

Nexus 9000クラウドスケールASICのCRCの識別およびトレース手順

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[該当ハードウェア](#)

[Cisco Nexus 9200 & 9300クラウドスケールCRCの識別およびトレース手順](#)

[NX-OSソフトウェアリリース10.2\(1\)以降](#)

[NX-OSソフトウェアリリース10.1\(2\)以前](#)

[手順1：物理インターフェイスのCRCカウンタの増加を特定する](#)

[ステップ2：物理インターフェイスをASIC、MACブロック、およびMACブロックサブポートにマッピングする](#)

[ステップ3：クラウドスケールASICレジスタのCRC関連カウンタの確認](#)

[Cisco Nexus 9500クラウドスケール：モジュラスイッチでのCRCの識別とトレース手順](#)

[ステップ1：ラインカードとファブリックモジュール間の内部リンクをマッピングする。](#)

[ステップ2:iEthリンクのCRCカウンタを確認し、破損したフレームの原因を追跡します。](#)

[例](#)

[シナリオ1：ストンプされたCRCを受信する物理インターフェイス](#)

[ステップ1: CRCの増加を確認する](#)

[ステップ2：物理インターフェイスのASIC、MACブロック、およびMACブロックサブポートへのマッピング](#)

[ステップ3：クラウドスケールASICレジスタのCRC関連カウンタの確認](#)

[シナリオ1まとめ](#)

[シナリオ2：物理インターフェイスが無効なCRCを持つ不正なフレームを受信する](#)

[ステップ1: CRCの増加を確認する](#)

[ステップ2：物理インターフェイスのASIC、MACブロック、およびMACブロックサブポートへのマッピング](#)

[ステップ3：クラウドスケールASICレジスタのCRC関連カウンタの確認](#)

[シナリオ2まとめ](#)

[シナリオ3:Nexus 9500のiEth CRCエラーSyslog](#)

[ステップ1：ファブリックモジュール上のiEthリンクを接続されたラインカードにマッピングする](#)

[ステップ2:iEthリンクで受信されたCRCが無効であるか、またはストンプされているかどうかを確認する](#)

[ステップ3：入カラインカードで無効なCRCを持つフレームの送信元を追跡する](#)

[シナリオ3まとめ](#)

[シナリオ4：出カインターフェイスで無効なCRCフレームの送信元を追跡する。](#)

[ステップ1：出カラインカードに無効なCRCフレームを送信しているファブリックモジュールを特定する](#)

[ステップ2：ファブリックモジュール上のiEthリンクを接続されたラインカードにマッピングし、](#)

[ストンプCRCをチェックする](#)

[ステップ3：入力モジュールで無効なCRCを持つフレームの送信元を追跡する](#)

[シナリオ4まとめ](#)

[関連情報](#)

概要

このドキュメントでは、一連のCisco Nexus 9000 Cloud Scale ASICモジュールの物理インターフェイスで見られるCRCエラーの原因をトレースするために使用する手順について説明します。このドキュメントでは、モジュラNexusスイッチの物理インターフェイスと内部ファブリックリンクで見られるストンプCRCエラーと非ストンプCRCエラーを区別するための手順についても説明します。

前提条件

要件

カットスルーおよびストアアンドフォワードスイッチングの基本を理解しておくことをお勧めします。また、イーサネットFCS（フレームチェックシーケンス）フィールドの基本と、FCSフィールドで使用されるCRC（巡回冗長検査）アルゴリズムを理解しておくことをお勧めします。詳細については、次の文書を参照してください。

- [低遅延環境向けのカットスルーおよびストアアンドフォワードイーサネットスイッチング](#)

使用するコンポーネント

このドキュメントの情報は、NX-OSソフトウェアリリース7.0(3)I7(8)を実行するCloud Scale ASICを搭載したCisco Nexus 9000シリーズスイッチに基づくものです。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。このドキュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。対象のネットワークが稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

背景説明

Cisco Nexus 9000シリーズスイッチは、デフォルトでカットスルースイッチングを使用します。カットスルースイッチングとは、スイッチがフレームに対してフォーワーディング決定を行い、有効なフォーワーディング決定を行うためにフレームのヘッダーがスイッチで十分に処理されるとすぐに、フレームの出カインターフェイスからのフォーワーディングを開始するスイッチングです。これは、出カインターフェイスからフレームを転送する前にフレーム全体をバッファするストアアンドフォワードスイッチングとは異なります。

イーサネットフレームのFCSフィールドは、フレームの完全性を検証し、フレームが転送中に破損していないことを確認します。イーサネットフレームのFCSフィールドは、フレームのペイロードの背後にあるイーサネットフレームの最後に配置されます。ストアアンドフォワードスイッチングモードで動作しているスイッチは、出カインターフェイスからフレームを転送する（または、FCSフィールドに無効な内容がある場合はフレームをドロップする）前に、FCSフィールド

を使用してイーサネットフレームの整合性を確認できます。ただし、カットスルースイッチングモードで動作しているスイッチは、出カインターフェイスからフレームを転送する前に、FCSフィールドを使用してイーサネットフレームの整合性を確認することはできません。つまり、カットスルースイッチがイーサネットフレームの完全性を検証できる時点で、イーサネットフレームの大部分はすでに出カインターフェイスから転送されています。

カットスルースイッチングモードで動作しているスイッチが、無効なFCSフィールドを持つイーサネットフレームを受信すると、スイッチは次のアクションを実行します。

1. イーサネットフレームのFCSフィールドを、現在の (誤った) FCSフィールド値のビット反転で書き換えます。フレームをルーティングする必要がある場合、現在の (誤った) FCSフィールドの値は、フレームのイーサネットヘッダーが書き換えられた後に計算されます。この動作は、CRCの「ストップ」と呼ばれます。
2. フレームに対する転送の決定に従って、イーサネットフレームの残りの部分を (ストップされたCRCとともに) 出カインターフェイスから転送します。
3. 入力インターフェイスの入力エラーカウンタまたはCRCエラーカウンタを増やします。

このドキュメントでは、入力インターフェイスに関連付けられたCRCカウンタが正常なCRC (通常、入力インターフェイスに接続されたリンク上の物理層の問題を示す) か、またはストップCRC (入力インターフェイスに接続されたデバイスがカットスルースイッチングモードでも動作していて、不正なイーサネットフレームを受信したことを示す) かを確認する手順について説明します。

該当ハードウェア

このドキュメントで説明する手順は、次のハードウェアにのみ適用できます。

- **Nexus 9200/9300固定構成スイッチ** 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
- **Nexus 9500モジュラスイッチラインカード** N9K-X97160YC-EXN9K-X9732C-EXN9K-X9736C-EXN9K-X97284YC-FXN9K-X9732C-FXN9K-X9788TC-FXN9K-X9716D-GX

Cisco Nexus 9200 & 9300クラウドスケールCRCの識別およびトレース手順

このセクションでは、Cisco Nexus 9200および9300シリーズスイッチの特定の物理インターフェイスEthernet1/1で見られるCRCエラーの発生源を特定する手順を説明します。

NX-OSソフトウェアリリース10.2(1)以降

NX-OSソフトウェアリリース10.2(1)以降、Cloud Scale ASICを搭載したNexusスイッチには、スイッチを通過するイーサネットフレームのFCSフィールドに、ストップされたCRCを持つパケット用の新しいインターフェイスカウンタがあります。show interfaceコマンドを使用すると、ゼロ

以外のCRCカウンタとストップされたCRCカウンタが増加している物理インターフェイスを特定できません。この例を次に示します。物理インターフェイスEthernet1/1のCRCカウンタとストップCRCカウンタがゼロです。これは、無効なCRCとストップCRCのフレームがこのインターフェイスで受信されたことを示します。

```
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
```

「CRC」カウンタが増加している点に注意してください。これは、フレームがCRCのストップまたはCRCの不正なストップを含めて受信されたことを示します。「stomped CRC」カウンタの増加は、CRCがストップされたフレームが受信されたことを示します。

または、**show interface counters errors non-zero**コマンドを使用して、インターフェイスエラーカウンタを表示することもできます。この例を次に示します。

```
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
```

パイプを使用してshow interfaceコマンドをjsonまたはjson-prettyコマンドに渡すと、構造化された形式でCRCおよびストンプされたCRCカウンタの統計情報を取得できます。この例を次に示します。

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

NX-API REST APIは、sys/intf/phys-[intf-id]/dbgEtherStats.jsonオブジェクトモデルを使用して、これらの同じ統計情報を取得するために使用できます。この例を次に示します。

```

/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",
          ...
        }
      }
    }
  ]
}

```

NX-OSソフトウェアリリース10.1(2)以前

10.2(1)より前のNX-OSソフトウェアリリースでは、インターフェイスでstomped CRCカウンタを使用できません。無効なCRCが検出された入力インターフェイスを特定し、CRCが無効であるかストンプであるかを検証するには、いくつかの手順が必要です。

手順1：物理インターフェイスのCRCカウンタの増加を特定する

show interfaceコマンドを使用して、ゼロ以外のCRCカウンタが増加している物理インターフェイスを特定します。次に示す例では、物理インターフェイスEthernet1/1にゼロ以外のCRCカウンタがあります。

```
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
```

または、**show interface counters errors non-zero**コマンドを使用して、ゼロ以外のエラーカウンタ（ゼロ以外のCRCカウンタを含む）を持つすべてのインターフェイスを表示することもできます。次に示す例では、物理インターフェイスEthernet1/1にゼロ以外のCRCカウンタがあり、FCS-Err列に表示されています。

```
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
```

ステップ2：物理インターフェイスをASIC、MACブロック、およびMACブロックサブポートにマッピングする

show interface hardware-mappingsコマンドを使用して、次の3つの主要な特性を特定します。

1. **Unit**：物理インターフェイスが接続するCloud Scale ASICのID。これは、ゼロベースの番号付けシステムを使用します（たとえば、最初のASICは0、2番目のASICは1など）。
2. **MacId**：物理インターフェイスが接続するMACブロックのID。これは、ゼロベースの番号付け方式を使用します（たとえば、最初のMACブロックは0、2番目のMACブロックは1など）。
3. **MacSP**：物理インターフェイスが接続するMACブロックサブポートのID。各MACブロックには4つのサブポートが関連付けられており、ゼロベースの番号付けシステムに従って値2ずつ増加します。したがって、最初のサブポートのインデックスは0、2番目のサブポートのインデックスは2、3番目のサブポートのインデックスは4、4番目のサブポートのインデックスは6になります。

これは、物理インターフェイスEthernet1/1がCloud Scale ASIC 0、MACブロック4、およびMACブロックサブポート0に関連付けられている例で示されています。

```
switch# 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

```

ステップ3：クラウドスケールASICレジスタのCRC関連カウンタの確認

slot {x} show hardware internal tah counters ASIC {y}コマンドを使用して、Cloud Scale ASICのレジスタカウンタを表示します。このコマンドには2つの変数が含まれています。

1. **{x}**：この値をラインカードスロット番号に置き換えます。トップオブブラックスイッチの場合、この値は常に1になります。エンドオブローのモジュラスイッチの場合、ラインカードスロット番号は物理インターフェイス名の最初の番号になります。たとえば、物理インターフェイスEthernet1/1のラインカードスロット番号は1ですが、物理インターフェイスEthernet4/24のラインカードスロット番号は4です。
2. **{y}** – この値を手順2で特定したCloud Scale ASIC識別子に置き換えます。たとえば、物理インターフェイスEthernet1/1の"Unit"列の値が0の場合、この変数の値は0になります。物理インターフェイスEthernet4/24の"Unit"列の値が3の場合、この変数の値は3になります。

この出力には表が表示されます。テーブルの各行は異なるASICレジスタです。テーブルの各列は、スイッチの物理インターフェイスに対応します。各列に使用される名前は、物理インターフェイスの名前ではなく、MACブロックとMACブロックサブポートの組み合わせです。列ヘッダーに使用される形式は次のとおりです。

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

この形式には、次の3つの変数があります。

1. **{A}**：この値をMACブロック番号に置き換えます。
2. **{B}**：この値をMACブロックサブポート番号に置き換えます。
3. **{InterfaceSpeed}**：この値は、インターフェイスの物理速度に対応します（10G、25G、40Gx4など）。

これについては、次の例で説明します。物理インターフェイスEthernet1/1は、ラインカードスロット番号1とCloud Scale ASIC 0に関連付けられていることを思い出してください。つまり、実行する必要があるコマンドは、**slot 1 show hardware internal tah counters ASIC 0**です。物理インターフェイスEthernet1/1に関連付けられたMACブロックは4で、物理インターフェイスEthernet1/1に関連付けられたMACブロックサブポートは0で、物理インターフェイスEthernet1/1は10Gインターフェイスです。したがって、探している列ヘッダーは**M4,0-10G**になります。

注：次のコマンドの出力は非常に長く、幅も広がっています。ターミナルセッション内でこの出力を読み取るのは困難な場合があります。シスコでは、**terminal width 511**コマンドを使用して端末の幅を最大化し、確認のためにこの出力を外部のテキストリーダー/エディタにコピーすることを推奨しています。

```

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

このコマンドの出力には、数十のレジスタカウンタが含まれます。自然なCRCエラーとストンプされたCRCの区別に関連する2つのキーレジスタカウンタがあります。

1. **02-RX Frm with FCS Err** : 無効だがストンプされていないCRCを含むフレームを受信したことを示します。

2. **16-RX Frm CRC Err(Stomp)**:CRCがストンプされたフレームが受信されたことを示します。これらのカウンタの値は16進数です。dec NX-OSコマンドは、次に示すように、16進数値を10進数値に変換できます。

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

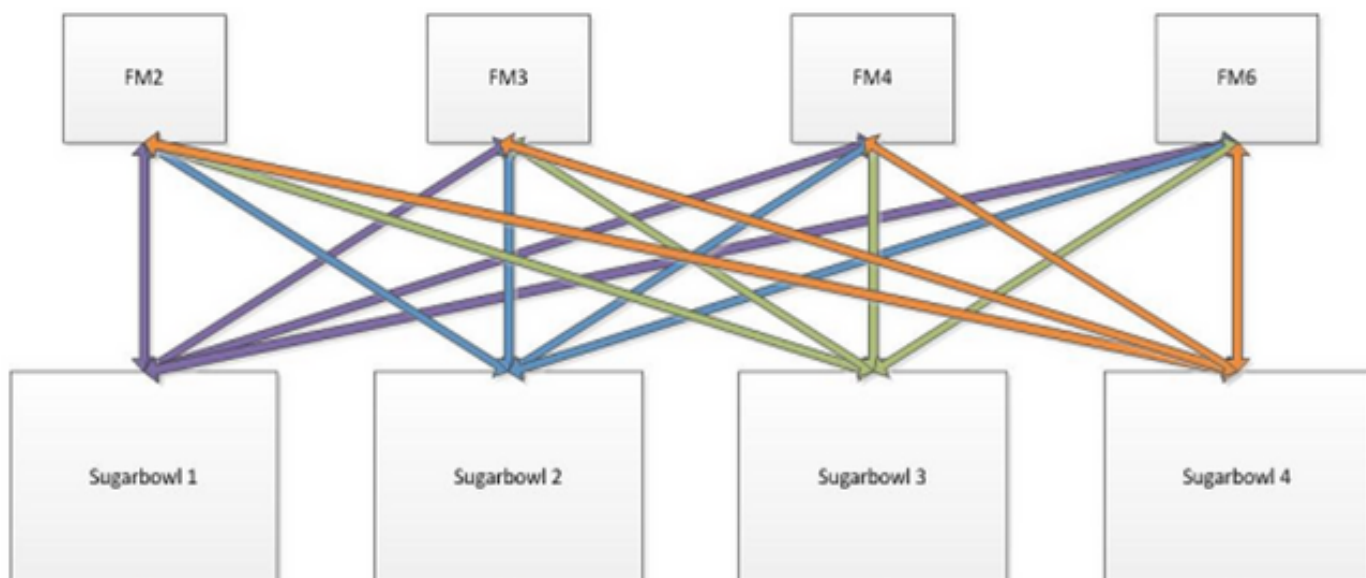
12

両方のレジスタカウンタを合わせた値は、**show interface**または**show interface counters errors non-zero**の出力によって物理インターフェイスで観察されたCRCの数と同じになります。

Cisco Nexus 9500クラウドスケール：モジュラスイッチでのCRCの識別とトレース手順

このセクションでは、Cisco Nexus 9500シリーズスイッチの特定の物理インターフェイス Ethernet1/1で見られるCRCエラーの発生源を特定する手順を説明します。

Nexus 9500シリーズスイッチの各ラインカードは、内部リンク(iEth)経由でファブリックモジュールに接続されます。各ラインカードの各ASICは、すべてのファブリックモジュールにフルメッシュ接続されています。次の例は、4つのSugarbowl ASICを搭載したラインカードを示しています。内部リンクは、モジュラ型のNexus 9500スイッチ内の4つのファブリックモジュールに接続されています。



ASICによって受信されたトラフィックが別のASICまたはラインカードから出力される必要がある場合、そのトラフィックはファブリックモジュールを介してファブリックに送信される必要があります。入力ASICは、パケットのヘッダーのハッシュとASICで使用可能なiEthリンクの数に基づいて、ファブリックモジュールへのiEthリンクの1つを選択します。

ステップ1: ラインカードとファブリックモジュール間の内部リンクをマッピングする。

`show system internal fabric connectivity module {x}` コマンド({x}はラインカードまたはファブリックモジュールの-slot番号)を使用して、指定したラインカードとすべてのファブリックモジュール間の内部リンクを表示します。この出力には、ラインカードの内部リンク(「LC-iEthLink」列の下)と各ファブリックモジュールの内部リンク(「FM-iEthLink」列の下)間の1対1のマッピングを各行に示す表が表示されます。この例を次に示します。これは、8つのラインカードと4つのファブリックモジュールが挿入されたNexus 9508スイッチから取得したものです。この出力は、スイッチの-slot8に挿入されたラインカードの各ASICインスタンスが、2つの内部リンクを介して、取り付けられた4つのファブリックモジュール(slot22、23、24、および26に挿入)のそれぞれに接続されていることを示しています。

```
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

同様に、iEthリンクマッピングは、ファブリックモジュールの観点からチェックできます。この例を次に示します。ここでは、slot22に挿入されたファブリックモジュールと、Nexus

9508シャーシに取り付けられた8枚の各ラインカード間の内部リンクが表示されています。

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	-

show system internal fabric link-state module {x}コマンドを使用して、内部ポートが稼働しているかどうか(「ST」列の下)と、特定の内部リンクの対応するASICスライスとMAC識別子(「MAC」列の下)を確認します。この例を次に示します。

```
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]

```

ステップ2:iEthリンクのCRCカウンタを確認し、破損したフレームの原因を追跡します。

モジュラ型Nexus 9500スイッチでは、次のシナリオで1つ以上のiEthリンクでCRCエラーが発生する場合があります。

1. スイッチがカットスルースイッチングモードで動作している場合、FCSフィールドに不正なCRC値を持つ破損したイーサネットフレームを受信したラインカードは、そのラインカードをローカルでは廃棄しません。代わりに、ラインカードは通常どおりパケットを転送します。パケットの出カインターフェイスが別のASICまたはラインカードに属している場合、入力ラインカードはパケットをファブリックモジュールに転送します。ファブリックモジュールもカットスルースイッチングモードで動作するため、ファブリックモジュールはパケットを出カラインカードに転送します。出カラインカードはパケットをネクストホップに転送し、出カインターフェイスの出カエラーカウンタを増加させます。
2. ハードウェアの障害が原因で内部リンクに障害が発生すると、ラインカードとファブリックモジュール間で内部リンクを通過するパケットが破損する可能性があります。

`show system internal fabric connectivity stats module {x}`コマンドを使用して、対応する内部リンクのCRCカウンタを確認します。次に例を示します。この例では、スロット22に挿入されたファブリックモジュールが、スイッチのスロット7に挿入されたラインカードのiEth26に接続されたiEth56で無効なCRCを持つパケットを受信します。これは、破損したイーサネットフレームが、スイッチのスロット7に挿入されたラインカードからファブリックモジュールで受信されていることを示します。

```

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

ラインカードまたはファブリックモジュールでslot {x} show hardware internal tah counters ASIC {y}コマンドを使用して、CRCエラーが無効であるか、CRCがストンプされているかどうかを判別します。無効なCRCエラーとストンプされたCRCエラーを区別する2つのレジスタカウンタは次のとおりです。

1. 02-RX Frm with FCS Err : 無効だがストンプされていないCRCを含むフレームを受信したことを示します。

2. 16-RX Frm CRC Err(Stomp):CRCがストンプされたフレームが受信されたことを示します。

次に例を示します。内部リンクiEth54を介してシャーシのスロット22に挿入されたファブリックモジュールで受信された破損フレームが、シャーシのスロット8に挿入されたラインカードに接続され、ストンプCRCで受信された場合です。

```
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)                144912733e
-----
                                         14491277d7
```

または、show hardware internal errors module {x}コマンドを使用して、特定のモジュールのASICエラーカウンタを表示します。この例を次に示します。この出力では、「Interface Inbound Errors (CRC,len,Algn Err)」カウンタが無効なCRCとストンプされたCRCの両方で増加し、「Interface Inbound CRC Error Stomped」カウンタがストンプされたCRCのみの増加であることに注意してください。

```
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
```

破損したフレームを受信している入力ラインカードを特定した後で、slot {x} show hardware internal tah counters ASIC {y}またはshow hardware internal errors module {x}コマンドを同様の方法で使用し、エラーを受信している入力インターフェイスを特定します。また、エラーが無効なCRCまたはストンプされたCRCとして受信されているかどうかも特定します。

まれに、ファブリックモジュールまたは出力ラインカードでiEthリンクにCRCエラーが表示されるものの、接続されたラインカードに入力CRCの兆候がない場合があります。この問題の根本原因は、通常、ファブリックモジュールのハードウェア障害です。この問題をさらにトラブルシューティングし、必要に応じてファブリックモジュールを交換するには、[Cisco TACでサポートケースをオープン](#)することを推奨します。

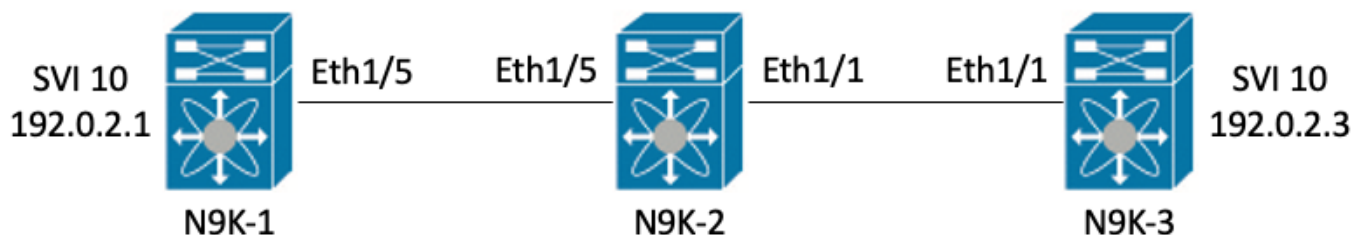
例

このセクションでは、上記の手順の一部の例について説明します。

シナリオ1：ストップされたCRCを受信する物理インターフェイス

この例では、物理インターフェイスのCRCエラーがストップされたCRCであることを識別する方法を示します。

次のトポロジについて考えます。



この例では、意図的にストップされたCRCエラーがスイッチN9K-1上で生成されます。スイッチN9K-1から、インターフェイスSVI 10 (IPアドレス192.0.2.1を所有) を送信元とし、N9K-3のインターフェイスSVI 10 (IPアドレス192.0.2.3を所有) を宛先とするジャンボサイズの8000バイトのICMPパケットがです。N9K-1、N9K-2、およびN9K-3はすべてNexus 93180YC-EXモデルスイッチです。

```
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
```

この例では、スイッチN9K-3の物理インターフェイスEthernet1/1でCRCエラーの増加が観察されています。

```
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 runs 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
```

ステップ1:CRCの増加を確認する

N9K-1のインターフェイスSVI 10 (IPアドレス192.0.2.1を所有) からN9K-3のインターフェイスSVI 10 (IPアドレス192.0.2.3を所有) を宛先として、ジャンボサイズの8000バイトのICMPパケットを生成することによって、物理インターフェイスEthernet1/1でCRCが増加していることを確認します。

```
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

```

ステップ2：物理インターフェイスのASIC、MACブロック、およびMACブロックサブポートへのマッピング

N9K-3で**show interface hardware-mappings**コマンドを使用して、物理インターフェイスEthernet1/1をASIC番号0、MACブロック4、およびMACブロックサブポート0にマッピングします。

```

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

```

ステップ3：クラウドスケールASICレジスタのCRC関連カウンタの確認

ステップ2の情報に基づいて、次の事実を確認します。

1. 物理インターフェイスEthernet1/1はASIC番号0にマッピングされます。
2. 物理インターフェイスEthernet1/1は、MACブロック4のMACブロックサブポート0にマッピングされる
3. N9K-3はトップオブラックのNexus 93180YC-EXモデルのスイッチであるため、使用可能なラインカードスロット番号は1だけであることがわかります
4. ステップ1で収集したshow interfaceの出力から、物理インターフェイスEthernet1/1の速度が10Gであることがわかります。

この情報を使用すると、**slot 1 show hardware internal tah counters ASIC 0**コマンドを使用して、すべての物理インターフェイスのASICレジスタカウンタを表示できます。具体的には、M4,0-10Gに関連するASICレジスタカウンタを探します。

```

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  ....          ....          ....          ....          ....
....          ....
```

レジスタ16にゼロ以外の16進数値0xcが表示されます。これは、この物理インターフェイスでCRCがストンプされたフレームが受信されたことを示します。dec 0xcコマンドを使用して、これを10進数の12に変換できます。12は、物理インターフェイスEthernet1/1でのCRCエラーの数と一致します。

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

シナリオ1まとめ

N9K-3が物理インターフェイスEthernet1/1でストンプされたCRCを持つフレームを受信していることを確認しました。これは、Ethernet1/1リンクのリモート側のデバイス(この場合はN9K-2)がこれらのフレームのCRCをストンプしていることを意味します。不正なフレームの根本原因は、Ethernet1/1に直接接続されているリンクではなく、さらに下流にあります。これらの不正なフレームの発生源を特定するには、ダウンストリームネットワークデバイスで追加のトラブルシューティングを実行する必要があります。

シナリオ2：物理インターフェイスが無効なCRCを持つ不正なフレームを受信する

この例では、直接接続されたリンク上の物理層の問題が原因で、不正なフレームが原因で物理インターフェイスのCRCエラーが増加していることを特定する方法を示します。

次のトポロジについて考えます。



この例では、スイッチN9K-1の物理インターフェイスEthernet1/40に接続されたトラフィックジェネレータが、意図的に誤ったCRCのフレームを生成しています。これは、Ethernet1/40に接続されたリンク上の物理層の問題(トランシーバの故障やケーブルの損傷など)をシミュレートします。N9K-1はこれらのフレームを受信し、CRCが無効であることを認識し、Ethernet1/40物理インターフェイスのCRCエラーカウンタを増やします。N9K-1はNexus 93180YC-EXモデルスイッチです。

```
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 runs  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
```

ステップ1:CRCの増加を確認する

show interfaceまたは**show interface counters non-zero**コマンドを使用して、N9K-1の物理インターフェイスEthernet1/40でCRCが増加していることを確認します。

```
N9K-1# show interface Ethernet1/40
<snip>
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
  14055 unicast packets  9873 multicast packets  0 broadcast packets
  23928 input packets  1676401 bytes
  0 jumbo packets  0 storm suppression bytes
```

```
0 runts 0 giants 14028 CRC 0 no buffer
14028 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
```

```
N9K-1# show interface counters errors non-zero
```

```
<snip>
```

```
-----
Port          Align-Err    FCS-Err    Xmit-Err    Rcv-Err    UnderSize  OutDiscards
-----
Eth1/40       26373       26373      0           26373      0          0
```

ステップ2：物理インターフェイスのASIC、MACブロック、およびMACブロックサブポートへのマッピング

N9K-1でshow interface hardware-mappingsコマンドを使用して、物理インターフェイスEthernet1/40をASIC番号0、MACブロック10、MACブロックサブポート6にマッピングします。

```
N9K-1# show interface hardware-mappings
```

```
<snip>
```

```
-----
Name          Ifindex  Smod Unit HPort FPort NPort VPort Slice SPort SrcId MacId MacSP VIF  Block
BlkSrcID
-----
Eth1/38       1a004a00 1    0    45   255  148  -1    1     5    10   10   2   149  0
10
Eth1/39       1a004c00 1    0    46   255  152  -1    1     6    12   10   4   153  0
12
Eth1/40       1a004e00 1    0    47   255  156  -1    1     7    14   10   6   157  0
14
Eth1/41       1a005000 1    0    76   255  160  -1    1    36   64   17   0   161  0
64
Eth1/42       1a005200 1    0    77   255  164  -1    1    37   66   17   2   165  0
66
```

ステップ3：クラウドスケールASICレジスタのCRC関連カウンタの確認

ステップ2の情報に基づいて、次の事実を確認します。

1. 物理インターフェイスEthernet1/40はASIC番号0にマッピングされます。
2. 物理インターフェイスEthernet1/40は、MACブロック10のMACブロックサブポート6にマッピングされます。
3. N9K-1はトップオブラックのNexus 93180YC-EXモデルのスイッチであるため、使用可能なラインカードスロット番号は1だけであることがわかります。
4. ステップ1で収集したshow interfaceの出力から、物理インターフェイスEthernet1/40の速度が10Gであることがわかります。

この情報を使用して、slot 1 show hardware internal tah counters asic 0コマンドを使用して、すべての物理インターフェイスのASICレジスタカウンタを表示できます。具体的には、M10,6-10Gに関連するASICレジスタカウンタを探します。

```
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) .....
.....
```

レジスタ2にゼロ以外の16進数値0x973eが表示される場合は、この物理インターフェイスで受信されたCRCが無効だがストップされていないフレームを示しています。dec 0x973eコマンドを使用して、これを10進数値の38,718に変換できます。この値は、物理インターフェイス Ethernet1/40のCRCエラー数と一致する（または、CRCが絶えず増加しているため、これより小さい）ものです。

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

シナリオ2まとめ

N9K-1が、物理インターフェイスEthernet1/40で無効だがストップされていないCRCを持つフレームを受信していることを確認しました。これは、Ethernet1/40（またはリンクのリモートエンドにあるデバイス）に直接接続されているリンクが、不正なフレームの原因である可能性が最も高いことを意味します。不正なフレームの根本原因を特定するには、このリンクの物理層でさらにトラブルシューティングを行う必要があります（損傷したケーブル配線のチェック、現在のトランシーバを既知の良品トランシーバに交換するなど）。

シナリオ3:Nexus 9500のiEth CRCエラーSyslog

この例では、内部インターフェイスのエラーを報告するsyslogがNexus 9500シリーズスイッチによって生成された場合に、iEth内部リンクのCRCエラーの原因を特定する方法を示します。このsyslogの例を次に示します。

```
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
```

このsyslogは、スイッチのスロット22に挿入されたファブリックモジュールのiEth56内部リンクでエラーが検出されたことを示しています。

ステップ1：ファブリックモジュール上のiEthリンクを接続されたラインカードにマッピングする

show system internal fabric connectivity stats module {x}コマンドを使用して、影響を受けるiEth内部リンクが接続しているラインカードを識別します。この例では、スイッチのスロット22に挿入されたファブリックモジュールのiEth56にエラーがあります。この例を次に示します。ここでは、スロット22に挿入されたファブリックモジュールのiEth56が、スイッチのスロット7に挿入されたラインカードのiEth26に接続されています。

```
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
```

show system internal fabric link-state module {x}コマンドを使用して、ファブリックモジュールのiEth56内部リンクに関連付けられたASICインスタンスとMAC識別子を特定します。この例を次に示します。ASICインスタンスが1で、MAC識別子が27です。

```
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]
```

ステップ2:iEthリンクで受信されたCRCが無効であるか、またはストンプされているかどうかを確認する

前のステップでは、ASICインスタンスIDが1で、スロット22に挿入されたファブリックモジュールに接続されたiEth56のMACIDが27であることが示されています。**slot {x} show hardware internal tah counters ASIC {y}**コマンドを使用して、syslogによって報告されたCRCが無効なCRCかストンプされたCRCかを識別します。この例を次に示します。M27,0-100Gx4列はMAC ID 27に関連付けられ、CRCがストンプされたことを示します。

```
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
```

または、**show hardware internal errors module {x}** コマンドを使用してこの同じ情報を収集します。この例を次に示します。

```
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
```

この出力では、「Interface Inbound Errors (CRC,len,Algn Err)」カウンタが無効なCRCとストンプされたCRCの両方で増加し、「Interface Inbound CRC Error Stomped」カウンタがストンプされたCRCのみで増加することを思い出してください。

ステップ3 : 入力ラインカードで無効なCRCを持つフレームの送信元を追跡する

これで、スイッチのスロット22に挿入されたファブリックモジュールに入るCRCが、スロット7に挿入されたラインカードからスイッチに入っていることがわかりました。この情報を使用して、**show interface counters errors module {x} non-zero**コマンドを使用して、該当するラインカードに属するインターフェイス上のゼロ以外のCRCカウンタを識別できます。この例を次に示します。

```
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
```

該当するラインカードでこのシナリオのステップ#2を繰り返して、ラインカードが無効なCRCまたはストンプされたCRCを受信しているかどうかを確認できます。

```
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 Stamped 0000051801011140 11:0
```

show interface hardware-mappingsコマンドを使用して、上記の出力のMacId:MacSP値11:0がマッピングされている前面パネルポートを特定します。この例を次に示します。11:0はフロントパネルポートEth7/32にマッピングされています。

```
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 32	1a300000	25	0	16	255	0	-1	0	16	32	4	0	1	0	
Eth7/2 24	1a300200	25	0	12	255	4	-1	0	12	24	3	0	5	0	
Eth7/3 16	1a300400	25	0	8	255	8	-1	0	8	16	2	0	9	0	
Eth7/4	1a300600	25	0	4	255	12	-1	0	4	8	1	0	13	0	8
Eth7/5 40	1a300800	25	0	60	255	16	-1	1	20	40	14	0	17	0	
Eth7/6 32	1a300a00	25	0	56	255	20	-1	1	16	32	13	0	21	0	
Eth7/7 24	1a300c00	25	0	52	255	24	-1	1	12	24	12	0	25	0	
Eth7/8 16	1a300e00	25	0	48	255	28	-1	1	8	16	11	0	29	0	
Eth7/9 24	1a301000	26	1	12	255	32	-1	0	12	24	3	0	33	0	
Eth7/10 16	1a301200	26	1	8	255	36	-1	0	8	16	2	0	37	0	
Eth7/11	1a301400	26	1	4	255	40	-1	0	4	8	1	0	41	0	8
Eth7/12	1a301600	26	1	0	255	44	-1	0	0	0	0	0	45	0	0
Eth7/13 40	1a301800	26	1	60	255	48	-1	1	20	40	14	0	49	0	
Eth7/14 32	1a301a00	26	1	56	255	52	-1	1	16	32	13	0	53	0	
Eth7/15 24	1a301c00	26	1	52	255	56	-1	1	12	24	12	0	57	0	
Eth7/16 16	1a301e00	26	1	48	255	60	-1	1	8	16	11	0	61	0	
Eth7/17 32	1a302000	27	2	16	255	64	-1	0	16	32	4	0	65	0	
Eth7/18 24	1a302200	27	2	12	255	68	-1	0	12	24	3	0	69	0	
Eth7/19 16	1a302400	27	2	8	255	72	-1	0	8	16	2	0	73	0	
Eth7/20	1a302600	27	2	4	255	76	-1	0	4	8	1	0	77	0	8
Eth7/21 40	1a302800	27	2	60	255	80	-1	1	20	40	14	0	81	0	
Eth7/22 32	1a302a00	27	2	56	255	84	-1	1	16	32	13	0	85	0	
Eth7/23 24	1a302c00	27	2	52	255	88	-1	1	12	24	12	0	89	0	
Eth7/24 16	1a302e00	27	2	48	255	92	-1	1	8	16	11	0	93	0	
Eth7/25 24	1a303000	28	3	12	255	96	-1	0	12	24	3	0	97	0	
Eth7/26 16	1a303200	28	3	8	255	100	-1	0	8	16	2	0	101	0	

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															

シナリオ3まとめ

Nexus 9500が物理インターフェイスEthernet7/32でストップされたCRCを持つフレームを受信していることを確認しました。これは、Ethernet7/32リンクのリモート側のデバイスがこれらのフレームのCRCをストップしていることを意味します。不正なフレームの根本原因は、Ethernet7/32に直接接続されているリンクではなく、さらにダウンストリームにあります。これらの不正なフレームの発生源を特定するには、ダウンストリームネットワークデバイスで追加のトラブルシューティングを実行する必要があります。

シナリオ4：出カインターフェイスで無効なCRCフレームの送信元を追跡する。

この例では、アップストリームスイッチがNexus 9500がCRCがストップされたフレームを生成していると報告した場合に、Nexus 9500スイッチで無効なCRCを持つフレームの送信元を追跡する方法を示します。このシナリオでは、アップストリームスイッチは前面パネルポートEthernet8/9を介して接続されています。

ステップ1：出カラインカードに無効なCRCフレームを送信しているファブリックモジュールを特定する

アップストリームスイッチに対して、ストップされたCRCを持つフレームを送信している出カインターフェイスはEthernet8/9であることがわかっています。まず、シャーシのスロット8に挿入されたラインカードにストップされたCRCを持つフレームを送信しているファブリックモジュールを判別する必要があります。このプロセスは、**show hardware internal errors module {x}**コマンドで開始します。この例を次に示します。

```
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
```

上記の出力のMacID:MacSP 9:0は、**show system internal fabric link-state module 8**コマンドを使用してソースファブリックモジュールにマッピングできます。この例を次に示します。

```
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]
```

スロット8に挿入されたラインカードのMAC識別子9が、シャーシのスロット22に挿入されたファブリックモジュールにマッピングされていることがわかります。内部リンクiEth10でCRCエラーが発生すると予想されます。これを検証するには、**show system internal fabric connectivity stats module 8**コマンドを使用します。この例を次に示します。

```
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

ステップ2 : ファブリックモジュール上のiEthリンクを接続されたラインカードにマッピングし、ストップCRCをチェックする

次に、シナリオ3と同じプロセスに従って、CRCを受信するiEth内部リンクをチェックし、ファブリックモジュールのASICに従ってCRCがストップされるかどうか、およびファブリックモジュールのiEth内部リンクに接続されているラインカードをチェックします。この例を次に示すには、それぞれ**show system internal fabric connectivity stats module {x}**コマンド、**show hardware internal errors module {x}**コマンド、および**show system internal fabric link-state module {x}**コマンドを使用します。

```
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]
```

ステップ3 : 入力モジュールで無効なCRCを持つフレームの送信元を追跡する

入力ラインカード (このシナリオでは、スロット22に挿入されたファブリックモジュールのiEth26からiEth56に接続されたスロット7に挿入されたラインカード) を判別した後、破損したフ

フレームがスイッチに入る入力ポートを特定します。これは、**show interface counters errors module {x} non-zero**コマンドで実行されます。show hardware internal errors module {x}コマンドとshow interface hardware-mappingsコマンドの出力では、受信したフレームが無効であるか、CRCがストップされているかどうかを検証できます。この例を次に示します。破損したフレームがフロントパネルインターフェイスEthernet7/32を通してスイッチに入ります。

```
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
```

シナリオ4まとめ

Nexus 9500が物理インターフェイスEthernet7/32でストップされたCRCを持つフレームを受信していることを確認しました。これは、Ethernet7/32リンクのリモート側のデバイスがこれらのフレームのCRCをストップしていることを意味します。不正なフレームの根本原因は、Ethernet7/32に直接接続されているリンクではなく、さらにダウンストリームにあります。これらの不正なフレームの発生源を特定するには、ダウンストリームネットワークデバイスで追加のトラブルシューティングを実行する必要があります。

関連情報

- [低遅延環境向けのカットスルーおよびストアアンドフォワードイーサネットスイッチング](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。