C8000vルータのパフォーマンス問題のトラブルシューティング

内容

はじめに

このドキュメントでは、パブリッククラウドとESXiのシナリオをまたがるC8000vエンタープライズルータのパフォーマンスの問題をトラブルシューティングする方法について説明します。

使用するコンポーネント

このドキュメントの情報は、次のハードウェアおよびソフトウェア コンポーネントに基づくものです。

- 17.12バージョンを実行するC8000v
- ESXIバージョン7.0 U3

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ました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

一般的なトラブルシューティング

C8000vを異なる環境でホストすることもできますが、C8000vがホストされている場所に関係な

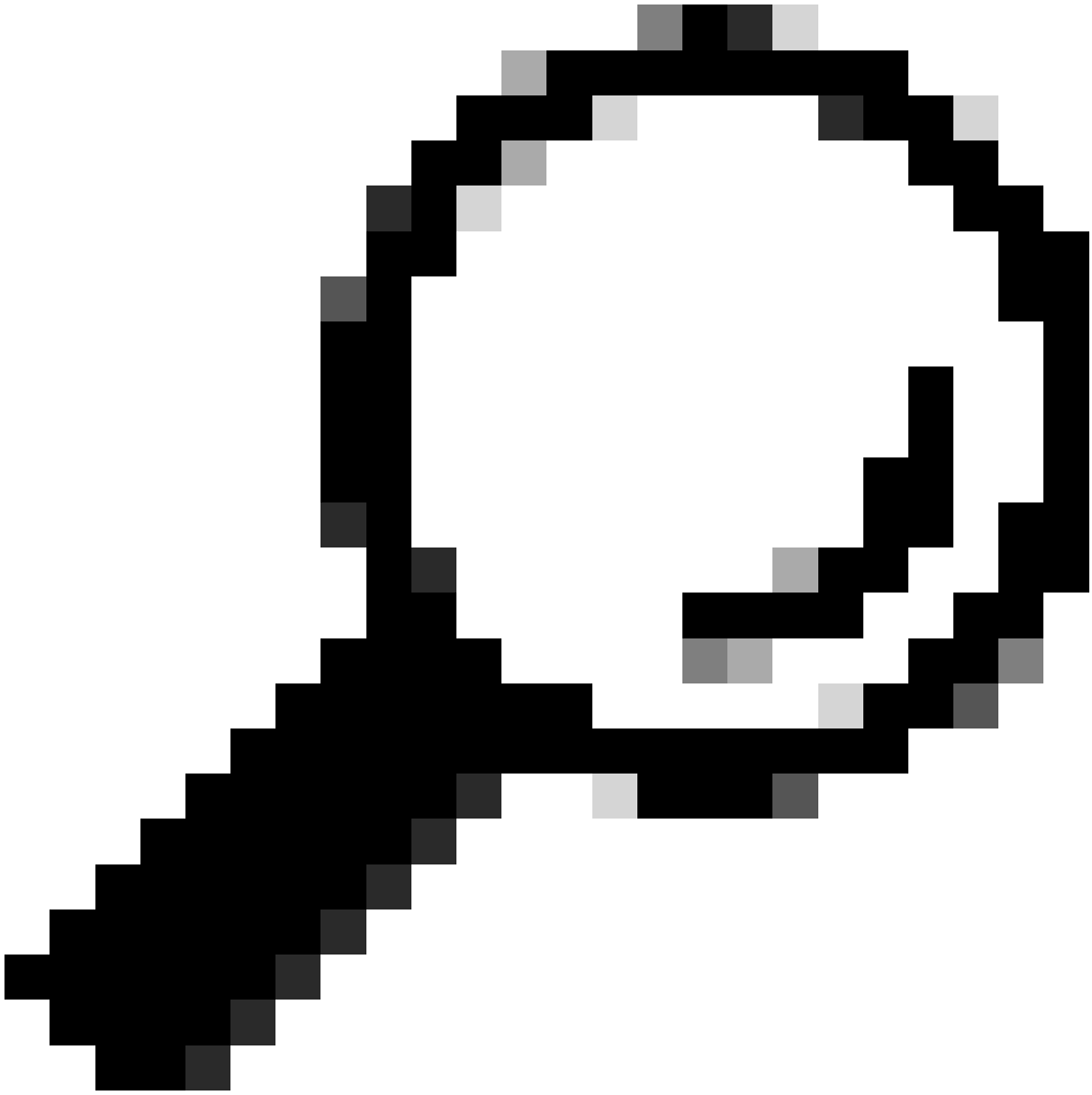
く、同じ手順を実行できるトラブルシューティング手順がいくつかあります。 まず基本から始めましょう。最初に確認する必要があるのは、デバイスが容量制限に達しているかどうかを確認することです。そのためには、最初に次の2つの出力を確認します。

1. show platform hardware qfp active datapath util summary：このコマンドを実行すると、C8000vがすべてのポートで送受信している入出力(I/O)の完全な情報が得られます。処理の負荷率に注意する必要があります。100 %に達しようとしているシナリオでは、キャパシティの限界に達していることを意味します

```
------------------ show platform hardware qfp active datapath utilization summary  ------------------

  CPP 0:                     5 secs        1 min        5 min       60 min
Input:     Total (pps)        93119        92938        65941        65131
                 (bps)    997875976   1000204000    708234904    699462016
Output:    Total (pps)        93119        92949        65944        65131
                 (bps)   1052264704   1054733128    746744264    737395744
Processing: Load (pct)           14           14           10           10
```

2. show platform hardware qfp active datapath infrastructure sw-cio：このコマンドは、上記のコマンドの詳細バージョンと見なしてください。QFP使用率の数値に含まれないIOおよび暗号化コアを含む、個々のコアに関するより詳細な情報を提供します。これは、特定のデータプレーンコアがボトルネックを引き起こしているかどうかを確認する場合に非常に便利です。



ヒント：このコマンドを使用する際には、非常に重要な詳細情報です。必ず2回実行してください。コマンドが実行された間のコア使用率のパーセンテージを計算します。

```
------------------ show platform hardware qfp active datapath infrastructure sw-cio  ------------------

Credits Usage:

  ID      Port  Wght  Global WRKR0  WRKR1  WRKR2  WRKR3  WRKR4  WRKR5  WRKR6  WRKR7  WRKR8  WRKR9  WRKR
   1      rcl0    16:    492      0      0      0      0      0      0      0      0      0      0
   1      rcl0    32:    496      0      0      0      0      0      0      0      0      0      0
   2       ipc     1:    489      0      0      0      0      0      0      0      0      0      0
   3 vxe_punti     4:    490      0      0      0      0      0      0      0      0      0      0
   4       Gi1     4:   1999      0      0      0      0      0      0      0      0      0      0
   5       Gi2     4:   1991      0      0      0      0      0      0      0      0      0      0
   6       Gi3     4:   1991      0      0      0      0      0      0      0      0      0      0
   7       Gi4     4:   1993      0      0      0      0      0      0      0      0      0      0
   8       Gi5     4:   2009      0      0      0      0      0      0      0      0      0      0
```

```
  9       Gi6     4:   2015     0      0      0      0      0      0      0      0      0      0
 10       Gi7     4:   2002     0      0      0      0      0      0      0      0      0      0
 11      vpg0   400:    490     0      0      0      0      0      0      0      0      0      0

Core Utilization over preceding 107352.2729 seconds
---------------------------------------------------
      ID:       0       1       2       3       4       5       6       7       8       9      10
    % PP:    2.98    2.01    1.81    1.67    1.60    1.53    1.35    1.30    1.25    1.19    2.19    1.
    % RX:    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.
    % TM:    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.
  % IDLE:   97.02   97.99   98.19   98.33   98.40   98.47   98.65   98.70   98.75   98.81   97.81   98.
```

これで、プラットフォームの制限に達しているかどうかを判断できました。次の手順では、ドロップを確認します。これらは本質的にパフォーマンスの問題に関連しています。ドロップが発生している場所に応じて、考慮できる3種類のドロップがあります。

- オーバーラン：このタイプのパケット廃棄はRx側で発生します。1つ以上のコアの処理能力を超えたために発生します。
- 機能ドロップ：このタイプのパケットドロップはPPEで発生します。これらは、ACLやQoSなどのルータの機能に関連しています。
- テールドロップ：このタイプのパケットドロップはTx側で発生します。これらは、Txバッファの輻輳が原因で発生します。

発生している廃棄を特定するには、次の出力を使用できます。

- show platform hardware qfp active drop state clear
- show interface
- show policy map interface（隠しコマンド）

発生しているドロップを特定する方法と、ドロップを軽減する方法を検証します。それにもかかわらず、この記事で最も焦点をあてるのは、テールドロップと呼ばれるドロップです。これは仮想ルータでは特にトラブルシューティングが難しいためです。

Overruns

Cisco IOS XEのオーバーランドロップは、ネットワークインターフェイスがパケットを処理またはバッファに格納できる速度よりも速く受信した場合に発生します。具体的には、インターフェイスの内部バッファ（FIFOキュー）がいっぱいになります。これは、着信データレートが、それを処理するハードウェアの能力を超えているためです。その結果、新しい着信パケットは保存できず、ドロップされ、overrunカウンタが増加します。これは本質的に、インターフェイスが一時的に過負荷になったために発生するパケット損失です。

このタイプのパケットドロップはRx側で発生します。これは、1つ以上のコアの処理能力を超過し、Rxスレッドが着信パケットを該当するPPスレッドに分散できず、入力バッファがすでに満杯であるために発生します。簡単な例え方として、レジの待ち行列が満杯になると、レジ係（インターフェイスハードウェア）が処理できる速度よりもパケットの到着速度が速くなります。キューがいっぱいになると、新しい顧客はサービスを受けずに退出する必要があります。これがオーバーラン廃棄です。

このセクションではハードウェアについて説明していますが、C8000vはソフトウェアベースのルータです。この場合、オーバーランは次の原因で発生する可能性があります。

- データプレーン使用率が高い：データプレーン使用率が高いと、パケットのポーリング速度が不十分になり、オーバーランが発生します。たとえば、「エレファントフロー」（大規模で継続的なデータフロー）があると、処理リソースが飽和状態になり、インターフェイスでオーバーランが発生する可能性があります。
- 不適切なデバイステンプレート：不適切なデバイステンプレートを使用すると、バッファ管理が非効率的になり、オーバーランが発生する可能性があります。これは、show platform software cpu allocコマンドで確認でき、platform resource <template>コマンドで変更できます。

各インターフェイスには限られたクレジットプールが割り当てられるため、このメカニズムによってインターフェイスのビジー状態やシステムリソースの過負荷が防止されます。新しいパケットがデータプレーンに到達するたびにクレジットが必要になります。パケット処理が完了すると、クレジットが返されるので、Rxスレッドはクレジットを再度使用できます。インターフェイスに使用可能なクレジットがない場合、パケットはインターフェイスのRxリングで待機する必要があります。 一般に、1つ以上のコアの処理能力を超過しているため、パフォーマンス制限に関連するドロップはRxオーバーランであると考えられます。

オーバーランを特定するには、通常、インターフェイスの統計情報で入力エラー、特にオーバーランカウンタを確認します。

- show platform hardware qfp active datapath infrastructure sw-cioコマンドを使用し、コアの使用率と、特定のインターフェイスが対応するクレジット数を超過しているかどうかを確認します。
- show interface <interface-name>コマンドを使用し、出力のオーバーランカウントを確認します。

オーバーランは入力エラーの一部として表示されます。次に例を示します。

```
Gig2 is up, line protocol is up
241302424557 packets input, 168997587698686 bytes, 0 no buffer
20150 input errors, 0 CRC, 0 frame, 20150 overrun, 0 ignored <<<<<<<<<<<
```

Gig2がオーバーランによるパフォーマンスの問題を観測している場合を考えてみましょう。このインターフェイスに関連付けられているワーカースレッドを判別するには、次のコマンドを使用します。

```
#show platform hardware qfp active datapath infra binding
Port Instance Bindings:

ID Port IOS Port WRKR 2
1 rcl0 rcl0 Rx+Tx
2 ipc ipc Rx+Tx
```

```
3 vxe_punti vxe_puntif Rx+Tx
4 Gi1 GigabitEthernet1 Rx+Tx
5 Gi2 GigabitEthernet2 Rx+Tx <<< in this case, WRKR2 is the thread responsible for both Rx and Tx
```

次に、このインターフェイスのRxトラフィックを担う特定のスレッドの使用率と、そのクレジット数を分析できます。

Gig2がオーバーランによるパフォーマンスの問題を観測しているシナリオでは、PP#2が常にフルに使用され（アイドル= 0 %）、インターフェイスGig2のクレジットが低いかゼロであるかが予想されます。

```
#show platform hardware qfp active datapath infrastructure sw-cio
Credits Usage:

ID Port Wght Global WRKR0 WRKR1 WRKR2 Total
1 rcl0 16: 487 0 0 25 512
1 rcl0 32: 496 0 0 16 512
2 ipc 1: 490 0 0 21 511
3 vxe_punti 4: 459 0 0 53 512
4 Gi1 4: 477 0 0 35 512
5 Gi2 4: 474 0 0 38 512 <<< low/zero credits for interface Gig2:

Core Utilization over preceding 1.0047 seconds
-----------------------------------------------
ID: 0 1 2
% PP: 0.77 0.00 0.00
% RX: 0.00 0.00 0.44
% TM: 0.00 0.00 5.63
% IDLE: 99.23 99.72 93.93 <<< the core ID relevant in this case would be PP#2
```

機能ドロップ

パケットは利用可能なデータプレーンスレッドで処理され、ソフトウェアRx機能(x86) – 負荷ベース分散(LBD)を介してQFPコアの可用性に厳密に基づいて分散されます。 PPEに到着したパケットは、特定のQFPドロップの理由でドロップできます。この理由は次の出力で確認できます。

```
#show drops
------------------ show platform hardware qfp active statistics drop detail ------------------

Last clearing of QFP drops statistics : never

-------------------------------------------------------------------------------
   ID  Global Drop Stats                          Packets                 Octets
-------------------------------------------------------------------------------
  319  BFDoffload                                    403                  31434
  139  Disabled                                      105                   7487
   61  Icmp                                          135                   5994
   94  Ipv4NoAdj                                       1                    193
   33  Ipv6NoRoute                                  2426                 135856
  215  UnconfiguredIpv4Fia                       1937573              353562196
```

```
  216  UnconfiguredIpv6Fia                        8046173              1057866418


------------------ show platform hardware qfp active interface all statistics drop_summary ------------

-----------------------------------------------------------------
Drop Stats Summary:
note: 1) these drop stats are only updated when PAL
         reads the interface stats.
      2) the interface stats include the subinterface

Interface                                      Rx Pkts             Tx Pkts
-------------------------------------------------------------------------
GigabitEthernet1                               9980371                   0
GigabitEthernet2                                  4012                   0
```

ドロップの理由は多岐にわたりますが、通常は説明の必要はありません。さらに調査するには、パケットトレースを使用できます。

テールドロップ

前述したように、テールドロップは、デバイスがパケットを送信しようとしているが、送信バッファがいっぱいであるときに発生します。

このサブセクションでは、このタイプの状況に直面したときに調査できる出力を見ていきます。これらの中に表示される値は何を意味し、問題を軽減するために何ができるかです。

まず、それらを識別する方法を知る必要があります。その1つとして、show interfaceコマンドの出力を確認するだけの方法があります。出力ドロップの増加に注意してください。

```
GigabitEthernet2 is up, line protocol is up
Hardware is vNIC, address is 0050.56ad.c777 (bia 0050.56ad.c777)
Description: Connected-To-ASR Cloud Gateway
Internet address is 10.6.255.81/29
MTU 1500 bytes, BW 10000000 Kbit/sec, DLY 10 usec,
reliability 255/255, txload 2/255, rxload 3/255
Encapsulation ARPA, loopback not set
Keepalive set (10 sec)
Full Duplex, 10000Mbps, link type is force-up, media type is Virtual
output flow-control is unsupported, input flow-control is unsupported
ARP type: ARPA, ARP Timeout 04:00:00
Last input 00:00:00, output 00:00:00, output hang never
Last clearing of "show interface" counters 03:16:21
Input queue: 0/375/0/0 (size/max/drops/flushes); Total output drops: 7982350 <<<<<<<<
Queueing strategy: fifo
Output queue: 0/40 (size/max)
5 minute input rate 150449000 bits/sec, 20461 packets/sec
5 minute output rate 89116000 bits/sec, 18976 packets/sec
```

このコマンドは、輻輳が発生しているかどうかを理解するのに特に役立ちます。

- show platform hardware qfp active datapath infrastructure:HQFは「Hierarchical Queueing Framework」の略です。これは、モジュラQoSコマンドラインインターフェイス(MQC)を使用して、異なるレベル（物理、論理、クラス）でのQuality of Service(QoS)管理を可能にする機能です。 現在のRXおよびTXコストが表示されます。 出力に示されているように、TXキューがいっぱいになった場合(full 1959)

```
pmd b1689fc0 device Gi1
RX: pkts 5663120 bytes 1621226335 return 0 badlen 0
Out-of-credits: Hi 0 Lo 0
pkts/burst 1 cycl/pkt 1565 ext_cycl/pkt 1173
Total ring read 12112962299, empty 12107695202
TX: pkts 8047873582 bytes 11241140363740
pri-0: pkts 8047873582 bytes 11241140363740
pkts/send 3
Total: pkts/send 3 cycl/pkt 452
send 2013612969 sendnow 1810842
forced 2013274797 poll 724781 thd_poll 0
blocked 2197451 retries 7401 mbuf alloc err 0
TX Queue 0: full 1959 current index 0 hiwater 224
```

この出力は、基盤となるハードウェアがパケットの送信に対応していないことを示しています。基盤となるインターフェイスをデバッグするには、C8000vの外部、およびC8000vが稼働している基盤となる環境を調査して、基盤となる物理インターフェイスに追加のエラーが報告されていないかどうかを確認する必要がある場合があります。

環境を確認するには、C8000vルータで実行されているハイパーバイザを確認する前に実行できる1つの手順があります。これは、show controllerコマンドの出力を調べることです。それにもかかわらず、各カウンターの意味や見るべき場所について自分自身を失うことができます。

まず、この出力を見る際に注意する必要のある重要な詳細の1つは、情報の大部分がvNIC自体から発信されているということです。各NICドライバには、使用する特定のカウンタのセットがあります。カウンタのセットはドライバによって自然に異なります。異なるハイパーバイザは、表示される内容に対してある種の影響を及ぼします。mbufカウンタなど、一部のカウンタはDPDKドライバからの統計情報です。これは、DPDKドライバによって異なります。実際のカウントは、一般にハイパーバイザによって仮想化レイヤで行われます。

```
GigabitEthernet2 - Gi2 is mapped to UIO on VXE
rx_good_packets 1590
tx_good_packets 1402515
rx_good_bytes 202860
tx_good_bytes 1857203911
rx_missed_errors 0
rx_errors 0
tx_errors 0
rx_mbuf_allocation_errors 0
rx_q0_packets 1590
rx_q0_bytes 202860
rx_q0_errors 0
tx_q0_packets 1402515
tx_q0_bytes 1857203911
```

```
rx_q0_drop_total 0
rx_q0_drop_err 0
rx_q0_drop_fcs 0
rx_q0_rx_buf_alloc_failure 0
tx_q0_drop_total 976999540797
tx_q0_drop_too_many_segs 0
tx_q0_drop_tso 0
tx_q0_tx_ring_full 30901211518
```

これらのカウンタの解釈方法と解釈方法については、ここで少し説明します。

1. 「subX」が表示される場合は、サブインターフェイス (メインインターフェイスの論理分割) であることを意味します。通常、sub0はプライマリ/デフォルトです。これらは、複数のVLANが関係する場合によく使用されます。
2. 次に、「rx = receiving」と「tx = transmitting」があります。
3. 最後に、q0は、そのインターフェイスで使用される最初の/デフォルトキューを指します

すべてのカウンタについて説明しているわけではありませんが、トラブルシューティングに関係する可能性がある一部のカウンタについて説明しています。

- 「RX_MISSED_ERRORS:」は、NICバッファ(Rx FIFO)がいっぱいになると表示されます。この状態は、ドロップと遅延の増加につながります。この問題を解決するには、NICバッファを増やすか (この場合は不可能)、NICドライバを変更します。
- 「tx_q0_drop_total」および「tx_q0_tx_ring_full」：これらは、ホストがパケットを廃棄しており、ホストがC8000vをバックプレッシャーしているため、C8000vでテールドロップが発生していることを示しています

上記の出力では、「rx_missed_errors」は表示されていません。ただし、テールドロップに焦点を当てているので、「tx_q0_drop_total」と「tx_q0_tx_ring_full」の両方が表示されます。 これにより、ホストの基盤となるハードウェアによって実際に輻輳が発生していると結論付けることができます。

前述のとおり、各ハイパーバイザは表示される内容に何らかの影響を与えます。この記事では、C8000vをホストできるさまざまなハイパーバイザ間の違いについて説明するため、次のセクションでこの点に焦点を当てます。また、それぞれの問題でこのタイプの問題を軽減するためのさまざまな推奨事項を見つけることができます。

ハイパーバイザ

ハイパーバイザは、CPU、メモリ、ストレージなどのハードウェアリソースを各VMに割り当てて管理することで、複数のオペレーティングシステム(仮想マシン(VM)を1つの物理ハードウェアホスト上で実行できるようにするソフトウェアレイヤです。これらの仮想マシンが互いに干渉することなく独立して動作することを保証します。

Cisco Catalyst 8000V(C8000v)では、ハイパーバイザはC8000v仮想マシンをホストするプラットフォームです。C8000vをホスティングしているハイパーバイザを調べるにはどうすればよいですか。この情報を提供する有用な出力があります。また、仮想ルータがアクセスできるリソースの種類も確認できます。

```
C8000v#show platform software system all
Processor Details
=================
Number of Processors : 8
Processor : 1 - 8
vendor_id : GenuineIntel
cpu MHz : 2593.906
cache size : 36608 KB
Crypto Supported : Yes
model name : Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz

Memory Details
==============
Physical Memory : 32817356KB

VNIC Details
============
Name Mac Address Driver Name Status Platform MTU
GigabitEthernet1 0022.480d.7a05 net_netvsc UP 1500
GigabitEthernet2 6045.bd69.83a0 net_netvsc UP 1500
GigabitEthernet3 6045.bd69.8042 net_netvsc UP 1500

Hypervisor Details
==================
Hypervisor: AZURE
Manufacturer: Microsoft Corporation
Product Name: Virtual Machine
Serial Number: 0000-0002-0201-5310-5478-4052-71
UUID: 8b06091c-f1d3-974c-85a5-a78dfb551bf2
Image Variant: None
```

VMware ESXi

ESXiは、VMwareによって開発されたタイプ1ハイパーバイザであり、仮想化を可能にするために物理サーバに直接インストールされます。ハードウェアリソースを抽象化して各VMに割り当てることにより、複数の仮想マシン(VM)を1台の物理サーバ上で実行できます。C8000vルータはこれらのVMの1つです。

輻輳が発生している一般的なシナリオを検討することから始めます。これは、tx_q0_tx_ring_full カウンタをチェックすることで確認できます。

例：

```
------------------ show platform software vnic-if interface-mapping ------------------

---------------------------------------------------------------
Interface Name Driver Name Mac Addr
---------------------------------------------------------------
GigabitEthernet3 net_vmxnet3 <-- 0050.5606.2239
GigabitEthernet2 net_vmxnet3 0050.5606.2238
GigabitEthernet1 net_vmxnet3 0050.5606.2237
```

```
-------------------------------------------------------------

GigabitEthernet3 - Gi3 is mapped to UIO on VXE
rx_good_packets 99850846
tx_good_packets 24276286
rx_good_bytes 78571263015
tx_good_bytes 14353154897
rx_missed_errors 0
rx_errors 0
tx_errors 0
rx_mbuf_allocation_errors 0
rx_q0packets 99850846
rx_q0bytes 78571263015
rx_q0errors 0
tx_q0packets 24276286
tx_q0bytes 14353154897
rx_q0_drop_total 0
rx_q0_drop_err 0
rx_q0_drop_fcs 0
rx_q0_rx_buf_alloc_failure 0
tx_q0_drop_total 160945155
tx_q0_drop_too_many_segs 0
tx_q0_drop_tso 0
tx_q0_tx_ring_full 5283588 <------
```

この輻輳は、C8000VがVMXNET3インターフェイス経由でパケットを送信しようとしたときに発生します。ただし、バッファリングにはすでにパケットがいっぱいになり、遅延またはドロップが発生します。

このような状況では、前述したように、これらのドロップはハイパーバイザ側で発生します。すべての推奨事項が満たされている場合、NICで何が起こっているのかをVMwareサポートに確認することをお勧めします。

パフォーマンスを向上させる方法に関する提案を次に示します。

- 専用のvSwitchとアップリンクを使用して最適なパフォーマンスを実現
- C8000Vを専用のvSwitchに割り当て、その物理アップリンクでバックアップすることで、そのトラフィックをノイズの多いネイバーから分離し、共有リソースのボトルネックを回避できます。

ESXi側から見る価値のあるコマンドがいくつかあります。たとえば、ESXiインターフェイスからのパケット損失を確認するには、次の手順を実行します。

1. SSH の有効化.
2. SSHを使用してESXiに接続します。
3. esxtopを実行します。
4. nと入力します。

esxtopコマンドでは、仮想マシンのネットワークドライバでRxバッファメモリが使い果たされた場合に、仮想スイッチでドロップされたパケットが表示される場合があります。esxtopは仮想スイッチでドロップされたパケットを示しますが、実際には仮想スイッチとゲストオペレーティングシステムドライバの間でドロップされます。

%DRPTXおよび%DRPRXでドロップされているパケットを検索します。

```
12:34:43pm up 73 days 16:05, 907 worlds, 9 VMs, 53 vCPUs; CPU load average: 0.42, 0.42, 0.42

PORT-ID USED-BY TEAM-PNIC DNAME PKTTX/s MbTX/s PSZTX PKTRX/s MbRX/s PSZRX %DRPTX %DRPRX
67108870 Management n/a vSwitch-to-9200 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
67108872 Shadow of vmnic1 n/a vSwitch-to-9200 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
67108876 vmk1 vmnic1 vSwitch-to-9200 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
67108890 2101719:c8kv-gw-mgmt vmnic1 vSwitch-to-9200 76724.83 792.35 1353.00 16180.39 9.30 75.00 0.00 0
100663305 Management n/a vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
100663307 Shadow of vmnic0 n/a vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
100663309 vmk0 vmnic0 vSwitch-to-Cisc 3.64 0.01 280.00 3.29 0.00 80.00 0.00 0.00
100663310 2100707:gsoaresc-On_Prem vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 2.43 0.00 60.00 0.00 0.00
100663311 2100993:cats-vmanage void vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
100663312 2100993:cats-vmanage void vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
100663313 2100993:cats-vmanage vmnic0 vSwitch-to-Cisc 5.38 0.01 212.00 9.71 0.01 141.00 0.00 0.00
100663314 2101341:cats-vsmart void vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
100663315 2101341:cats-vsmart vmnic0 vSwitch-to-Cisc 2.60 0.00 164.00 6.94 0.01 124.00 0.00 0.00
100663316 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00
100663317 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 0.00 0.00 0.00 0.00 100.00
100663318 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 4.33 0.01 174.00 7.80 0.01 162.00 0.00 0.00
100663319 2101522:cats-vbond vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 4.16 0.00 90.00 0.00 0.00
100663320 2101547:gdk-backup vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
100663321 2101703:sevvy vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
100663323 2101719:c8kv-gw-mgmt vmnic0 vSwitch-to-Cisc 16180.91 9.09 73.00 76755.87 792.44 1353.00 0.00 0
100663324 2137274:telemetry-server vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
100663335 2396721:netlab vmnic0 vSwitch-to-Cisc 0.00 0.00 0.00 3.12 0.00 77.00 0.00 0.00
2214592519 vmnic1 - vSwitch-to-9200 76727.26 792.38 1353.00 16182.64 9.30 75.00 0.00 0.00
2248146954 vmnic0 - vSwitch-to-Cisc 16189.05 9.32 75.00 76736.97 792.38 1353.00 0.00 0.00
```

次のコマンドは、ホスト上で設定されているすべてのNICを一覧表示します。

```
esxcli network nic list
Name PCI Device Driver Admin Status Link Status Speed Duplex MAC Address MTU Description
------ ------------ ------ ------------ ----------- ----- ------ ----------------- ---- -----------
vmnic0 0000:01:00.0 igbn Up Up 1000 Full fc:99:47:49:c5:0a 1500 Intel(R) I350 Gigabit Network Connectio
vmnic1 0000:01:00.1 igbn Up Up 1000 Full fc:99:47:49:c5:0b 1500 Intel(R) I350 Gigabit Network Connectio
vmnic2 0000:03:00.0 ixgben Up Up 1000 Full a0:36:9f:1c:1f:cc 1500 Intel(R) Ethernet Controller 10 Gigab
vmnic3 0000:03:00.1 ixgben Up Up 1000 Full a0:36:9f:1c:1f:ce 1500 Intel(R) Ethernet Controller 10 Gigab
```

特定のVMに割り当てられたvNICのステータスを確認する便利なコマンドもあります。

```
esxcli network vm list
World ID Name Num Ports Networks
-------- ---------------- --------- --------
2137274 telemetry-server 1 Cisco Backbone 10.50.25.0/24
2101703 sevvy 1 Cisco Backbone 10.50.25.0/24
2396721 netlab 1 Cisco Backbone 10.50.25.0/24
```

```
2101547 gdk-backup 1 Cisco Backbone 10.50.25.0/24
2101522 cats-vbond 4 VPN0, VPN0, VPN0, VPN0
2101719 c8kv-gw-mgmt 2 c8kv-to-9200l, c8kv-to-cisco
2100707 gsoaresc-On_Prem 1 Cisco Backbone 10.50.25.0/24
2100993 cats-vmanage 3 VPN0, VPN0, VPN0
2101341 cats-vsmart 2 VPN0, VPN0
[root@localhost:~]
```

c8kv-gw-mgmt(C8000v VM)を見ると、2つのネットワークが割り当てられています。

- c8kv-to-9200l
- c8kvとシスコ

ワールドIDを使用して、このVMに関する詳細を検索できます。

```
[root@localhost:~] esxcli network vm port list -w 2101719
Port ID: 67108890
vSwitch: vSwitch-to-9200L
Portgroup: c8kv-to-9200l
DVPort ID:
MAC Address: 00:0c:29:31:a6:b6
IP Address: 0.0.0.0
Team Uplink: vmnic1
Uplink Port ID: 2214592519
Active Filters:

Port ID: 100663323
vSwitch: vSwitch-to-Cisco
Portgroup: c8kv-to-cisco
DVPort ID:
MAC Address: 00:0c:29:31:a6:ac
IP Address: 0.0.0.0
Team Uplink: vmnic0 <----
Uplink Port ID: 2248146954
Active Filters:
[root@localhost:~]
```

この情報を入手したら、vSwitchが割り当てられているネットワークを特定できます。

vSwitchに割り当てられた物理NICのトラフィック統計情報を確認するには、次のコマンドを使用します。

```
# esxcli network nic stats get -n <vmnic>
```

このコマンドは、受信されたパケット、受信されたバイト数、ドロップされたパケット数、受信されたエラー数などの情報を表示します。これは、NICでドロップが発生しているかどうかを特定するのに役立ちます。

```
[root@localhost:~] esxcli network nic stats get -n vmnic0
NIC statistics for vmnic0
Packets received: 266984237
Packets sent: 123640666
Bytes received: 166544114308
Bytes sent: 30940114661
Receive packets dropped: 0
Transmit packets dropped: 0
Multicast packets received: 16773454
Broadcast packets received: 36251726
Multicast packets sent: 221108
Broadcast packets sent: 1947649
Total receive errors: 0
Receive length errors: 0
Receive over errors: 0
Receive CRC errors: 0
Receive frame errors: 0
Receive FIFO errors: 0
Receive missed errors: 0
Total transmit errors: 0
Transmit aborted errors: 0
Transmit carrier errors: 0
Transmit FIFO errors: 0
Transmit heartbeat errors: 0
Transmit window errors: 0
```

ホストと仮想マシンの設定を変更することで、ESXi環境で実行されるCisco Catalyst 8000Vのパフォーマンスを向上させることができる検証すべき設定がいくつかあります。

- Virtual Hardware: CPU reservation設定をMaximumに設定します。
- 仮想ハードウェア：メモリですべてのゲストメモリを予約します。
- [Virtual Hardware: SCSI Controller]から[VMware Paravirtual]を選択します。
- Virtual Hardware: Network Adapter: Adapter Typeオプションから、サポートされているNICに対してSR-IOVを選択します
- General Guest OS Version > VM OptionsオプションをOther 3.x or later Linux (64-bit)に設定します。
- Advanced Latency SensitivityのVM OptionsオプションをHighに設定します。
- VM Options > Advanced Edit Configurationで、SRIOV NICと同じNUMAノードに「numa.nodeAffinity」を追加します
- ハイパーバイザのパフォーマンス設定を有効にします。
- サポートされている物理NICでSR-IOVを有効にして、vSwitchのオーバーヘッドを制限します。
- 物理NICと同じNUMAノードで実行するようにVMのvCPUを設定します。
- VM Latency SensitivityをHighに設定します。

AWS

C8000vは、Amazon Virtual Private Cloud (VPC)内でAmazon Machine Image (AMI)として起動することで、AWSでのデプロイをサポートします。これにより、ユーザーはネットワークリソース

用にAWSクラウドの論理的に分離されたセクションをプロビジョニングできます。

マルチTXキュー

AWSで実行されているC8000vの主な機能は、マルチTXキュー（マルチTXQ）の使用です。 これらのキューは、内部処理のオーバーヘッドを削減し、拡張性を向上させるのに役立ちます。複数のキューを使用すると、着信パケットと発信パケットを適切な仮想CPU(vCPU)にすばやく簡単に割り当てることができます。

RX/TXキューがvCPUごとに割り当てられる一部のシステムとは異なり、C8000vでは、これらのキューはインターフェイスごとに割り当てられます。RX（受信）キューとTX（送信）キューは、Catalyst 8000VアプリケーションとAWSインフラストラクチャまたはハードウェア間の接続ポイントとして機能し、ネットワークトラフィックの送受信方法を管理します。AWSは、インスタンスタイプに応じて、各インターフェイスで利用可能なRX/TXキューの数と速度を制御します。

複数のTXキューを作成するには、Catalyst 8000Vに複数のインターフェイスが必要です。複数のTXキューが有効な場合、デバイスはフローの5タプル（送信元IP、宛先IP、送信元ポート、宛先ポート、およびプロトコル）に基づくハッシュ方式を使用して、パケットフローの順序を維持します。 このハッシングにより、各フローに使用するTXキューが決定されます。

ユーザは、AWSインスタンスに接続されている同じ物理ネットワークインターフェイスカード(NIC)を使用して、Catalyst 8000V上に複数のインターフェイスを作成できます。これは、ループバックインターフェイスを設定するか、セカンダリIPアドレスを追加することによって行われます。

マルチTXQでは、発信トラフィックを処理するために複数の送信キューがあります。この例では、12個のTxキュー（番号0 ~ 11）があります。 この設定では、各キューを個別に監視して、キューがいっぱいになっているかどうかを確認できます。

出力を見ると、TX Queue 8の「full」カウンタ(56,406,998)が非常に高く、バッファが頻繁に満杯になっていることがわかります。その他のTXキューは「full」カウンタにゼロを示し、輻輳していないことを示します。

```
Router#show platform hardware qfp active datapath infrastructure sw-cio
pmd b17a2f00 device Gi2
RX: pkts 9525 bytes 1229599 return 0 badlen 0
Out-of-credits: Hi 0 Lo 0
pkts/burst 1 cycl/pkt 560 ext_cycl/pkt 360
Total ring read 117322273, empty 117312792
TX: pkts 175116324 bytes 246208197526
pri-0: pkts 157 bytes 10238
pkts/send 1
pri-1: pkts 75 bytes 4117
pkts/send 1
pri-2: pkts 91 bytes 6955
pkts/send 1
pri-3: pkts 95 bytes 8021
```

```
pkts/send 1
pri-4: pkts 54 bytes 2902
pkts/send 1
pri-5: pkts 75 bytes 4082
pkts/send 1
pri-6: pkts 104 bytes 8571
pkts/send 1
pri-7: pkts 74 bytes 4341
pkts/send 1
pri-8: pkts 175115328 bytes 246208130411
pkts/send 2
pri-9: pkts 85 bytes 7649
pkts/send 1
pri-10: pkts 106 bytes 5784
pkts/send 1
pri-11: pkts 82 bytes 7267
pkts/send 1
Total: pkts/send 2 cycl/pkt 203
send 68548581 sendnow 175024880
forced 1039215617 poll 1155226129 thd_poll 0
blocked 2300918060 retries 68534370 mbuf alloc err 0
TX Queue 0: full 0 current index 0 hiwater 0
TX Queue 1: full 0 current index 0 hiwater 0
TX Queue 2: full 0 current index 0 hiwater 0
TX Queue 3: full 0 current index 0 hiwater 0
TX Queue 4: full 0 current index 0 hiwater 0
TX Queue 5: full 0 current index 0 hiwater 0
TX Queue 6: full 0 current index 0 hiwater 0
TX Queue 7: full 0 current index 0 hiwater 0
TX Queue 8: full 56406998 current index 224 hiwater 224 <<<<<<<<<<
TX Queue 9: full 0 current index 0 hiwater 0
TX Queue 10: full 0 current index 0 hiwater 0
TX Queue 11: full 0 current index 0 hiwater 0
```

TXキューの「フル」カウンタを監視すると、送信キューが過負荷になっているかどうかを確認するのに役立ちます。特定のTXキューの「フル」カウントが増加し続けている場合は、デバイスに負荷のかかるトラフィックフローを示しています。これに対処するには、トラフィックバランシング、設定の調整、またはパフォーマンスを向上させるためのリソースの拡張が必要です。

メトリックの超過

AWSは、インスタンスレベルで一定のネットワーク制限を設定し、異なるインスタンスサイズ間で一貫した高品質のネットワークパフォーマンスを保証します。これらの制限により、すべてのユーザに対して安定したネットワークを維持できます。

ご使用のデバイスでshow controllersコマンドを使用すると、これらの制限と関連する統計情報をチェックできます。出力には多くのカウンタが含まれていますが、ここではネットワークパフォーマンスの監視に最も重要なカウンタのみに焦点を当てます。

```
c8kv-2#sh control | inc exceed
<snipped>
bw_in_allowance_exceeded 0
bw_out_allowance_exceeded 0
```

```
pps_allowance_exceeded 0
conntrack_allowance_exceeded 0
linklocal_allowance_exceeded 0
<snipped>
```

これらのカウンタが正確に参照している内容を確認できます。

- bw_in_allowance_exceeded:受信帯域幅がインスタンスの制限を超えたためにキューに入れられたか、ドロップされたパケットの数。
- bw_out_allowance_exceeded:発信帯域幅がインスタンスの制限を超えたためにキューに入れられたか、ドロップされたパケットの数。
- pps_allowance_exceeded:1秒あたりのパケット総数(PPS)がインスタンスの制限を超えたためにキューに入れられたか、ドロップされたパケットの数。
- conntrack_allowance_exceeded:追跡された接続のうち、インスタンスタイプで許可されている最大数に達した接続の数。
- linklocal_allowance_exceeded：ローカルプロキシサービス（Amazon DNS、インスタンスメタデータサービス、Time Syncサービスなど）へのトラフィックがネットワークインターフェイスのPPS制限を超えたためにドロップされたパケットの数。これは、カスタムDNSリゾルバには影響しません。

これは、C8000vのパフォーマンスにとって何を意味しますか。

- これらのカウンタが増加していて、パフォーマンスの問題が発生している場合は、必ずしもC8000vルータに問題があるとは限りません。その代わりに、使用しているAWSインスタンスが容量制限に達したことを示します。AWSインスタンスの仕様を確認して、トラフィックのニーズに対応できることを確認できます。

Microsoft Azure

このセクションでは、Microsoft AzureとCisco C8000v仮想ルータを組み合わせて、スケーラブルで安全かつ高性能な仮想ネットワーキングソリューションをクラウドで提供する方法について説明します。

Accelerated Networking(AN)とパケットフラグメンテーションがパフォーマンスに与える影響について説明します。また、Microsoft Azureでサポートされているインスタンスタイプを使用することの重要性を確認します。

ネットワークの高速化

C8000vがMicrosoft Azureクラウドでホストされている場合のパフォーマンスの問題です。見落とすことができない点の1つは、Accelerated Networkが有効になっているかどうかです。ルータのパフォーマンスが大幅に向上します。つまり、高速ネットワーキングにより、Cisco Catalyst 8000V VMなどのVM上でシングルルートI/O仮想化(SR-IOV)が可能になります。高速化されたネットワーキングパスは、仮想スイッチをバイパスし、ネットワークトラフィックの速度を向上させ、ネットワークパフォーマンスを向上させ、ネットワーク遅延とジッタを削減します。

Accelerated Network（ASP；高速ネットワーク）が有効になっているかどうかを確認する簡単な

方法があります。これは、show controllersの出力をチェックして、特定のカウンタが存在するかどうかを確認することです。

```
------------------ show controllers ------------------

GigabitEthernet1 - Gi1 is mapped to UIO on VXE
  rx_good_packets 6497723453
  tx_good_packets 14690462024
  rx_good_bytes 2271904425498
  tx_good_bytes 6276731371987




  rx_q0_good_packets 58576251
  rx_q0_good_bytes 44254667162




  vf_rx_good_packets 6439147188
  vf_tx_good_packets 14690462024
  vf_rx_good_bytes 2227649747816
  vf_tx_good_bytes 6276731371987
```

探しているカウンタは、vf_rx_good_packetsのようにvfで始まるカウンタです。 これらのカウンタが存在することを確認すれば、高速ネットワーキングがイネーブルになっていることを確信できます。

Azureとフラグメンテーション

フラグメンテーションは、パフォーマンスに悪影響を及ぼす可能性があります。パフォーマンスに影響する主な理由の1つは、パケットのフラグメンテーションと再構成によるCPU/メモリの影響です。ネットワークデバイスがパケットをフラグメント化する必要がある場合、フラグメント化を実行するためにCPU/メモリリソースを割り当てる必要があります。

パケットが再構成された場合も同じことが起こります。ネットワークデバイスは、すべてのフラグメントを受信するまで保存し、元のパケットに再構成できるようにする必要があります。

Azureは、高速ネットワーキングを使用してフラグメント化されたパケットを処理しません。VMがフラグメント化されたパケットを受信すると、高速化されていないパスがそれを処理します。その結果、フラグメント化されたパケットは、低遅延、ジッタの減少、毎秒のパケット数の増加など、高速ネットワーキングの利点を失います。このため、可能であればフラグメンテーションを回避することをお勧めします。

Azureはデフォルトで、フラグメント化されたパケットがVMに順番どおりに届かないようにドロップします。これは、パケットがソースエンドポイントからの送信シーケンスと一致しないことを意味します。この問題は、パケットがインターネットまたはその他の大規模なWANを通過するときに発生する可能性があります。

Microsoft Azureでサポートされているインスタンスの種類

C8000vが、シスコの標準に従ってサポートされているインスタンスタイプを使用していることが重要です。詳細は、『Cisco Catalyst 8000Vエッジソフトウェアインストールおよびコンフィギュレーションガイド』を参照してください。

この理由は、このリストのインスタンスタイプがC8KVが適切にテストされたインスタンスタイプであるためです。次に、リストにないインスタンスタイプでC8000vが動作するかどうかを確認する質問があります。答えは「イエス」です。ただし、パフォーマンスの問題と同じくらい複雑な問題をトラブルシューティングする場合は、別の未知の要因を問題に追加する必要はありません。そのため、Cisco TACでは、サポートされているインスタンスタイプを常に使用することをお勧めします。

関連情報

パフォーマンスの問題が実際にトラブルシューティングされるのは、その瞬間に発生した場合だけです。ただし、これは常に発生する可能性があるため、捕捉が困難な場合があります。そのため、このEEMスクリプトを提供します。パケットがドロップされ始めた瞬間に重要な出力をキャプチャし、パフォーマンスの問題を解決するのに役立ちます。

```
ip access-list extended TAC
permit ip host  host



permit ip host


          host



conf t
event manager applet CONNECTIONLOST1 authorization bypass
event track 100 state down maxrun 500
action 0010 syslog msg "Logging information to file bootflash:SLA-DROPS.txt and bootflash:FIASLA_Decode
action 0020 cli command "enable"
action 0021 cli command "term length 0"
action 0022 cli command "term exec prompt timestamp"
action 0023 cli command "term exec prompt expand"
action 0095 cli command "show clock | append bootflash:SLA-DROPS.txt"
action 0096 cli command "show platform hardware qfp active statistics drop detail | append bootflash:SL
action 0097 cli command "show logging | append bootflash:SLA-DROPS.txt"
action 0099 cli command "show interfaces summary | append bootflash:SLA-DROPS.txt"
action 0100 cli command "show interfaces | append bootflash:SLA-DROPS.txt"
```

```
action 0101 cli command "show platform hardware qfp active statistics drop clear"
action 0102 cli command "debug platform packet-trace packet 2048 fia-trace"
action 0103 cli command "debug platform packet-trace copy packet both"
action 0104 cli command "debug platform condition ipv4 access-list TAC both"
action 0105 cli command "debug platform condition start"
action 0106 cli command "show platform hardware qfp active data infrastructure sw-cio | append bootflas
action 0110 wait 60
action 0111 cli command "debug platform condition stop"
action 0112 cli command "show platform packet-trace packet all decode | append bootflash:FIASLA_Decode.
action 0120 cli command "show platform packet-trace statistics | append bootflash:FIASLA_Decode.txt"
action 0121 cli command "show platform packet-trace summary | append bootflash:FIASLA_Decode.txt"
action 0122 cli command "show platform hardware qfp active datapath utilization summary | append bootfl
action 0123 cli command "show platform hardware qfp active statistics drop detail | append bootflash:SL
action 0124 cli command "show platform hardware qfp active infrastructure bqs queue output default all
action 0125 cli command "show platform software status control-processor brief | append bootflash:SLA-D
action 0126 cli command "show platform hardware qfp active datapath infrastructure sw-pktmem | append b
action 0127 cli command "show platform hardware qfp active infrastructure punt statistics type per-caus
action 0128 cli command "show platform hardware qfp active statistics drop | append bootflash:SLA-DROPS
action 0129 cli command "show platform hardware qfp active infrastructure bqs queue output default all
action 0130 cli command "show platform hardware qfp active data infrastructure sw-hqf config 0 0 | appe
action 0131 cli command "show platform hardware qfp active feature lic-bw oversubscription | append boo
action 0132 cli command "show platform hardware qfp active data infrastructure sw-hqf config 0 0 | appe
action 0133 cli command "show platform hardware qfp active data infrastructure sw-cio | append bootflas
action 0134 cli command "show platform hardware qfp active data infrastructure sw-hqf sched | append bo
action 0135 cli command "show platform hardware qfp active data infrastructure sw-dist | append bootfla
action 0136 cli command "show platform hardware qfp active data infrastructure sw-nic | append bootflas
action 0137 cli command "show platform hardware qfp active data infrastructure sw-pktmem | append bootf
action 0138 cli command "show controllers | append bootflash:SLA-DROPS.txt"
action 0139 cli command "show platform hardware qfp active datapath pmd controllers | append bootflash:
action 0140 cli command "show platform hardware qfp active datapath pmd system | append bootflash:SLA-D
action 0141 cli command "show platform hardware qfp active datapath pmd static-if-config | append bootf
action 0150 cli command "clear platform condition all"
action 0151 cli command "clear platform packet-trace statistics"
action 0152 cli command "clear platform packet-trace configuration"
action 0153 cli command "show log | append bootflash:througput_levelinfoSLA.txt"
action 0154 cli command "show version | append bootflash:througput_levelinfoSLA.txt"
action 0155 cli command "show platform software system all | append bootflash:througput_levelinfoSLA.tx
action 0156 syslog msg "EEM script and FIA trace completed."
action 0180 cli command "conf t"
action 0181 cli command "no event manager applet CONNECTIONLOST1"
end
```

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。