0 storm suppression packets
```

```
    0 runs  0 giants  579204 CRC/FCS  0 no buffer
```

```
    579204 input error  0 short frame  0 overrun  0 underrun  0 ignored
```

```
    0 watchdog  0 bad etype drop  0 bad proto drop  0 if down drop
```

```
    0 input with dribble  0 input discard
```

```
    0 Rx pause
```

CRC 錯誤數在 show interface counters 錯誤輸出訊息中，亦可以非零「FCS-Err」計數器顯示。此命令輸出中的「Rcv-Err」計數器也可以具有非零值，該值表示介面收到的所有輸入錯誤（CRC或其他）的總和。以下是一個示例：

```
<#root>
```

```
switch#
```

```
show interface counters errors
```

```
<snip>
```

```
-----  
Port            Align-Err    FCS-Err    Xmit-Err    Rcv-Err    UnderSize  OutDiscards  
-----  
Eth1/1          0           579204     0           579204     0           0
```

## 直通網路裝置的輸入和輸出錯誤數

當以直通轉發模式運行的網路裝置開始接收幀時，網路裝置可以對幀報頭做出轉發決定，並在接收到足夠的幀做出有效的轉發決定後立即開始從介面傳送幀。如果訊框和封包標頭位於訊框開頭，則裝置通常會先進行此轉送決定，才會接收到訊框的裝載。

乙太網路訊框的 FCS 欄位位於訊框結尾，緊接於訊框裝載的後方。因此，在直通轉發模式下運行的網路裝置在能夠計算幀的CRC時，可能已經開始從另一個介面傳輸幀。如果網路裝置針對訊框計算出的CRC，與FCS欄位中顯示的CRC值不符合，即表示網路裝置已將損毀的訊框轉送至網路。發生這種情況時，網路裝置可以增加兩個計數器：

1. 最初接收到損毀的訊框之介面上的「輸入錯誤數」計數器。
2. 傳輸損毀的訊框之所有介面上的「輸出錯誤數」計數器。對於單播流量，這通常是單個介面-但是，對於廣播、組播或未知單播流量，這可以是一個或多個介面。

這裡顯示了一個例子，其中show interface命令的輸出表明在網路裝置的乙太網1/1上收到多個損壞的幀，並且由於網路裝置的直通轉發模式而從Ethernet1/2傳送出去：

```
<#root>
```

```
switch#
```

```
show interface
```

```
<snip>
```

```
Ethernet1/1 is up
```

```
RX
```

```
46739903 unicast packets 29596632 multicast packets 0 broadcast packets  
76336535 input packets 6743810714 bytes  
15 jumbo packets 0 storm suppression bytes  
0 runts 0 giants 47294 CRC 0 no buffer  
47294 input error 0 short frame 0 overrun 0 underrun 0 ignored  
0 watchdog 0 bad etype drop 0 bad proto drop 0 if down drop  
0 input with dribble 0 input discard  
0 Rx pause
```

```
Ethernet1/2 is up
```

```
TX
```

```
46091721 unicast packets 2852390 multicast packets 102619 broadcast packets  
49046730 output packets 3859955290 bytes  
50230 jumbo packets
```

```
47294 output error 0 collision 0 deferred 0 late collision
0 lost carrier 0 no carrier 0 babble 0 output discard
0 Tx pause
```

CRC 錯誤數在 show interface counters 錯誤的輸出內容中，亦可在輸入介面上以非零「FCS-Err」計數器顯示，並在輸出介面上以非零「Xmit-Err」計數器顯示。此命令輸出中入口介面上的「Rcv-Err」計數器也可以具有非零值，該值是該介面收到的所有輸入錯誤（CRC或其他）的總和。以下是一個示例：

```
<#root>
```

```
switch#
```

```
show interface counters errors
```

```
<snip>
```

```
-----  
Port          Align-Err
```

```
FCS-Err
```

```
Xmit-Err
```

```
Rcv-Err
```

```
UnderSize OutDiscards  
-----
```

```
Eth1/1
```

```
0
```

```
47294
```

```
0
```

```
47294
```

```
0
```

```
0
```

```
Eth1/2
```

```
0
```

```
0
```

```
47294
```

```
0
```

```
0
```

```
0
```

網路裝置還可以以特定方式修改幀的FCS欄位中的CRC值，以指示上游網路裝置此幀已損壞。此行為就是「按下並標記」CRC。修改CRC的精確方式因平台而異，但通常情況下，它會反轉幀的FCS欄位中存在的當前CRC值。以下是一個示例：

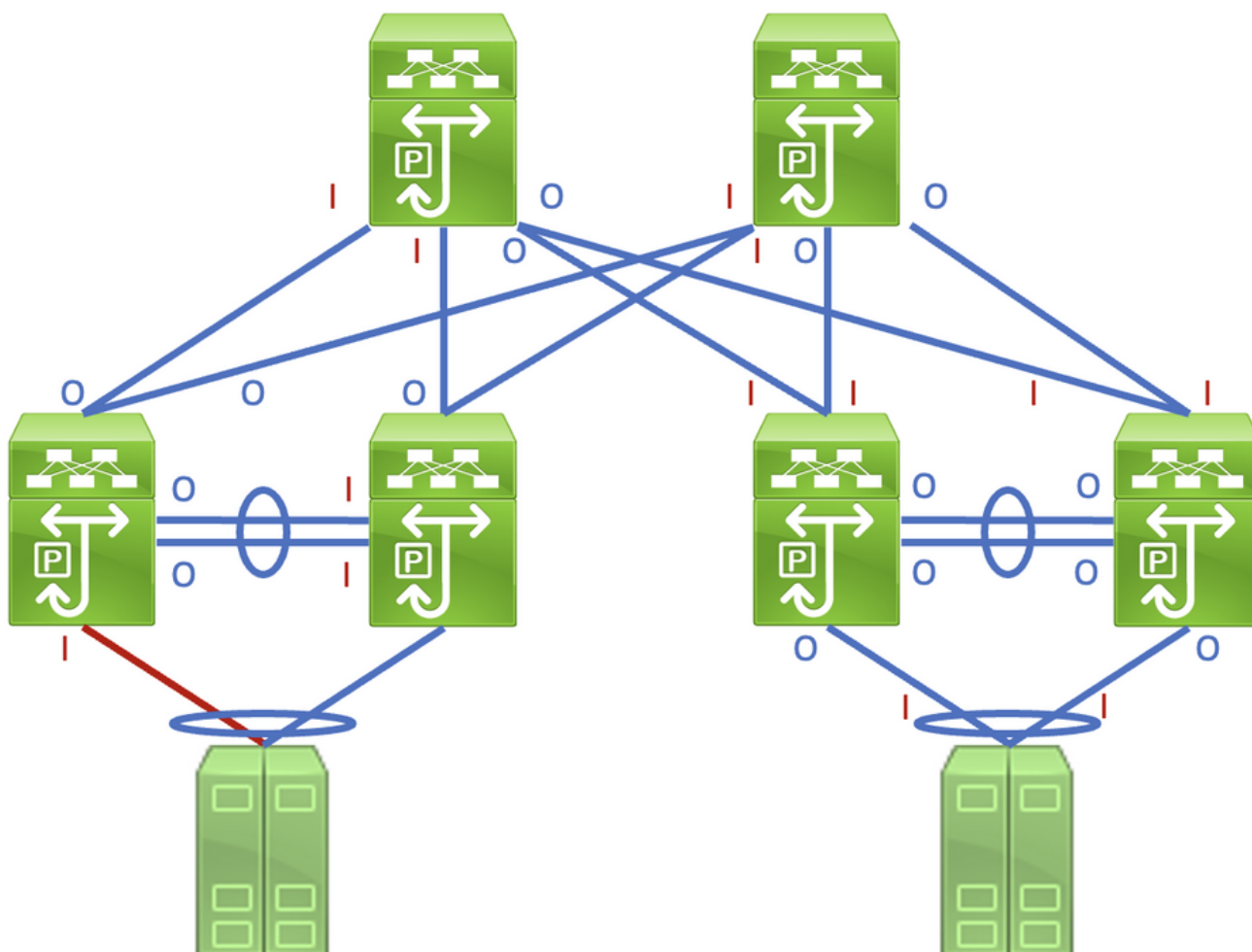
Original CRC: 0xABCD (1010101111001101)  
Stomped CRC: 0x5432 (0101010000110010)

由於此行為，以直通轉送方式運作的網路裝置可透過網路傳播損毀的訊框。如果網路包含多個以直通轉送模式運作的網路裝置，則出現一個損毀的訊框，可能會造成網路內多個網路裝置的輸入錯誤數計數器和輸出錯誤數計數器的數字增加。

## 追蹤和隔離 CRC 錯誤

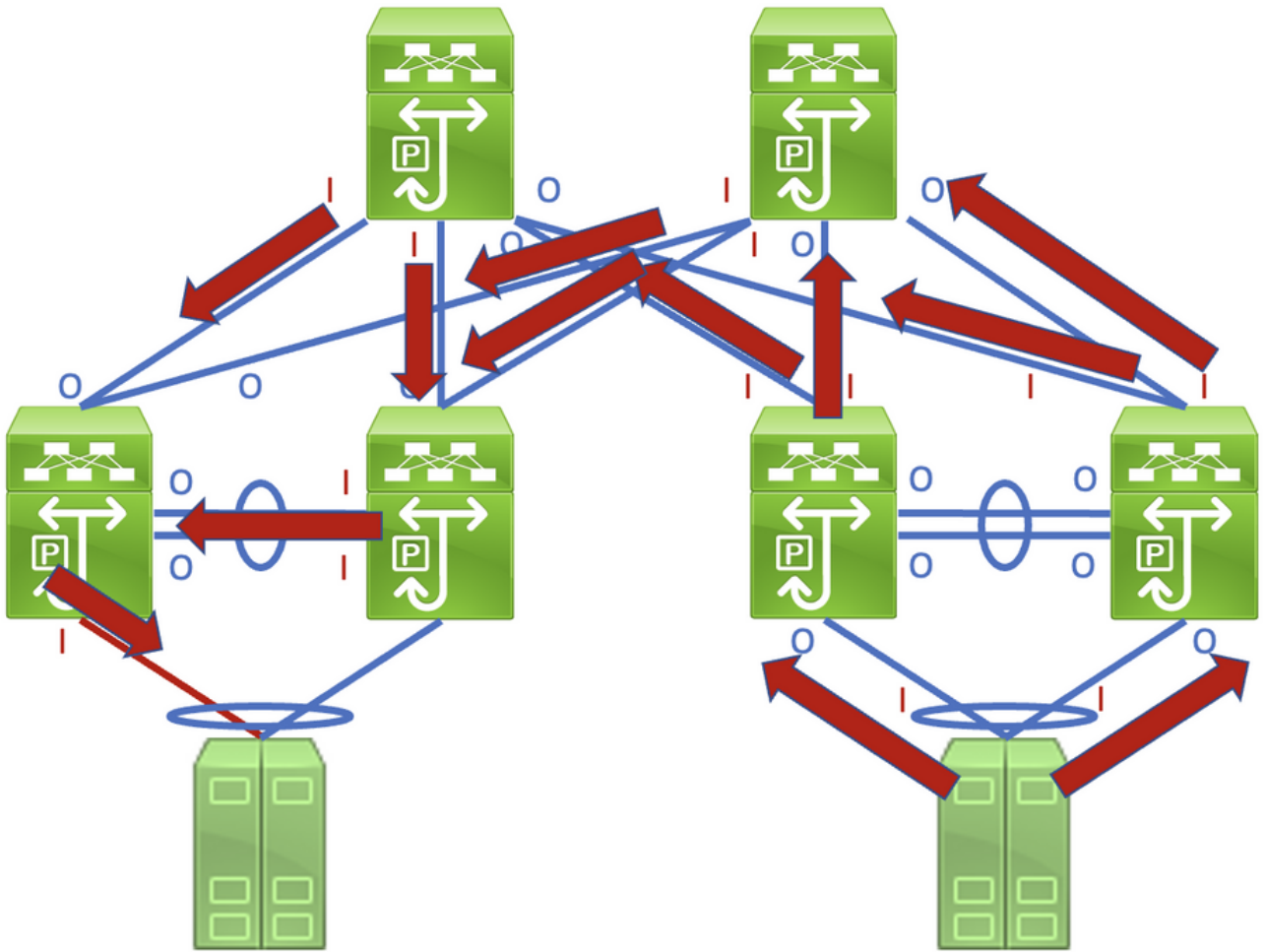
要確定並解決CRC錯誤的根本原因，第一步是將CRC錯誤的來源隔離到網路中兩台裝置之間的特定鏈路。連線到此鏈路的一個裝置可以有值為零或不遞增的介面輸出錯誤計數器，而連線到此鏈路的另一個裝置可以有非零或遞增的介面輸入錯誤計數器。這表示從某個裝置完好無損的介面流出的流量在傳輸到遠端裝置時已損壞，並且鏈路上另一個裝置的入口介面將其計為輸入錯誤。

在由以儲存轉發模式運行的網路裝置組成的網路中辨識此鏈路是一項簡單的任務。但是，如果您在包含以直通轉發模式運行的網路裝置的網路中標識此鏈路，則難度會更大，因為許多網路裝置可能具有非零輸入和輸出錯誤計數器。此現象的一個示例可以在此拓撲中看到，以紅色突出顯示的鏈路受損，因此經過該鏈路的流量受損。標示紅色「I」文字的介面，表示其可能具有非零輸入錯誤數；而標示藍色「O」文字的介面，則表示其可能具有非零輸出錯誤數。

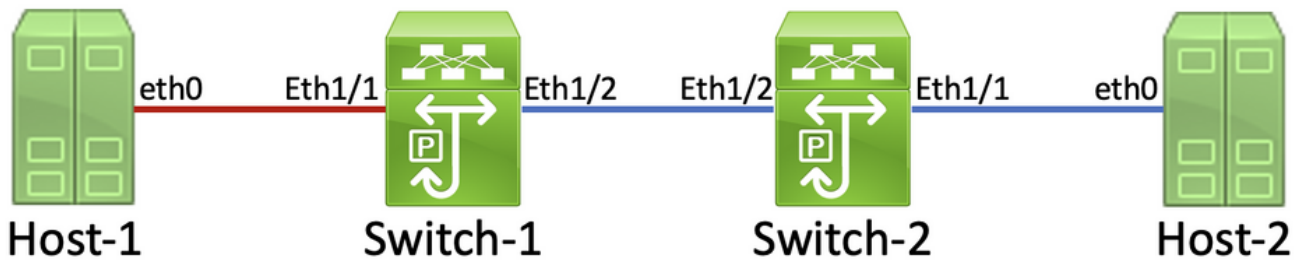




本文檔介紹在介面計數器上觀察到的循環冗餘檢查(CRC)錯誤以及Cisco Nexus交換機的統計資訊。



透過示例可以最好地演示用於跟蹤和辨識損壞鏈路的詳細過程。考量以下拓撲：



在此拓撲中，名為 Switch-1 之 Nexus 交換器的介面 Ethernet1/1 會透過 Host-1 的網路介面卡 (NIC) eth0 連線至名為 Host-1 的主機。Switch-1 的介面 Ethernet1/2 會透過 Switch-2 的介面 Ethernet1/2，連線至第二個名為 Switch-2 的 Nexus 交換器。Switch-2 的 Ethernet1/1 介面透過 Host-2 NIC eth0 連線到名為 Host-2 的主機。

透過 Switch-1 的 Ethernet1/1 介面的 Host-1 和 Switch-1 之間的鏈路損壞，導致透過該鏈路的流量間歇性損壞。但是，目前尚不清楚鏈路是否損壞。您必須透過非零或遞增的輸入和輸出錯誤計數器來跟蹤損壞的幀在網路中離開的路徑，以定位該網路中損壞的鏈路。

在本示例中，主機2 NIC 報告其正在接收CRC錯誤。

```
<#root>
```

```
Host-2$
```

```
ip -s link show eth0
```

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000  
    link/ether 00:50:56:84:8f:6d brd ff:ff:ff:ff:ff:ff
```

```
RX:
```

```
bytes packets
```

```
errors
```

```
dropped overrun mcast  
32246366102 444908978
```

```
478920
```

```
647 0 419445867
```

```
TX: bytes packets errors dropped carrier collsns  
3352693923 30185715 0 0 0 0  
altname enp11s0
```

您知道Host-2 NIC透過介面Ethernet1/1連線到Switch-2。您可使用 `show interface` 命令，確認介面 Ethernet1/1 具有非零輸出錯誤數計數器。

```
<#root>
```

```
Switch-2#
```

```
show interface
```

```
<snip>
```

```
Ethernet1/1 is up
```

```
admin state is up, Dedicated Interface
```

```
RX
```

```
30184570 unicast packets 872 multicast packets 273 broadcast packets
```

```
30185715 input packets 3352693923 bytes
```

```
0 jumbo packets 0 storm suppression bytes
```

```
0 runs 0 giants 0 CRC 0 no buffer
```

```
0 input error 0 short frame 0 overrun 0 underrun 0 ignored
```

```
0 watchdog 0 bad etype drop 0 bad proto drop 0 if down drop
```

```
0 input with dribble 0 input discard
```

```
0 Rx pause
```

```
TX
```

```
444907944 unicast packets 932 multicast packets 102 broadcast packets
```

```
444908978 output packets 32246366102 bytes
```

```
0 jumbo packets
```

```
478920 output error 0 collision 0 deferred 0 late collision
```

```
0 lost carrier 0 no carrier 0 babble 0 output discard
```

```
0 Tx pause
```

由於介面 Ethernet1/1 的輸出錯誤數計數器非零值，因此 Switch-2 的另一個介面極可能具有非零輸入錯誤數計數器。您可以使用 `show interface counters errors non-zero` 命令，以識別 Switch-2 的

任何介面是否具有非零輸入錯誤數計數器。

```
<#root>
```

```
Switch-2#
```

```
show interface counters errors non-zero
```

```
<snip>
```

```
-----  
Port          Align-Err
```

```
FCS-Err
```

```
Xmit-Err
```

```
Rcv-Err
```

```
UnderSize OutDiscards
```

```
-----  
Eth1/1          0          0
```

```
478920
```

```
0          0          0  
Eth1/2          0
```

```
478920
```

```
0
```

```
478920
```

```
0          0
```

```
-----  
Port          Single-Col Multi-Col Late-Col Exces-Col Carri-Sen Runts
```

```
-----  
Port          Giants SQETest-Err Deferred-Tx IntMacTx-Er IntMacRx-Er Symbol-Err
```

```
-----  
Port          InDiscards
```

您會發現 Switch-2 的 Ethernet1/2 具有非零輸入錯誤數計數器。如此表示 Switch-2 會在此介面上接收到損毀的流量。您可透過 Cisco Discovery Protocol (CDP) 或連結層探索通訊協定 (LLDP) 功能確認哪個裝置連線至 Switch-2 之 Ethernet1/2。以下顯示使用 show cdp neighbors 命令的範例。

```
<#root>
```

```
Switch-2#
```

```
show cdp neighbors
```

<snip>

Capability Codes: R - Router, T - Trans-Bridge, B - Source-Route-Bridge  
S - Switch, H - Host, I - IGMP, r - Repeater,  
V - VoIP-Phone, D - Remotely-Managed-Device,  
s - Supports-STP-Dispute

Device-ID                    Local Intrfce   Hldtme   Capability   Platform                    Port ID

Switch-1

(FD012345678)

Eth1/2

125      R S I s      N9K-C93180YC-

Eth1/2

您現在知道，交換器2正在其乙太網路1/2介面上從交換器1的乙太網路1/2介面接收損毀的流量，但您尚未知道，交換器1的乙太網路1/2與交換器2的乙太網路1/2之間的連結是否已損毀並導致損毀，或者交換器1是轉送損毀流量的直通交換器，但您仍不知道該交換器是否正在接收損毀的流量。您必須登入 Switch-1 以確認以上情況。

您可使用 show interfaces 命令，確認 Switch-1 的 Ethernet1/2 介面具有非零輸出錯誤數計數器。

<#root>

Switch-1#

show interface

<snip>

Ethernet1/2 is up

admin state is up, Dedicated Interface

RX

30581666 unicast packets 178 multicast packets 931 broadcast packets

30582775 input packets 3352693923 bytes

0 jumbo packets 0 storm suppression bytes

0 runts 0 giants 0 CRC 0 no buffer

0 input error 0 short frame 0 overrun 0 underrun 0 ignored

0 watchdog 0 bad etype drop 0 bad proto drop 0 if down drop

0 input with dribble 0 input discard

0 Rx pause

TX

454301132 unicast packets 734 multicast packets 72 broadcast packets

454301938 output packets 32246366102 bytes

0 jumbo packets

478920 output error

0 collision 0 deferred 0 late collision

0 lost carrier 0 no carrier 0 babble 0 output discard

0 Tx pause

您會發現 Switch-1 的 Ethernet1/2 具有非零輸出錯誤數計數器。這表示交換器1的Ethernet1/2和 Switch-2的Ethernet1/2之間的連結沒有損毀；相反，Switch-1是直通交換器，會轉送它在某些其他介面上收到損毀的流量。正如之前在Switch-2上演示的那樣，您可以使用show interface counters errors non-zero命令來辨識Switch-1的任何介面是否具有非零輸入錯誤計數器。

```
<#root>
```

```
Switch-1#
```

```
show interface counters errors non-zero
```

```
<snip>
```

```
-----
```

Port	Align-Err
<b>FCS-Err</b>	
<b>Xmit-Err</b>	
<b>Rcv-Err</b>	
	UnderSize OutDiscards
-----	
Eth1/1	0
<b>478920</b>	
	0
<b>478920</b>	
Eth1/2	0 0 0
<b>478920</b>	
	0 0 0

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

```
-----
```

Port	Single-Col	Multi-Col	Late-Col	Exces-Col	Carri-Sen	Runts
-----						
-----						
Port	Giants	SQETest-Err	Deferred-Tx	IntMacTx-Er	IntMacRx-Er	Symbol-Err
-----						
-----						
Port	InDiscards					
-----						

```
-----
```

```
-----
```

您會發現 Switch-1 的 Ethernet1/1 具有非零輸入錯誤數計數器。如此表示 Switch-1 會在此介面上接收到損毀的流量。您知道此介面連線到Host-1的eth0 NIC。您可以檢視Host-1的eth0 NIC介面統計資訊，以確認Host-1是否從此介面傳送損壞的幀。

```
<#root>
```

```
Host-1$
```

```
ip -s link show eth0
```

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000  
    link/ether 00:50:56:84:8f:6d brd ff:ff:ff:ff:ff:ff  
    RX: bytes  packets  errors  dropped overrun mcast  
       73146816142 423112898 0        0        0        437368817
```

```
TX:
```

```
bytes  packets
```

```
errors
```

```
dropped carrier collsns  
3312398924 37942624
```

```
0
```

```
0        0        0
```

```
altname enp11s0
```

Host-1的eth0 NIC統計資料表明主機不傳輸損壞的流量。如此表示 Host-1 的 eth0 和 Switch-1 的 Ethernet1/1 之間的連結已損壞，且為此流量損毀的來源。您需要對此連結進行故障排除，以辨識導致此損壞的故障元件並更換它。

#### CRC 錯誤的根本原因

最常見的 CRC 錯誤根本原因為兩個裝置間之實體連結的元件損壞或故障。範例包括：

- 故障或損壞的實體媒體（銅纜或光纖）或直接連接纜線 (DAC)。
- 故障或損壞的收發器/光纖。
- 故障或損壞的跳線面板連接埠。
- 網路裝置硬體故障（包括特定埠、板卡專用積體電路[ASIC]、介質訪問控制[MAC]、交換矩陣模組等）。
- 插入主機的故障網路介面卡。

一或多個設定錯誤的裝置亦可能意外造成網路內的 CRC 錯誤。其中一個範例是網路中兩個或多個裝置之間的最大傳輸單位(MTU)組態不相符，導致大型封包被錯誤截斷。當您辨識並解決此配置問題時，它還可以更正網路中的CRC錯誤。

#### 解決 CRC 錯誤

您可透過排除程序識別特定的故障元件：

- 將實體媒體（銅纜或光纖）或 DAC，更換為已知正常之相同類型的實體媒體。
- 將插入一個裝置介面的收發器更換為相同型號的已知良好收發器。如果這不能解決CRC錯誤，請用相同型號的已知良好的收發器替換插入其它裝置介面的收發器。
- 如果任何跳線面板使用於損壞連結的一部份，請將該連結移動至跳線面板上已知正常的連接埠。或者，如果可能的話，也可以透過連線沒有配線面板的連結來消除配線面板作為潛在的根本原因。
- 將損壞的連結移動至各裝置上其他已知正常的連接埠。您可能需要測試多個不同的埠，以隔離MAC、ASIC或板卡故障。
- 如果損壞的連結涉及主機，請將連結移動至主機上的其他 NIC。或者，將損壞的鏈路連線到已知良好的主機，以隔離主機網絡卡故障。

如果故障元件是思科產品（例如思科網路裝置或收發器），並且受有效支援合約的覆蓋，則您可以[向思科TAC提交支援請求](#)，包括您的問題詳細資訊，並透過退貨授權(RMA)更換故障元件。

#### 相關資訊

- [Nexus 9000 Cloud Scale ASIC CRC 識別和追蹤程序](#)
- [思科技術支援與下載](#)

## 關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。