

了解 Nexus 交换机的循环冗余校验错误

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[适用硬件](#)

[CRC定义](#)

[CRC错误定义](#)

[CRC错误的常见症状](#)

[在Windows主机上收到错误](#)

[Linux主机上的RX错误](#)

[网络设备上的CRC错误](#)

[存储转发网络设备上的输入错误](#)

[直通网络设备的输入和输出错误](#)

[跟踪和隔离CRC错误](#)

[CRC错误的根本原因](#)

[解决CRC错误](#)

[相关信息](#)

简介

本文档介绍在接口计数器上观察到的循环冗余校验(CRC)错误以及Cisco Nexus交换机的统计信息。

先决条件

要求

思科建议您了解以太网交换和Cisco NX-OS命令行界面(CLI)的基础知识。有关详细信息，请参阅以下适用文档之一：

- [Cisco Nexus 9000 NX-OS基础配置指南，版本10.2\(x\)](#)
- [Cisco Nexus 9000系列NX-OS基础配置指南，版本9.3\(x\)](#)
- [Cisco Nexus 9000系列NX-OS基础配置指南9.2\(x\)版](#)
- [Cisco Nexus 9000系列NX-OS基础配置指南7.x版](#)
- [排除以太网故障](#)

使用的组件

本文档中的信息基于以下软件和硬件版本：

- 从NX-OS软件版本9.3(8)开始的Nexus 9000系列交换机
- 从NX-OS软件版本9.3(8)开始的Nexus 3000系列交换机

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

本文档介绍有关在Cisco Nexus系列交换机上的接口计数器上观察到的循环冗余校验(CRC)错误的详细信息。本文档介绍什么是CRC，它如何在以太网帧的帧校验序列(FCS)字段中使用，CRC错误如何在Nexus交换机上显示，以及CRC错误如何在存储转发交换中交互。本文还介绍了直通交换场景、CRC错误最可能的根本原因以及如何对CRC错误进行故障排除和解决。

适用硬件

本文档中的信息适用于所有Cisco Nexus系列交换机。本文档中的某些信息也适用于其他思科路由和交换平台，例如Cisco Catalyst路由器和交换机。

CRC定义

CRC是一种错误检测机制，通常用于计算机和存储网络，用于识别传输过程中发生更改或损坏的数据。当连接到网络的设备需要传输数据时，该设备会根据产生固定长度数字的数据运行基于循环代码的计算算法。这个固定长度的数字称为CRC值，但通常简称为CRC。此CRC值附加到数据上，并通过网络传输到另一台设备。此远程设备对数据运行相同的循环代码算法，并将结果值与附加到数据的CRC进行比较。如果两个值都匹配，则远程设备会假定数据是通过网络传输而没有损坏。如果两者的值不匹配，则远程设备会认为数据在通过网络传输时已损坏。此损坏的数据不可信，将被丢弃。

CRC用于多种计算机网络技术的错误检测，例如以太网（有线和无线两种类型）、令牌环、异步传输模式(ATM)和帧中继。以太网帧在插入32位CRC值的帧末尾有一个32位帧校验序列(FCS)字段（紧接帧的有效负载之后）。

例如，假设名为Host-A和Host-B的两台主机通过其网络接口卡(NIC)直接相连。主机A需要通过网络向主机B发送句子“这是一个示例”。主机A设计了一个以有效负载“这是一个示例”为目标到主机B的以太网帧，并计算出该帧的CRC值为十六进制值0xABCD。Host-A在以太网帧的FCS字段中插入CRC值0xABCD，然后将以太网帧从Host-A NIC传输到Host-B。

当主机B收到此帧时，它可以使用与主机A完全相同的算法来计算帧的CRC值。主机B计算帧的CRC值为0xABCD，它向主机B表明该以太网帧在传输到主机B时未损坏。

CRC错误定义

当设备（网络设备或连接到网络的主机）收到以太网帧时，该帧的FCS字段中的CRC值与该设备为

该帧计算出的CRC值不匹配，就会发生CRC错误。

此概念最好通过示例进行说明。假设名为Host-A和Host-B的两台主机通过其网络接口卡(NIC)直接相连。主机A需要通过网络向主机B发送句子“这是一个示例”。主机A设计一个以有效负载“这是一个示例”为目的的主机B的以太网帧，并计算该帧的CRC值为十六进制值0xABCD。Host-A在以太网帧的FCS字段中插入CRC值0xABCD，然后将以太网帧从Host-A NIC传输到Host-B。

但是，连接主机A和主机B的物理介质损坏会损坏帧的内容，使得帧中的句子更改为“这是一个示例”，而不是期望的有效负载“这是一个示例”。

当主机B收到此帧时，它可以计算该帧的CRC值，并在计算中包含损坏的负载。主机B计算帧的CRC值为0xDEAD的十六进制值，这与以太网帧的FCS字段中的0xABCD CRC值不同。CRC值的差异告知主机B，该以太网帧在传输到主机B时已损坏。因此，主机B无法信任此以太网帧的内容，因此可以丢弃该帧。主机B通常也会在其网络接口卡(NIC)上增加某种类型的错误计数器，例如“输入错误”、“CRC错误”或“RX错误”计数器。

CRC错误的常见症状

CRC错误通常表现为以下两种方式之一：

1. 网络连接设备接口上的错误计数器递增或非零错误。
2. 由于网络设备丢弃损坏的帧，导致流经网络的流量丢失数据包/帧。

这些错误的表现方式略有不同，具体取决于您当时使用的设备。以下小节针对每种类型的设备进行了详细介绍。

在Windows主机上收到错误

Windows主机上的CRC错误通常表现为非零的Received Errors计数器，该计数器显示在来自命令提示符的netstat -e命令的输出中。Windows主机命令提示符中的非零接收错误计数器的示例如下：

```
<#root>
```

```
>
```

```
netstat -e
```

```
Interface Statistics
```

```
Received
```

| | Sent | |
|---------------------|------------|------------|
| Bytes | 1116139893 | 3374201234 |
| Unicast packets | 101276400 | 49751195 |
| Non-unicast packets | 0 | 0 |
| Discards | 0 | 0 |
| Errors | 47294 | |

NIC及其各自的驱动程序必须支持由NIC接收的CRC错误的记帐，以便由netstat -e命令报告的已接收错误数是准确的。大多数现代NIC及其各自的驱动程序都支持对NIC收到的CRC错误进行准确的统计。

Linux主机上的RX错误

Linux主机上的CRC错误通常显示为ifconfig命令输出中显示的非零“RX错误”计数器。Linux主机上的非零RX错误计数器的示例如下：

```
<#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主机上的CRC错误也可能表现为在ip -s link show命令的输出中显示的一个非零“RX错误”计数器。Linux主机上的非零RX错误计数器的示例如下：

```
<#root>
```

```
$
```

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

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
    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
```

为了使由ifconfig或ip -s link show命令报告的RX错误数量准确，NIC及其各自的驱动程序必须支持对

NIC接收到的CRC错误进行记帐。大多数现代NIC及其各自的驱动程序都支持对NIC收到的CRC错误进行准确的统计。

网络设备上的CRC错误

网络设备工作在两种转发模式之一：

- 存储转发转发模式
- 直通转发模式

网络设备处理收到的CRC错误的方式因转发模式而异。本节小节介绍每种转发模式的具体行为。

存储转发网络设备上的输入错误

当在存储转发模式下运行的网络设备收到帧时，网络设备可以在验证帧的CRC值之前缓冲整个帧（“存储”），对帧做出转发决策，并将帧从接口发送出去（“转发”）。因此，当在存储转发模式下运行的网络设备在特定接口上收到包含错误CRC值的损坏帧时，它可能会丢弃该帧并增加接口上的“输入错误”计数器。

换句话说，损坏的以太网帧不会由运行在存储转发模式下的网络设备转发；它们会在入口丢弃。

Cisco Nexus 7000和7700系列交换机在存储转发模式下运行。Nexus 7000或7700系列交换机的非零输入错误计数器和非零CRC/FCS计数器的示例如下：

```
<#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错误还可能会在show interface counters errors的输出中显示为非零“FCS-Err”计数器。此命令输出中的“Rcv-Err”计数器也可能具有非零值，该值是接口收到的所有输入错误（CRC或其他错误）的总和。相应示例如下：

```
<#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 |

直通网络设备的输入和输出错误

当以直通转发模式运行的网络设备开始接收帧时，网络设备可以对帧报头做出转发决策，并在收到足够帧做出有效转发决策后立即开始将帧从接口发送出去。由于帧头和数据包报头位于帧的开始位置，因此通常会在收到帧的负载之前做出转发决策。

以太网帧的FCS字段位于帧的末尾，紧跟在该帧的负载之后。因此，在直通转发模式下运行的网络设备在能够计算帧的CRC时，可能已经从另一个接口开始传输帧。如果网络设备为帧计算出的CRC与FCS字段中的CRC值不匹配，则意味着网络设备将损坏的帧转发到网络。发生这种情况时，网络设备可增加两个计数器：

1. 最初收到损坏帧的接口上的“Input Errors”计数器。
2. 传输损坏帧的所有接口上的“输出错误”计数器。对于单播流量，这通常可以是单个接口-但是对于广播、组播或未知单播流量，这可以是一个或多个接口。

这里显示了一个示例，其中show interface命令的输出表明，由于网络设备的直通转发模式，在网络设备的Ethernet1/1上收到多个损坏的帧并从Ethernet1/2发送出去：

```
<#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错误还可能会将自己显示为show interface counters错误输出中，输入接口上的非零“FCS-Err”计数器和输出接口上的非零“Xmit-Err”计数器。此命令输出中入口接口上的“Rcv-Err”计数器也可能有一个非零值，该值是接口收到的所有输入错误（CRC或其他）的总和。相应示例如下：

```
<#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
```

网络设备也可以以特定方式修改帧的FCS字段中的CRC值，这表示上游网络设备此帧已损坏。此行为称为“堆叠”CRC。CRC的精确修改方式因平台而异，但通常情况下，它会反转帧的FCS字段中出现的当前CRC值。以下是一个示例：

```
Original CRC: 0xABCD (1010101111001101)
```

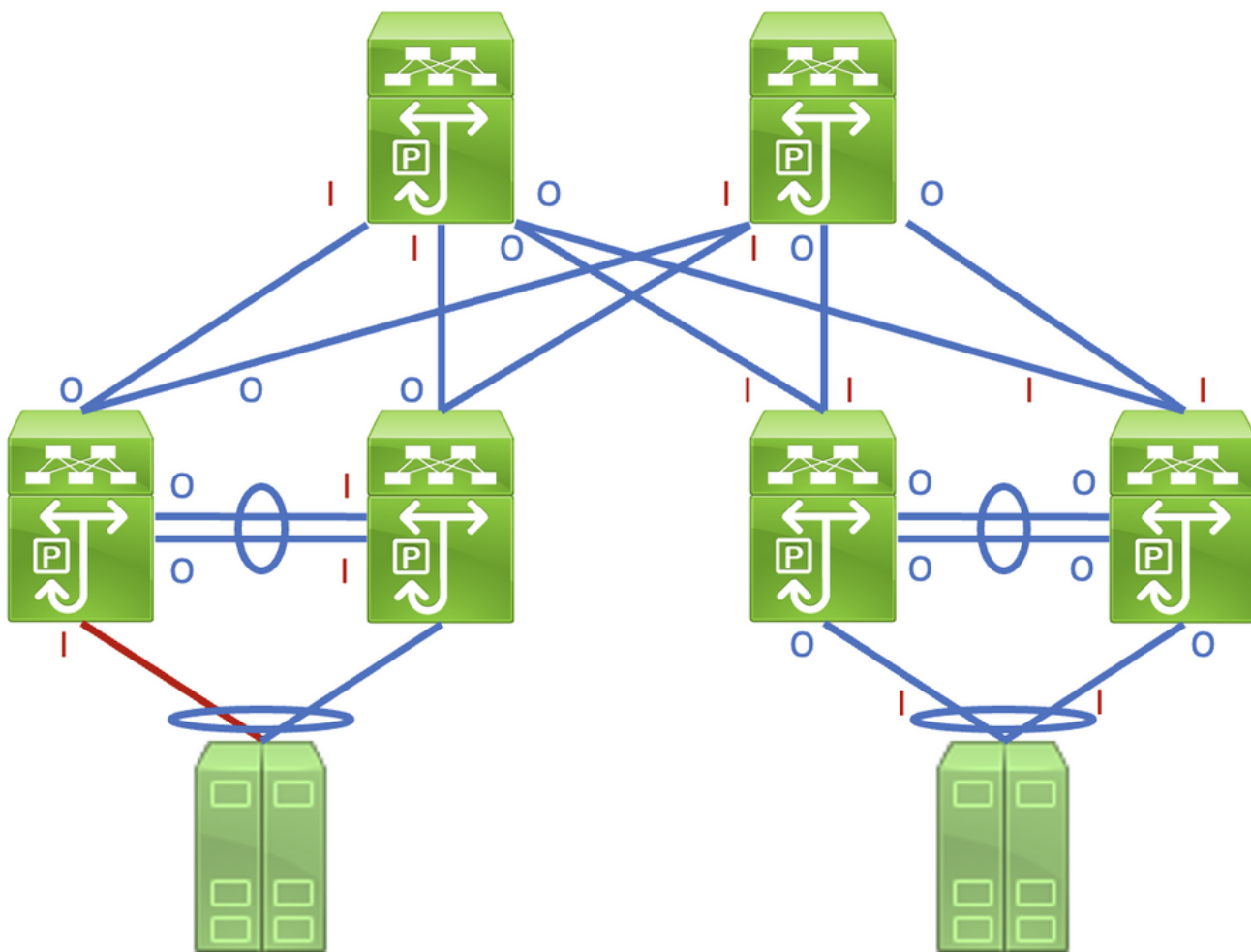
```
Stomped CRC: 0x5432 (0101010000110010)
```

由于此行为，在直通转发模式下运行的网络设备可能会在整个网络中传播损坏的帧。如果网络由工作在直通转发模式下的多个网络设备组成，则单个损坏的帧可能会导致网络中的多个网络设备上的输入错误和输出错误计数器增加。

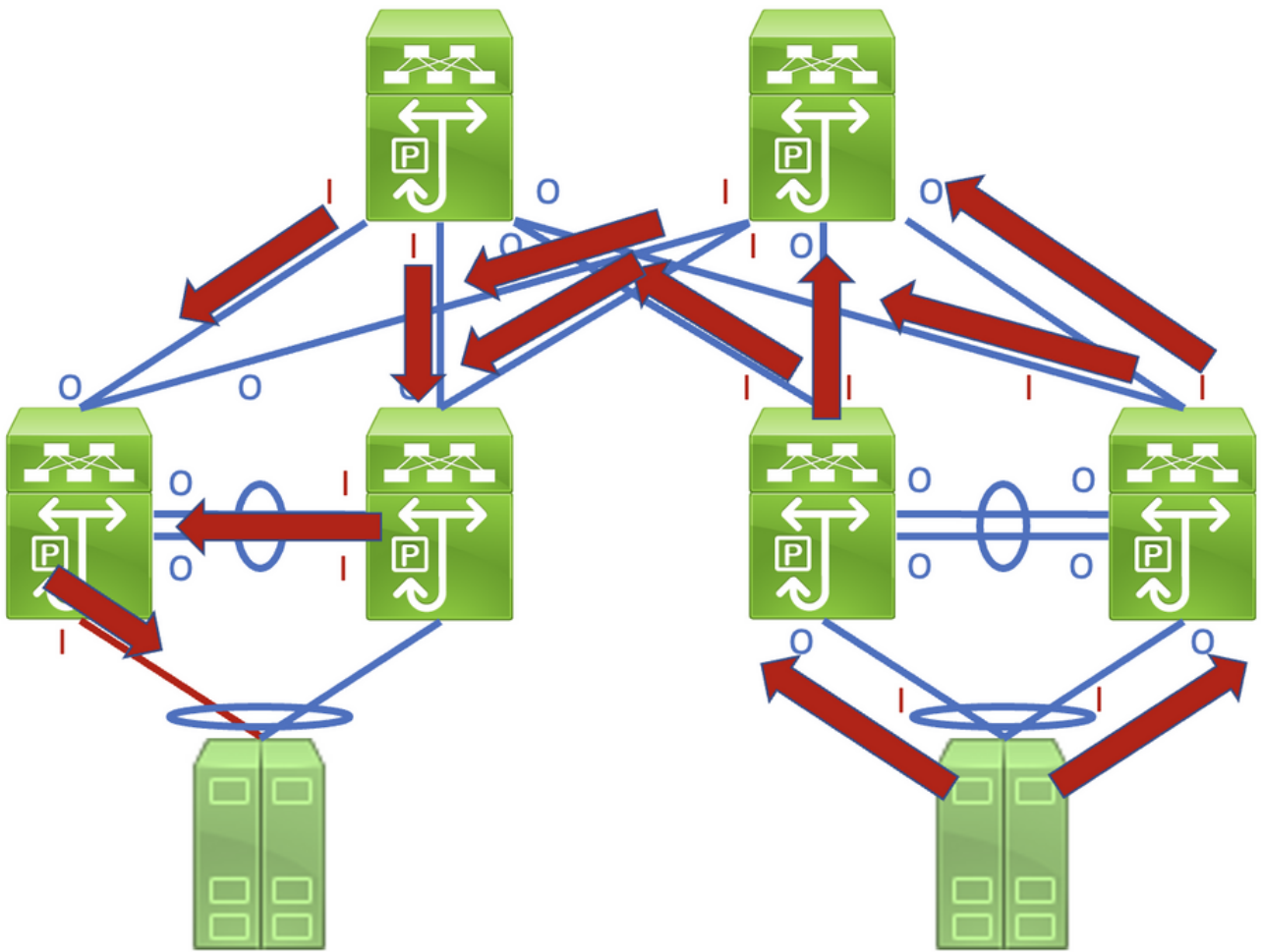
跟踪和隔离CRC错误

要确定并解决CRC错误的根本原因，第一步是将CRC错误的来源隔离到网络中两台设备之间的特定链路。连接到此链路的一台设备的接口输出错误计数器的值为零或未增加，而连接到此链路的另一台设备的接口输入错误计数器的值为非零或增加。这表明，将一个设备完整接口流出的流量在向远程设备传输时已损坏，链路上另一设备的入口接口将其计为输入错误。

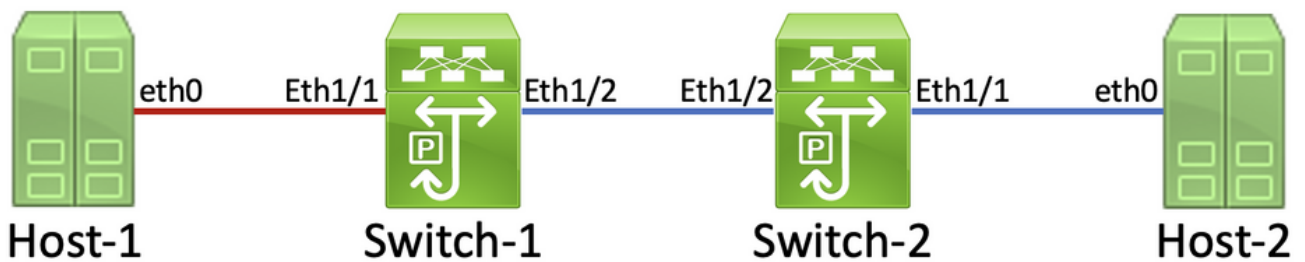
在由存储转发转发模式下运行的网络设备组成的网络中识别此链路是一项非常简单的任务。但是，如果您在由以直通转发模式运行的网络设备组成的网络中识别此链路，则会更加困难，因为许多网络设备可能具有非零输入和输出错误计数器。此现象的一个示例可以在此处的拓扑中看到，其中以红色突出显示的链路已损坏，因此流经该链路的流量已损坏。标有红色“|”的接口表示可能有非零输入错误的接口，标有蓝色“O”的接口表示可能有非零输出错误的接口。



本文档介绍在接口计数器上观察到的循环冗余校验(CRC)错误以及Cisco Nexus交换机的统计信息。



通过示例最好地演示了用于跟踪和识别已损坏链路的详细过程。请思考以下拓扑：



在此拓扑中，名为Switch-1的Nexus交换机的接口Ethernet1/1通过Host-1的网络接口卡(NIC) eth0连接到名为Host-1的主机。Switch-1的接口Ethernet1/2通过Switch-2的接口Ethernet1/2连接到第二台Nexus交换机，名为Switch-2。Switch-2的接口Ethernet1/1通过Host-2的NIC eth0连接到主机Host-2。

通过交换机1的Ethernet1/1接口的主机1和交换机1之间的链路已损坏，导致流经该链路的流量间歇性损坏。但是，目前还不清楚链路是否损坏。您必须通过非零或递增的输入和输出错误计数器来跟踪损坏帧在网络中留下的路径，以定位此网络中损坏的链路。

在本例中，主机2网卡报告收到CRC错误。

<#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的网卡通过接口Ethernet1/1连接到Switch-2。您可以使用show interface命令确认接口Ethernet1/1是否具有非零输出错误计数器。

<#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 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

```
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
```

由于接口Ethernet1/1的输出错误计数器不是零，因此交换机2的另一个接口很可能有非零输入错误计数器。您可以使用show interface counters errors non-zero命令识别交换机2的任何接口是否具有非零输入错误计数器。

```
<#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  
-----
```

您可以看到Switch-2的Ethernet1/2有非零输入错误计数器。这表明Switch-2正在此接口上接收损坏的流量。您可以通过Cisco发现协议(CDP)或链路本地发现协议(LLDP)功能确认哪个设备连接到Switch-2的Ethernet1/2。这里通过show cdp neighbors命令可以显示一个相关示例。

```
<#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
```

现在您已经知道，交换机2的以太网接口1/2正在接收来自交换机1的以太网接口1/2的损坏流量，但是您还不知道交换机1的以太网接口1/2和交换机2的以太网接口1/2之间的链路是否损坏并导致了损坏，或者交换机1是转发所接收损坏流量的直通交换机。您必须登录Switch-1进行验证。

您可以使用show interfaces命令确认Switch-1的Ethernet1/2接口有非零输出错误计数器。

<#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 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

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

您可以看到Switch-1的Ethernet1/2有非零输出错误计数器。这表明，交换机1的以太网接口1/2和交换机2的以太网接口1/2之间的链路未损坏；相反，交换机1是直通交换机，转发其它接口上接收的损坏流量。正如之前在交换机2上演示的一样，您可以使用show interface counters errors non-zero命令来识别交换机1的任何接口是否具有非零输入错误计数器。

<#root>

Switch-1#

show interface counters errors non-zero

<snip>

Port Align-Err

FCS-Err

Xmit-Err

Rcv-Err

 UnderSize OutDiscards

Eth1/1 0

478920

0

478920

0

0

Eth1/2 0 0

0

0

478920

0

0

0

Port Single-Co1 Multi-Co1 Late-Co1 Exces-Co1 Carri-Sen Runts

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

Port InDiscards

您可以看到Switch-1的Ethernet1/1有非零输入错误计数器。这表示交换机1正在此接口上接收损坏的流量。您知道此接口连接到Host-1的eth0 NIC。您可以查看Host-1的eth0 NIC接口统计信息，以确认Host-1是否从此接口发送损坏的帧。

<#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统计信息表明主机未传输损坏的流量。这表明Host-1的eth0和Switch-1的Ethernet1/1之间的链路已损坏，并且是此流量损坏的来源。您需要对此链接进行故障排除，找出导致此损坏的故障组件并更换它。

CRC错误的根本原因

CRC错误最常见的根本原因是两台设备之间物理链路的损坏或故障。示例包括：

- 物理介质（铜缆或光纤）或直接连接电缆(DAC)出现故障或损坏。
- 收发器/光纤故障或损坏。
- 配线面板端口出现故障或损坏。
- 网络设备硬件故障（包括特定端口、板卡专用集成电路[ASIC]、介质访问控制[MAC]、交换矩阵模块等）。
- 插入主机的网络接口卡出现故障。

一台或多台配置错误的设备也可能在无意中造成网络中的CRC错误。例如，网络中两台或多台设备之间的最大传输单元(MTU)配置不匹配，导致大数据包被错误截断。当您确定和解决此配置问题时，它还可以更正网络中的CRC错误。

解决CRC错误

您可以通过消除过程来识别特定故障组件：

- 使用同类已知良好的物理介质更换物理介质（铜缆或光纤）或DAC。
- 将插入一个设备接口的收发器更换为相同型号的已知良好收发器。如果这无法解决CRC错误，请使用相同型号的已知良好的收发器替换插入其他设备接口的收发器。
- 如果损坏的链路使用了任何配线面板，请将链路移至配线面板上已知良好的端口。或者，如果可能的话，通过连接链路而不连接配线面板，来消除配线面板作为潜在根本原因的可能性。
- 将损坏的链路移到每台设备上另一个已知良好的端口。您可能需要测试多个不同端口来隔离MAC、ASIC或线卡故障。
- 如果损坏的链路涉及主机，请将链路移到主机上的其他网卡。或者，将损坏的链路连接到已知良好的主机，以隔离主机网卡故障。

如果故障组件是思科产品（例如思科网络设备或收发器），受有效支持合同覆盖，您可以[向思科TAC提交支持案例](#)，包括您的问题详细信息，并通过退货授权(RMA)更换故障组件。

相关信息

- [Nexus 9000云规模ASIC CRC识别和跟踪程序](#)
- [思科技术支持和下载](#)

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。