

Identificação e procedimento de rastreamento de ASIC CRC em escala de nuvem do Nexus 9000

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Hardware aplicável](#)

[Identificação e procedimento de rastreamento de CRC em escala de nuvem do Cisco Nexus 9200 e 9300](#)

[Software NX-OS versão 10.2\(1\) e posterior](#)

[Software NX-OS versão 10.1\(2\) e anterior](#)

[Etapa 1. Identificar o incremento de contadores CRC em interface\(s\) física\(is\)](#)

[Etapa 2. Mapear a interface física para a subporta ASIC, bloco MAC e bloco Mac](#)

[Etapa 3. Verificar Registros ASIC de Escala de Nuvem para Contadores Relacionados a CRC](#)

[Escala de nuvem do Cisco Nexus 9500 - Procedimento de identificação e rastreamento de CRC em switches modulares](#)

[Etapa 1. Mapeie os links internos entre as placas de linha e os módulos de estrutura.](#)

[Etapa 2. Verifique os contadores CRC nos enlaces Eth e rastreie a origem dos quadros corrompidos.](#)

[Examples](#)

[Cenário 1. Interface Física Recebendo CRCs Estampados](#)

[Etapa 1. Confirmar Incrementação de CRCs](#)

[Etapa 2. Mapear a interface física para a subporta ASIC, bloco MAC e bloco MAC](#)

[Etapa 3. Verificar Registros ASIC de Escala de Nuvem para Contadores Relacionados a CRC](#)

[Conclusão do cenário 1](#)

[Cenário 2. A Interface Física Recebeu Quadros Malformados com CRC Inválido](#)

[Etapa 1. Confirmar Incrementação de CRCs](#)

[Etapa 2. Mapear a interface física para a subporta ASIC, bloco MAC e bloco MAC](#)

[Etapa 3. Verificar Registros ASIC de Escala de Nuvem para Contadores Relacionados a CRC](#)

[Conclusão do cenário 2](#)

[Cenário 3. Syslog de erros do Nexus 9500 Eth CRC](#)

[Etapa 1. Mapear o Eth Link no Módulo de Estrutura para a Placa de Linha Conectada](#)

[Etapa 2. Verificar se os CRCs Recebidos no Enlace Eth são Inválidos ou Estão Parados](#)

[Etapa 3. Rastreie a origem dos quadros com CRCs inválidos na placa de linha de ingresso](#)

[Conclusão do cenário 3](#)

[Cenário 4. Rastreie a origem de quadros CRC inválidos com interface de saída.](#)

[Etapa 1. Identificar o módulo de estrutura que envia quadros CRC inválidos à placa de linha de saída](#)

[Etapa 2. Mapear o enlace Eth no módulo de estrutura para a placa de linha conectada e verificar se há CRCs estampados](#)

[Etapa 3. Rastrear a origem dos quadros com CRCs inválidos no módulo de entrada](#)

[Conclusão do cenário 4](#)

[Informações Relacionadas](#)

Introduction

Este documento descreve as etapas usadas para rastrear a origem de erros de CRC observados em interfaces físicas em uma série de módulos Cisco Nexus 9000 Cloud Scale ASIC. Este documento também descreve o procedimento usado para diferenciar erros de CRC estompados e não estompados observados em interfaces físicas e links de estrutura interna de switches Nexus modulares.

Prerequisites

Requirements

A Cisco recomenda que você compreenda os conceitos básicos de switching cut-through e store-and-forward. A Cisco também recomenda que você compreenda os conceitos básicos do campo Ethernet FCS (Frame Check Sequence) e do algoritmo CRC (Cyclic Redundancy Check) usados pelo campo FCS. Para obter mais informações, consulte os seguintes documentos:

- [Switching Ethernet cut-through e store-and-forward para ambientes de baixa latência](#)

Componentes Utilizados

As informações neste documento são baseadas nos switches Cisco Nexus 9000 Series com o ASIC de escala de nuvem executando o software NX-OS versão 7.0(3)I7(8).

As informações apresentadas neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. 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.

Informações de Apoio

Os switches Cisco Nexus 9000 Series usam switching cut-through por padrão. O switching cut-through é onde um switch toma uma decisão de encaminhamento em um quadro e começa a encaminhar o quadro a partir de uma interface de saída assim que o switch tiver processado o suficiente do cabeçalho do quadro para tomar uma decisão de encaminhamento válida. Isso difere do switching store-and-forward, em que um switch armazena o quadro inteiro em buffer antes de encaminhá-lo para fora de uma interface de saída.

O campo FCS de um quadro Ethernet valida a integridade do quadro e garante que o quadro não seja corrompido em trânsito. O campo FCS de um quadro Ethernet está localizado no final do quadro Ethernet atrás do payload do quadro. Um switch que opera em um modo de switching store-and-forward pode verificar a integridade de um quadro Ethernet com o campo FCS antes de encaminhar o quadro de uma interface de saída (ou descartar o quadro se o campo FCS tiver conteúdo inválido). No entanto, um switch que opera em modo de comutação cut-through não é capaz de verificar a integridade de um quadro Ethernet com o campo FCS antes de encaminhar o

quadro para fora de uma interface de saída; em outras palavras, quando um switch cut-through for capaz de verificar a integridade de um quadro Ethernet, a maioria do quadro Ethernet já terá sido encaminhada de uma interface de saída.

Se um switch que opera em um modo de switching cut-through receber um quadro Ethernet com um campo FCS inválido, o switch executará as seguintes ações:

1. Reescreva o campo FCS do quadro Ethernet com o inverso bit a bit do valor atual (incorreto) do campo FCS. Se o quadro precisar ser roteado, o valor atual (incorreto) do campo FCS será calculado depois que o cabeçalho Ethernet do quadro for reescrito. Essa ação é conhecida como "pisar" no CRC.
2. Encaminhe o restante do quadro Ethernet (junto com o CRC estombado) para fora da interface de saída de acordo com a decisão de encaminhamento tomada no quadro.
3. Incremente o contador de erros de entrada e/ou o contador de erros de CRC na interface de entrada.

Este documento descreve as etapas para verificar se os contadores CRC associados a uma interface de entrada são CRCs normais (que geralmente indicam problemas de camada física no link conectado à interface de entrada) ou CRCs estompados (que indicam que o dispositivo conectado à interface de entrada também está operando em um modo de switching cut-through e recebeu um quadro Ethernet malformado).

Hardware aplicável

O procedimento abordado neste documento é aplicável somente a este hardware:

- **Switches fixos Nexus 9200/9300** N9K-C92160YC-XN9K-C92300YCN9K-C92304QCN9K-C92348GC-XN9K-C9236CN9K-C9272QN9K-C9332CN9K-C9364CN9K-C93108TC-EXN9K-C93108TC-EX-24N9K-C93180LC-EXN9K-C93180YC-EXN9K-C93180YC-EX-24N9K-C93108TC-FXN9K-C93108TC-FX-24N9K-C93180YC-FXN9K-C93180YC-FX-24N9K-C9348GC-FXPN9K-C93240YC-FX2N9K-C93216TC-FX2N9K-C9336C-FX2N9K-C9336C-FX2-EN9K-C93360YC-FX2N9K-C93180YC-FX3N9K-C93108TC-FX3PN9K-C93180YC-FX3SN9K-C9316D-GXN9K-C93600CD-GXN9K-C9364C-GXN9K-C9364D-GX2AN9K-C9332D-GX2B
- **Placas de linha do switch modular Nexus 9500** N9K-X97160YC-EXN9K-X9732C-EXN9K-X9736C-EXN9K-X97284YC-FXN9K-X9732C-FXN9K-X9788TC-FXN9K-X9716D-GX

Identificação e procedimento de rastreamento de CRC em escala de nuvem do Cisco Nexus 9200 e 9300

Esta seção do documento descreverá instruções passo a passo para identificar a origem de erros de CRC observados em uma interface física específica Ethernet1/1 nos switches Cisco Nexus 9200 e 9300 Series.

Software NX-OS versão 10.2(1) e posterior

Começando com o software NX-OS versão 10.2(1), os switches Nexus equipados com o ASIC Cloud Scale têm um novo contador de interface para pacotes com um CRC estombado no campo FCS de quadros Ethernet que atravessam o switch. Você pode usar o comando **show interface**

para identificar interfaces físicas com o aumento de contadores CRC diferentes de zero e CRC estompados. Um exemplo disso é mostrado aqui, onde a interface física Ethernet1/1 tem um contador de CRC zero e um contador de CRC estompado diferente de zero, que indica que os quadros com um CRC inválido e estompado foram recebidos nessa interface.

```
switch# show interface
<snip>
Ethernet1/1 is up
admin state is up, Dedicated Interface
Hardware: 100/1000/10000/25000 Ethernet, address: 00d7.8f86.2bbe (bia 00d7.8f86.2bbe)
MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, medium is broadcast
Port mode is trunk
full-duplex, 10 Gb/s, media type is 10G
Beacon is turned off
Auto-Negotiation is turned on FEC mode is Auto
Input flow-control is off, output flow-control is off
Auto-mdix is turned off
Rate mode is dedicated
Switchport monitor is off
EtherType is 0x8100
EEE (efficient-ethernet) : n/a
admin fec state is auto, oper fec state is off
Last link flapped 04:09:21
Last clearing of "show interface" counters 00:50:37
0 interface resets
RX
 8 unicast packets 253 multicast packets 2 broadcast packets
1832838280 input packets 2199405650587 bytes
 0 jumbo packets 0 storm suppression bytes
 0 runts 0 giants 1832838019 CRC 0 no buffer
1832838019 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
1832838019 Stomped CRC
TX
908 unicast packets 323 multicast packets 3 broadcast packets
1234 output packets 113342 bytes
 0 jumbo packets
 0 output error 0 collision 0 deferred 0 late collision
 0 lost carrier 0 no carrier 0 babble 0 output discard
 0 Tx pause
```

Observe que um contador de "CRC" incremental indica que um quadro foi recebido com CRC estompada ou com CRC inválida, mas sem estompo. Um aumento incremental do contador de "CRC estompada" indica que um quadro com CRC estompada foi recebido.

Como alternativa, o comando **show interface counters errors non-zero** pode ser usado para ver os contadores de erros de interface. Um exemplo disso é mostrado aqui.

```
switch# show interface counters errors non-zero
-----
Port          Align-Err    FCS-Err    Xmit-Err    Rcv-Err    UnderSize  OutDiscards
-----
Eth1/1       1790348828 1790348828          0    1790348828          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
-----
-----
Port          Stomped-CRC
-----
-----
Eth1/1        1790348828

```

Você pode canalizar o comando **show interface** para os comandos **json** ou **json-pretty** para obter estatísticas de contador CRC e CRC estompadas em um formato estruturado. Um exemplo disso é mostrado aqui.

```

switch# show interface Ethernet1/1 | json-pretty | include ignore-case crc
      "eth_crc": "828640831",
      "eth_stomped_crc": "828640831",

```

A API REST NX-API pode ser usada para recuperar essas mesmas estatísticas usando o modelo de objeto **sys/intf/phys-[intf-id]/dbgEtherStats.json**. Um exemplo disso é mostrado aqui.

```

/api/node/mo/sys/intf/phys-[eth1/1]/dbgEtherStats.json
{
  "totalCount": "1",
  "imdata": [
    {
      "rmonEtherStats": {
        "attributes": {
          "CRCAlignErrors": "26874272810",
          "dn": "sys/intf/phys-[eth1/1]/dbgEtherStats",
          "dropEvents": "0",
          "rXNoErrors": "26874276337",
          "stompedCRCAlignErrors": "26874272810",
          ...
        }
      }
    }
  ]
}

```

Software NX-OS versão 10.1(2) e anterior

Para versões do software NX-OS anteriores à 10.2(1), o contador CRC estompado não está disponível nas interfaces. Várias etapas são necessárias para determinar a interface de entrada onde CRCs inválidos são observados e validar se os CRCs são inválidos ou estompados.

Etapa 1. Identificar o incremento de contadores CRC em interface(s) física(is)

Use o comando **show interface** para identificar interfaces físicas com incremento de contadores CRC diferentes de zero. Um exemplo disso é mostrado aqui, onde a interface física Ethernet1/1 tem um contador CRC diferente de zero.

```

switch# show interface

```

```
<snip> Ethernet1/1 is up admin state is up, Dedicated Interface Hardware: 100/1000/10000/25000
Ethernet, address: 00d7.8f86.2bbe (bia 00d7.8f86.2bbe) MTU 1500 bytes, BW 10000000 Kbit, DLY 10
usec reliability 255/255, txload 1/255, rxload 1/255 Encapsulation ARPA, medium is broadcast
Port mode is trunk full-duplex, 10 Gb/s, media type is 10G Beacon is turned off Auto-Negotiation
is turned on FEC mode is Auto Input flow-control is off, output flow-control is off Auto-mdix is
turned off Rate mode is dedicated Switchport monitor is off EtherType is 0x8100 EEE (efficient-
ethernet) : n/a admin fec state is auto, oper fec state is off Last link flapped 04:09:21 Last
clearing of "show interface" counters 00:50:37 0 interface resets RX 3 unicast packets 3087
multicast packets 0 broadcast packets 3097 input packets 244636 bytes 7 jumbo packets 0 storm
suppression bytes 0 runts 7 giants 7 CRC 0 no buffer
    7 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
```

Como alternativa, você pode usar o comando **show interface counters errors non-zero** para exibir todas as interfaces com contadores de erro diferentes de zero (que incluem contadores CRC diferentes de zero). Um exemplo disso é mostrado aqui, onde a interface física Ethernet1/1 tem um contador CRC diferente de zero exibido pela coluna FCS-Err.

```
switch# show interface counters errors non-zero
<snip>
-----
Port                Align-Err    FCS-Err    Xmit-Err    Rcv-Err    UnderSize  OutDiscards
-----
Eth1/1                7            7            0            7            0            0
```

Etapa 2. Mapear a interface física para a subporta ASIC, bloco MAC e bloco Mac

Use o comando **show interface hardware-mappings** para identificar três características principais:

1. **Unidade** - O identificador do ASIC de escala de nuvem ao qual a interface física se conecta. Usa um sistema numérico baseado em zero (por exemplo, o primeiro ASIC é 0, o segundo ASIC é 1 etc.)
2. **MacId** - O identificador do bloco MAC ao qual a interface física se conecta. Usa um sistema numérico de base zero (por exemplo, o primeiro bloco MAC é 0, o segundo bloco MAC é 1, etc.)
3. **MacSP** - O identificador da subporta de bloco MAC à qual a interface física se conecta. Cada bloco MAC tem quatro subportas associadas a ele, que seguem um sistema de numeração de base zero e incrementam em um valor de 2. Portanto, a primeira subporta terá um índice de 0, a segunda subporta terá um índice de 2, a terceira subporta terá um índice de 4 e a quarta subporta terá um índice de 6.

Isso é demonstrado no exemplo aqui, onde a interface física Ethernet1/1 está associada à escala de nuvem ASIC 0, bloco MAC 4 e bloco MAC subporta 0.

```
switch# show interface hardware-mappings
<snip>
-----
-----
Name          IFindeX   Smod Unit HPort FPort NPort VPort Slice SPort SrcId MacId MacSP VIF   Block
BlkSrcID
-----
-----
```

Eth1/1 32	1a000000	1	0	16	255	0	-1	0	16	32	4	0	1	0
Eth1/2 34	1a000200	1	0	17	255	4	-1	0	17	34	4	2	5	0
Eth1/3 36	1a000400	1	0	18	255	8	-1	0	18	36	4	4	9	0
Eth1/4 38	1a000600	1	0	19	255	12	-1	0	19	38	4	6	13	0
Eth1/5 24	1a000800	1	0	12	255	16	-1	0	12	24	3	0	17	0

Etapa 3. Verificar Registros ASIC de Escala de Nuvem para Contadores Relacionados a CRC

Use o comando `slot {x} show hardware internal tah counters asic {y}` para exibir os contadores de registro para o ASIC de Escala de Nuvem. Esse comando contém duas variáveis:

1. **{x}** - Substitua esse valor pelo número do slot da placa de linha. Para switches de topo de rack, sempre será o valor 1. Para switches modulares de fim de linha, o número do slot da placa de linha será o primeiro número no nome da interface física. Por exemplo, a interface física Ethernet1/1 teria um slot de placa de linha número 1, enquanto a interface física Ethernet4/24 teria um slot de placa de linha número 4.
2. **{y}** - Substitua esse valor pelo identificador ASIC da Escala de Nuvem identificado na Etapa 2. Por exemplo, se a coluna "Unit" da interface física Ethernet1/1 tiver um valor de 0, o valor dessa variável será 0. Se a coluna "Unit" da interface física Ethernet4/24 tiver um valor de 3, o valor dessa variável será 3.

Esta saída exibirá uma tabela. Cada linha da tabela é um registro ASIC diferente. Cada coluna da tabela corresponde a uma interface física no switch. O nome usado para cada coluna não é o nome da interface física, mas é uma combinação do bloco MAC e da subporta do bloco MAC. O formato usado para o cabeçalho da coluna é o seguinte:

`M{A}, {B} - {InterfaceSpeed}`

Há três variáveis neste formato, que são as seguintes:

1. **{A}** - Substitua esse valor pelo número do bloco MAC.
2. **{B}** - Substitua esse valor pelo número da subporta do bloco MAC.
3. **{InterfaceSpeed}** - Este valor corresponderá à velocidade física da interface (por exemplo, 10G, 25G, 40Gx4, etc.)

Isso é demonstrado no exemplo aqui. Lembre-se de que a interface física Ethernet1/1 está associada ao slot de placa de linha número 1 e à Cloud Scale ASIC 0, o que significa que o comando que devemos executar é `slot 1 show hardware internal tah counters asic 0`. O bloco MAC associado à interface física Ethernet1/1 é 4, a subporta do bloco MAC associada à interface física Ethernet1/1 é 0 e a interface física Ethernet1/1 é uma interface 10G. Portanto, o cabeçalho da coluna que estamos procurando será **M4,0-10G**.

Note: A saída do comando abaixo é muito longa e ampla. Pode ser difícil ler essa saída em uma sessão de terminal. A Cisco recomenda maximizar a largura do seu terminal com o comando `terminal width 511` e copiando essa saída para um leitor/editor de texto externo para revisão.

```
switch# slot 1 show hardware internal tah counters asic 0
<snip>
```

```

***** PER MAC/CH SRAM COUNTERS *****
REG_NAME          M4,0-10G    M4,2-10G    M4,4-10G    M4,6-10G    M5,0-40Gx4    M6,0-
40Gx4    M7,0-40Gx4    M8,0-10G
-----
02-RX Frm with FCS Err    ....    ....    ....    ....    ....    ....
....    ....
16-RX Frm CRC Err(Stomp) c    ....    ....    ....    ....    ....
....    ....

```

A saída desse comando conterá várias dezenas de contadores de registro. Há dois contadores de registro de chave relacionados à diferenciação de erros de CRC naturais de CRCs estompados:

1. **Formulário 02-RX com Erro de FCS** - Indica um quadro com um CRC inválido, mas sem piscar, recebido.
2. **16-RX From CRC Err(Stomp)** - Indica que um quadro com CRC estompado foi recebido.

O valor desses contadores é hexadecimal. O comando **dec NX-OS** pode converter um valor hexadecimal em um valor decimal, como mostrado aqui.

```
N9K-C93180YC-EX-2# dec 0xc
```

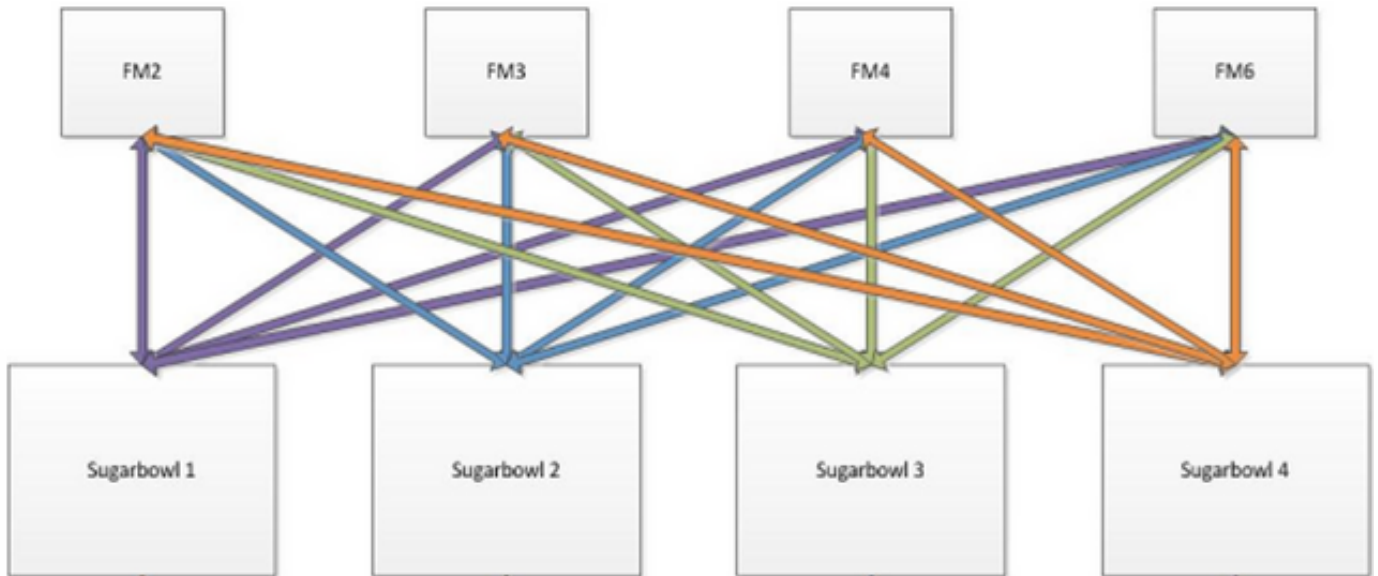
```
12
```

Os valores combinados de ambos os contadores de registro serão equivalentes ao número de CRCs observados na interface física através da saída de **show interface** ou **show interface counters errors non-zero**.

Escala de nuvem do Cisco Nexus 9500 - Procedimento de identificação e rastreamento de CRC em switches modulares

Esta seção do documento descreverá instruções passo a passo para identificar a origem de erros de CRC observados em uma interface física específica Ethernet1/1 nos switches Cisco Nexus 9500 Series.

Cada placa de linha em um switch Nexus 9500 series é conectada via link interno (Ethernet) aos módulos de estrutura. Cada ASIC de cada placa de linha tem conectividade full-mesh para todos os módulos de estrutura. O exemplo mostra uma placa de linha com quatro ASICs do Sugarbowl com links internos que se conectam a quatro módulos de estrutura em um switch Nexus 9500 modular.



Quando o tráfego recebido por um ASIC precisa sair de outro ASIC ou placa de linha, esse tráfego precisa ser enviado à estrutura através de um módulo de estrutura. O ASIC de entrada selecionará um dos enlaces Ethernet para os módulos de estrutura com base em um hash dos cabeçalhos do pacote e no número de enlaces Ethernet disponíveis para o ASIC.

Etapa 1. Mapeie os links internos entre as placas de linha e os módulos de estrutura.

Use o comando **show system internal fabric connectivity module {x}** (onde {x} é o número do slot da placa de linha ou do módulo de malha) para exibir os links internos entre a placa de linha especificada e todos os módulos de malha. Esta saída exibirá uma tabela em que cada linha mostra um mapeamento um para um entre os links internos da placa de linha (na coluna "LC-EthLink") para os links internos de cada módulo de estrutura (na coluna "FM-EthLink"). Um exemplo disso é mostrado aqui, tirado de um switch Nexus 9508 com 8 placas de linha e 4 módulos de estrutura inseridos. A saída aqui mostra que cada instância ASIC da placa de linha inserida no slot 8 do switch está conectada a cada um dos 4 módulos de estrutura instalados (inseridos nos slots 22, 23, 24 e 26) por meio de 2 links internos.

```
Nexus9500# show system internal fabric connectivity module 8
Internal Link-info Linecard slot:8
```

LC-Slot	LC-Unit	LC-iEthLink	MUX	FM-Slot	FM-Unit	FM-iEthLink
8	0	iEth01	-	22	0	iEth18
8	0	iEth02	-	22	1	iEth50
8	0	iEth03	-	23	0	iEth18
8	0	iEth04	-	23	1	iEth50
8	0	iEth05	-	24	0	iEth18
8	0	iEth06	-	24	1	iEth50
8	0	iEth07	-	26	0	iEth18
8	0	iEth08	-	26	1	iEth50
8	1	iEth09	-	22	0	iEth03
8	1	iEth10	-	22	1	iEth35
8	1	iEth11	-	23	0	iEth03
8	1	iEth12	-	23	1	iEth35
8	1	iEth13	-	24	0	iEth03
8	1	iEth14	-	24	1	iEth35
8	1	iEth15	-	26	0	iEth03
8	1	iEth16	-	26	1	iEth35

8	2	iEth17	-	22	0	iEth32
8	2	iEth18	-	22	1	iEth53
8	2	iEth19	-	23	0	iEth32
8	2	iEth20	-	23	1	iEth53
8	2	iEth21	-	24	0	iEth32
8	2	iEth22	-	24	1	iEth53
8	2	iEth23	-	26	0	iEth32
8	2	iEth24	-	26	1	iEth53
8	3	iEth25	-	22	0	iEth31
8	3	iEth26	-	22	1	iEth54
8	3	iEth27	-	23	0	iEth31
8	3	iEth28	-	23	1	iEth54
8	3	iEth29	-	24	0	iEth31
8	3	iEth30	-	24	1	iEth54
8	3	iEth31	-	26	0	iEth31
8	3	iEth32	-	26	1	iEth54

Da mesma forma, o mapeamento do enlace Eth pode ser verificado da perspectiva de um módulo de estrutura. Um exemplo disso é mostrado aqui, onde os links internos entre o módulo de estrutura inserido no slot 22 e cada uma das 8 placas de linha instaladas no chassi do Nexus 9508 são exibidos.

Nexus9500# **show system internal fabric connectivity module 22**

Internal Link-info Fabriccard slot:22

FM-Slot	FM-Unit	FM-iEthLink	LC-Slot	LC-Unit	LC-EthLink	MUX
22	0	iEth09	1	0	iEth01	-
22	0	iEth06	1	1	iEth11	-
22	0	iEth25	1	2	iEth21	-
22	0	iEth26	1	3	iEth31	-
22	0	iEth10	2	0	iEth01	-
22	0	iEth05	2	1	iEth11	-
22	0	iEth23	2	2	iEth21	-
22	0	iEth24	2	3	iEth31	-
22	0	iEth12	3	0	iEth01	-
22	0	iEth11	3	1	iEth11	-
22	0	iEth21	3	2	iEth21	-
22	0	iEth22	3	3	iEth31	-
22	0	iEth14	4	0	iEth01	-
22	0	iEth13	4	1	iEth11	-
22	0	iEth07	4	2	iEth21	-
22	0	iEth08	4	3	iEth31	-
22	0	iEth16	5	0	iEth01	-
22	0	iEth15	5	1	iEth11	-
22	0	iEth01	5	2	iEth21	-
22	0	iEth04	5	3	iEth31	-
22	0	iEth20	6	0	iEth01	-
22	0	iEth17	6	1	iEth11	-
22	0	iEth28	6	2	iEth21	-
22	0	iEth27	6	3	iEth31	-
22	0	iEth19	7	0	iEth01	-
22	0	iEth02	7	1	iEth09	-
22	0	iEth30	7	2	iEth17	-
22	0	iEth29	7	3	iEth25	-
22	0	iEth18	8	0	iEth01	-
22	0	iEth03	8	1	iEth09	-
22	0	iEth32	8	2	iEth17	-
22	0	iEth31	8	3	iEth25	-
22	1	iEth41	1	0	iEth02	-
22	1	iEth38	1	1	iEth12	-
22	1	iEth59	1	2	iEth22	-

22	1	iEth60	1	3	iEth32	-
22	1	iEth42	2	0	iEth02	-
22	1	iEth37	2	1	iEth12	-
22	1	iEth62	2	2	iEth22	-
22	1	iEth61	2	3	iEth32	-
22	1	iEth44	3	0	iEth02	-
22	1	iEth43	3	1	iEth12	-
22	1	iEth64	3	2	iEth22	-
22	1	iEth63	3	3	iEth32	-
22	1	iEth46	4	0	iEth02	-
22	1	iEth45	4	1	iEth12	-
22	1	iEth39	4	2	iEth22	-
22	1	iEth40	4	3	iEth32	-
22	1	iEth48	5	0	iEth02	-
22	1	iEth47	5	1	iEth12	-
22	1	iEth36	5	2	iEth22	-
22	1	iEth33	5	3	iEth32	-
22	1	iEth52	6	0	iEth02	-
22	1	iEth49	6	1	iEth12	-
22	1	iEth57	6	2	iEth22	-
22	1	iEth58	6	3	iEth32	-
22	1	iEth34	7	0	iEth02	-
22	1	iEth51	7	1	iEth10	-
22	1	iEth55	7	2	iEth18	-
22	1	iEth56	7	3	iEth26	-
22	1	iEth50	8	0	iEth02	-
22	1	iEth35	8	1	iEth10	-
22	1	iEth53	8	2	iEth18	-
22	1	iEth54	8	3	iEth26	-

Use o comando **show system internal fabric link-state module {x}** para verificar se a porta interna está ativa ou não (nas colunas "ST") e qual é a porção ASIC correspondente e o identificador MAC de um link interno específico (na coluna "MAC"). Um exemplo disso é mostrado aqui.

```
Nexus9500# show system internal fabric link-state module 8
cli : mod = 8
module number = 8
=====
Module number = 8
=====
=====
[LC] [ INST:SLI:MAC:GLSRC] [IETH] [ST] <=====> [FM] [ INST:SLI:MAC:GLSRC]
[IETH] [ST]
=====
=====
[ 8] [ 0 : 0 : 7 : 0x38] [iEth01] [UP] <=====> [22] [ 0 : 3 : 21 :
0x18] [iEth18] [UP]
[ 8] [ 0 : 1 : 9 : 0x0] [iEth02] [UP] <=====> [22] [ 1 : 3 : 21 :
0x18] [iEth50] [UP]
[ 8] [ 0 : 0 : 6 : 0x30] [iEth03] [UP] <=====> [23] [ 0 : 3 : 21 :
0x18] [iEth18] [UP]
[ 8] [ 0 : 1 : 16 : 0x38] [iEth04] [UP] <=====> [23] [ 1 : 3 : 21 :
0x18] [iEth50] [UP]
[ 8] [ 0 : 0 : 8 : 0x40] [iEth05] [UP] <=====> [24] [ 0 : 3 : 21 :
0x18] [iEth18] [UP]
[ 8] [ 0 : 1 : 15 : 0x30] [iEth06] [UP] <=====> [24] [ 1 : 3 : 21 :
0x18] [iEth50] [UP]
[ 8] [ 0 : 0 : 5 : 0x28] [iEth07] [UP] <=====> [26] [ 0 : 3 : 21 :
0x18] [iEth18] [UP]
[ 8] [ 0 : 1 : 17 : 0x40] [iEth08] [UP] <=====> [26] [ 1 : 3 : 21 :
0x18] [iEth50] [UP]
[ 8] [ 1 : 0 : 7 : 0x38] [iEth09] [UP] <=====> [22] [ 0 : 0 : 4 :
```

```

0x20] [iEth03] [UP]
[ 8] [ 1 : 1 : 9 : 0x0] [iEth10] [UP] <=====> [22] [ 1 : 0 : 4 :
0x20] [iEth35] [UP]
[ 8] [ 1 : 0 : 6 : 0x30] [iEth11] [UP] <=====> [23] [ 0 : 0 : 4 :
0x20] [iEth03] [UP]
[ 8] [ 1 : 1 : 16 : 0x38] [iEth12] [UP] <=====> [23] [ 1 : 0 : 4 :
0x20] [iEth35] [UP]
[ 8] [ 1 : 0 : 8 : 0x40] [iEth13] [UP] <=====> [24] [ 0 : 0 : 4 :
0x20] [iEth03] [UP]
[ 8] [ 1 : 1 : 15 : 0x30] [iEth14] [UP] <=====> [24] [ 1 : 0 : 4 :
0x20] [iEth35] [UP]
[ 8] [ 1 : 0 : 5 : 0x28] [iEth15] [UP] <=====> [26] [ 0 : 0 : 4 :
0x20] [iEth03] [UP]
[ 8] [ 1 : 1 : 17 : 0x40] [iEth16] [UP] <=====> [26] [ 1 : 0 : 4 :
0x20] [iEth35] [UP]
[ 8] [ 2 : 0 : 7 : 0x38] [iEth17] [UP] <=====> [22] [ 0 : 5 : 35 :
0x28] [iEth32] [UP]
[ 8] [ 2 : 1 : 9 : 0x0] [iEth18] [UP] <=====> [22] [ 1 : 4 : 24 :
0x0] [iEth53] [UP]
[ 8] [ 2 : 0 : 6 : 0x30] [iEth19] [UP] <=====> [23] [ 0 : 5 : 35 :
0x28] [iEth32] [UP]
[ 8] [ 2 : 1 : 16 : 0x38] [iEth20] [UP] <=====> [23] [ 1 : 4 : 24 :
0x0] [iEth53] [UP]
[ 8] [ 2 : 0 : 8 : 0x40] [iEth21] [UP] <=====> [24] [ 0 : 5 : 35 :
0x28] [iEth32] [UP]
[ 8] [ 2 : 1 : 15 : 0x30] [iEth22] [UP] <=====> [24] [ 1 : 4 : 24 :
0x0] [iEth53] [UP]
[ 8] [ 2 : 0 : 5 : 0x28] [iEth23] [UP] <=====> [26] [ 0 : 5 : 35 :
0x28] [iEth32] [UP]
[ 8] [ 2 : 1 : 17 : 0x40] [iEth24] [UP] <=====> [26] [ 1 : 4 : 24 :
0x0] [iEth53] [UP]
[ 8] [ 3 : 0 : 7 : 0x38] [iEth25] [UP] <=====> [22] [ 0 : 5 : 34 :
0x20] [iEth31] [UP]
[ 8] [ 3 : 1 : 9 : 0x0] [iEth26] [UP] <=====> [22] [ 1 : 4 : 25 :
0x8] [iEth54] [UP]
[ 8] [ 3 : 0 : 6 : 0x30] [iEth27] [UP] <=====> [23] [ 0 : 5 : 34 :
0x20] [iEth31] [UP]
[ 8] [ 3 : 1 : 16 : 0x38] [iEth28] [UP] <=====> [23] [ 1 : 4 : 25 :
0x8] [iEth54] [UP]
[ 8] [ 3 : 0 : 8 : 0x40] [iEth29] [UP] <=====> [24] [ 0 : 5 : 34 :
0x20] [iEth31] [UP]
[ 8] [ 3 : 1 : 15 : 0x30] [iEth30] [UP] <=====> [24] [ 1 : 4 : 25 :
0x8] [iEth54] [UP]
[ 8] [ 3 : 0 : 5 : 0x28] [iEth31] [UP] <=====> [26] [ 0 : 5 : 34 :
0x20] [iEth31] [UP]
[ 8] [ 3 : 1 : 17 : 0x40] [iEth32] [UP] <=====> [26] [ 1 : 4 : 25 :
0x8] [iEth54] [UP]

```

Etapa 2. Verifique os contadores CRC nos enlaces Eth e rastreie a origem dos quadros corrompidos.

Em um switch Nexus 9500 modular, você pode ver erros de CRC em um ou mais links Ethernet nos seguintes cenários:

1. Quando o switch estiver operando em um modo de switching cut-through, uma placa de linha que recebe um quadro Ethernet corrompido com um valor de CRC incorreto no campo FCS não descartará a placa de linha localmente. Em vez disso, a placa de linha encaminhará o pacote como de costume. Se a interface de saída do pacote pertencer a outro ASIC ou placa de linha, a placa de linha de entrada encaminhará o pacote para um módulo de estrutura. Os módulos de estrutura também operam em um modo de switching

cut-through, de modo que o módulo de estrutura encaminhará o pacote para a placa de linha de saída. A placa de linha de saída encaminhará o pacote para o próximo salto e incrementará o contador de erros de saída na interface de saída.

2. Se um link interno estiver falhando devido a hardware defeituoso, os pacotes que passam pelo link interno podem ser corrompidos entre uma placa de linha e o módulo de estrutura.

Use o comando **show system internal fabric connectivity stats module {x}** para verificar o contador de CRC dos links internos correspondentes. Um exemplo disso é mostrado aqui, onde o módulo de estrutura inserido no slot 22 recebe pacotes com um CRC inválido no Eth56 conectado ao Eth26 da placa de linha inserida no slot 7 do switch. Isso indica que quadros Ethernet corrompidos estão sendo recebidos pelo módulo de estrutura da placa de linha inserida no slot 7 do switch.

```
Nexus9500# show system internal fabric connectivity stats module 22
```

```
Internal Link-info Stats Fabriccard slot:22
```

FM-Slot	FM-Unit	FM-iEthLink	LC-Slot	LC-Unit	LC-EthLink	MUX	CRC
22	0	iEth09	1	0	iEth01	-	0
22	0	iEth06	1	1	iEth11	-	0
22	0	iEth25	1	2	iEth21	-	0
22	0	iEth26	1	3	iEth31	-	0
22	0	iEth10	2	0	iEth01	-	0
22	0	iEth05	2	1	iEth11	-	0
22	0	iEth23	2	2	iEth21	-	0
22	0	iEth24	2	3	iEth31	-	0
22	0	iEth12	3	0	iEth01	-	0
22	0	iEth11	3	1	iEth11	-	0
22	0	iEth21	3	2	iEth21	-	0
22	0	iEth22	3	3	iEth31	-	0
22	0	iEth14	4	0	iEth01	-	0
22	0	iEth13	4	1	iEth11	-	0
22	0	iEth07	4	2	iEth21	-	0
22	0	iEth08	4	3	iEth31	-	0
22	0	iEth16	5	0	iEth01	-	0
22	0	iEth15	5	1	iEth11	-	0
22	0	iEth01	5	2	iEth21	-	0
22	0	iEth04	5	3	iEth31	-	0
22	0	iEth20	6	0	iEth01	-	0
22	0	iEth17	6	1	iEth11	-	0
22	0	iEth28	6	2	iEth21	-	0
22	0	iEth27	6	3	iEth31	-	0
22	0	iEth19	7	0	iEth01	-	0
22	0	iEth02	7	1	iEth09	-	0
22	0	iEth30	7	2	iEth17	-	0
22	0	iEth29	7	3	iEth25	-	0
22	0	iEth18	8	0	iEth01	-	0
22	0	iEth03	8	1	iEth09	-	0
22	0	iEth32	8	2	iEth17	-	0
22	0	iEth31	8	3	iEth25	-	0
22	1	iEth41	1	0	iEth02	-	0
22	1	iEth38	1	1	iEth12	-	0
22	1	iEth59	1	2	iEth22	-	0
22	1	iEth60	1	3	iEth32	-	0
22	1	iEth42	2	0	iEth02	-	0
22	1	iEth37	2	1	iEth12	-	0
22	1	iEth62	2	2	iEth22	-	0
22	1	iEth61	2	3	iEth32	-	0
22	1	iEth44	3	0	iEth02	-	0
22	1	iEth43	3	1	iEth12	-	0

22	1	iEth64	3	2	iEth22	-	0
22	1	iEth63	3	3	iEth32	-	0
22	1	iEth46	4	0	iEth02	-	0
22	1	iEth45	4	1	iEth12	-	0
22	1	iEth39	4	2	iEth22	-	0
22	1	iEth40	4	3	iEth32	-	0
22	1	iEth48	5	0	iEth02	-	0
22	1	iEth47	5	1	iEth12	-	0
22	1	iEth36	5	2	iEth22	-	0
22	1	iEth33	5	3	iEth32	-	0
22	1	iEth52	6	0	iEth02	-	0
22	1	iEth49	6	1	iEth12	-	0
22	1	iEth57	6	2	iEth22	-	0
22	1	iEth58	6	3	iEth32	-	0
22	1	iEth34	7	0	iEth02	-	0
22	1	iEth51	7	1	iEth10	-	0
22	1	iEth55	7	2	iEth18	-	0
22	1	iEth56	7	3	iEth26	-	1665601166
22	1	iEth50	8	0	iEth02	-	0
22	1	iEth35	8	1	iEth10	-	0
22	1	iEth53	8	2	iEth18	-	0
22	1	iEth54	8	3	iEth26	-	0

Use o comando `slot {x} show hardware internal tah counters ASIC {y}` em uma placa de linha ou módulo de malha para determinar se os erros de CRC são inválidos ou CRCs estompados. Os dois contadores de registro que diferenciam erros de CRC inválidos de erros de CRC estompados são:

1. **Formulário 02-RX com Erro de FCS** - Indica um quadro com um CRC inválido, mas sem piscar, recebido.
2. **16-RX From CRC Err(Stomp)** - Indica que um quadro com CRC estompado foi recebido.

Um exemplo disso é mostrado aqui, onde os quadros corrompidos recebidos no módulo de estrutura inserido no slot 22 do chassi através do link interno Eth54 conectam-se à placa de linha inserida no slot 8 do chassi são recebidos com CRC estompada:

```
Nexus9500# slot 22 show hardware internal tah counters ASIC 1
REG_NAME                M24,0-
100Gx4                    M25,0-100Gx4
-----
02-RX Frm with FCS Err
....
03-RX Frm with any Err
....
16-RX Frm CRC Err(Stomp)
....
```

Como alternativa, use o comando `show hardware internal errors module {x}` para ver os contadores de erro ASIC de um módulo específico. Um exemplo disso é mostrado aqui. Observe que nesta saída, o contador "Erros de entrada de interface (CRC, len, Erro de alinhamento)" incrementa tanto para CRCs inválidos quanto para CRCs estompados, enquanto o contador "Erro de CRC de entrada de interface estompado" incrementa somente para CRCs estompados.

```
Nexus9500# show hardware internal errors module 22
-----
| Device:Lacrosse                Role:MAC                Mod:22                |
| Last cleared @ Tue Jul 6 04:10:45 2021 |
| Device Statistics Category :: ERROR |
-----
```

```

Instance:0
ID      Name                               Value                               Ports
--      ----                               -
Instance:1
ID      Name                               Value                               Ports
--      ----                               -
196635  Interface Inbound Errors (CRC,len,Algn Err)  0000053053264536  27:0
1048603 Interface Inbound CRC Error Stomped         0000053053264535  27:0

```

Após identificar a placa de linha de entrada da qual os quadros corrompidos estão sendo recebidos, use os comandos **slot {x} show hardware internal tah counters ASIC {y}** ou **show hardware internal errors module {x}** de maneira semelhante para identificar a interface de entrada na qual os erros estão sendo recebidos, bem como se os erros são recebidos como CRCs inválidos ou CRCs estompados.

Um cenário raro é possível onde um módulo de estrutura ou placa de linha de saída mostra erros de CRC em um enlace Ethernet, mas a placa de linha conectada não tem sinais de CRCs de entrada. A causa raiz desse problema é geralmente uma falha de hardware do módulo de estrutura. A Cisco recomenda abrir um [caso de suporte com o Cisco TAC](#) para fazer troubleshooting adicional desse problema e substituir o módulo de estrutura, se necessário.

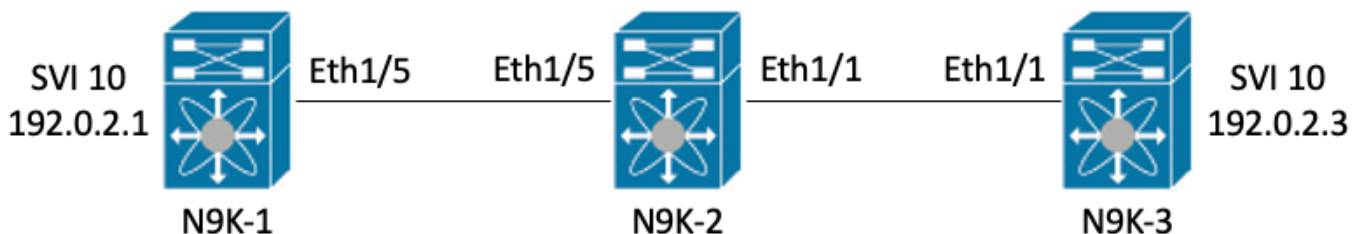
Examples

Esta seção do documento abordará o procedimento acima em alguns exemplos.

Cenário 1. Interface Física Recebendo CRCs Estompados

Este exemplo demonstra como identificar que os erros de CRC em uma interface física são CRCs estompados.

Considere a seguinte topologia:



Neste exemplo, erros de CRC propositalmente estompados são gerados no switch N9K-1 através de pacotes ICMP de tamanho gigante de 8000 bytes originados da interface SVI 10 (que possui o endereço IP 192.0.2.1) destinados à interface SVI 10 do N9K-3 (que possui o endereço IP 192.0.2.3), que tem uma MTU de 1500 bytes. N9K-1, N9K-2 e N9K-3 são todos switches modelo Nexus 93180YC-EX.

```

N9K-3# ping 192.0.2.3 count 5 packet-size 8000
PING 192.0.2.3 (192.0.2.3): 8000 data bytes
Request 0 timed out
Request 1 timed out
Request 2 timed out
Request 3 timed out
Request 4 timed out

```

Request 5 timed out

--- 192.0.2.3 ping statistics ---

5 packets transmitted, 0 packets received, 100.00% packet loss

Neste exemplo, erros de CRC de incremento são observados na interface física Ethernet1/1 do switch N9K-3.

```
N9K-3# show interface Ethernet1/1
```

```
<snip>
```

```
Ethernet1/1 is up
```

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

```
Hardware: 100/1000/10000/25000 Ethernet, address: 00d7.8f86.2bbe (bia 00d7.8f86.2bbe)
```

```
MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec
```

```
reliability 255/255, txload 1/255, rxload 1/255
```

```
Encapsulation ARPA, medium is broadcast
```

```
Port mode is trunk
```

```
full-duplex, 10 Gb/s, media type is 10G
```

```
Beacon is turned off
```

```
Auto-Negotiation is turned on FEC mode is Auto
```

```
Input flow-control is off, output flow-control is off
```

```
Auto-mdix is turned off
```

```
Rate mode is dedicated
```

```
Switchport monitor is off
```

```
EtherType is 0x8100
```

```
EEE (efficient-ethernet) : n/a
```

```
admin fec state is auto, oper fec state is off
```

```
Last link flapped 06:13:44
```

```
Last clearing of "show interface" counters 02:55:00
```

```
0 interface resets
```

```
RX
```

```
9 unicast packets 10675 multicast packets 0 broadcast packets
```

```
10691 input packets 816924 bytes
```

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

```
0 runts 7 giants 7 CRC 0 no buffer
```

```
7 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
```

Etapa 1. Confirmar Incrementação de CRCs

Confirme se os CRCs estão aumentando na interface física Ethernet1/1, gerando pacotes ICMP de tamanho gigante de 8000 bytes originados da interface SVI 10 da N9K-1 (que possui o endereço IP 192.0.2.1) destinados à interface SVI 10 da N9K-3 (que possui o endereço IP 192.0.2.3).

```
N9K-1# ping 192.0.2.3 count 5 packet-size 8000
```

```
PING 192.0.2.3 (192.0.2.3): 8000 data bytes
```

```
Request 0 timed out
```

```
Request 1 timed out
```

```
Request 2 timed out
```

```
Request 3 timed out
```

```
Request 4 timed out
```

```
Request 5 timed out
```

```
--- 192.0.2.3 ping statistics ---
```

```
5 packets transmitted, 0 packets received, 100.00% packet loss
```



```
N9K-3# show interface Ethernet1/1
```

```
Ethernet1/1 is up
admin state is up, Dedicated Interface
Hardware: 100/1000/10000/25000 Ethernet, address: 00d7.8f86.2bbe (bia 00d7.8f86.2bbe)
MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ARPA, medium is broadcast
Port mode is trunk
full-duplex, 10 Gb/s, media type is 10G
Beacon is turned off
Auto-Negotiation is turned on FEC mode is Auto
Input flow-control is off, output flow-control is off
Auto-mdix is turned off
Rate mode is dedicated
Switchport monitor is off
EtherType is 0x8100
EEE (efficient-ethernet) : n/a
  admin fec state is auto, oper fec state is off
Last link flapped 06:52:57
Last clearing of "show interface" counters 03:34:13
0 interface resets
RX
  11 unicast packets 13066 multicast packets 0 broadcast packets
  13089 input packets 1005576 bytes
  12 jumbo packets 0 storm suppression bytes
  0 runts 12 giants 12 CRC 0 no buffer
  12 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
```

Etapa 2. Mapear a interface física para a subporta ASIC, bloco MAC e bloco MAC

Use o comando **show interface hardware-mappings** em N9K-3 para mapear a interface física Ethernet1/1 para o número ASIC 0, bloco MAC 4 e sub-porta 0 do bloco MAC.

```
N9K-3# show interface hardware-mappings
```

```
<snip>
```

```
-----
-----
Name          Ifindex  Smod  Unit  HPort  FPort  NPort  VPort  Slice  SPort  SrcId  MacId  MacSP  VIF  Block
BlkSrcID
-----
-----
Eth1/1        1a000000 1     0     16     255    0      -1     0     16     32     4     0     1     0
32
Eth1/2        1a000200 1     0     17     255    4      -1     0     17     34     4     2     5     0
34
Eth1/3        1a000400 1     0     18     255    8      -1     0     18     36     4     4     9     0
36
Eth1/4        1a000600 1     0     19     255    12     -1     0     19     38     4     6     13    0
38
Eth1/5        1a000800 1     0     12     255    16     -1     0     12     24     3     0     17    0
24
```

Etapa 3. Verificar Registros ASIC de Escala de Nuvem para Contadores Relacionados a CRC

Com base nas informações da Etapa 2, sabemos os seguintes fatos:

1. A interface física Ethernet1/1 é mapeada para o número ASIC 0.
2. A interface física Ethernet1/1 é mapeada para a sub-porta 0 do bloco MAC 4
3. Como o N9K-3 é um switch de modelo Nexus 93180YC-EX de topo de rack, sabemos que o único número de slot de placa de linha possível é 1
4. A partir da saída de show interface coletada na Etapa 1, sabemos que a velocidade da interface física Ethernet1/1 é 10G.

Usando essas informações, podemos usar o comando **slot 1 show hardware internal tah counters ASIC 0** para exibir os contadores de registro ASIC para todas as interfaces físicas. Especificamente, procuraremos contadores de registro ASIC associados a M4,0-10G.

```
N9K-3# slot 1 show hardware internal tah counters ASIC 0
<snip>
***** PER MAC/CH SRAM COUNTERS *****
REG_NAME          M4,0-10G      M4,2-10G      M4,4-10G      M4,6-10G      M5,0-40Gx4
M6,0-40Gx4      M7,0-40Gx4      M8,0-10G
-----
02-RX Frm with FCS Err      ....      ....      ....      ....      ....
.....
16-RX Frm CRC Err(Stomp) c      ....      ....      ....      ....      ....
.....
```

Podemos ver um valor hexadecimal diferente de zero de 0xc para o registro 16, que indica que os quadros com um CRC estompado foram recebidos nessa interface física. Podemos usar o comando **dec 0xc** para converter isso em um valor decimal de 12, que corresponde ao número de erros de CRC na interface física Ethernet1/1.

```
N9K-3# dec 0xc
12
```

Conclusão do cenário 1

Confirmamos que o N9K-3 está recebendo quadros com um CRC pisoteado na interface física Ethernet1/1. Isso significa que o dispositivo no lado remoto do link Ethernet1/1 (neste caso, N9K-2) está pisoteando o CRC desses quadros; a causa raiz dos quadros malformados não é o link diretamente conectado à Ethernet1/1, mas é mais downstream. Troubleshooting adicional deve ser executado no dispositivo de rede downstream para determinar a origem desses quadros malformados.

Cenário 2. A Interface Física Recebeu Quadros Malformados com CRC Inválido

Este exemplo demonstra como identificar que os erros de CRC em uma interface física estão aumentando devido a quadros malformados causados por um problema de camada física em um link conectado diretamente.

Considere a seguinte topologia:



Neste exemplo, um gerador de tráfego conectado à interface física Ethernet1/40 do switch N9K-1 está propositalmente gerando quadros com um CRC incorreto. Isso simula um problema de camada física no link conectado à Ethernet1/40, como um transceptor com defeito ou um cabo danificado. O N9K-1 recebe esses quadros, reconhece que o CRC é inválido e incrementa o contador de erros de CRC na interface física Ethernet1/40. O N9K-1 é um switch modelo Nexus 93180YC-EX.

```
N9K-1# show interface Ethernet1/40
```

```
Ethernet1/40 is up
admin state is up, Dedicated Interface
  Hardware: 100/1000/10000/25000 Ethernet, address: 00d7.8f86.2bbe (bia 00d7.8f86.2c02)
  MTU 1500 bytes, BW 10000000 Kbit, DLY 10 usec
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, medium is broadcast
  Port mode is trunk
  full-duplex, 10 Gb/s, media type is 10G
  Beacon is turned off
  Auto-Negotiation is turned on FEC mode is Auto
  Input flow-control is off, output flow-control is off
  Auto-mdix is turned off
  Rate mode is dedicated
  Switchport monitor is off
  EtherType is 0x8100
  EEE (efficient-ethernet) : n/a
    admin fec state is auto, oper fec state is off
  Last link flapped 06:13:44
  Last clearing of "show interface" counters 02:55:00
  0 interface resets
  RX
    1710 unicast packets  9873 multicast packets  0 broadcast packets
    11583 input packets  886321 bytes
    0 jumbo packets  0 storm suppression bytes
    0 runts  0 giants  1683 CRC  0 no buffer
    1683 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
```

Etapa 1. Confirmar Incrementação de CRCs

Confirme se os CRCs estão incrementando na interface física Ethernet1/40 de N9K-1 através dos comandos **show interface** ou **show interface counters non-zero**.

Etapa 3. Verificar Registros ASIC de Escala de Nuvem para Contadores Relacionados a CRC

Com base nas informações da Etapa 2, sabemos os seguintes fatos:

1. A interface física Ethernet1/40 é mapeada para o número ASIC 0.
2. A interface física Ethernet1/40 é mapeada para a subporta 6 do bloco MAC 10.
3. Como o N9K-1 é um switch de modelo Nexus 93180YC-EX de topo de rack, sabemos que o único número de slot de placa de linha possível é 1.
4. A partir da saída de **show interface** reunida na Etapa 1, sabemos que a velocidade da interface física Ethernet1/40 é 10G.

Usando essas informações, podemos usar o comando `slot 1 show hardware internal tah counters asic 0` para exibir os contadores de registro ASIC para todas as interfaces físicas. Especificamente, procuraremos contadores de registro ASIC associados a M10,6-10G.

```
N9K-1# slot 1 show hardware internal tah counters asic 0

***** PER MAC/CH SRAM COUNTERS *****
REG_NAME          M8,2-10G      M8,4-10G      M8,6-10G      M9,0-40Gx4      M10,0-10G
M10,2-10G         M10,4-10G      M10,6-10G
-----
02-RX Frm with FCS Err   ....          ....          ....          ....          ....
....                ....          973e
16-RX Frm CRC Err(Stomp) ....          ....          ....          ....          ....
....                ....          ....
```

Podemos ver um valor hexadecimal diferente de zero de 0x973e para o registro 2, que indica que foram recebidos quadros com um CRC inválido, mas sem piscar. Podemos usar o comando `dec 0x973e` para converter isso em um valor decimal de 38.718, que corresponde (ou é menor que, já que os CRCs estão constantemente incrementando) o número de erros de CRC na interface física Ethernet1/40.

```
N9K-1# dec 0x973e
38718
```

Conclusão do cenário 2

Confirmamos que o N9K-1 está recebendo quadros com um CRC inválido, mas sem piscar na interface física Ethernet1/40. Isso significa que o link diretamente conectado à Ethernet1/40 (ou o dispositivo na extremidade remota do link) é a origem mais provável dos quadros malformados. Mais troubleshooting deve ser realizado na camada física desse link para isolar a causa raiz dos quadros malformados (como verificação de cabeamento danificado, substituição dos transceptores atuais por transceptores em boas condições, etc.).

Cenário 3. Syslog de erros do Nexus 9500 Eth CRC

Este exemplo demonstra como identificar a origem de erros de CRC em um enlace interno Eth quando um syslog relatando erros em uma interface interna é gerado por um switch Nexus 9500 series. Um exemplo desse syslog é mostrado aqui.

```
Nexus9500# show logging logfile
```

```
<snip>
```

```
2021 Jul 9 05:51:19 Nexus9500 %DEVICE_TEST-SLOT22-3-INTERNAL_PORT_MONITOR_CRC_ERRORS_DETECTED:
Module 22 received tx errors on internal interface ii22/1/56 since last run TXErr=36836897
TotalTXErr=50781987904
```

Esse syslog indica que foram detectados erros no link interno Eth56 do módulo de estrutura inserido no slot 22 do switch.

Etapa 1. Mapear o Eth Link no Módulo de Estrutura para a Placa de Linha Conectada

Use o comando **show system internal fabric connectivity stats module {x}** para identificar a que placa de linha o enlace interno Ethernet afetado se conecta. Neste exemplo, o Eth56 do módulo de estrutura inserido no slot 22 do switch tem erros. Um exemplo disso é mostrado aqui, onde Eth56 do módulo de estrutura inserido no slot 22 é conectado à Eth26 da placa de linha inserida no slot 7 do switch.

```
Nexus9500# show system internal fabric connectivity stats module 22 | include Eth56|FM-Slot
FM-Slot  FM-Unit  FM-iEthLink  LC-Slot  LC-Unit  LC-EthLink  MUX  CRC
      22      1      iEth56      7      3      iEth26      -      603816174
```

Use o comando **show system internal fabric link-state module {x}** para localizar a instância ASIC e o identificador MAC associados ao link interno Eth56 do módulo de estrutura. Um exemplo disso é mostrado aqui, onde a instância ASIC é 1 e o identificador MAC é 27.

```
Nexus9500# show system internal fabric link-state module 22 | include MAC|iEth56
[FM] [ INST:SLI:MAC:GLSRC] [IETH] [ST] <=====> [LC] [ INST:SLI:MAC:GLSRC]
[IETH] [ST]
[22] [ 1 : 4 : 27 : 0x18] [iEth56] [UP] <=====> [ 7] [ 3 : 1 : 9 :
0x0] [iEth26] [UP]
```

Etapa 2. Verificar se os CRCs Recebidos no Enlace Eth são Inválidos ou Estão Parados

A etapa anterior mostra que o identificador de instância do ASIC é 1 e o identificador MAC é 27 para o Eth56 conectado ao módulo de estrutura inserido no slot 22. Use o comando **slot {x} show hardware internal tah counters ASIC {y}** para identificar se os CRCs relatados pelo syslog são CRCs inválidos ou CRCs estompados. Um exemplo disso é mostrado aqui, em que a coluna M27,0-100Gx4 é associada ao nosso identificador MAC de 27 e indica que os CRCs estão empilhados.

```
Nexus9500# slot 22 show hardware internal tah counters ASIC 1
REG_NAME          M27,0-100Gx4
-----
```

```
02-RX Frm with FCS Err      ....
16-RX Frm CRC Err(Stomp)    be9cb9bd6
```

Como alternativa, use o comando **show hardware internal errors module {x}** para obter as mesmas informações. Um exemplo disso é mostrado aqui.

```
Nexus9500# show hardware internal errors module 22 | include CRC|Stomp|Inst
Instance:1
196635 Interface Inbound Errors (CRC,len,Algn Err)    0000051587084851 27:0
1048603 Interface Inbound CRC Error Stomped          0000051587084850 27:0
```

Lembre-se de que nesta saída, o contador "Erros de entrada de interface (CRC,len,Erro de

Alinhamento)" incrementa tanto para CRCs inválidos quanto para CRCs estompados, enquanto o contador "Erro de CRC de entrada de interface estompado" incrementa somente para CRCs estompados.

Etapa 3. Rastreie a origem dos quadros com CRCs inválidos na placa de linha de ingresso

Agora sabemos que os CRCs que entram no módulo de estrutura inserido no slot 22 do switch estão entrando no switch a partir da placa de linha inserida no slot 7. Com essas informações, podemos usar o comando **show interface counters errors module {x} non-zero** para identificar contadores CRC diferentes de zero nas interfaces pertencentes à placa de linha relevante. Um exemplo disso é mostrado aqui.

```
Nexus9500# show interface counters errors module 7 non-zero
<snip>
```

Port	Align-Err	FCS-Err	Xmit-Err	Rcv-Err	UnderSize	OutDiscards
Eth7/32	0	0	0	1195309745	0	0

Podemos repetir a Etapa #2 deste cenário na placa de linha relevante para verificar se a placa de linha está recebendo CRCs inválidos ou CRCs estourados.

```
Nexus9500# show hardware internal errors module 7 | include ignore-case CRC|Stomp|Inst
Instance:3
```

```
196619 Interface Inbound Errors (CRC,len,Algn Err) 0000051801011139 11:0
1048587 Interface Inbound CRC Error Stomped 0000051801011140 11:0
```

Use o comando **show interface hardware-mappings** para identificar a porta do painel frontal para a qual o valor MacId:MacSP de 11:0 na saída acima está mapeado. Um exemplo disso é mostrado aqui, onde 11:0 mapeia para a porta Eth7/32 do painel frontal.

```
Nexus9500# show interface hardware-mappings | include Name|Eth7
<snip>
```

Name	Ifindex	Smod	Unit	HPort	FPort	NPort	VPort	Slice	SPort	SrcId	MacId	MacSP	VIF	Block
BlkSrcID														
Eth7/1	1a300000	25	0	16	255	0	-1	0	16	32	4	0	1	0
32														
Eth7/2	1a300200	25	0	12	255	4	-1	0	12	24	3	0	5	0
24														
Eth7/3	1a300400	25	0	8	255	8	-1	0	8	16	2	0	9	0
16														
Eth7/4	1a300600	25	0	4	255	12	-1	0	4	8	1	0	13	0
Eth7/5	1a300800	25	0	60	255	16	-1	1	20	40	14	0	17	0
40														
Eth7/6	1a300a00	25	0	56	255	20	-1	1	16	32	13	0	21	0
32														
Eth7/7	1a300c00	25	0	52	255	24	-1	1	12	24	12	0	25	0
24														
Eth7/8	1a300e00	25	0	48	255	28	-1	1	8	16	11	0	29	0
16														
Eth7/9	1a301000	26	1	12	255	32	-1	0	12	24	3	0	33	0
24														
Eth7/10	1a301200	26	1	8	255	36	-1	0	8	16	2	0	37	0
16														
Eth7/11	1a301400	26	1	4	255	40	-1	0	4	8	1	0	41	0
Eth7/12	1a301600	26	1	0	255	44	-1	0	0	0	0	0	45	0
Eth7/13	1a301800	26	1	60	255	48	-1	1	20	40	14	0	49	0

40																
Eth7/14	1a301a00	26	1	56	255	52	-1	1	16	32	13	0	53	0		
32																
Eth7/15	1a301c00	26	1	52	255	56	-1	1	12	24	12	0	57	0		
24																
Eth7/16	1a301e00	26	1	48	255	60	-1	1	8	16	11	0	61	0		
16																
Eth7/17	1a302000	27	2	16	255	64	-1	0	16	32	4	0	65	0		
32																
Eth7/18	1a302200	27	2	12	255	68	-1	0	12	24	3	0	69	0		
24																
Eth7/19	1a302400	27	2	8	255	72	-1	0	8	16	2	0	73	0		
16																
Eth7/20	1a302600	27	2	4	255	76	-1	0	4	8	1	0	77	0	8	
Eth7/21	1a302800	27	2	60	255	80	-1	1	20	40	14	0	81	0		
40																
Eth7/22	1a302a00	27	2	56	255	84	-1	1	16	32	13	0	85	0		
32																
Eth7/23	1a302c00	27	2	52	255	88	-1	1	12	24	12	0	89	0		
24																
Eth7/24	1a302e00	27	2	48	255	92	-1	1	8	16	11	0	93	0		
16																
Eth7/25	1a303000	28	3	12	255	96	-1	0	12	24	3	0	97	0		
24																
Eth7/26	1a303200	28	3	8	255	100	-1	0	8	16	2	0	101	0		
16																
Eth7/27	1a303400	28	3	4	255	104	-1	0	4	8	1	0	105	0	8	
Eth7/28	1a303600	28	3	0	255	108	-1	0	0	0	0	0	109	0	0	
Eth7/29	1a303800	28	3	60	255	112	-1	1	20	40	14	0	113	0		
40																
Eth7/30	1a303a00	28	3	56	255	116	-1	1	16	32	13	0	117	0		
32																
Eth7/31	1a303c00	28	3	52	255	120	-1	1	12	24	12	0	121	0		
24																
Eth7/32	1a303e00	28	3	48	255	124	-1	1	8	16	11	0	125	0		
16																

Conclusão do cenário 3

Confirmamos que o Nexus 9500 está recebendo quadros com um CRC pisoteado na interface física Ethernet7/32. Isso significa que o dispositivo no lado remoto do link Ethernet7/32 está estombando o CRC desses quadros; a causa raiz dos quadros malformados não é o link diretamente conectado à Ethernet7/32, mas é mais downstream. Troubleshooting adicional deve ser executado no dispositivo de rede downstream para determinar a origem desses quadros malformados.

Cenário 4. Rastreie a origem de quadros CRC inválidos com interface de saída.

Este exemplo demonstra como rastrear a origem de quadros com CRCs inválidos em um switch Nexus 9500 quando um switch upstream relata que o Nexus 9500 está gerando quadros com CRCs estropiadas. Neste cenário, o switch upstream é conectado através da porta Ethernet8/9 do painel frontal.

Etapa 1. Identificar o módulo de estrutura que envia quadros CRC inválidos à placa de linha de saída

Sabemos que a interface de saída que envia quadros com CRCs estropiados em direção aos switches upstream é a Ethernet8/9. Primeiro, precisamos determinar o módulo de estrutura que

está enviando quadros com CRCs estropiados para a placa de linha inserida no slot 8 do chassi. Iniciamos esse processo com o comando **show hardware internal errors module {x}**. Um exemplo disso é mostrado aqui.

```
Nexus9500# show hardware internal errors module 8 | i CRC|Inst
<snip>
Instance:1
196617 Interface Inbound Errors (CRC,len,Algn Err)      0000091499464650  9:0
1048585 Interface Inbound CRC Error Stomped            0000091499464651  9:0
```

MacID:MacSP 9:0 na saída acima pode ser mapeado para o módulo de estrutura de origem com o comando **show system internal fabric link-state module 8**. Um exemplo disso é mostrado aqui.

```
Nexus9500# show system internal fabric link-state module 8
cli : mod = 8
module number = 8

=====
Module number = 8
=====
[LC] [ INST:SLI:MAC:GLSRC] [IETH] [ST] <=====> [FM] [ INST:SLI:MAC:GLSRC]
[IETH] [ST]
=====
...
[ 8] [ 1 : 1 : 9 : 0x0] [iEth10] [UP] <=====> [22] [ 1 : 0 : 4 :
0x20] [iEth35] [UP]
```

Vemos que o identificador MAC 9 na placa de linha inserida no slot 8 está mapeado para o módulo de estrutura inserido no slot 22 do chassi. Esperamos ver erros de CRC no Eth10 de link interno. Podemos validar isso com o comando **show system internal fabric connectivity stats module 8**. Um exemplo disso é mostrado aqui.

```
Nexus9500# show system internal fabric connectivity stats module 8

Internal Link-info Stats Linecard slot:8
-----
LC-Slot  LC-Unit  LC-iEthLink  MUX  FM-Slot  FM-Unit  FM-iEthLink  CRC
-----
8         0         iEth01      -    22       0        iEth18       0
8         0         iEth02      -    22       1        iEth50       0
8         0         iEth03      -    23       0        iEth18       0
8         0         iEth04      -    23       1        iEth50       0
8         0         iEth05      -    24       0        iEth18       0
8         0         iEth06      -    24       1        iEth50       0
8         0         iEth07      -    26       0        iEth18       0
8         0         iEth08      -    26       1        iEth50       0
8         1         iEth09      -    22       0        iEth03       0
8         1         iEth10      -    22       1        iEth35      1784603561
```

Etapa 2. Mapear o enlace Eth no módulo de estrutura para a placa de linha conectada e verificar se há CRCs estampados

Em seguida, seguimos o mesmo processo do Cenário 3 verificando os enlaces internos Eth que recebem CRCs, se esses CRCs estão ou não estropiados de acordo com o ASIC do módulo de estrutura e qual placa de linha está conectada ao enlace interno Eth do módulo de estrutura. Um

exemplo disso é mostrado aqui usando o comando **show system internal fabric connectivity stats module {x}**, o comando **show hardware internal errors module {x}** e o comando **show system internal fabric link-state module {x}**, respectivamente.

```
Nexus9500# show system internal fabric connectivity stats module 22
Internal Link-info Stats Fabriccard slot:22
-----
FM-Slot  FM-Unit  FM-iEthLink  LC-Slot  LC-Unit  LC-EthLink  MUX  CRC
   22      1      iEth56      7        3      iEth26      -    1171851894
Nexus9500# show hardware internal errors module 22 | i CRC|Stomp|Inst
Instance:1
196635 Interface Inbound Errors (CRC,len,Algn Err)  0000054593935847  27:0
1048603 Interface Inbound CRC Error Stomped          0000054593935846  27:0
Nexus9500# show system internal fabric link-state module 22 | i MAC|iEth56

[FM]  [ INST:SLI:MAC:GLSRC]  [IETH]  [ST]  <=====>  [LC]  [ INST:SLI:MAC:GLSRC]
[IETH]  [ST]
[22]  [ 1 : 4 : 27 : 0x18]  [iEth56]  [UP]  <=====>  [ 7]  [ 3 : 1 : 9 :
0x0]  [iEth26]  [UP]
```

Etapa 3. Rastrear a origem dos quadros com CRCs inválidos no módulo de entrada

Após determinar a placa de linha de entrada (neste cenário, a placa de linha inserida no slot 7 conectada pela Ethernet26 à Ethernet56 do módulo de estrutura inserido no slot 22), identificamos qual porta de entrada os quadros corrompidos entram no switch. Isso é feito com o comando **show interface counters errors module {x} non-zero**. A saída do comando **show hardware internal errors module {x}** e do comando **show interface hardware-mappings** pode validar se os quadros recebidos são CRCs inválidos ou estompados. Um exemplo disso é mostrado aqui, onde quadros corrompidos estão entrando no switch através da interface Ethernet7/32 do painel frontal.

```
Nexus9500# show interface counters errors module 7 non-zero
<snip>
-----
Port          Align-Err  FCS-Err  Xmit-Err  Rcv-Err  UnderSize  OutDiscards
-----
Eth7/32          0          0          0 4128770335          0          0
-----
Port          Stomped-CRC
-----
Eth7/32          4129998971
Nexus9500# show hardware internal errors module 7 | i i CRC|Stomp|Inst
<snip>
Instance:3
196619 Interface Inbound Errors (CRC,len,Algn Err)  0000054901402307  11:0
1048587 Interface Inbound CRC Error Stomped          0000054901402308  11:0
Nexus9500# show interface hardware-mappings | i Name|Eth7
<snip>
Name          Ifindex  Smod Unit  HPort  FPort  NPort  VPort  Slice  SPort  SrcId  MacId  MacSP  VIF  Block
BlkSrcID
...
Eth7/32    1a303e00 28   3    48    255   124   -1    1     8     16    11    0    125  0
16
```

Conclusão do cenário 4

Confirmamos que o Nexus 9500 está recebendo quadros com um CRC pisoteado na interface física Ethernet7/32. Isso significa que o dispositivo no lado remoto do link Ethernet7/32 está estombando o CRC desses quadros; a causa raiz dos quadros malformados não é o link diretamente conectado à Ethernet7/32, mas é mais downstream. Troubleshooting adicional deve ser executado no dispositivo de rede downstream para determinar a origem desses quadros malformados.

Informações Relacionadas

- [Switching Ethernet cut-through e store-and-forward para ambientes de baixa latência](#)

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.