

Nexus 9500-R、Nexus 3000-R:入力廃棄のトラブルシューティング

内容

[概要](#)

[背景説明](#)

[Ingress Traffic Manager\(ITM\)](#)

[入力VOQバッファ](#)

[スケジューリングおよびフロー制御](#)

[よくある原因](#)

[該当ハードウェア](#)

[入力廃棄のトラブルシューティング](#)

[一般的なシナリオ – 10Gから1Gへのトラフィックフロー – 継続的なドロップ :](#)

[ステップ1: 入力廃棄インターフェイスに影響するキューを確認します。](#)

[ステップ2: 診断に使用されるBroadcomのカウンタのグラフィック表示を確認します。](#)

[ステップ3: 入力廃棄が発生している前面パネルポートが属するASICとJerichoポートを見つけます。](#)

[ステップ4: 入力ポートのVOQおよびVOQコネクタについて理解します。](#)

[ステップ5: BCMの観点から確認します。キューは特に空ではありません。つまり、輻輳です。](#)

[ステップ6: 空でない\[Queue\]値から出力輻輳ポートを見つけます。](#)

[ステップ7: 前の検出結果に基づいて、ASIC 1にある前面パネルポートとJerichoポート9にマッピングを確認します。](#)

[追加コマンド](#)

[追加のラボテスト :](#)

[ステップ1: 複数の出力輻輳インターフェイスによる入力廃棄](#)

[ステップ2: SPANによる入力廃棄](#)

[ステップ3: トラフィックヘアピンによる入力廃棄](#)

[ステップ4: 宛先IPが不明のパケットを送信する。](#)

[ステップ5: アクセス/トランクポートがSTPフォワーディングステートに移行する間の入力廃棄](#)

[ステップ6: Eth1/9超過ラインレートによる入力廃棄](#)

概要

このドキュメントでは、Cisco Nexus 9500-R EoRおよびNexus 3000-R ToRの入力廃棄の原因と解決策について説明します。入力廃棄は、輻輳のために入力キューで廃棄されたパケットの数を示します。この数には、テールドロップとWeighted Random Early Detection (WRED ; 重み付けランダム早期検出) によって引き起こされるドロップが含まれます。

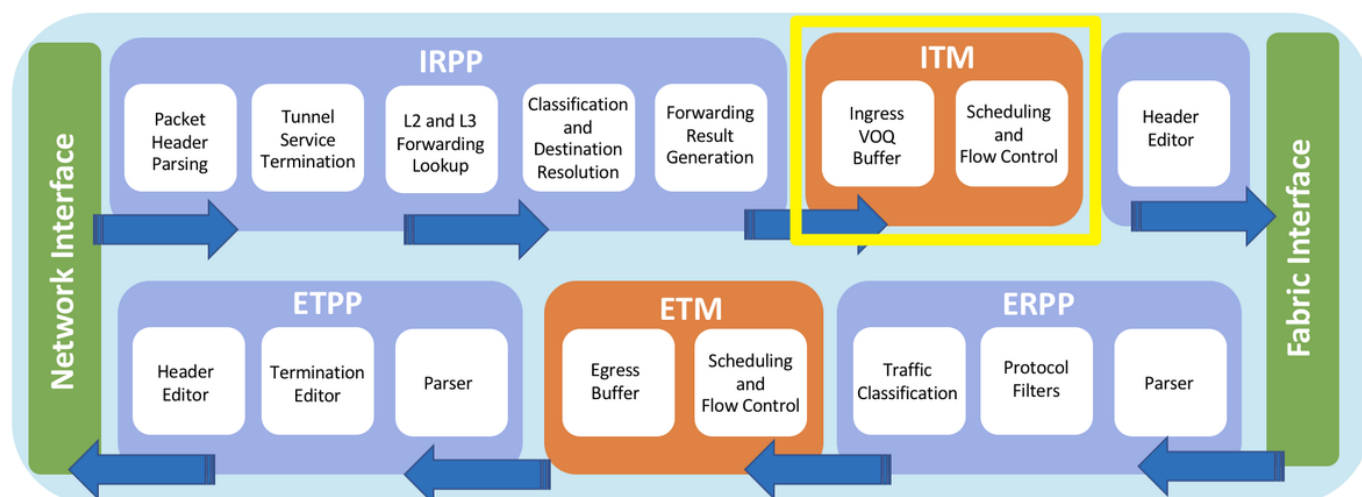
ランダム/散発性/履歴 (つまり発生していない) ドロップが発生した場合は、Cisco TACに連絡して、詳細な調査を依頼してください。このワークスルーは、入力廃棄が頻繁に増加する場合に便利です。

背景説明

Rシリーズは入力VOQアーキテクチャを使用します。VOQアーキテクチャは、入力バッファの出力キューを仮想キューでエミュレートします。各出力ポートには、ユニキャストトラフィック用の8つのキューと、マルチキャストトラフィック用の8つのキューがあります。トラフィックは、パケットのClass-of-Service(CoS)またはDifferentiated Services Code Point(DSCP)値に基づいてトラフィッククラスに分類され、そのトラフィッククラスに対応する仮想キューにキューイングされます。

Rシリーズは、分散クレジットメカニズムを使用して、ファブリック上でトラフィックを転送します。パケットがVOQから送出されるようにスケジュールされる前に、入力バッファスケジューラは、出力バッファの特定のポートとプライオリティのクレジットを要求します。クレジットは、宛先ポートとプライオリティの入力クレジットスケジューラから要求されます。バッファスペースが使用可能な場合、出力スケジューラはアクセスを許可し、クレジット認可を入力バッファスケジューラに送信します。出力バッファに使用可能なバッファスペースがない場合、出力スケジューラはクレジットを付与せず、次のクレジットが使用可能になるまでトラフィックがVOQにバッファリングされます。

次に、Rプラットフォームの packets 転送パイプラインを示します。この記事では、Ingress Traffic Managerコンポーネントに焦点を当てます。アーキテクチャの詳細については、このリンクを参照してください [こちら](#)



Ingress Traffic Manager(ITM)

入力トラフィックマネージャ(ITM)は、入力パイプラインのブロックです。VOQへのトラフィックのキューイング、ファブリック上での送信のためのトラフィックのスケジュール、クレジットの管理に関連する手順を実行します。

入力VOQバッファ

入力VOQバッファブロックは、オンチップバッファとオフチップパケットバッファの両方を管理します。どちらのバッファもVOQアーキテクチャを使用し、トラフィックはIRPP(Ingress Receiver Packet Processor)からの情報に基づいてキューイングされます。ユニキャストおよびマルチキャストトラフィックでは、合計96,000のVOQを使用できます。

スケジュールおよびフロー制御

パケットが入力パイプラインから送信される前に、パケットがファブリック経由で転送されるようにスケジュールする必要があります。入力スケジューラは、出力トラフィックマネージャブロ

ックにある出カスケジューラにクレジット要求を送信します。入力トラフィックマネージャは、クレジットを受信すると、入力送信パケットプロセッサへのトラフィックの送信を開始します。出力バッファがいっぱいになると、トラフィックは出力ポートとトラフィッククラスで表される専用キューにバッファリングされます。

よくある原因

一般的に、さまざまなNexusハードウェアで次のような理由で入力廃棄が発生する可能性があります

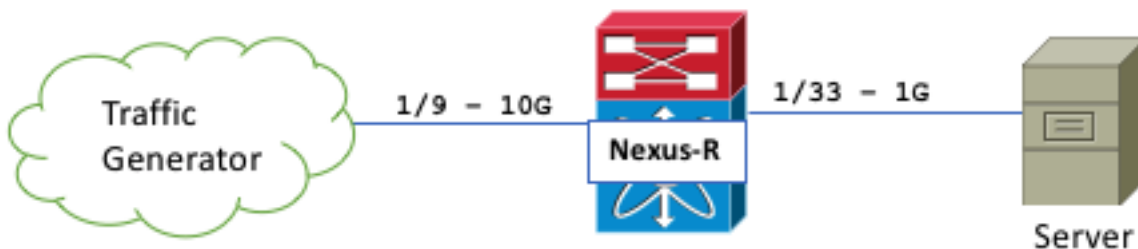
- トラフィックフローが出カインターフェイス (10Gの入力と1Gの出力など)
- オーバーサブスクライブSPAN宛先ポート：特定のハードウェアタイプに適用されます。

該当ハードウェア

PID
N9K-X9636C-R
N9K-X9636Q-R
N9K-X9636C-RX
N9K-X96136YC-R
N3K-C36180YC-R
N3K-C3636C-R

入力廃棄のトラブルシューティング

一般的なシナリオ – 10Gから1Gへのトラフィックフロー – 継続的なドロップ：



この記事では、「input discards」のカウンタと同じ値を参照するHW内部カウンタの値は、テスト中にエラーが増加していたことと同じように変化し、関連するコマンドをライブで取得する必要があります。

ステップ1：入力廃棄インターフェイスに影響するキューを確認します。

この手順は後で役立ちます。

この例では、キュー7、デフォルトキューです。入力には合計8つのキューがあります。

```
Nexus-R# show system internal qos queuing stats interface e1/9 | beg "QUEUE: 7"
QUEUE: 7
=====
ingress dropped packets: 113503981
ingress dropped bytes: 113503981000
enqueued packet count: 74115825
enqueued byte count: 74115825000
```

ステップ2 : 診断に使用されるBroadcomのカウンタのグラフィック表示を確認します。

```
Nexus-R# bcm-shell mod 1 "diag counters g"
```

```

/\
|
| J E R I C H O   N E T W O R K   I N T E
R F A C E
|
| \
|
+-----+
|
| NBI
|
| RX_TOTAL_BYTE_COUNTER = 10,616,663,796
TX_TOTAL_BYTE_COUNTER = 41,136
| RX_TOTAL_PKT_COUNTER = 10,659,301
TX_TOTAL_PKT_COUNTER = 606
| RX_TOTAL_DROPPED_EOPS = 0
|
+-----+
| IRE
|
| EPNI
| CPU_PACKET_COUNTER = 606
|
| NIF_PACKET_COUNTER = 10,659,302
EPE_BYTES_COUNTER = 41,136
| OAMP_PACKET_COUNTER = 0
EPE_PKT_COUNTER = 606
| OLP_PACKET_COUNTER = 0
EPE_DSCRD_PKT_CNT = 0
| RCY_PACKET_COUNTER = 0
|
| IRE_FDT_INTRFACE_CNT = 0
|
+-----+
| IDR
|
| EGQ
|
| MMU_IDR_PACKET_COUNTER = 10,659,302
FQP_PACKET_COUNTER = 606
| IDR_OCB_INTERFACE_COUNTER = 0
PQP_UNICAST_PKT_CNT = 606
|
| PQP_DSCRD_UC_PKT_CNT = 0
|
| PQP_UC_BYTES_CNT = 48,408
+-----+

```

| | | | |
|-------------------------------------|--------------------|--|--------------------------|
| PQP_MC_PKT_CNT | = 0 | | |
| | IQM | | |
| PQP_DSCRD_MC_PKT_CNT | = 0 | | |
| | | | |
| PQP_MC_BYTES_CNT | = 0 | | |
| ENQUEUE_PKT_CNT | = 1,403,078 | | |
| EHP_UNICAST_PKT_CNT | = 606 | | |
| DEQUEUE_PKT_CNT | = 1,403,078 | | |
| EHP_MC_HIGH_PKT_CNT | = 0 | | |
| DELETED_PKT_CNT | = 0 | | |
| EHP_MC_LOW_PKT_CNT | = 0 | | |
| ENQ_DISCARDED_PACKET_COUNTER | = 9,256,829 | | |
| DELETED_PKT_CNT | = 0 | | |
| Rejects: PORT_AND_PG_STATUS | | | |
| | | | |
| | | | |
| RQP_PKT_CNT | = 606 | | |
| | | | |
| RQP_DSCRD_PKT_CNT | = 0 | | |
| | | | |
| PRP_PKT_DSCRD_TDM_CNT | = 0 | | |
| | | | |
| PRP_SOP_DSCRD_UC_CNT | = 0 | | |
| | | | |
| PRP_SOP_DSCRD_MC_CNT | = 0 | | |
| | | | |
| PRP_SOP_DSCRD_TDM_CNT | = 0 | | |
| | | | |
| EHP_MC_HIGH_DSCRD_CNT | = 0 | | |
| | | | |
| EHP_MC_LOW_DSCRD_CNT | = 0 | | |
| | | | |
| ERPP_LAG_PRUNING_DSCRD_CNT | = 0 | | |
| | | | |
| ERPP_PMF_DISCARDS_CNT | = 0 | | |
| | | | |
| ERPP_VLAN_MBR_DSCRD_CNT | = 0 | | |
| +-----+-----+-----+ | | | |
| -----+-----+-----+ | | | |
| | | | |
| FDA | | | |
| | | | |
| CELLS_IN_CNT_P1 | = 0 | | CELLS_OUT_CNT_P1 = 0 |
| | | | |
| CELLS_IN_CNT_P2 | = 0 | | CELLS_OUT_CNT_P2 = 0 |
| | | | |
| +-----+-----+-----+ | | | |
| CELLS_IN_CNT_P3 | = 0 | | CELLS_OUT_CNT_P3 = 0 |
| | | | IPT |
| CELLS_IN_TDM_CNT | = 0 | | CELLS_OUT_TDM_CNT = 0 |
| | | | |
| CELLS_IN_MESHMC_CNT | = 0 | | CELLS_OUT_MESHMC_CNT = 0 |
| EGQ_PKT_CNT | | | = 606 |
| CELLS_IN_IPT_CNT | = 606 | | CELLS_OUT_IPT_CNT = 606 |
| ENQ_PKT_CNT | | | = 1,403,084 |
| EGQ_DROP_CNT | = 0 | | |
| FDT_PKT_CNT | | | = 1,402,472 |
| EGQ_MESHMC_DROP_CNT | = 0 | | |
| CRC_ERROR_CNT | | | = 0 |
| EGQ_TDM_OVF_DROP_CNT | = 0 | | |
| CFG_EVENT_CNT | | | = 606 * |
| | | | |
| CFG_BYTE_CNT | | | = 48,408 |
| | | | |
| +-----+-----+-----+ | | | |

```

-----+-----+-----+
|                                     FDT                                     |
FDR |                                     |
| IPT_DESC_CELL_COUNTER              = 5,609,892                          |
P1_CELL_IN_CNT                       = 0                                |
| IRE_DESC_CELL_COUNTER              = 0                                |
P2_CELL_IN_CNT                       = 0                                |
|                                     |
P3_CELL_IN_CNT                       = 0                                |
| TRANSMITTED_DATA_CELLS_COUNTER      = 5,609,892                          |
CELL_IN_CNT_TOTAL                    = 0                                |
-----+-----+-----+

```

```

/|\
|                                     |
|                                     J E R I C H O   F A B R I C   I N T E R
F A C E |                                     |
|                                     \|\

```

0より大きいQUEUE_DELETED_PACKET_COUNTERは、パケットがIQM(Ingress Queueing Manager)によって削除されたことを示します。これは、アクティブなキューがクレジットを受け取っていないため、スケジューリング方式の設定ミスを示唆します。これはbcm-shell mod X "getReg IQM_QUEUE_DELETED_PACKET_COUNTER"で確認できます

ENQ_DISCARDED_PACKET_COUNTERは、エンキュー前にパケットが廃棄されたことを意味します。このカウンタはBCMでも設定されています (読み取り時にコマンドがクリアされます)。

```

Nexus-R# bcm-shell mod 1 "g iqm_reject_status_bmp" | i i PG|IQM0|IQM1
IQM_REJECT_STATUS_BMP.IQM0[0x1a7]=0x20000000: <VSQF_WRED_STATUS=0,
QNUM_OVF_STATUS=0,PORT_AND_PG_STATUS=1,OCUPIED_BD_STATUS=0,
IQM_REJECT_STATUS_BMP.IQM1[0x1a7]=0: <VSQF_WRED_STATUS=0,VSQF_MX_SZ_STATUS=0,
PORT_AND_PG_STATUS=0,OCUPIED_BD_STATUS=0,MULTICAST_ERROR_STATUS=0,

```

show hardware internal errors module X(コマンドが読み取り時にクリアされる)を使用すると、常に次のことがわかります。

```

Nexus-R# show hardware internal errors module 1

```

```

slot 1
=====
|-----|
| Device:Forwarding ASIC Role:MAC Mod: 1 |
| Device Statistics Category :: ERROR    |
|-----|
Instance:0

IQM
-----
ENQ_DISCARDED_PACKET_COUNTER = 8,233,862
Rejects: PORT_AND_PG_STATUS

Instance:1

```

ステップ3：入力廃棄が発生している前面パネルポートが属するASICとJericoポートを見つけます。

```
Nexus-R# show interface hardware-mappings | i i Eth1/9|--|Name|Eth1/33
HName - Hardware port name. None means N/A
-----
Name          Ifindex  Smod Unit HPort HName FPort NPort VPort SrcId
-----
Eth1/9        1a001000 0    0    9    xe9   255   8    -1    0    << ASIC 0, Jericho Port 9
Eth1/33       1a004000 2    1    9    xe9   32    -1    0    << ASIC 1, Jericho Port 9
```

この例では、Eth1/33を表示しています。実際のネットワークでは、輻輳した出力ポートを認識していません。

ステップ4：入力ポートのVOQおよびVOQコネクタについて理解します。

```
Nexus-R# attach module 1
module-1# show hardware internal jer-usd info voq asic 0 port 9

+-----+
|Unit|JerPort| Voq| VoqConn| SE    |   HR   |CreditBal|
+-----+
| 0  |    9  | 104|    176| 82213 |    72   |    16a . |
+-----+
```

このコマンドは、特定のポートの入力VoQのフローの詳細を表示します。また、VoQの現在の信用残高も示します。

ポートのVOQは次のように生成されます。

LCは0ベース：モジュール1は0、モジュール2は1など
LCごとに256のシステムポートIDがあります

$$ID = (LC * \text{システムポートID}) + \text{FP番号}$$
$$\text{Eth1/9} = (0 * 256) + 9 = 9$$
$$\text{VOQ ID} = 32 + (\text{システムポートID} * 8)$$
$$\text{Eth1/9} = 32 + (9 * 8) = 104$$

したがって、Eth1/9のVOQは、以前に収集した出力と一致する104になります

```
module-1# show hardware internal jer-usd ingress-vsqs buffer-occupancy front-port 9
```

```
+-----+
|                                     VSQF BUFFER OCCUPANCY                                     |
+-----+
|                                     Front port 9                                     |
|max global shared                    | 157286 |
|max ocb buffer occupancy              | 0     |
+-----+
|                                     COSQ 0                                         |
+-----+
|rate class                            | 4     |
|granted buffers per port              | 3280  |
|shared buffers occupied               | 127792| <<<<
|granted buffers occupied              | 3280  |
|shared buffer max occupancy          | 127792| <<<<
+-----+
```

ステップ5:BCMの観点から確認します。キューは特に空ではありません。つまり、輻輳です。

```
Nexus-R# bcm-shell mod 1 "diag cosq non_empty_queue"
```

```
Core 0:
Ingress VOQs Sizes (format: [queue_id(queue_size)]):
[303(191338496B)] << the Queue ID belongs to your Egress CONGESTED port!

Core 1:
<empty>
```

ステップ6：空でない[Queue]値から出力輻輳ポートを見つけます。

キューが303の場合、これらのキューは実際には範囲であるため、 $303 + 7$ または $303 - 7$ になります。問題は、296-303または303-310の範囲で一致するVOQがあるポートはどれですか。

Eth1/9のキュー7が輻輳していることがわかっているため、実際には303が範囲の中で最も高いため、296 ~ 303の範囲は十分に知識のある推測です。

```
module-1# show hardware internal jer-usd info voq asic 1
```

```
+-----+
|Unit|JerPort| Voq| VoqConn| SE| HR|CreditBal|
+-----+
| 1| 1| 232| 56| 81957| 8| 3ffff|
| 1| 2| 240| 72| 81989| 16| 3ffff|
| 1| 3| 248| 88| 82021| 24| 3ffff|
| 1| 4| 256| 104| 82053| 32| 3ffff|
| 1| 5| 264| 120| 82085| 40| 3ffff|
| 1| 6| 272| 136| 82117| 48| 3ffff|
| 1| 7| 280| 152| 82149| 56| 3ffff|
| 1| 8| 288| 168| 82181| 64| 3ffff|
| 1| 9| 296| 184| 82213| 72| 3a5| <<< 296 +7 would give us 303
| 1| 10| 304| 200| 82245| 80| 3ffff| << It cannot be this one as 303 is not included
| 1| 11| 312| 216| 82277| 88| 3ffff|
<snip>
```

ASIC 0についても同じ内容を表示します。簡略化のためにはここでは示しません。Voq列の下に、対象の範囲がそのASICに含まれていないことがわかります

上記の出力には、いくつかの点があります。

- 出力輻輳ポートはASIC 1にあります。
- 出力輻輳ポートのVOQは296で、303はそのポートのキュー7に相当します。
- [Credit Balance]列に注目してください。このインターフェイスには付与するクレジットが非常に少ないため、入力Eth1/9がバッファリングを開始します。

ステップ7：前の検出結果に基づいて、ASIC 1にある前面パネルポートとJerichoポート9にマッピングを確認します。

```
Nexus-R# show interface hardware-mappings | i i Eth1/9|--|Name|Eth1/33
      HName - Hardware port name. None means N/A
-----
Name      Ifindex  Smod Unit HPort HName FPort NPort VPort SrcId
-----
Eth1/9    1a001000 0    0    9    xe9   255  8    -1    0    << ASIC 0, Jericho Port 9
Eth1/33   1a004000 2    1    9    xe9   32   -1    0     << ASIC 1, Jericho Port 9
```

この時点で、出力の輻輳ポートが見つかりました。ネットワークに誤ってバーストしているかどうかや、SPANを設定し、宛先ポートが1Gで、1つ以上の10Gインターフェイスを発信しているか、ボトルネックや設計の問題かどうかを確認します。

追加コマンド

これらは、より高度です。通常のシナリオでは、Egress Congestedポートを検出する必要はありません。

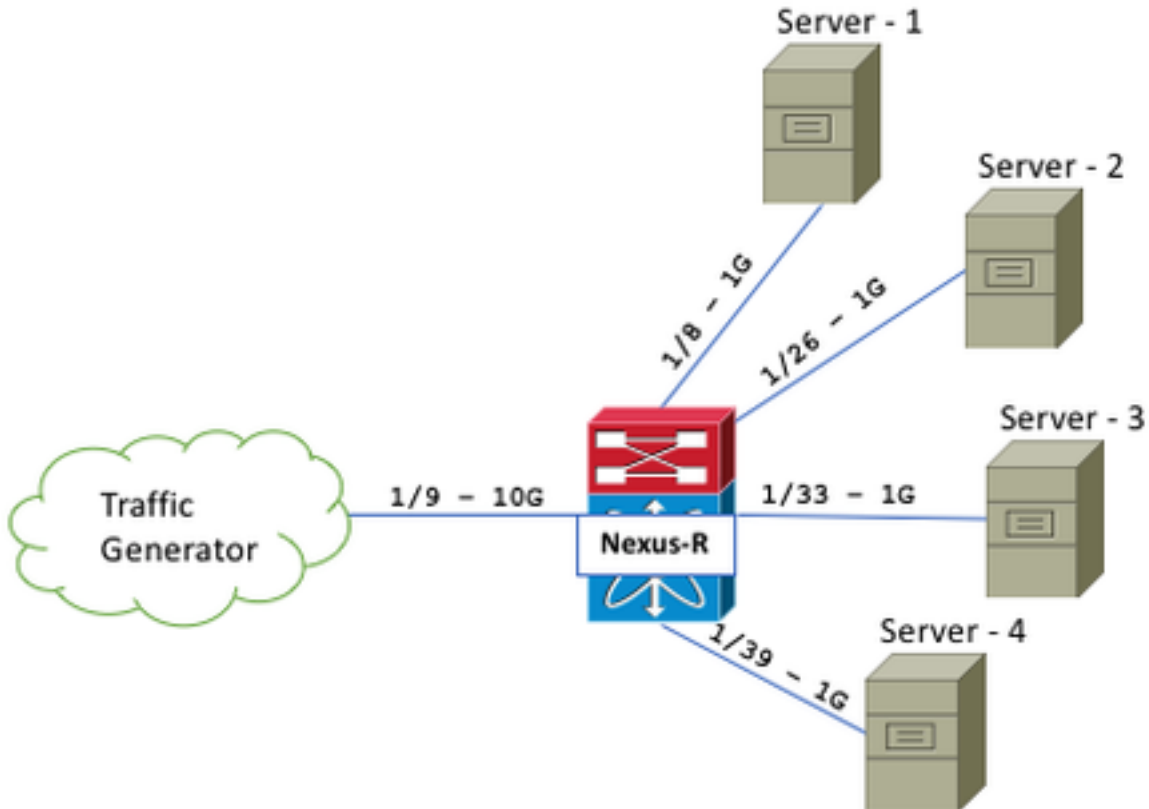
```
attach module X
show hardware internal jer-usd tm_debug asic <slot> module <module>
show hardware internal jer-usd info voq [ asic <instance> ] [ port <port> ] [ ]
show hardware internal jer-usd info non-empty voq asic [ <instance> ] [ ]
show hardware internal jer-usd info voq-profile { QueueThreshold drop_p <dp> | OCBThreshold } [
asic <instance> ] [ port<port> ] [ ]
show hardware internal jer-usd info voq-connector front-port <port> [ ]
show hardware internal jer-usd stats vsq { front-port <port> | inband asic <slot> | recycle-port
<port> asic <slot> }
show hardware internal jer-usd ingress-vsqueue buffer-occupancy front-port <port>
show hardware internal jer-usd info IQM { counter | rate } asic <instance> dst-port <port> [
interval <int> ] [ ]
show hardware internal jer-usd info SCH { counter | rate } asic <instance> dst-port <port> [
interval <int> ] [ ]
```

```
bcm-shell mod X
diag cosq print_flow_and_up dest_id=<flow_id>
diag cosq voq id=<voqid> detailed=1
diag cosq qpair e2e ps=<id>
cosq conn ing
cosq conn egr
dump IPS_CR_BAL_TABLE <voqID>
getReg IQM_QUEUE_MAXIMUM_OCCUPANCY_QUEUE_SIZE
```

追加のラボテスト：

ステップ1：複数の出力輻輳インターフェイスによる入力廃棄

トラフィックジェネレータが各サーバに2Gのトラフィックを送信する次のトポロジを考えます。



どのキューが空でないかを迅速にチェックします。4つがあることに注意してください。

```
Nexus-R# bcm-shell mod 1 "diag cosq non_empty_queue"
Core 0:
Ingress VOQs Sizes (format: [queue_id(queue_size)]):
[103(29475840B)]      [247(29379584B)]      [303(56452096B)]      [351(76020736B)]
```

これらのキューが属するインターフェイスを判別します。最初にASIC 0をチェックします (1つのインターフェイスでのみ示されます)。

```
module-1# show hardware internal jer-usd info voq asic 0
+-----+
|Unit|JerPort| Voq| VoqConn| SE| HR|CreditBal|
+-----+
| 0| 1| 40| 48| 81957| 8| 3ffff|
| 0| 2| 48| 64| 81989| 16| 3ffff|
| 0| 3| 56| 80| 82021| 24| 3ffff|
...
| 0| 8| 96| 160| 82181| 64| 7b| << 96 + 7 = 103, this is port Eth1/8
<snip>

`show interface hardware-mappings`
+-----+
Name      Ifindex  Smod  Unit  HPort  HName  NPort  VPort  SrcId
+-----+
Eth1/8    1a000e00 0     0     8     xe8    7     -1     0
```

他の3つのキュー値についても、同じプロセスを繰り返します。247、303、351。

ステップ2:SPANによる入力廃棄

Eth1/33をSPAN宛先ポートとして設定し、Eth1/9をRX方向のSPAN送信元ポートとして設定します

```
Nexus-R# show run mon
monitor session 1
description SPAN TEST INPUT DISCARDS
source interface Ethernet1/9 rx
destination interface Ethernet1/33
no shut

Nexus-R# show int e1/9 | i i input.disc
0 input with dribble 9314306 input discard
```

ステップ3 : トラフィックヘアピンによる入力廃棄

Eth1/9が10.10.10.1/24にあるSRC 10.10.10.10およびDEST 192.168.10.10を使用してパケットを送信する – これはInput Discard (INPUT ; 入力廃棄) にはなりません。ただし、次のカウンタが表示されます。

```
Nexus-R# bcm-shell mod 1 "diag counters g"
```

```
| /|\
| J E R I C H O N E T W O R K I N T E R F A C E |
\|/ |
```

```
-----+-----+-----+
| _PACKET_COUNTER = 0 | DELETED_PKT_CNT = 12,027,201 |
| | Discards: INVALID_OTM SRC_EQUAL_DEST
```

ステップ4 : 宛先IPが不明のパケットを送信する。

SRC 10.10.10.10およびDEST 192.168.10.10を使用してパケットを送信します。ここで、Eth1/9は10.10.10.1/24に存在し、Eth1/33は172.16.0.1/30サブネットのL3ポートです。ドロップカウンタは発生せず、宛先が不明な場合でも入力廃棄はありません。

ステップ5 : アクセス/トランクポートがSTPフォワーディングステートに移行する間に入力廃棄

Eth1/9が単なるワイドトランク (またはアクセスポート) である場合にパケットを送信する : ポートがSTPフォワーディングステートに移行する間、これはInput Discard (INPUT ; 入力廃棄) として登録されます。

```
Nexus-R(config)# int e1/9
Nexus-R(config-if)# switchport mode trunk
Nexus-R# bcm-shell mod 1 "diag counters g" | i i --|IQM|ENQ_DISCARD|Rejects
```

```

-----+-----
+-----+-----+-----
-----+-----
+-----+-----+-----
PQP_MC_PKT_CNT = 1,678,949 |
| IQM |
PQP_DSCRD_MC_PKT_CNT = 11,369,033 |
| ENQ_DISCARDED_PACKET_COUNTER = 1,289,182 |
DELETED_PKT_CNT = 11,369,081 |
| Rejects: QUEUE_NOT_VALID_STATUS |
Discards: SRC_EQUAL_DEST |
-----+-----
-----+-----

```

Nexus-R# **show span int e1/9**

```
Vlan Role Sts Cost Prio.Nbr Type
```

```
-----
```

```
VLAN0001 Desg BLK 2 128.9 P2p
```

```
VLAN0010 Desg BLK 2 128.9 P2p
```

```
<snip>
```

QUEUE_NOT_VALID_STATUSは、パケットプロセッサ(PP)によるドロップの決定またはパケットプロセッサ(PP)ブロックから受信した無効な宛先によるドロップです。

ステップ6:Eth1/9超過ラインレートによる入力廃棄

Eth1/9に10G+を送信すると、最初の場所でEth1/9を上限としているため、異なるタイプのドロップが発生します。それでもInput Discardとしてカウントされます。

bcm-shell.0> **diag counters g**

```

/|\
|
| J E R I C H O N E T W O R K I N T E
R F A C E |
| \|\
|
-----+-----
-----+-----
| NBI
|
| RX_TOTAL_BYTE_COUNTER = 53,913,106,009 |
TX_TOTAL_BYTE_COUNTER = 1,164,231 |
| RX_TOTAL_PKT_COUNTER = 54,145,395 |
TX_TOTAL_PKT_COUNTER = 17,029 |
| RX_TOTAL_DROPPED_EOPS = 0 |
|
-----+-----
-----+-----
| IRE
EPNI |
| CPU_PACKET_COUNTER = 17,010 |
|
| NIF_PACKET_COUNTER = 54,145,476 |
EPE_BYTES_COUNTER = 5,721,307 |
| OAMP_PACKET_COUNTER = 0 |
EPE_PKT_COUNTER = 50,703 |
| OLP_PACKET_COUNTER = 0 |
EPE_DSCRD_PKT_CNT = 0 |
| RCY_PACKET_COUNTER = 16,837 |

```

```

|
|   IRE_FDT_INTRFACE_CNT                = 0
|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|
|                                     IDR
EGQ |
|
|   MMU_IDR_PACKET_COUNTER              = 54,128,577
FQP_PACKET_COUNTER                     = 50,703
|   IDR_OCB_INTERFACE_COUNTER           = 0
PQP_UNICAST_PKT_CNT                    = 50,683
|
PQP_DSCRD_UC_PKT_CNT                   = 0
|
PQP_UC_BYTES_CNT                        = 5,216,716
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
PQP_MC_PKT_CNT                          = 20
|                                     IQM
PQP_DSCRD_MC_PKT_CNT                    = 20
|
PQP_MC_BYTES_CNT                        = 2,079
|   ENQUEUE_PKT_CNT                     = 5,463,323
EHP_UNICAST_PKT_CNT                     = 50,683
|   DEQUEUE_PKT_CNT                     = 5,594,400
EHP_MC_HIGH_PKT_CNT                     = 20
|   DELETED_PKT_CNT                      = 0
EHP_MC_LOW_PKT_CNT                      = 0
|   ENQ_DISCARDED_PACKET_COUNTER       = 48,716,055
DELETED_PKT_CNT                          = 40
|   Rejects: VOQ_MX_QSZ_STATUS
|
<snip>

```