

Nexus 9500-R, Nexus 3000-R: Solución de problemas de descartes de entrada

Contenido

[Introducción](#)

[Antecedentes](#)

[Administrador de tráfico de entrada \(ITM\)](#)

[Búfer VOQ de ingreso](#)

[Programación y control de flujo](#)

[Causas comunes](#)

[Hardware aplicable](#)

[Solución de problemas de descartes de entrada](#)

[Situación común: flujo de tráfico de 10 G a 1 G: caídas constantes:](#)

[Paso 1. Verifique qué cola está impactada en su interfaz de descartes de entrada.](#)

[Paso 2. Verifique la representación gráfica de contadores de Broadcom utilizados para el diagnóstico:](#)

[Paso 3. Descubra qué ASIC y qué puerto Jericho pertenece al puerto del panel frontal que experimenta descartes de entrada:](#)

[Paso 4. Comprenda qué tienen el VoQ y el conector VOQ en su puerto de entrada.](#)

[Paso 5. Verifique desde la perspectiva de BCM, que cola específicamente no está vacía; es decir, congestionado.](#)

[Paso 6. Busque el puerto congestionado de salida del valor de cola no vacío:](#)

[Paso 7. Verifique qué puerto del panel frontal está en ASIC 1 y mapea a Jericho Port 9 según su hallazgo anterior.](#)

[Comandos adicionales](#)

[Pruebas adicionales de laboratorio:](#)

[Paso 1. Descartes de entrada con varias interfaces congestionadas de salida.](#)

[Paso 2. Descartes de entrada debido a SPAN.](#)

[Paso 3. Descartes de entrada debido al PIN del capó de tráfico.](#)

[Paso 4. Enviar paquetes con una IP de destino desconocida.](#)

[Paso 5. Descartes de entrada mientras un puerto de acceso/enlace troncal pasa al estado de reenvío STP](#)

[Paso 6. Descartes de entrada debido a una velocidad de línea superior a Eth1/9.](#)

Introducción

Este documento describe las causas y soluciones de los descartes de entrada para Cisco Nexus 9500-R EoR y Nexus 3000-R ToR. Un descarte de entrada indica el número de paquetes descartados en la cola de entrada debido a la congestión. Este número incluye caídas causadas por la caída de cola y la Detección temprana aleatoria ponderada (WRED).

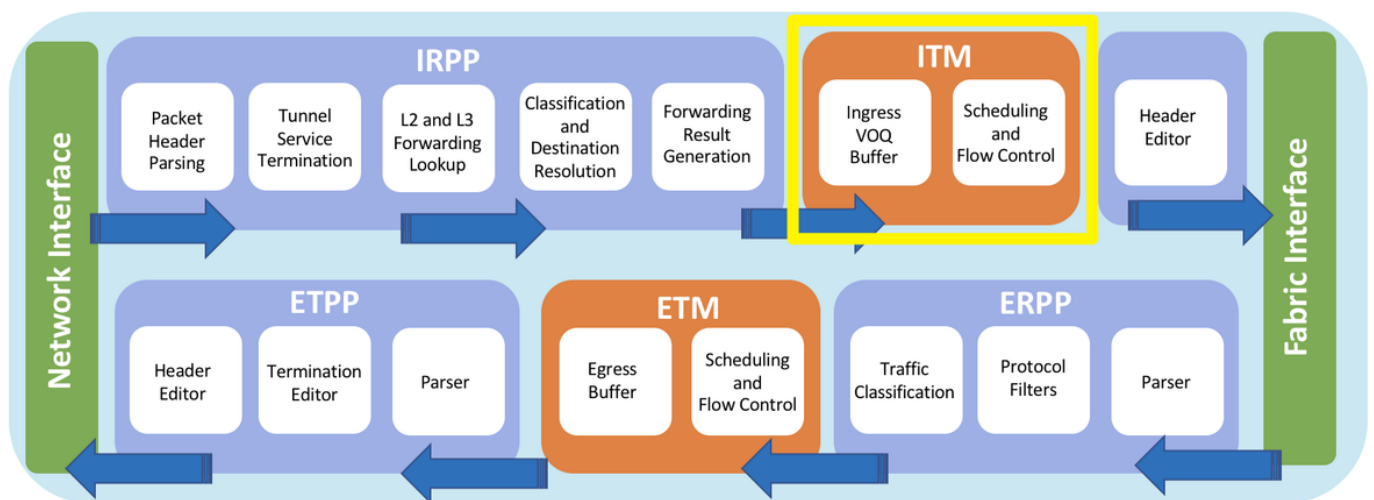
Si experimenta caídas aleatorias/espóricas/históricas (es decir, que ya no se producen), póngase en contacto con el TAC de Cisco para obtener más información. Este paso a través es útil cuando los descartes de entrada se incrementan con frecuencia.

Antecedentes

La serie R utiliza la arquitectura VOQ de ingreso. La arquitectura VOQ emula las colas de salida en el búfer de ingreso con las colas virtuales. Cada puerto de salida tiene ocho colas para el tráfico de unidifusión y ocho colas para el tráfico de multidifusión. El tráfico se puede clasificar en clases de tráfico basadas en el valor de Clase de servicio (CoS) o Punto de código de servicios diferenciados (DSCP) en los paquetes y, a continuación, se puede poner en cola en la cola virtual correspondiente para esa clase de tráfico.

La serie R utiliza un mecanismo de crédito distribuido para transferir tráfico a través del fabric. Antes de que se programe un paquete para salir de VOQ, el programador de búfer de ingreso solicita un crédito para el puerto específico y la prioridad en el búfer de salida. Se solicita crédito a un planificador de crédito de ingreso para el puerto de destino y la prioridad. Si el espacio del búfer está disponible, el programador de egreso concede acceso y envía la concesión de crédito al programador del búfer de ingreso. Si no hay espacio de búfer disponible en el búfer de salida, el programa de egreso no concede un crédito y el tráfico se almacena en el búfer en el VOQ hasta que el siguiente crédito esté disponible.

A continuación se muestra la canalización de reenvío de paquetes para la plataforma -R. En este artículo, se centra en el componente **Ingress Traffic Manager**. Más detalles sobre la arquitectura en este [enlace](#)



Administrador de tráfico de entrada (ITM)

El administrador de tráfico de entrada (ITM) es un bloque en el operador de comparación de entrada. Realiza los pasos relacionados con el tráfico en cola en VOQ, programa el tráfico para la transmisión a través del fabric y administra los créditos.

Búfer VOQ de ingreso

El bloque de memoria intermedia VOQ de ingreso administra tanto el buffer en el chip como el buffer de paquetes fuera del chip. Ambos buffers utilizan la arquitectura VOQ y el tráfico se coloca en cola según la información de IRPP (Procesador de paquetes de receptor de entrada). Hay un total de 96.000 VOQ disponibles para el tráfico de unidifusión y multidifusión.

Programación y control de flujo

Antes de que un paquete se transmita desde la canalización de ingreso, el paquete debe programarse para su transferencia a través del entramado. El programador de ingreso envía una solicitud de crédito al programador de egreso ubicado en el bloque del administrador de tráfico de egreso. Cuando el administrador de tráfico de ingreso recibe el crédito, comienza a enviar tráfico al procesador de paquetes de transmisión de ingreso. Si el búfer de salida está lleno, el tráfico se almacenará en la cola dedicada representada por el puerto de salida y la clase de tráfico.

Causas comunes

Por lo general, los descartes de entrada se pueden ver por los motivos siguientes en diversos hardware Nexus

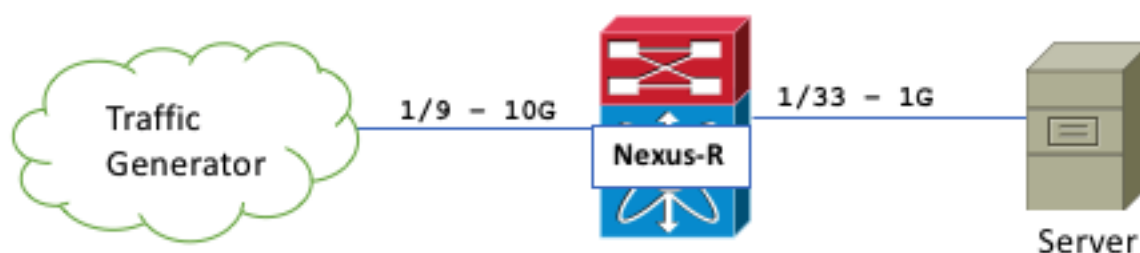
- Flujos de tráfico que congestionan las interfaces de salida (como entrada 10G y salida 1G)
- Puerto de destino SPAN con suscripción excesiva: se aplica a tipos de hardware específicos.

Hardware aplicable

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

Solución de problemas de descartes de entrada

Situación común: flujo de tráfico de 10 G a 1 G: caídas constantes:



A lo largo de este artículo, el valor para el contador de "descartes de entrada" y cualquier contador interno de hardware que haga referencia a lo mismo cambiará a medida que los errores aumenten mientras se realizan las pruebas y los comandos relevantes se deben tomar en vivo.

Paso 1. Verifique qué cola está impactada en su interfaz de descartes de entrada.

Este paso resulta útil más tarde.

En nuestro caso, es la cola 7, la cola predeterminada - Hay 8 colas en total en el ingreso:

```
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
```

Paso 2. Verifique la representación gráfica de contadores de Broadcom utilizados para el diagnóstico:

```
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 |                                         \|/
|

```

Un **QUEUE_DELETED_PACKET_COUNTER** mayor que cero indicaría que los paquetes fueron **ELIMINADOS** por el IQM (Administrador de almacenamiento en cola de entrada) después de la cola. Esto se debe a que una cola activa no recibe ningún crédito que sugiera una configuración incorrecta del esquema de programación. Esto se verificaría a través del **mod bcm-shell X "getReg IQM_QUEUE_DELETED_PACKET_COUNTER"**

ENQ_DISCARDED_PACKET_COUNTER significa que los paquetes se descartaron **ANTES** de la cola. También puede ver este contador establecido en BCM (el comando se borra al leer):

```

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,

```

Siempre puede notar estos rápidamente con **show hardware internal errors module X** (el comando se borra al leer):

```

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

```

Paso 3. Descubra qué ASIC y qué puerto Jericho pertenece al puerto del panel frontal que experimenta descartes de entrada:

```
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
```

Mostrando Eth1/33 para este ejemplo. En una red real, todavía no conocerá el puerto de salida congestionado.

Paso 4. Comprenda qué tienen el VoQ y el conector VOQ en su puerto de entrada.

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

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

Este comando nos muestra los detalles del flujo para VoQ de ingreso para un puerto específico. Además, nos muestra el saldo de crédito actual de VoQ.

El VOQ del puerto se deriva de esta manera:

Las LC se basan en 0 - el módulo 1 es 0, el módulo 2 es 1, etc
Hay 256 ID de puerto del sistema por LC

ID = (LC * ID de puerto del sistema) + número FP

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

ID de VOQ = 32 + (ID de puerto del sistema * 8)

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

Nuestro VOQ para Eth1/9 será por lo tanto 104 que coincide con la salida previamente recopilada

```
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| <<<<
+-----+-----+
```

Paso 5. Verifique desde la perspectiva de BCM, que cola específicamente no está vacía; es decir, congestionado.

```
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>
```

Paso 6. Busque el puerto congestionado de salida del valor de cola no vacío:

Si la cola es 303, recuerde que estas colas son en realidad un rango para que pueda ser 303 + 7 o 303-7 - La pregunta es, ¿qué puerto tiene un VOQ que coincida en un rango de 296-303 o, alternativamente, 303-310?

Se sabe que la cola 7 en Eth1/9 está congestionada, por lo que 303 en realidad es la más alta en su rango, así que el rango de 296-303 es una suposición bien educada.

```
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>
```


Muestra lo mismo para ASIC 0 - No se muestra aquí para la brevedad; en la columna Voq, notará que su rango de interés no está en ese ASIC

Observe algunas cosas en el resultado anterior:

- Nuestro puerto congestionado de salida está en ASIC 1.
- Nuestro puerto congestionado de salida tiene un VOQ de 296 y 303 equivaldría a la Cola 7 en ese puerto.
- Observe la columna Balance de Crédito - Quedan muy pocos créditos en esta interfaz para conceder, razón por la cual nuestro Eth1/9 de ingreso comienza a almacenar en búfer.

Paso 7. Verifique qué puerto del panel frontal está en ASIC 1 y mapea a Jericho Port 9 según su hallazgo anterior.

```
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
```

En este punto, ha encontrado el puerto congestionado de salida - Determine si hay alguna ráfaga incorrecta en la red, ha configurado SPAN y su puerto de destino es 1G mientras proporciona una o más interfaces 10G o si se trata de un problema de cuellos de botella/diseño.

Comandos adicionales

Estos son más avanzados: no se necesitan para encontrar el puerto congestionado de salida en escenarios normales.

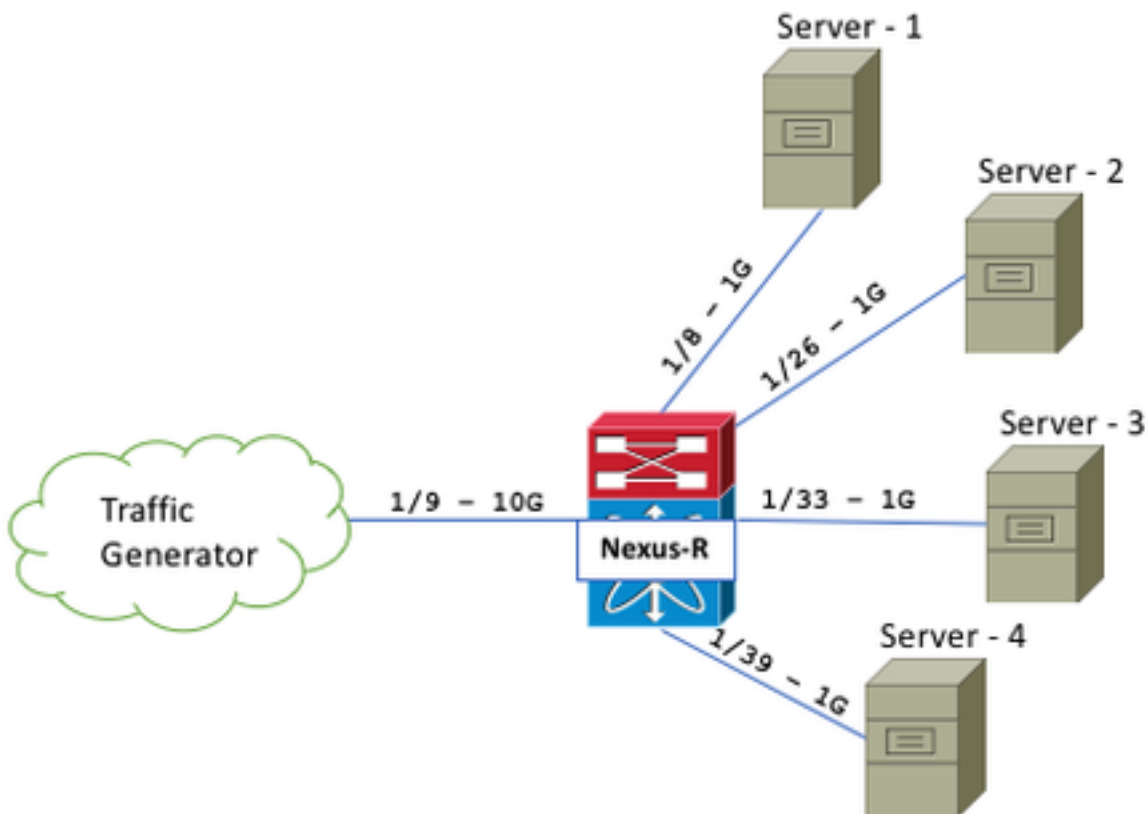
```
attach module X
show hardware internal jer-usd tm_debug asic <slot> module <module>
show hardware internal jer-usd info voq [ asic <instance> ] [ port <port> ] [ ]
show hardware internal jer-usd info non-empty voq asic [ <instance> ] [ ]
show hardware internal jer-usd info voq-profile { QueueThreshold 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-vsq 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
```

Pruebas adicionales de laboratorio:

Paso 1. Descartes de entrada con varias interfaces congestionadas de salida.

Tenga en cuenta esta topología en la que el Generador de tráfico envía tráfico 2G a cada servidor:



Verifique rápidamente qué colas no están vacías - Observe que hay 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)]
```

Determine a qué interfaces pertenecen estas colas - Verifique primero ASIC 0 (sólo se muestra con una interfaz):

```

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

```

Repita el mismo proceso para los otros tres valores de cola: 247, 303 y 351.

Paso 2. Descartes de entrada debido a SPAN.

Configuración de Eth1/33 como puerto de destino SPAN mientras se configura Eth1/9 como puerto de origen SPAN en la dirección RX

```

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

```

Paso 3. Descartes de entrada debido al PIN del capó de tráfico.

Envío de paquetes con SRC 10.10.10.10 y DEST 192.168.10.10 donde Eth1/9 está en 10.10.10.1/24 - Esto no resulta en un descarte de entrada; sin embargo, puede ver este contador:

```

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
+-----+
+-----+

```

Paso 4. Enviar paquetes con una IP de destino desconocida.

Envíe paquetes con SRC 10.10.10.10 y DEST 192.168.10.10 donde Eth1/9 está en 10.10.10.1/24 y Eth1/33 es un puerto L3 en la subred 172.16.0.1/30 - Sin contador de caídas, sin descartes de entrada incluso cuando el destino es **desconocido**.

Paso 5. Descartes de entrada mientras un puerto de acceso/enlace troncal pasa al estado de reenvío STP

Enviar paquetes donde Eth1/9 es sólo un tronco ancho (o puerto de acceso) - Esto se registra como un descarte de entrada mientras el puerto pasa a un estado de reenvío STP.

```
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 es un descarte debido a la decisión del procesador de paquetes (PP) de descartar o a un destino no válido recibido de los bloques de procesador de paquetes (PP).

Paso 6. Descartes de entrada debido a una velocidad de línea superior a Eth1/9.

Envía 10G+ a Eth1/9 resultaría en un tipo diferente de caída, ya que está realizando el envío de Eth1/9 en primer lugar - Todavía cuenta como descarte de entrada:

```
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	
PQP_DSCRD_MC_PKT_CNT		IQM	
		= 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>