

Pesquisa defeitos edições do encaminhamento de hardware em 7000 Series Switch do nexa

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Troubleshooting](#)

[Pesquisa defeitos ELAM nos módulos do F3 Series sem cabos breakout](#)

[Pesquisa defeitos ELAM nos módulos do F3 Series com cabos breakout](#)

Introdução

Este documento descreve como pesquisar defeitos edições do encaminhamento de hardware nos módulos do F3 Series para o Switches Cisco Nexus série 7000.

Pré-requisitos

Requisitos

Cisco recomenda que você tem uma familiaridade com o sistema operacional do nexa de Cisco (NX-OS) e a arquitetura básica do nexa antes que você continue com a informação que está descrita neste documento.

[Componentes Utilizados](#)

As informações neste documento são baseadas nestas versões de software e hardware:

- Switches Cisco Nexus série 7000 (N7K)
- Módulos de Ethernet Gigabit (N7K-F312FQ-25, 12-Port 10/40) dos módulos do F3 Series de Cisco N7K
- Versões 6.2.8a e mais recente do Cisco NX-OS

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se sua rede está viva, assegure-se de que você compreenda o impacto potencial do comando any.

Informações de Apoio

Este documento focaliza primeiramente em algumas das ferramentas incorporados que estão usadas para o Troubleshooting de hardware quando você esgotou sua parte de software da tabela do forwarding ou controla o plano. Uma tal ferramenta é o módulo encaixado do analisador de lógica (ELAM), que é uns circuitos integrados do aplicativo específicos (ASIC) esses captações um pacote único e mostras como o pacote de ingresso aparece no barramento de dados (DBUS) e no BARRAMENTO do resultado (RBUS) após a transmissão.

O ASIC é encaixado dentro do encanamento da transmissão, e pode capturar um pacote no tempo real sem rompimentos ao desempenho ou aos recursos de controle plano. Isto ajuda a responder a perguntas como:

- O pacote alcançou o Forwarding Engine (FE)?
- Em que porta e VLAN o pacote é recebido?
- Como o pacote aparece (camada 2 (L2) ou para mergulhar 4 dados (L4))?
- Como o pacote é alterado, e de onde é enviado?

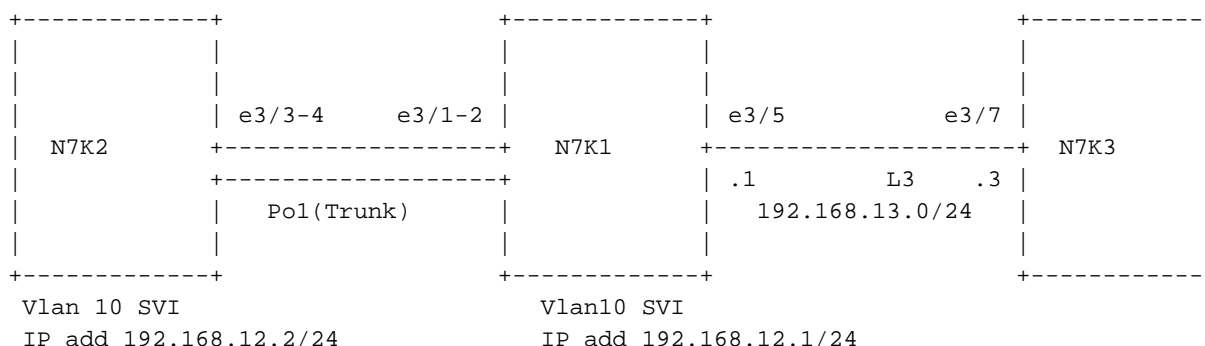
O ELAM é uma ferramenta poderosa, granulada, e sem intrusão que seja a mais de uso geral pelos coordenadores do centro de assistência técnica da Cisco (TAC) que trabalham em Plataformas do switching de hardware. Contudo, é importante saber que a ferramenta ELAM captura somente um pacote no tempo. Isto é, o primeiro pacote que é recebido depois que o ELAM é provocado.

Troubleshooting

Esta seção descreve como pesquisar defeitos ELAM em um módulo do F3 Series nas disposições que não envolvem o uso de um cabo breakout, assim como nas disposições que usam cabos breakout.

Pesquise defeitos ELAM nos módulos do F3 Series sem cabos breakout

Esta é a topologia que é usada para os exemplos durante todo esta seção:



Estão aqui algumas notas sobre esta topologia:

- A versão 6.2.8a da corrida NX-OS N7Ks.
- Os sibilos são enviados da relação N7K2 VLAN10 a um endereço IP remoto de 192.168.12.1.
- O ELAM captura pacotes no N7K1.

- Um N7K-F312FQ-25 é usado, que seja um módulo de Ethernet Gigabit 12-Port 10/40 introduzido no entalhe 3.

Antes que você comece a pesquisar defeitos seu sistema, você deve confirmar a conectividade básica:

```
N7K2# ping 192.168.13.3
PING 192.168.13.3 (192.168.13.3): 56 data bytes
64 bytes from 192.168.13.3: icmp_seq=0 ttl=253 time=1.513 ms
64 bytes from 192.168.13.3: icmp_seq=1 ttl=253 time=1.062 ms
64 bytes from 192.168.13.3: icmp_seq=2 ttl=253 time=0.822 ms
64 bytes from 192.168.13.3: icmp_seq=3 ttl=253 time=0.830 ms
64 bytes from 192.168.13.3: icmp_seq=4 ttl=253 time=0.845 ms

--- 192.168.13.3 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 0.822/1.014/1.513 ms
```

```
N7K2# show ip route 192.168.13.3
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'***' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>
```

```
192.168.13.0/24, ubest/mbest: 1/0
 *via 192.168.12.1, [1/0], 01:20:36, static
```

!--- The next command verifies the Address Resolution Protocol (ARP) for the next hop.

```
N7K2# show ip arp 192.168.12.1
----SNIP----
IP ARP Table
Total number of entries: 1
Address      Age      MAC Address      Interface
192.168.12.1 00:10:29 e4c7.2210.a142  Vlan10
```

Você deve igualmente verificar o endereço de controle de acesso de mídia (MAC) que aprende no Supervisor Engine (Sup) e no módulo para o salto seguinte:

```
N7K2# show mac address-table address e4c7.2210.a142
```

!--- This command output shows the MAC learning on the Sup (software).

Legend:

* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen, + - primary entry using vPC Peer-Link,
(T) - True, (F) - False

VLAN	MAC Address	Type	age	Secure	NTFY	Ports/SWID	SSID	LID
* 10	e4c7.2210.a142	dynamic	120	F	F	Po1		

Esta saída mostra a aprendizagem MAC no módulo/hardware; contudo, a fim conhecer a relação, você deve converter o deslocamento predeterminado:

```
N7K2# show hardware mac address-table 3 address e4c7.2210.a142
FE | Valid| PI|  BD |      MAC      | Index | Stat| SW  | Modi| Age| Tmr| GM| Sec|
  |     |  |   |             |       |    |   |   |   |   |   |   |   |
---|-----|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|
 1  1  1  41  e4c7.2210.a142  0x00a2a  0  0x089  1  185  1  0  0
```

```

| TR| NT| RM| RMA| Cap| Fld| Always| PV | RD| NN| UC| PI_E8| VIF | SWID| SSWID| LID |
| AP| FY|   |   | ture|   | Learn|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
0  0  0  0  0  0  0  0x00 0  0  1  0  0x000 0x000 0x000 0x00a2a

```

```
N7K2# show system internal pixm info ltl 0x00a2a
```

!--- This is the index that was received in the previous output.

---SNIP---

PC_TYPE	PORT	LTL	RES_ID	LTL_FLAG	CB_FLAG	MEMB_CNT
Normal	Po1	0x0a2a	0x16000000	0x00000000	0x00000002	2

```

Member rbh rbh_cnt
Eth3/4 0x000000f0 0x04
Eth3/3 0x0000000f 0x04

```

---SNIP---

Incorpore estes comandos a fim obter o número do contexto do dispositivo virtual (VDC) (neste exemplo, é 3) e verificar o MAC diretamente no módulo:

```
N7K2# show vdc
```

---SNIP---

vdc_id	vdc_name	state	mac	type	lc
3	N7K2	active	e4:c7:22:10:a1:43	Ethernet	f3

```
module-3#attach module 3
```

```
module-3# vdc 3
```

!--- This data is obtained from the previous command output.

```
module-3# show mac address-table address e4c7.2210.a142
```

Legend:

```

* - primary entry, G - Gateway MAC, (R) - Routed MAC, (d) - dec
Age - seconds since last seen,,+ - primary entry using vPC Peer-Link
(T) - True, (F) - False, h - hex, d - decimal

```

VDC = 3

FE	VLAN/BD	MAC Address	Type	Age	Secure	NTFY	Ports/SWID.SSID.LID(d)
* 1	10	e4c7.2210.a142	dynamic	360	F	F	Po1

Determine o link no Canal de porta 1 que é usado a fim enviar o tráfego no Sup de N7K2, assim como o link que está usado a fim enviar uma resposta de N7K3 quando o Canal de porta 1 é usado de N7K1 a N7K2:

```
N7K2# show port-channel load-balance forwarding-path interface port-channel 1 src-ip
192.168.12.2 dst-ip 192.168.13.3 module 3
```

Module 3: Missing params will be substituted by 0's.

Load-balance Algorithm: src-dst ip

RBH: 0xd2 Outgoing port id: Ethernet3/3

```
N7K1# show port-channel load-balance forwarding-path interface port-channel 1 src-ip 192.168.13.3 dst-ip 192.168.12.2 module 3
```

Module 3: Missing params will be substituted by 0's.

Load-balance Algorithm: src-dst ip

RBH: 0xd2 Outgoing port id: Ethernet3/1

Envie um sibilo de N7K2 (endereço IP 192.168.12.2) e capture os pacotes em N7K1 na direção de ingresso a fim confirmar que os pacotes estão enviados a N7K3 (endereço IP de Um ou Mais Servidores Cisco ICM NT 192.168.13.3).

Antes que você envie o sibilo, você deve ter o conhecimento do acúmulo do hardware. Termine estas etapas a fim compreender o acúmulo:

1. Anexe o módulo:

```
N7K1# attach module 3
Attaching to module 3 ...
To exit type 'exit', to abort type '$.'
```

2. Identifique o exemplo do *flanker*. O flanker é um interruptor na microplaqueta (SOC) ASIC para o módulo do F3 Series. Cada flanker é traçado a duas portas externas no módulo (as alterações de informação pelo tipo de módulo e são específicas ao N7K-F312FQ-25).

Há 12 portas no módulo, e mapas cada ASIC a duas portas no painel dianteiro, assim que significa que há (0-5) uns exemplos do flanker 6 disponíveis no módulo (a contagem do exemplo zero-é baseada). Nota: Assegure-se de que você tenha privilégios administrativos da rede antes que você comece. Porque você captura o pacote que chega de N7K2 através do Canal de porta 1 em N7K1, procure as portas (e3/1 e e3/2) que são traçadas a cada exemplo:

```
module-3# show hardware internal dev-port-map
-----
CARD_TYPE:           12 port 40G
>Front Panel ports:12
-----
Device name           Dev role           Abbr num_inst:
-----
>Flanker Eth Mac Driver DEV_ETHERNET_MAC    MAC_0   6
>Flanker Fwd Driver    DEV_LAYER_2_LOOKUP L2LKP   6

!--- Check for the L2LKP number for ports 1 and 2.

>Flanker Xbar Driver   DEV_XBAR_INTF       XBAR_INTF 6
>Flanker Queue Driver  DEV_QUEUEING        QUEUE   6
>Sacramento Xbar ASIC  DEV_SWITCH_FABRIC  SWICHF  1
>Flanker L3 Driver     DEV_LAYER_3_LOOKUP L3LKP   6
>EDC                   DEV_PHY             PHYS    2
+-----+
+-----++FRONT PANEL PORT TO ASIC INSTANCE MAP+++-----+
+-----+
FP port |  PHYS | MAC_0 | L2LKP | L3LKP | QUEUE | SWICHF
      1 |       |    0  |    0  |    0  |    0  |    0
      2 |       |    0  |    0  |    0  |    0  |    0
      3 |       |    1  |    1  |    1  |    1  |    0
      4 |       |    1  |    1  |    1  |    1  |    0
      5 |    0  |    2  |    2  |    2  |    2  |    0

!--- The L2LKP for both ports is 0, so both belong to instance 0.
```

```

6      0      2      2      2      2      0
7      1      3      3      3      3      0
8      1      3      3      3      3      0
9              4      4      4      4      0
10             4      4      4      4      0
11             5      5      5      5      0
12             5      5      5      5      0

```

```

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

```

3. Selecione o exemplo, ajuste o disparador, e comece a captação. É importante compreender, contudo, que há muitas opções que podem ser usadas com o disparador ELAM:

```

module-3# elam asic flanker instance 0
module-3(fln-elam)# layer2
module-3(fln-l2-elam)# trigger ?
  dbus  Pre L2 BUS
  rbus  Post L2 BUS

```

-----SNIP----- Estas duas opções são importantes se você quer incluir o DBUS na captação (o pacote que é recebido pelo interruptor). Este é o pacote cru que não é sujeitado a uma consulta. O RBUS mostra os resultados de consulta no hardware para um DBUS. Para um ELAM e uma análise completos, você deve capturar o RBUS e o DBUS.

A saída seguinte mostra os tipos de pacotes que você pode capturar com a opção do DBUS. Neste exemplo, o pacote da versão 4 do protocolo de internet (IPv4) é selecionado:

```

module-3(fln-l2-elam)# trigger dbus ?
  arp    ARP Frame Format
  fc     Fc hdr Frame Format
  ipv4   IPV4 Frame Format
  ipv6   IPV6 Frame Format
  mpls   MPLS
  other  L2 hdr Frame Format
  pup    PUP Frame Format
  rarp   RARP Frame Format

```

valid On valid packet **Estão aqui algumas opções adicionais que você pode escolher se usar:**

```

module-3(fln-l2-elam)# trigger dbus ipv4 ?
  egress          Egress packets

```

!--- Capture packets in egress (outbound from the port).

```

  if              If Trigger Condition
  ingress         Ingress packets

```

!--- Capture packets in ingress (inbound to the port).

```

  multicast          Multicast packet

```

multicast-replication Multicast replication **Neste exemplo, se o punho é usado a fim selecionar uma circunstância para a captação. A maioria das opções que mostrado na saída seguinte são baseados nos encabeçamentos L2, L3, e L4. Os endereços IP de origem e de destino são usados igualmente para a captação.**

```

module-3(fln-l2-elam)# trigger dbus ipv4 ingress if ?
<CR>
  acos              Acos

```

block-capture	Capture l2 blocks
bpdu	Bpdu
bundle-port	Bundle-port
ccc	Ccc
copp	Copp
da-type	Da-type
de-cfi	De cfi
destination-index	Destination-index
destination-ipv4-address	destination ipv4 address
destination-mac-address	Destination-mac-address
destination-vif	Destination-vif
df	df
dfst	Dfst
dft	Dft
disable-index-learn	Disable-index-learn
disable-new-learn	Disable-new-learn
dont-forward	Dont-forward
dont-learn	Dont-learn
dtag-ftag	Dtag-ftag
dtag-ttl	Dtag-ttl
dti-type-vpnid	Dti type vpnid
error	Error
erspan-kpa-valid	Erspan kpa valid
ff	Ff
frag	frag
header-type	Header type
ib-length-bundle	Ib length bundle
ids-check-fail	Ids-check-fail
ignore-acli	Ignore-acli
ignore-aclo	Ignore-aclo
ignore-qosi	Ignore-qosi
ignore-qoso	Ignore-qoso
inband-flow-creation-deletion	Inband-flow-creation-deletion
index-direct	Index-direct
inner-cos	Inner-cos
inner-de-valid	Inner de valid
inner-drop-eligibility	Inner-drop-eligibility
ip-da-multicast	Ip-da-multicast
ip-multicast	Ip-multicast
ip-multicast-control	Ip-multicast-control
ipv6	Ipv6
l2	L2
l2-frame-type	L2-frame-type
l2-length-check	L2 length check
l2lu-mode	L2lu-mode
l3-packet-length	l3 packet length
l4-protocol	l4 protocol
label-count	Label count
last-ethertype	Last-ethertype
lbl0-eos	Lbl0 eos
lbl0-exp	Lbl0 exp
lbl0-lbl	Lbl0 lbl
lbl0-ttl	Lbl0 ttl
lbl0-valid	Lbl0 valid
lbl1-exp	Lbl1 exp
lbl1-ttl	Lbl1 ttl
mac-in-mac-valid	Mac-in-mac-valid
mc	Mc
md-acos	Md acos
md-destination-table-index	Md destination table index
md-fwd-only	Md fwd only
md-lif	Md lif
md-mark-enable	Md mark enable
md-multicast-bridge-disable	Md multicast bridge disable

md-preserve-acos	Md preserve acos
md-qos-group-id	Md qos group id
md-replication-packet	Md replication packet
md-router-mac	Md router mac
md-ttl-err	Md-ttl-err
md-version	Md version
mf	mf
mim-destination-mac-address	Mim-destination-mac-address
mim-source-mac-address	Mim-source-mac-address
mlh-type	Mlh-type
no-stats	No-stats
notify-index-learn	Notify-index-learn
notify-new-learn	Notify-new-learn
null-label-exp	Null label exp
null-label-ttl	Null label ttl
null-label-valid	Null label valid
option	option
outer-cos	Outer-cos
outer-drop-eligibility	Outer-drop-eligibility
ovl-mlh-bndl	Ovl mlh bndl
ovl-ulh-bndl	Ovl ulh bndl
ovl-ulh-bndl-1	Ovl-ulh-bndl-1
ovl-ulh-bndl-2	Ovl-ulh-bndl-2
packet-length	Packet-length
packet-type	Packet type
pd-tag-gt-2	Pdt-tag-gt-2
pd-tag0	Pdt-tag0
pd-tag1	Pdt-tag1
pd-valid	Pdt-valid
pd-value	Pdt-value
port-id	Port-id
rbh	Rbh
rdt	Rdt
recirc-shim-vxlan-src-peer-id	Recirc shim vxlan src peer id
recirc-acos	Recirc acos
recirc-bypass-ife	Recirc bypass ife
recirc-bypass-l2	Recirc bypass l2
recirc-destination-table-index	Recirc destination table index
recirc-forward-only	Recirc forward only
recirc-l2-tunnel-encap	Recirc l2 tunnel encap
recirc-lif	Recirc lif
recirc-ls-hash	Recirc ls hash
recirc-mark-enable	Recirc mark enable
recirc-multicast-bridge-disable	Recirc multicast bridge disable
recirc-preserve-acos	Recirc preserve acos
recirc-preserve-ls-hash	Recirc preserve ls hash
recirc-preserve-rbh	Recirc preserve rbh
recirc-qos-group-id	Recirc qos group id
recirc-replication-packet	Recirc replication packet
recirc-router-mac	Recirc router mac
recirc-ttl-err	Recirc ttl err
recirc-valid	Recirc-valid
recirc-version	Recirc version
redirect	Redirect
repl-bypass-ife	Repl bypass ife
repl-bypass-l2	Repl bypass l2
repl-disable-local-bridge	Repl disable local bridge
repl-fwd-only	Repl fwd only
repl-l2-tunnel-encap	Repl l2 tunnel encap
repl-l2-tunnel-info	Repl l2 tunnel info
repl-lif	Repl lif
repl-mark-enable	Repl mark enable
repl-met-lif	Repl met lif
repl-ml3	Repl ml3

repl-preserve-acos	Repl preserve acos
repl-preserve-rbh	Repl preserve rbh
repl-qos-group-id	Repl qos group id
repl-replication-packet	Repl replication packet
repl-router-mac	Repl router mac
repl-ttl-err	Repl ttl err
repl-version	Repl version
rf	Rf
second-inner-cos	Second inner cos
segment-id	Segment id
segment-id-valid	Segment id valid
sequence-number	Sequence-number
sg-tag	Sg-tag
shim-valid	Shim valid
source-index	Source-index
source-ipv4-address	source ipv4 address
source-mac-address	Source-mac-address
source-vif	Source-vif
status-ce-1q	Status-ce-1q
status-is-1q	Status-is-1q
sup-eid	Sup-eid
tos	tos
traceroute	Traceroute
trig	Any of previous elam triggered
trill-encap	Trill-encap
ttl	ttl
tunnel-bundle	Tunnel bundle
tunnel-type	Tunnel type
ulh-type	Ulh-type
valid	VALID
v1	V1
vlan	Vlan
vn-p	Vn p
vn-valid	Vn-valid
vqi	Vqi
vqi-valid	Vqi-valid
vsl-num	Vsl-num

Esta saída mostra a opção final do disparador:

```

module-3# elam asic flanker instance 0
module-3(fln-elam)# layer2
module-3(fln-l2-elam)# trigger dbus ipv4 ingress if source-ipv4-address 192.168.12.2
destination-ipv4-address 192.168.13.3

```

```

module-3(fln-l2-elam)# trigger rbus ingress if trig

```

Nota: A configuração RBUS não é geralmente complexa e mantida simples.

4. A fim verificar o disparador, inscrever o comando **status**, começar o processo da captação, e iniciar um sibilo de N7K2 a N7K3 (192.168.12.1 a 192.168.13.3):

```

module-3(fln-l2-elam)# stat
ELAM Slot 3 instance 0: L2 DBUS Configuration: trigger dbus ipv4 ingress if
source-ipv4-address 192.168.12.2 destination-ipv4-address 192.168.13.3
L2 DBUS: Configured
ELAM Slot 3 instance 0: L2 RBUS Configuration: trigger rbus ingress if trig
L2 RBUS: Configured

```

```

module-3(fln-l2-elam)# start
module-3(fln-l2-elam)# status

```

!--- The status shows as Armed because the process has begun.

```

ELAM Slot 3 instance 0: L2 DBUS Configuration: trigger dbus ipv4 ingress if

```

```

source-ipv4-address 192.168.12.2 destination-ipv4-address 192.168.13.1
L2 DBUS: Armed
ELAM Slot 3 instance 0: L2 RBUS Configuration: trigger rbus ingress if trig
L2 RBUS: Armed
module-3(fln-l2-elam)#

```

```

module-3(fln-l2-elam)# status

```

!--- If the packet is captured, the status shows Triggered.

```

ELAM Slot 3 instance 0: L2 DBUS Configuration: trigger dbus ipv4 ingress if
source-ipv4-address 192.168.12.2 destination-ipv4-address 192.168.13.3
L2 DBUS: Triggered
ELAM Slot 3 instance 0: L2 RBUS Configuration: trigger rbus ingress if trig
L2 RBUS: Triggered
module-3(fln-l2-elam)#

```

5. Se as mostras do estado **provocadas**, verificam então se os RBUS e o DBUS têm o mesmo número de sequência a fim confirmar que são para o mesmo pacote. Neste exemplo, **0x55** é usado, mas a coluna que mostra o número de sequência é diferente:

```

module-3(fln-l2-elam)# show dbus | in seq
sequence-number      : 0x6b          vl      : 0x0

```

!--- The sequence number is the same (0x6b).

```

module-3(fln-l2-elam)# show rbus | in seq
l2-rbus-trigger      : 0x1          sequence-number      : 0x6b

```

6. Entre no **dbus da mostra e mostre** comandos do **rbus** a fim verificar o DBUS e o RBUS. Procure o *deslocamento predeterminado de fonte na saída do comando do DBUS* e o *deslocamento predeterminado do destino no comando RBUS output*:

```

module-3(fln-l2-elam)# show dbus
cp = 0x1007db4c, buf = 0x1007db4c, end = 0x10089e9c

```

Flanker Instance 00 - Capture Buffer On L2 DBUS:

```

Status(0x0102), TriggerWord(0x000), SampleStored(0x005),CaptureBufferPointer(0x005)

```

```

is_l2_egress: 0x0000, data_size: 0x023
[000]: 5902a000 08010000 00000000 0cc01400 00145800 00000000 01800100 00000000
00000000 00000000 003931c8 842850b9 31c88428 50c00000 01ac0000 00000000 00000000
00000000 00000000 00000000 00000005 80005000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00605406 01605406 8180008f f0054608 00000000

```

Printing packet 0

L2 DBUS PRS MLH IPV4

```

label-count          : 0x0          mc          : 0x0
null-label-valid     : 0x0          null-label-exp : 0x0
null-label-ttl       : 0x0          lb10-vld     : 0x0
lb10-eos             : 0x0          lb10-lb1     : 0x0
lb10-exp             : 0x0          lb10-ttl     : 0x0
lb11-exp             : 0x0          lb11-ttl     : 0x0
ipv4                 : 0x0          ipv6         : 0x0
l4-protocol          : 0x1          df           : 0x0
mf                   : 0x0          frag        : 0x0
ttl                  : 0xff         l3-packet-length : 0x54
option               : 0x0          tos         : 0x0
sup-eid              : 0x0          header-type  : 0x1

```

```

error : 0x0 redirect : 0x0
port-id : 0x0 last-ethertype : 0x800
l2-frame-type : 0x0 da-type : 0x0
packet-type : 0x0 l2-length-check : 0x0
ip-da-multicast : 0x0 ip-multicast : 0x0
ip-multicast-control : 0x0 ids-check-fail : 0x0
traceroute : 0x0 outer-cos : 0x0
inner-cos : 0x0 vqi-valid : 0x0
vqi : 0x0 packet-length : 0x66
vlan : 0xa destination-index : 0x0
source-index : 0xa2c bundle-port : 0x0
acos : 0x0 outer-drop-eligibility : 0x0
inner-drop-eligibility : 0x0 sg-tag : 0x0
rbh : 0x0 vsl-num : 0x0
inband-flow-creation-deletion : 0x0 ignore-qoso : 0x0
ignore-qosi : 0x0 ignore-aclo : 0x0
ignore-acli : 0x0 index-direct : 0x0
no-stats : 0x0 dont-forward : 0x0
notify-index-learn : 0x1 notify-new-learn : 0x1
disable-new-learn : 0x0 disable-index-learn : 0x0
dont-learn : 0x0 bpdu : 0x0
ff : 0x0 rf : 0x0
ccc : 0x0 l2 : 0x0
rdt : 0x0 dft : 0x0
dfst : 0x0 status-ce-lq : 0x0
status-is-lq : 0x1 trill-encap : 0x0
mim-valid : 0x0 dtag-ttl : 0x0
dtag-ftag : 0x0 valid : 0x1
erspan-kpa-valid : 0x0 recir-shim-vxlan-src-peer-id : 0x0
vn-valid : 0x0 source-vif : 0x0
destination-vif : 0x0 vn-p : 0x0
sequence-number : 0x6b vl : 0x0
inner-de-valid : 0x0 de-cfi : 0x0
second-inner-cos : 0x0 tunnel-type : 0x0
shim-valid : 0x0
segment-id-valid : 0x0 copp : 0x0
dti-type-vpnid : 0x0 segment-id : 0x0
ib-length-bundle : 0x58000 mlh-type : 0x5
ulh-type : 0x6
source-ipv4-address : 192.168.12.2
destination-ipv4-address : 192.168.13.3
mim-destination-mac-address : 0000.0000.0000
mim-source-mac-address : 0000.0000.0000
destination-mac-address : e4c7.2210.a142
source-mac-address : e4c7.2210.a143

```

```

module-3(fln-l2-elam)# show rbus
cp = 0x100a2548, buf = 0x100a2548, end = 0x100ae898

```

```

-----
Flanker Instance 00 - Capture Buffer On L2 RBUS:

```

```

Status(0x0102), TriggerWord(0x000), SampleStored(0x005), CaptureBufferPointer(0x005)

```

```

is_l2_egress: 0x0000, data_size: 0x018
[000]: 0059d930 0000000c c0000000 03580000 00000000 00000000 0000001f 57b00021
fdfc0000 00000000 02000000 14001402 8b000105 00000000 68200000 00000000 00000000
00000400 00008000 005b0000 00fe0e4c 7220850a 210000a0 000000b6

```

```

Printing packet 0

```

```

-----
L2 RBUS INGRESS CONTENT
-----

```

```

pad : 0x16764 valid : 0x1
l2-rbus-trigger : 0x1 sequence-number : 0x6b
rit-ipv4-id : 0x0 ipv4-tunnel-encap : 0x0
rit-mpls-rw : 0x0 ml2-ptr : 0x0
ml3-ptr : 0x0 mark : 0x0
result-cap3 : 0x0 dil-v5-delta-length : 0x0
dil-v5-delta-length-plus: 0x0 dil-v4-delta-length : 0x0
dil-v4-delta-length-plus: 0x0 di2-delta-length : 0x0
di2-delta-length-plus: 0x0 ml2-delta-length : 0x0
ml2-delta-length-plus: 0x0 ml3-delta-length : 0x0
ml3-delta-length-plus: 0x0 s-vector : 0x0
lcpu-ff-valid : 0x0 sup-di-vqi : 0x0
erspan-term-index-dir: 0x0 erspan-buffer-check : 0x0
l2-tunnel-decapped : 0x0 l3-delta-length : 0x0
rit-crc16-valid : 0x1 rit-crc16 : 0xf57b
vntag-p : 0x0 frr-recirc : 0x0
ingress-lif : 0x1 earl-proxy-vld : 0x0
md-di-vld : 0x0 rc : 0x0
segment-id-valid : 0x0 ttl-out : 0xfe
ttl-mid : 0xfe tos-out : 0x0
tos-in : 0x0 orig-vlan1 : 0x0
vlan1 : 0x0 source-peer-id : 0x0
final-ignore-qoso : 0x0 port-id : 0x0
cr-type : 0x1 pup-packet : 0x0
bpdu : 0x0 vdc : 0x0
traceroute : 0x0 de : 0x0
cos : 0x0 inner-drop-eligibility: 0x0
inner-cos : 0x0 acos : 0x0
di-ltl-index : 0x50 l3-multicast-di : 0x50
source-index : 0xa2c vlan : 0x0
index-direct : 0x0 dil-valid : 0x1
vqi : 0x50 di2-valid : 0x0
v5-fpoe-idx : 0x0 di2-fpoe-idx : 0x0
l3-multicast-v5 : 0x0 dft : 0x0
dfst : 0x0 l3-learning-ff : 0x0
result-rbh : 0xd0 di2-cr-type : 0x0
result-2 : 0x1 dtag-ftag : 0x0
dtag-ttl : 0x20 mac-in-mac-op : 0x0
dvif : 0x0 result-cap1 : 0x0
result-cap2 : 0x0 erspan-term : 0x0
erspan-decap : 0x0 dont-learn : 0x0
routed-frame : 0x1 copy-cause : 0x0
l2-copy-cause : 0x0 l3-rit-ptr : 0x5b
sg-tag : 0x0 trill-nh-id : 0x0
ttl-in : 0xfe fc-up : 0x0
up-did : 0x0 did : 0xe4c722
up-sid : 0x0 sid : 0x10a144
shim-l2-tunnel-encap: 0x0 shim-ls-hash : 0x8
shim-rc : 0x0 shim-lif : 0x1
shim-replication-pkt: 0x0 shim-router-mac : 0x1
shim-mark-enable : 0x0 shim-qos-group-id : 0x0
shim-destination-table-index: 0x5b shim-acos-preserve : 0x0
mim-destination-mac-address : 0000.0000.0000
mim-source-mac-address : 0000.0000.0000

```

7. Verifique o deslocamento predeterminado do destino e o deslocamento predeterminado de fonte no Sup:

```
N7K1# show system internal pixm info ltl 0xa2c
```

PC_TYPE	PORT	LTL	RES_ID	LTL_FLAG	CB_FLAG	MEMB_CNT
Normal	Po1	0x0a2c	0x16000000	0x00000000	0x00000002	2

```
Member rbh rbh_cnt
Eth3/2 0x000000f0 0x04
Eth3/1 0x0000000f 0x04
```

CBL Check States: Ingress: Enabled; Egress: Enabled

```
VLAN| BD| BD-St | CBL St & Direction:
-----
1 | 0x15 | INCLUDE_IF_IN_BD | FORWARDING (Both)
10 | 0x19 | INCLUDE_IF_IN_BD | FORWARDING (Both)
```

Member info

```
-----
Type LTL
-----
PORT_CHANNEL Po1
FLOOD_W_FPOE 0x8019
FLOOD_W_FPOE 0x8015
```

```
N7K1# show system internal pixm info ltl 0x50
0x0050 is in DCE/FC pool
```

Member info

```
-----
Type LTL
-----
PHY_PORT Eth3/5
```

Esta saída confirma que o pacote esteve recebido no Canal de porta 1 (Po1) e enviado através de Eth3/5.

8. Verifique a lógica de alvo local (LTL) no módulo para a programação correta:

```
module-3# show system internal pixmc info ltl-cb ltl 0xa2c
ltl |ltl_type|if_index|lc_type| vdc |v4_fpoelv5_fpoel base_fpoel_idx | flag
0x0a2c | 4 | Po1 | 2 | 2 | 0x00 | 0x00 | 0x0000 | 0x0
, local ports:
VDCs the entry is part of:
```

LTL HW programming info

```
.....
-----
|Index | ec |drop|span_vec|SOM|ucr_fab|
|-----|
|[ a2c]| 1| 0| 0| 0| 0|
| RBH | VQI | PS(INST:LPOE)
|-----|
0, 40 0 : 1
1, 40 0 : 1
2, 40 0 : 1
3, 40 0 : 1
4, 44 0 : 10
5, 44 0 : 10
6, 44 0 : 10
7, 44 0 : 10
8, 0 0 : 1
9, 0 0 : 1
a, 0 0 : 1
b, 0 0 : 1
c, 0 0 : 10
d, 0 0 : 10
```

```
e,      0      0 : 10
f,      0      0 : 10
```

```
module-3# show system internal pixmc info ltl-cb ltl 0x50
```

```
ltl |ltl_type|if_index|lc_type| vdc |v4_fpoe | v5_fpoe| base_fpoe_idx | flag
0x0050 | 5 |Eth3/5 | 2 | 2 | 0x00 | 0x00 | 0x0000 | 0x0
```

```
, local ports:
```

```
VDCs the entry is part of:
```

```
LTL HW programming info
```

```
.....
```

```
-----
|Index | ec |drop|span_vec|SOM|ucr_fab|
|-----
```

```
| [ 50] | 1 | 0 | 0 | 0 | 0 |
```

```
| RBH | VQI | PS
```

```
|-----
```

```
ALL RBH | 50 | 2 : 1
```

9. Capture o pacote ELAM em cima da saída. A fim capturar o pacote, envie uma resposta do sibilo de IP address 192.168.13.3 a 192.168.12.2. Você deve ajustar a captura com a palavra-chave da **saída nas** relações do Canal de porta 1 (e3/1-2). As relações pertencem para citar como exemplo **0**, como descrito anteriormente.

```
N7K1# att mo 3
```

```
Attaching to module 3 ...
```

```
To exit type 'exit', to abort type '$.'
```

```
module-3# el asic flanker instance 0
```

```
module-3(fln-elam)# layer2
```

```
module-3(fln-l2-elam)# trigger dbus ipv4 egress if source-ipv4-address 192.168.13.3
```

```
destination-ipv4-address 192.168.12.2
```

```
module-3(fln-l2-elam)# trigger rbus egress if trig
```

```
module-3(fln-l2-elam)# status
```

```
ELAM Slot 3 instance 0: L2 DBUS Configuration: trigger dbus ipv4 egress if
source-ipv4-address 192.168.13.3 destination-ipv4-address 192.168.12.2
```

```
L2 DBUS: Configured
```

```
ELAM Slot 3 instance 0: L2 RBUS Configuration: trigger rbus egress if trig
```

```
L2 RBUS: Configured
```

```
module-3(fln-l2-elam)# start
```

```
module-3(fln-l2-elam)# status
```

```
ELAM Slot 3 instance 0: L2 DBUS Configuration: trigger dbus ipv4 egress if
source-ipv4-address 192.168.13.3 destination-ipv4-address 192.168.12.2
```

```
L2 DBUS: Armed
```

```
ELAM Slot 3 instance 0: L2 RBUS Configuration: trigger rbus egress if trig
```

```
L2 RBUS: Armed
```

```
module-3(fln-l2-elam)# status
```

```
ELAM Slot 3 instance 0: L2 DBUS Configuration: trigger dbus ipv4 egress if
source-ipv4-address 192.168.13.3 destination-ipv4-address 192.168.12.2
```

```
L2 DBUS: Triggered
```

```
ELAM Slot 3 instance 0: L2 RBUS Configuration: trigger rbus egress if trig
```

```
L2 RBUS: Triggered
```

```
module-3(fln-l2-elam)#
```

```
module-3(fln-l2-elam)# show dbus | in seq
sequence-number      : 0x8d          vl          : 0x3
```

!--- The sequence number is the same.

```
module-3(fln-l2-elam)# show rbus | in seq
vl                   : 0x0          sequence-number : 0x8d
```

```
module-3(fln-l2-elam)# show dbus
cp = 0x1007db4c, buf = 0x1007db4c, end = 0x10089e9c
```

Flanker Instance 00 - Capture Buffer On L2 DBUS:

Status(0x0102), TriggerWord(0x000), SampleStored(0x005),CaptureBufferPointer(0x005)

```
is_l2_egress: 0x0000, data_size: 0x023
[000]: 48c22000 08210000 40020800 0cc01414 5800a000 00001a40 01030000 00000000
00000000 00000000 003931c8 842850f9 31c88428 50800000 02358000 00000000 00000000
00000000 00000000 00000000 00000000 00005000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00605406 81e05406 0100008f e0054600 00000000
```

Printing packet 0

L2 DBUS PRS MLH IPV4

label-count	: 0x0	mc	: 0x0
null-label-valid	: 0x0	null-label-exp	: 0x0
null-label-ttl	: 0x0	lbl0-vld	: 0x0
lbl0-eos	: 0x0	lbl0-lbl	: 0x0
lbl0-exp	: 0x0	lbl0-ttl	: 0x0
lbl1-exp	: 0x0	lbl1-ttl	: 0x0
ipv4	: 0x0	ipv6	: 0x0
l4-protocol	: 0x1	df	: 0x0
mf	: 0x0	frag	: 0x0
ttl	: 0xfe	l3-packet-length	: 0x54
option	: 0x0	tos	: 0x0
sup-eid	: 0x0	header-type	: 0x1
error	: 0x0	redirect	: 0x0
port-id	: 0x1	last-ethertype	: 0x800
l2-frame-type	: 0x0	da-type	: 0x0
packet-type	: 0x1	l2-length-check	: 0x0
ip-da-multicast	: 0x0	ip-multicast	: 0x0
ip-multicast-control	: 0x0	ids-check-fail	: 0x0
traceroute	: 0x0	outer-cos	: 0x0
inner-cos	: 0x0	vqi-valid	: 0x1
vqi	: 0x40	packet-length	: 0x66
vlan	: 0xa	destination-index	: 0xa2c
source-index	: 0x50	bundle-port	: 0x0
acos	: 0x0	outer-drop-eligibility	: 0x0
inner-drop-eligibility	: 0x0	sg-tag	: 0x0
rbh	: 0xd2	vsl-num	: 0x0
inband-flow-creation-deletion	: 0x0	ignore-qoso	: 0x0
ignore-qosi	: 0x0	ignore-aclo	: 0x0
ignore-acli	: 0x0	index-direct	: 0x0
no-stats	: 0x0	dont-forward	: 0x0
notify-index-learn	: 0x1	notify-new-learn	: 0x0
disable-new-learn	: 0x0	disable-index-learn	: 0x0
dont-learn	: 0x0	bpdu	: 0x0
ff	: 0x0	rf	: 0x1
ccc	: 0x4	l2	: 0x0
rdt	: 0x0	dft	: 0x0
dfst	: 0x0	status-ce-lq	: 0x0

```

status-is-lq      : 0x0          trill-encap      : 0x0
mim-valid        : 0x0          dtag-ttl         : 0x0
dtag-ftag        : 0x0          valid            : 0x1
erspan-kpa-valid : 0x0          recir-shim-vxlan-src-peer-id: 0x0
vn-valid         : 0x0          source-vif       : 0x0
destination-vif  : 0x0          vn-p             : 0x0
sequence-number  : 0x8d         vl               : 0x3
inner-de-valid   : 0x0          de-cfi          : 0x0
second-inner-cos : 0x0          tunnel-type      : 0x0
shim-valid       : 0x0
segment-id-valid : 0x0          copp            : 0x0
dti-type-vpnid  : 0x0          segment-id       : 0x0
ib-length-bundle : 0x0          mlh-type        : 0x5
ulh-type         : 0x6
source-ipv4-address: 192.168.13.3
destination-ipv4-address: 192.168.12.2
mim-destination-mac-address : 0000.0000.0000
mim-source-mac-address : 0000.0000.0000
destination-mac-address : e4c7.2210.a143

```

source-mac-address : e4c7.2210.a142 Como mostrado, a fonte e os deslocamentos predeterminados do destino são parte do DBUS (desigual que mostrado na captação do ingresso).

```

module-3(fln-l2-elam)# show rbus
cp = 0x100a2548, buf = 0x100a2548, end = 0x100ae898

```

Flanker Instance 00 - Capture Buffer On L2 RBUS:

Status(0x1102), TriggerWord(0x000), SampleStored(0x008), CaptureBufferPointer(0x000)

```

is_l2_egress: 0x0001, data_size: 0x018
[000]: 0048ea00 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 0c000000 00000000 04014008 00005000 00000000
00000726 3910850a 1b931c88 42850800 00000000 00000000 0000008d

```

Printing packet 0

L2 RBUS EGRESS CONTENT

```

pad          : 0x0          valid           : 0x1
trig         : 0x1          reserved        : 0x0
vn-tag-p     : 0x0          cbl-vlan-valid : 0x0
vft-hop-count : 0x0          vft-vsan       : 0x0
vft-up       : 0x0          vft-valid      : 0x0
copp         : 0x0          segment-id-valid : 0x0
segment-id-23 : 0x0          vsl-num        : 0x0
inner-cos    : 0x0          inner-drop-eligibility: 0x0
cos          : 0x0          drop-eligibility : 0x0
dce-mode     : 0x0          flood-to-bd     : 0x0
pt-bit-en   : 0x1          cpu-port        : 0x0
vlan-id      : 0xa          ip-tos          : 0x0
result-rbh   : 0x1          met-ptr         : 0x2000
packet-type  : 0x1          sg-tag          : 0x0
dtag-ftag    : 0x0          vdc             : 0x0
vn-tag-src-vif : 0x0          vn-tag-dst-vif  : 0x0
vn-tag-l     : 0x0          dc3-tr          : 0x0
vl           : 0x0          sequence-number : 0x8d
destination-mac-valid: 0x0
source-mac-valid: 0x0
mim-destination-mac-address : 0000.0000.0000
destination-mac-address : e4c7.2210.a143

```


source-mac-address : e4c7.2210.a142

mim-source-mac-address : 0000.0000.0000 Os endereços IP de origem e de destino estão corretos, como descodificado após a captura de pacote de informação do ingresso ELAM; contudo, o sentido está definidamente oposto a quando comparado ao ingresso ELAM, como o tráfego de retorno é capturado.

10. Verifique o Colorado - a lógica baseada (CBL) para o módulo 3 do Canal de porta 1 em N7K1 a fim determinar se o VLAN10 trafica para a frente através dele. O CBL é relação por-física uma lógica baseada, assim que você deve incorporar o número da interface membro do Canal de porta 1 em N7K1, não o número de Canal de porta. Na saída seguinte, você pode ver esse VLAN10 para a frente ele como esperado.

O CBL é usado a fim determinar o estado do Spanning Tree Protocol (STP) de uma porta dentro do hardware. É possível que a relação mostra a transmissão quando você verifica o STP para ver se há um VLAN no Sup, mas o módulo obstrui o tráfego. Nota: Você deve verificar o CBL individualmente para ver se há ambas as interfaces membro (e3/1 e e3/2).

```
.module-3# show hardware internal mac port 1 table cbl vlan
```

```
-----  
|                                     INGRESS  
| Disabled State   | 0,2-9,11-4031,4036-4095  
| Forwarding State| 1,10,4032-4035  
| Blocked State   |  
| Learning State  |  
-----
```

```
-----  
|                                     EGRESS  
| Disabled State   | 0,2-9,11-4031,4036-4095  
| Forwarding State| 1,10,4032-4035  
| Blocked State   |  
| Learning State  |  
-----
```

Nota: O

comando precedente é para o Canal de porta 1 (o módulo 3 está em e3/1).

```
.module-3# show hardware internal mac port 2 table cbl vlan
```

```
-----  
|                                     INGRESS  
| Disabled State   | 0,2-9,11-4031,4036-4095  
| Forwarding State| 1,10,4032-4035  
| Blocked State   |  
| Learning State  |  
-----
```

```
-----  
|                                     EGRESS  
| Disabled State   | 0,2-9,11-4031,4036-4095  
| Forwarding State| 1,10,4032-4035  
| Blocked State   |  
| Learning State  |  
-----
```

Nota:

Similarmente, este comando verifica o CBL para ver se há o Canal de porta 2 (e3/2).

Pesquise defeitos ELAM nos módulos do F3 Series com cabos breakout

O procedimento ELAM para um módulo do F3 Series quando um cabo breakout é conectado não difere dos procedimentos ELAM em uma porta do módulo regular. Contudo, há algumas mudanças com respeito à verificação do gerente do deslocamento predeterminado de porta (PIXM) durante tentativas de converter o deslocamento predeterminado ao número do painel

dianteiro, neste caso as relações são recebidas do cabo breakout.

Está aqui a topologia que é usada para os exemplos durante todo esta seção:

```
module-3# show hardware internal mac port 2 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-9,11-4031,4036-4095 |  
| Forwarding State | 1,10,4032-4035          |  
| Blocked State   |                          |  
| Learning State  |                          |  
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-9,11-4031,4036-4095 |  
| Forwarding State | 1,10,4032-4035          |  
| Blocked State   |                          |  
| Learning State  |                          |  
-----
```

Para este exemplo, um cabo breakout é conectado à interface Ethernet 3/8, que quebra a porta 40-Gigabit em quatro portas 10-Gigabit. A configuração requerida é fornecida nesta seção para a referência.

```
N7K3(config)# interface breakout module 3 port 8 map 10g-4x
```

```
N7K3(config)# show interface brief
```

```
---SNIP---
```

```
-----  
Ethernet      VLAN    Type Mode  Status Reason  Speed  Port  
Interface                                           Ch #  
-----  
Eth3/7        --     eth  routed up    none   40G(D) --  
Eth3/8/1      1     eth  trunk  up    none   10G(D) 2
```

```
!--- From 3/8/1 to 3/8/4.
```

```
Eth3/8/2      1     eth  trunk  up    none   10G(D) 2  
Eth3/8/3      1     eth  trunk  up    none   10G(D) 2  
Eth3/8/4      1     eth  trunk  up    none   10G(D) 2
```

Na saída precedente, você pode ver que a interface Ethernet 3/7 é ainda uma porta 40-Gigabit; contudo, a interface Ethernet 3/8 é agora quebrada em quatro portas 10-Gigabit, que podem ser configuradas individualmente:

```
N7K3# show run interface e3/8/1 - 4
```

```
!Command: show running-config interface Ethernet3/8/1-4
```

```
!Time: Mon May 4 01:46:28 2015
```

```
version 6.2(8a)
```

```
interface Ethernet3/8/1  
  switchport  
  switchport mode trunk  
  switchport trunk allowed vlan 10,20  
  no shutdown
```

```
interface Ethernet3/8/2  
  switchport  
  switchport mode trunk
```

```
switchport trunk allowed vlan 30,40
no shutdown
```

```
interface Ethernet3/8/3
switchport
switchport mode trunk
switchport trunk allowed vlan 50
no shutdown
```

```
interface Ethernet3/8/4
switchport
switchport mode trunk
no shutdown
```

Comece a captura de pacote de informação do endereço IP de Um ou Mais Servidores Cisco ICM NT 20 do Switched Virtual Interface N7K3 (SVI) (192.168.20.3) 4500 ao endereço IP de Um ou Mais Servidores Cisco ICM NT SVI 20 (192.168.20.1). O pacote será capturado em N7K3 em cima da saída a 4500, e a resposta é enviada de 4500 a N7K3.

Como descrito na seção anterior, você deve ter o conhecimento do exemplo do flanker a fim aplicar o disparador. Esta saída mostra o acessório do módulo 3:

```
N7K3# attach module 3
Attaching to module 3 ...
To exit type 'exit', to abort type '$.'
```

```
module-3# show hardware internal dev
dev-port-map dev-version
module-3# show hardware internal dev-port-map
```

```
-----
CARD_TYPE:          12 port 40G
>Front Panel ports:12
-----
```

```
-----
Device name                Dev role                Abbr num_inst:
-----
>Flanker Eth Mac Driver    DEV_ETHERNET_MAC        MAC_0  6
>Flanker Fwd Driver        DEV_LAYER_2_LOOKUP      L2LKP  6
>Flanker Xbar Driver       DEV_XBAR_INTF           XBAR_INTF 6
>Flanker Queue Driver     DEV_QUEUEING            QUEUE   6
>Sacramento Xbar ASIC     DEV_SWITCH_FABRIC       SWICHF  1
>Flanker L3 Driver        DEV_LAYER_3_LOOKUP      L3LKP  6
>EDC                      DEV_PHY                 PHYS    2
-----
```

```
+-----+
+-----+++FRONT PANEL PORT TO ASIC INSTANCE MAP+++-----+
+-----+
```

FP port	PHYS	MAC_0	L2LKP	L3LKP	QUEUE	SWICHF
1		0	0	0	0	0
2		0	0	0	0	0
3		1	1	1	1	0
4		1	1	1	1	0
5	0	2	2	2	2	0
6	0	2	2	2	2	0
7	1	3	3	3	3	0
8	1	3	3	3	3	0

!--- The port 8 L2LKP column shows a value of 3.

9		4	4	4	4	0
10		4	4	4	4	0
11		5	5	5	5	0
12		5	5	5	5	0

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

Nesta saída, a porta 8 está no exemplo 3. *do* flanker agora que você conhece o exemplo, você pode colocar o disparador através dos endereços IP de origem e de destino. Porque você capturará a solicitação de ping de N7K3 a 4500, será uma saída ELAM.

```
N7K3# attach module 3
Attaching to module 3 ...
To exit type 'exit', to abort type '$.'
```

```
module-3# show hardware internal dev
dev-port-map dev-version
module-3# show hardware internal dev-port-map
```

```
-----+
CARD_TYPE:          12 port 40G
>Front Panel ports:12
```

```
-----+
Device name          Dev role          Abbr num_inst:
-----+
>Flanker Eth Mac Driver DEV_ETHERNET_MAC   MAC_0  6
>Flanker Fwd Driver    DEV_LAYER_2_LOOKUP L2LKP  6
>Flanker Xbar Driver   DEV_XBAR_INTF      XBAR_INTF 6
>Flanker Queue Driver  DEV_QUEUEING       QUEUE    6
>Sacramento Xbar ASIC  DEV_SWITCH_FABRIC  SWICHF  1
>Flanker L3 Driver     DEV_LAYER_3_LOOKUP L3LKP  6
>EDC                   DEV_PHY            PHYS    2
```

```
-----+
+-----+++FRONT PANEL PORT TO ASIC INSTANCE MAP+++-----+
-----+
FP port |  PHYS |  MAC_0 |  L2LKP |  L3LKP |  QUEUE | SWICHF
```

FP port	PHYS	MAC_0	L2LKP	L3LKP	QUEUE	SWICHF
1		0	0	0	0	0
2		0	0	0	0	0
3		1	1	1	1	0
4		1	1	1	1	0
5	0	2	2	2	2	0
6	0	2	2	2	2	0
7	1	3	3	3	3	0
8	1	3	3	3	3	0

```
!--- The port 8 L2LKP column shows a value of 3.
```

9		4	4	4	4	0
10		4	4	4	4	0
11		5	5	5	5	0
12		5	5	5	5	0

```
-----+
O sibilo é iniciado de N7K3 a 4500:
```

```
N7K3# ping 192.168.20.1
PING 192.168.20.1 (192.168.20.1): 56 data bytes
36 bytes from 192.168.20.3: Destination Host Unreachable
Request 0 timed out
64 bytes from 192.168.20.1: icmp_seq=1 ttl=254 time=6.49 ms
64 bytes from 192.168.20.1: icmp_seq=2 ttl=254 time=6.518 ms
64 bytes from 192.168.20.1: icmp_seq=3 ttl=254 time=7.936 ms
64 bytes from 192.168.20.1: icmp_seq=4 ttl=254 time=7.945 ms
```

```
--- 192.168.20.1 ping statistics ---
5 packets transmitted, 4 packets received, 20.00% packet loss
round-trip min/avg/max = 6.49/7.222/7.945 ms
```

```
Está aqui o estado ELAM:
```

```

module-3(fln-l2-elam)# status
ELAM Slot 3 instance 3: L2 DBUS Configuration: trigger dbus ipv4 egress if
source-ipv4-address 192.168.20.3 destination-ipv4-address 192.168.20.1
L2 DBUS: Triggered
ELAM Slot 3 instance 3: L2 RBUS Configuration: trigger rbus egress if trig
L2 RBUS: Triggered

```

Verifique que os números de sequência são os mesmos:

```

module-3(fln-l2-elam)# show dbus | in seq
sequence-number      : 0x27          vl          : 0x3
module-3(fln-l2-elam)# show rbus | in seq
vl                   : 0x0          sequence-number : 0x27

```

Os números de sequência são os mesmos. Agora você pode verificar a informação do DBUS e RBUS:

```

module-3(fln-l2-elam)# show dbus
cp = 0x1011033c, buf = 0x1011033c, end = 0x1011c68c

```

Flanker Instance 03 - Capture Buffer On L2 DBUS:

Status(0x0102), TriggerWord(0x000), SampleStored(0x004),CaptureBufferPointer(0x004)

```

is_l2_egress: 0x0000, data_size: 0x023
[000]: 4c1ea000 20a10000 40021040 0cc02801 04080000 00000000 08100000 00000000
00000000 00000000 003c1fc1 8732dff9 31c88428 51000000 009d8000 00000000 00000000
00000000 00000000 00000000 00000000 00005000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 0060540a 01e0540a 0080008f f0054608 00000000

```

Printing packet 0

```

-----
L2 DBUS PRS MLH IPV4
-----
label-count          : 0x0          mc          : 0x0
null-label-valid    : 0x0          null-label-exp : 0x0
null-label-ttl      : 0x0          lb10-vld     : 0x0
lb10-eos            : 0x0          lb10-lb1    : 0x0
lb10-exp            : 0x0          lb10-ttl    : 0x0
lb11-exp            : 0x0          lb11-ttl    : 0x0
ipv4                 : 0x0          ipv6         : 0x0
l4-protocol         : 0x1          df           : 0x0
mf                   : 0x0          frag        : 0x0
ttl                  : 0xff         l3-packet-length : 0x54
option              : 0x0          tos         : 0x0
sup-eid              : 0x1          header-type  : 0x0
error                : 0x0          redirect    : 0x0
port-id              : 0x5          last-ethertype : 0x800
l2-frame-type       : 0x0          da-type     : 0x0
packet-type         : 0x1          l2-length-check : 0x0
ip-da-multicast     : 0x0          ip-multicast : 0x0
ip-multicast-control : 0x0          ids-check-fail : 0x0
traceroute          : 0x0          outer-cos   : 0x0
inner-cos           : 0x0          vqi-valid   : 0x1
vqi                  : 0x82         packet-length : 0x66
vlan                : 0x14         destination-index : 0x82
source-index        : 0x400         bundle-port       : 0x0
acos                : 0x0          outer-drop-eligibility: 0x0
inner-drop-eligibility: 0x0          sg-tag      : 0x0
rbh                  : 0x0          vsl-num     : 0x0
inband-flow-creation-deletion: 0x0          ignore-qoso : 0x0
ignore-qosi          : 0x0          ignore-aclo : 0x0
ignore-acli          : 0x0          index-direct : 0x1

```

```

no-stats          : 0x0          dont-forward      : 0x0
notify-index-learn : 0x0          notify-new-learn  : 0x0
disable-new-learn  : 0x0          disable-index-learn : 0x0
dont-learn        : 0x1          bpdu              : 0x0
ff                : 0x0          rf                : 0x0
ccc               : 0x0          l2                : 0x0
rdt               : 0x0          dft               : 0x0
dfst              : 0x0          status-ce-lq      : 0x0
status-is-lq      : 0x0          trill-encap       : 0x0
mim-valid         : 0x0          dtag-ttl          : 0x0
dtag-ftag         : 0x0          valid             : 0x1
erspan-kpa-valid  : 0x0          recir-shim-vxlan-src-peer-id: 0x0
vn-valid          : 0x0          source-vif        : 0x0
destination-vif   : 0x0          vn-p              : 0x0
sequence-number   : 0x27         vl                : 0x3
inner-de-valid    : 0x0          de-cfi            : 0x0
second-inner-cos  : 0x0          tunnel-type       : 0x0
shim-valid        : 0x0
segment-id-valid  : 0x0          copp              : 0x0
dti-type-vpnid   : 0x0          segment-id        : 0x0
ib-length-bundle  : 0x0          mlh-type          : 0x5
ulh-type          : 0x6
source-ipv4-address: 192.168.20.3
destination-ipv4-address: 192.168.20.1
mim-destination-mac-address : 0000.0000.0000
mim-source-mac-address  : 0000.0000.0000
destination-mac-address : f07f.061c.cb7f
source-mac-address  : e4c7.2210.a144

```

```

module-3(fln-l2-elam)#
module-3(fln-l2-elam)#
module-3(fln-l2-elam)#
module-3(fln-l2-elam)# show rbus
cp = 0x10134d38, buf = 0x10134d38, end = 0x10141088

```

Flanker Instance 03 - Capture Buffer On L2 RBUS:

Status(0x1102), TriggerWord(0x000), SampleStored(0x008),CaptureBufferPointer(0x000)

```

is_l2_egress: 0x0001, data_size: 0x018
[000]: 004c4780 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 0c001000 00000000 80028010 00009000 00000000
00000783 f830e65b fb931c88 42851000 00000000 00000000 00000027

```

Printing packet 0

L2 RBUS EGRESS CONTENT

```

pad          : 0x0          valid             : 0x1
trig         : 0x1          reserved          : 0x0
vn-tag-p     : 0x1          cbl-vlan-valid   : 0x0
vft-hop-count : 0x0          vft-vsant        : 0x0
vft-up       : 0x0          vft-valid        : 0x0
copp         : 0x0          segment-id-valid  : 0x0
segment-id-23 : 0x0          vsl-num          : 0x0
inner-cos    : 0x0          inner-drop-eligibility: 0x0
cos          : 0x0          drop-eligibility : 0x0
dce-mode     : 0x0          flood-to-bd      : 0x0
pt-bit-en   : 0x20         cpu-port         : 0x0
vlan-id      : 0x14         ip-tos           : 0x0
result-rbh   : 0x2          met-ptr          : 0x4000
packet-type  : 0x1          sg-tag           : 0x0

```

```

dtag-ftag          : 0x0          vdc                : 0x0
vn-tag-src-vif     : 0x0          vn-tag-dst-vif     : 0x0
vn-tag-l           : 0x0          dc3-tr             : 0x0
vl                 : 0x0          sequence-number    : 0x27
destination-mac-valid: 0x0
source-mac-valid: 0x0
mim-destination-mac-address : 0000.0000.0000
destination-mac-address : f07f.061c.cb7f
source-mac-address : e4c7.2210.a144
mim-source-mac-address : 0000.0000.0000

```

Converta os deslocamentos predeterminados do destino e de fonte em portas do painel dianteiro a fim confirmar o fluxo:

```

N7K3# show system internal pixm info ltl 0x400
0x0400 is in SUP In-band LTL range

```

Esta saída mostra o deslocamento predeterminado de fonte. Você sabe que está correta devido aos sibilos que vêm a N7K3 do Sup. A saída seguinte mostra a interface de saída (e3/8/1), que é uma das duas relações no N7K que permite o VLAN20. A outra relação é e3/8/4, que é obstruído em 4500 devido ao STP.

```

N7K3# show system internal pixm info ltl 0x82
0x0082 is in DCE/FC pool

```

Member info

```

-----
Type          LTL
-----
PHY_PORT      Eth3/8/1
FLOOD_W_FPOE  0x8039
FLOOD_W_FPOE  0x803f

```

Verifique o CBL para as portas que foram criadas com o cabo breakout no N7K. A fim verificar o CBL, você deve ter os números de porta de hardware para todas as portas novo-formadas.

Nota: A relação e3/8 não existe no interruptor. Somente as portas novo-formadas aparecem.

```

N7K3# show interface e3/8
^
% Incomplete command at '^' marker.
N7K3#

```

Porque o cabo breakout é usado e a relação e3/8 é inexistente no interruptor, o cálculo que é usado a fim obter o número de porta de hardware muda. Para todo o módulo que apoiar a fuga, a numeração da porta de hardware é diferente. Você deve primeira verificação se uma porta apoia a fuga:

```

N7K3# show int e3/7 capabilities
Ethernet3/7
  Model:          N7K-F312FQ-25
  Type (SFP capable): QSFP-40G-CR4
  Speed:          10000,40000
  Duplex:         full
  ---SNIP---
  PFC capable:    yes
  Breakout capable: yes

```

Como mostrado, a porta e3/7 apoia a fuga, assim que significa que sua largura de banda pode se quebrar em quatro portas 10-Gigabit. Similarmente, outros módulos do F3 Series que têm as portas 100-Gigabit podem quebrar-se em dez movem cada um com 10-Gigabits, ou três portas com o 40-Gigabits com sobreassinatura. Isto depende do módulo.

Desde que o módulo do F3 Series neste exemplo tem as portas 40-Gigabit, e cada porta pode se quebrar em quatro move cada um, os números de porta de hardware para cada porta são 0-3, 4-7, 8-11...40-43, 44-47 em uma escala zero-baseada. Se você tem um cabo breakout em uma porta para o primeiro exemplo, a seguir sua numeração da porta de hardware será 0, 1, 2 e 3. Se você não tem um cabo breakout, a seguir sua numeração da porta de hardware será 0 (1, 2, e 3 não será ativos).

Porque a porta do pai é e3/8, seu número de porta de hardware será 28 se é usado sem o cabo breakout, e será 28, 29, 30, e 31 se é usado com o cabo breakout. Esta saída do comando mostra as portas de hardware ativas (zero baseado):

```
N7K3# show system internal ifindex info mod 3
```

```
Init DB dump follows:
```

```
module_num_bitmask = 0x3ffff
```

```
Slot:3, Proc:1, breakout_factor:0, sw_card_id:0, active_cfg_ports:, broken_fp_ports:
```

```
Slot:3, Proc:2, breakout_factor:4, sw_card_id:155, active_cfg_ports:0,4,8,12,16,20,24,28-32,36,40,44, broken_fp_ports:28
```

```
Lookup DB dump follows:
```

```
Slot:3, breakout_factor:4
```

O número de porta de hardware quebrado da porta é **28**, que é rachado agora em quatro (28-32). Agora você pode anexar o módulo 3 e verificar o CBL no hardware:

```
N7K3# attach module 3
```

```
Attaching to module 3 ...
```

```
To exit type 'exit', to abort type '$.'
```

```
module-3#
```

O módulo do F3 Series espera o número de porta ser formatado de acordo com uma escala um-baseada. Por este motivo, você deve incorporar 29, 30, 31, e 32:

```
module-3# show hardware internal mac port ?
```

```
<1-96> Port number (1-based)
```

```
!--- This is context sensitive, so it helps to say the port number is 1-based.
```

Está aqui a configuração running para a interface Ethernet 3/8/1 a fim verificar e confirmar o estado de encaminhamento VLAN:

```
module-3# show hardware internal mac port ?
```

```
<1-96> Port number (1-based)
```

```
!--- This is context sensitive, so it helps to say the port number is 1-based.module-3# show hardware internal mac port 29 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-9,11-19,21-4031,4036-4095 |  
| Forwarding State | 10,20,4032-4035 |  
| Blocked State   | 1 |  
| Learning State  | |  
-----
```

```
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-9,11-19,21-4031,4036-4095 |  
| Forwarding State | 10,20,4032-4035 |  
| Blocked State   | 1 |  
| Learning State  | |  
-----
```

Está aqui a configuração running para a interface Ethernet 3/8/2 a fim verificar e confirmar o

estado de encaminhamento VLAN:

```
module-3# show hardware internal mac port 29 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-9,11-19,21-4031,4036-4095 |  
| Forwarding State| 10,20,4032-4035                |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

```
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-9,11-19,21-4031,4036-4095 |  
| Forwarding State| 10,20,4032-4035                |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

-----module-3# show

```
hardware internal mac port 30 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-29,31-39,41-4031,4036-4095 |  
| Forwarding State| 30,40,4032-4035                |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

```
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-29,31-39,41-4031,4036-4095 |  
| Forwarding State| 30,40,4032-4035                |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

Está aqui a configuração running para a interface Ethernet 3/8/3 a fim verificar e confirmar o estado de encaminhamento VLAN:

```
module-3# show hardware internal mac port 30 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-29,31-39,41-4031,4036-4095 |  
| Forwarding State| 30,40,4032-4035                |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

```
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-29,31-39,41-4031,4036-4095 |  
| Forwarding State| 30,40,4032-4035                |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

-----module-3# show

```
hardware internal mac port 31 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-49,51-4031,4036-4095 |  
| Forwarding State| 50,4032-4035                  |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

```
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-49,51-4031,4036-4095 |  
| Forwarding State| 50,4032-4035                  |  
| Blocked State   | 1                               |  
| Learning State  |                               |  
-----
```

| Learning State |

Está aqui a configuração running para a interface Ethernet 3/8/4 a fim verificar e confirmar o estado de encaminhamento VLAN (todos os VLAN configurados são permitidos):

```
module-3# show hardware internal mac port 31 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-49,51-4031,4036-4095 |  
| Forwarding State | 50,4032-4035             |  
| Blocked State    | 1                         |  
| Learning State   |                           |  
-----
```

```
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-49,51-4031,4036-4095 |  
| Forwarding State | 50,4032-4035             |  
| Blocked State    | 1                         |  
| Learning State   |                           |  
-----
```

-----module-3# show

```
hardware internal mac port 32 table cbl vlan
```

```
-----  
|                                     INGRESS                                     |  
| Disabled State   | 0,2-9,11-19,21-29,31-39,41-49,51-59,61-669,671-4031 |  
| Disabled State   | 4036-4095                 |  
| Forwarding State | 1,20,30,40,50,60,670,4032-4035 |  
| Blocked State    | 10                         |  
| Learning State   |                           |  
-----
```

```
-----  
|                                     EGRESS                                     |  
| Disabled State   | 0,2-9,11-19,21-29,31-39,41-49,51-59,61-669,671-4031 |  
| Disabled State   | 4036-4095                 |  
| Forwarding State | 1,20,30,40,50,60,670,4032-4035 |  
| Blocked State    | 10                         |  
| Learning State   |                           |  
-----
```

O CBL mostra que os VLAN corretos estão enviados.

Você pode usar o comando do **number**> do <module do módulo do erro interno do hardware da mostra a fim obter o número de porta de hardware. Este comando é útil quando você deve verificar todas as gotas internas que não aparecerem na saída do comando do x/y da relação da mostra. Aqui está um exemplo:

```
N7K2# show hardware internal errors module 3
```

```
---SNIP---
```

```
Instance:1
```

Cntr	Name	Value	Ports
3836	igr rx pl: cbl drops	0000000000000001	10 -
4636	igr rx pl: cbl drops	0000000000000001	14 -

```
Instance:2
```

Cntr	Name	Value	Ports
423	igr in upm: pkts with symbol/sequence error rcvd	0000000000000478	18 -
455	igr in upm: pkts with symbol/sequence error rcvd	0000000000000478	17 -
487	igr in upm: pkts with symbol/sequence error rcvd	0000000000000478	19 -
519	igr in upm: pkts with symbol/sequence error rcvd	0000000000000478	20 -

Instance:3

Cntr	Name	Value	Ports
423	igr in upm: pkts with symbol/sequence error rcvd	0000000000000745	26 -
455	igr in upm: pkts with symbol/sequence error rcvd	0000000000000745	25 -
487	igr in upm: pkts with symbol/sequence error rcvd	0000000000000745	27 -
519	igr in upm: pkts with symbol/sequence error rcvd	0000000000000745	28 -
550	igr in upm: pkts rcvd, with RCODE violation	0000359810913821	30 -
551	igr in upm: pkts with symbol/sequence error rcvd	0000425092490108	30 -
552	igr in upm: pkts with error	0000000000176136	30 -
582	igr in upm: pkts rcvd, with RCODE violation	0000000000292641	29 -
583	igr in upm: pkts with symbol/sequence error rcvd	0000000000114014	29 -
614	igr in upm: pkts rcvd, with RCODE violation	0000133362265995	31 -
615	igr in upm: pkts with symbol/sequence error rcvd	0000146701474013	31 -
616	igr in upm: pkts with error	0000000000157479	31 -
646	igr in upm: pkts rcvd, with RCODE violation	0000000002160959	32 -
647	igr in upm: pkts with symbol/sequence error rcvd	0000000003722562	32 -
648	igr in upm: pkts with error	0000000000000002	32 -