

Nexus 9500-R, Nexus 3000-R: Problemen oplossen bij invoerkaarten

Inhoud

[Inleiding](#)

[Achtergrondinformatie](#)

[IP-verkeersbeheer \(ITM\)](#)

[Ingress VOQ-buffer](#)

[Schedule en Flow Control](#)

[Vaak voorkomende oorzaken](#)

[Toepasselijke hardware](#)

[Problemen oplossen bij invoerkaarten](#)

[Common Scenario - 10G tot 1G Traffic Flow - constante druppels:](#)

[Stap 1. Controleer welke wachtrij op uw interface voor invoerkaarten is ingeschakeld.](#)

[Stap 2. Controleer de grafische weergave van Broadcom van tellers die voor diagnostiek worden gebruikt:](#)

[Stap 3. Zoek naar welke ASIC en welke Jericho-poort in de poort van uw Frontpaneel-poort de invoerkaarten behoren:](#)

[Stap 4. Begrijp wat de VOQ- en VOQ-connector van uw Inbelpoort heeft.](#)

[Stap 5. Controleer vanuit het perspectief van BCM, welke wachtrij specifiek niet-leeg is; bijv. verstopt.](#)

[Stap 6. Vind uw uitgaande geconvergeerde poort op de waarde van de niet-lege wachtrij:](#)

[Stap 7. Controleer welke poort op het voorpaneel in ASIC 1 en kaarten naar Jericho poort 9 is gebaseerd op uw vorige zoekopdracht.](#)

[Aanvullende opdrachten](#)

[Aanvullende laboratoriumtests:](#)

[Stap 1. Invoerkaarten met meerdere uitgaande interfaces.](#)

[Stap 2. Teruggooi van ingangen vanwege SPAN.](#)

[Stap 3. Teruggooi van ingangen door verkeershelf.](#)

[Stap 4. Verzend pakketten met een niet bekende bestemming van IP.](#)

[Stap 5. Invoerkaarten worden verwijderd terwijl een Access/Trunk-poort wordt overgeschakeld op STP-doorsturen](#)

[Stap 6. Invoerterruggooi door Ethernet1/9 te overschrijden lijnsnelheid.](#)

Inleiding

Dit document beschrijft de oorzaken van en de oplossingen voor invoerweggooien voor Cisco Nexus 9500-R EoR en Nexus 3000-R ToR. Een invoerterruggooi geeft het aantal pakketten aan dat in de invoerwachtrij is gevallen vanwege congestie. Dit getal bevat ook druppels die worden veroorzaakt door zogenoemde 'tail druppels' en Weighted Random Early Detection (WRED).

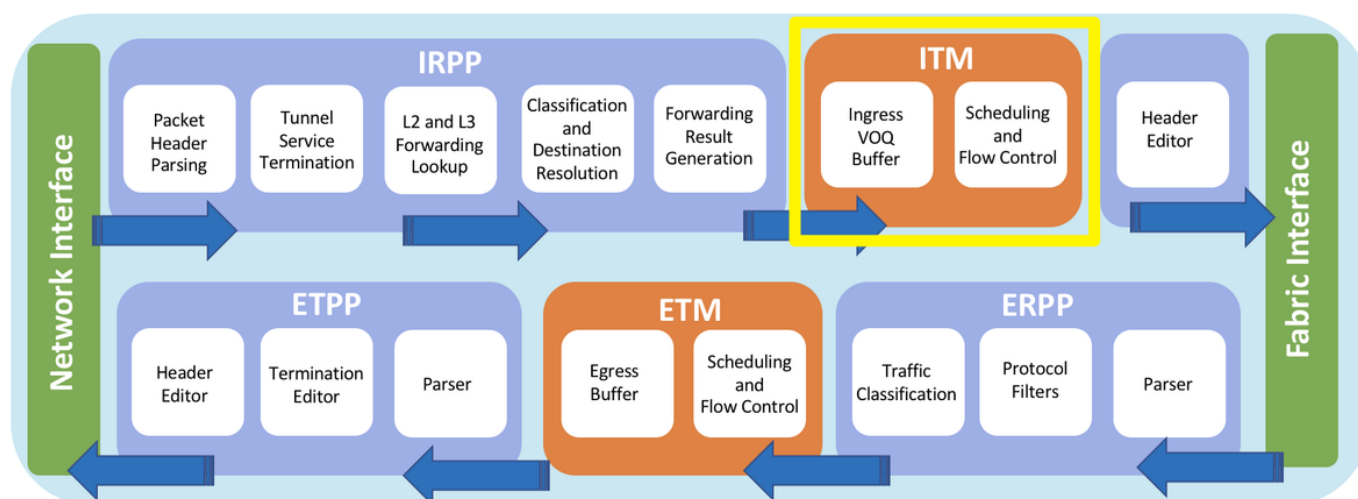
Als u lijdt aan willekeurige/sporadische/historische (d.w.z. niet langer voorkomende) vallen, kunt u contact opnemen met Cisco TAC voor verder onderzoek. Deze wandeling is handig wanneer invoerkaarten regelmatig worden verhoogd.

Achtergrondinformatie

De R-Series maakt gebruik van de inkomende VOQ-architectuur. VOQ-architectuur emuleert noodrijen in de ingress buffer met virtuele wachtrijen. Elke poort heeft acht rijen voor eenastverkeer en acht rijen voor multicast verkeer. Het verkeer kan in verkeersklassen worden geclassificeerd op basis van de waarde van de Klasse-of-Service (CoS) of het Gedifferentieerde Punt van de Servicescode (DSCP) in de pakketten en dan in de overeenkomstige virtuele rij voor die verkeersklasse in de wachtrij geplaatst worden.

De R-Series gebruikt een gedistribueerd kredietmechanisme om verkeer over de stof te verplaatsen. Voordat een pakket gepland is om de VOQ te verlaten, vraagt de server van de ingress buffer een krediet voor de specifieke poort en prioriteit in de egress buffer. Het krediet wordt gevraagd van een gevestigde kredietplanner voor de doelhaven en prioriteit. Als bufferruimte beschikbaar is, geeft de server toegang en stuurt de kredietsubsidie naar de server van de ingangsbuffer. Als er geen bufferruimte in de egress buffer beschikbaar is, kent de uitsparing geen krediet toe en wordt het verkeer in de VOQ gebufferd totdat het volgende krediet beschikbaar is.

Hieronder zie je de Packet Forwarding Pipeline voor het -R platform. In dit artikel richt u zich op de component **Ingress Traffic Manager**. Meer informatie over de architectuur op deze [link](#)



IP-verkeersbeheer (ITM)

De indringer (ITM) is een blok in de indringleiding. Het voert stappen uit verbonden met rijverkeer in VOQ, verkeer voor transmissie over de stof te plannen en kredieten te beheren.

Ingress VOQ-buffer

Het ingress VOQ-bufferblok beheert zowel de op-chip-buffer als de off-chip pakketbuffer. Beide buffers gebruiken VOQ-architectuur en verkeer wordt in de wachtrij geplaatst op basis van de informatie van de IRPP (Ingress ontvanger Packet Processor). Er zijn in totaal 96.000 VOQ's beschikbaar voor eenvoudig en multicast verkeer.

Schedule en Flow Control

Voordat een pakje uit de ingangspijplijn wordt verzonden, moet het pakje op een geplande datum

voor overdracht via het weefsel worden geplaatst. De server verstuurt een kredietaanvraag naar de server die in de gevangenis ligt. Wanneer de ingangsverkeersmanager het krediet ontvangt, begint het verkeer naar de ingress te verzenden pakkeprocessor. Als de egress buffer vol is, wordt het verkeer gebufferd in de specifieke wachtrij die wordt weergegeven door de poort en de verkeersklasse.

Vaak voorkomende oorzaken

Over het algemeen kunnen teruggooi van input om de volgende redenen gezien worden door verschillende Nexus hardware

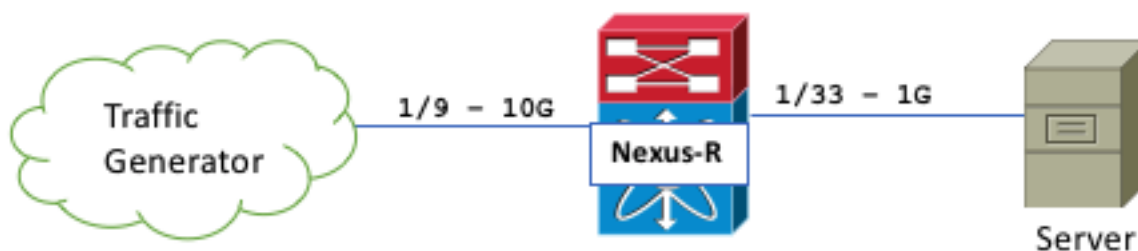
- Verkeersstromen congesteringinterfaces (zoals 10G ingangen en 1G stress)
- OverSubscriber SPAN-doelpoort - is van toepassing op specifieke hardwaretypen.

Toepasselijke hardware

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

Problemen oplossen bij invoerkaarten

Common Scenario - 10G tot 1G Traffic Flow - constante druppels:



In dit artikel zal de waarde voor de teller van "input disards" en elke HW interne teller die hetzelfde aangeeft, veranderen aangezien de fouten toenamen terwijl het testen en relevante opdrachten live moeten worden gegrepen.

Stap 1. Controleer welke wachtrij op uw interface voor invoerkaarten is ingeschakeld.

Deze stap is later handig.

In ons geval is het Wachtrij 7, de standaardrij - er zijn 8 rijen in totaal bij ingangen:

```
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
```

Stap 2. Controleer de grafische weergave van Broadcom van tellers die voor diagnostiek worden gebruikt:

```
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

```



```

+-----+-----+-----+
|                                         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                               \|\
|

```

Een QUEUE_DELETED_PACKET_COUNTER die groter is dan nul zou betekenen dat pakketten na de wachtrij werden GEDELETEERD door de IQM (Ingress Queueing Manager). Dit zou het gevolg zijn van een actieve wachtrij die geen kredieten ontvangt die zouden wijzen op een verkeerde configuratie van de planningsregeling. Je zou dit controleren via **bcm-shell mod X "getReg IQM_QUEUE_DELETED_PACKET_COUNTER"**

ENQ_DISCARDED_PACKET_COUNTER betekent dat pakketten werden weggegooid VOORDAT u de informatie in de wachtrij bracht. U kunt deze teller ook in BCM zien instellen (de opdracht wordt gewist bij lezen):

```

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,OCCUPIED_BD_STATUS=0,
IQM_REJECT_STATUS_BMP.IQM1[0x1a7]=0: <VSQF_WRED_STATUS=0,VSQF_MX_SZ_STATUS=0,
PORT_AND_PG_STATUS=0,OCCUPIED_BD_STATUS=0,MULTICAST_ERROR_STATUS=0,

```

U kunt deze altijd snel opmerken met **show hardware interne foutmodule X** (opdracht wordt bij lezen gewist):

```
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
```

Stap 3. Zoek naar welke ASIC en welke Jericho-poort in de poort van uw Frontpaneel-poort de invoerkaarten behoren:

```
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
```

Het tonen van Eth1/33 voor dit voorbeeld. In een echt netwerk, zal je de gecongeneerde poort nog niet kennen.

Stap 4. Begrijp wat de VOQ- en VOQ-connector van uw Inbelpoort heeft.

```
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 .  |
+-----+
```

Deze opdracht toont ons details voor de flow voor VoQ voor een specifieke poort. Daarnaast toont het ons het huidige kredietsaldo van het VoQ.

Het VOQ van de haven is op deze manier afgeleid:

LC's zijn 0 gebaseerd - module 1 is 0, module 2 is 1, enz

Er zijn 256 systeempoortadapters per LC

ID = (LC * Systeem poort-ID + FP-nummer

Eth1/9 = (0 * 256) + 9 = 9

VOQ-id = 32 + (systeempoortadapter-ID * 8)

Eth1/9 = 32 + (9 * 8) = 104

Ons VOQ for th1/9 zal daarom 104 zijn die de eerder verzamelde productie aanpassen

```
module-1# show hardware internal jer-usd ingress-vsqr 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 <<<< |

Stap 5. Controleer vanuit het perspectief van BCM, welke wachtrij specifiek niet-leeg is; bijv. verstopt.

```
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>

Stap 6. Vind uw uitgaande geconvergeerde poort op de waarde van de niet-lege wachtrij:

Als de Wachtrij 303 is, moet je dan bedenken dat deze rijen feitelijk een bereik hebben zodat het 303 + 7 of 303-7 kan zijn. De vraag is: welke haven heeft een VOQ die overeenkomt met een bereik van 296-303 of, anders, 303-310?

Het is bekend dat wachtrij 7 op nummer 1/9 verstopt is, dus 303 is feitelijk het hoogste in zijn bereik, dus het bereik 296-303 is een goed opgeleide gok.


```

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>

```

Dit is gelijk aan basis 0, niet hier voor de beknoptheid; U merkt onder de Voq-kolom dat uw bereik niet in die ASIC zit

Let op een paar dingen bij de bovenstaande uitvoer:

- Onze enorme gested haven is op ASIC 1.
- Onze enorme overbelaste haven heeft een VOQ van 296 en 303 zou gelijk zijn aan Wachtrij 7 op die haven.
- Let op de kolom Credit Balance. Er zijn maar weinig credits over op deze interface om te verstrekken, daarom begint onze etentje 9 te bufferen.

Stap 7. Controleer welke poort op het voorpaneel in ASIC 1 en kaarten naar Jericho poort 9 is gebaseerd op uw vorige zoekopdracht.

```

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    0    << ASIC 1, Jericho Port 9

```

Op dit punt, hebt u de gras gecongested port gevonden - Bepaal of er iets ten onrechte in het netwerk is gebrand, u hebt SPAN ingesteld en uw doelpoort is 1G terwijl u een of meer 10G interface aanschaft of als dit een knelpunt/ontwerp probleem is.

Aanvullende opdrachten

Deze zijn geavanceerder - niet nodig om in normale scenario's een Groot Gegste haven te vinden.

```

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-vsqs 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> ] [ ]

```

```

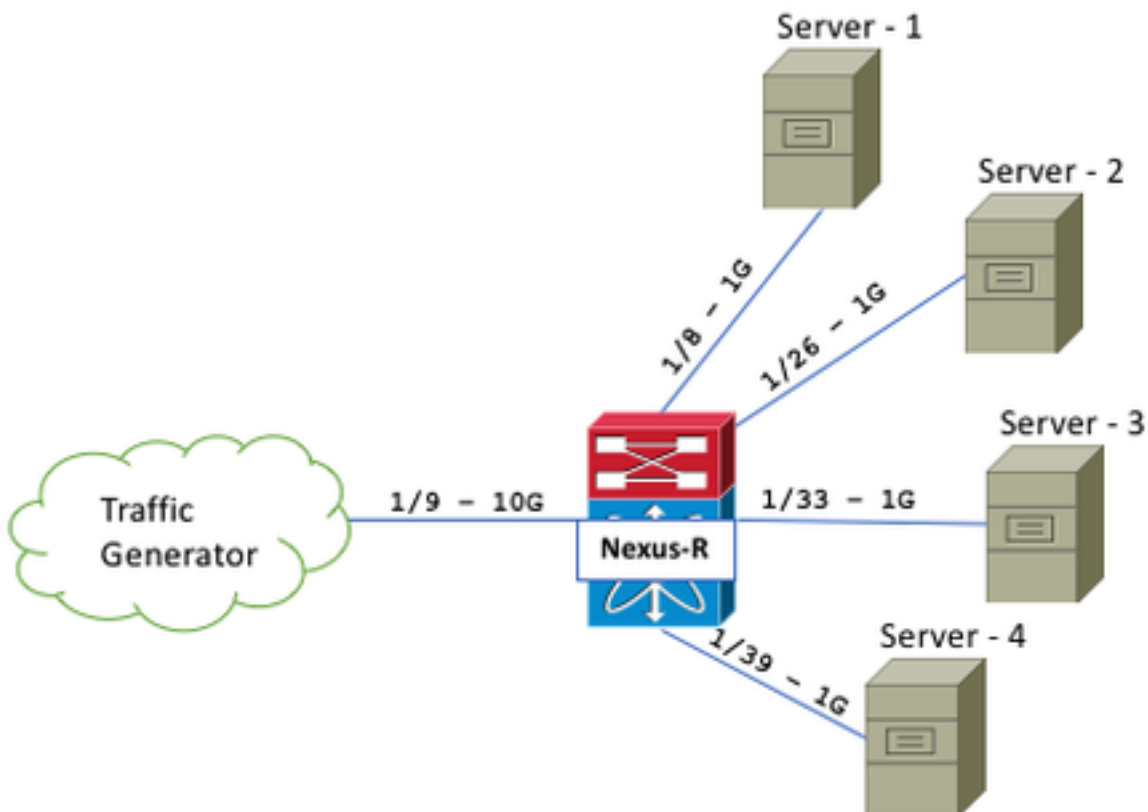
bcm-shell mod X
diag cosq print_flow_and_up dest_id=<flow_id>
diag cosq voq id=<voqid> detailed=1
diag cosq qpair e2e ps=<id>
cosq conn ing
cosq conn egr
dump IPS_CR_BAL_TABLE <voqID>
getReg IQM_QUEUE_MAXIMUM_OCCUPANCY_QUEUE_SIZE

```

Aanvullende laboratoriumtests:

Stap 1. Invoerkaarten met meerdere uitgaande interfaces.

Neem deze topologie in overweging waar de Generator van het verkeer 2 G verkeer naar elke Server stuurt:



Snel controleren welke wachtrijen niet leeg zijn - Opmerking: 4:

```

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)]

```

Bepaal welke interfaces deze wachtrijen behoren tot - Controleer ASIC 0 eerst (het demonstreert

alleen met één interface):

```
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
```

Herhaal hetzelfde proces voor de andere drie waarden van de wachtrij: 247, 303 en 351.

Stap 2. Teruggooi van ingangen vanwege SPAN.

th1/33 instellen als een SPAN-doelpoort terwijl Eth1/9 wordt ingesteld als een SPAN-bronpoort in de RX-richting

```
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
```

Stap 3. Teruggooi van ingangen door verkeersshelf.

pakketten verzenden met SRC 10.10.10.10 en DEST 192.168.10.10, waarbij Eth1/9 zich op 10.10.10.1/24 bevindt - Dit leidt niet tot een invoerkaart; u ziet deze teller echter :

```
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
+-----+
+-----+
```

Stap 4. Verzend pakketten met een niet bekende bestemming van IP.

Verzend pakketten met SRC 10.10.10.10 en DEST 192.168.10.10 waar Eth1/9 in 10.10.10.1/24 is en Eth1/33 een L3 poort in 172.16.0.1/30 subnet - Geen druppelteller, geen input teruggooi zelfs wanneer de bestemming **onbekend** is.

Stap 5. Invoerkaarten worden verwijderd terwijl een Access/Trunk-poort wordt overgeschakeld op STP-doorsturen

Verzend pakketten waar Eth1/9 slechts een brede boomstam (of toegangshaven) is - Dit is geregistreerd als een Vergiet van de Invoer terwijl de haven in een STP door middel van staat overgaat.

```
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 is een druppel vanwege de PP-beslissing (Packet Processor) van de Packet Processor- (PP) blokkering of een ongeldige bestemming die is ontvangen van de Packet Processor- (PP) blokken.

Stap 6. Invoerteruggooi door Ethernet1/9 te overschrijden lijnsnelheid.

Verzenden van 10G+ naar Eth1/9 zou resulteren in een ander type daling aangezien u op de eerste plaats Eth1/9 uitvouwt - telt nog steeds als een invoerkaart:

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

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