

Dépannage des problèmes de transfert matériel sur les commutateurs de la gamme Nexus 7000

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Informations générales](#)

[Dépannage](#)

[Dépannage d'ELAM sur les modules de la gamme F3 sans câbles de dérivation](#)

[Dépannage d'ELAM sur les modules de la gamme F3 avec câbles séparés](#)

Introduction

Ce document décrit comment résoudre les problèmes de transfert matériel sur les modules de la gamme F3 pour les commutateurs Cisco Nexus 7000.

Conditions préalables

Conditions requises

Cisco vous recommande de vous familiariser avec le système d'exploitation Cisco Nexus (NX-OS) et l'architecture Nexus de base avant de continuer avec les informations décrites dans ce document.

Components Used

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Commutateurs Cisco Nexus 7000 (N7K)
- Modules de la gamme Cisco N7K F3 (modules N7K-F312FQ-25, 12 ports 10/40 Gigabit Ethernet)
- Cisco NX-OS versions 6.2.8a et ultérieures

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

Ce document se concentre principalement sur certains des outils intégrés qui sont utilisés pour le dépannage matériel lorsque vous avez épuisé la partie logicielle de la table de transfert ou du plan de contrôle. Un de ces outils est le module ELAM (Embedded Logic Analyzer Module), qui est un circuit intégré spécifique à l'application (ASIC) qui capture un seul paquet et montre comment le paquet d'entrée apparaît sur le DBUS (Data BUS) et le BUS de résultat (RBUS) après transmission.

L'ASIC est intégré au pipeline de transfert et peut capturer un paquet en temps réel sans perturber les performances ou les ressources du plan de contrôle. Cela permet de répondre à des questions telles que :

- Le paquet a-t-il atteint le moteur de transfert (FE) ?
- Sur quel port et quel VLAN le paquet est-il reçu ?
- Comment le paquet apparaît-il (données de couche 2 (L2) ou de couche 4 (L4)) ?
- Comment le paquet est-il modifié et où est-il envoyé ?

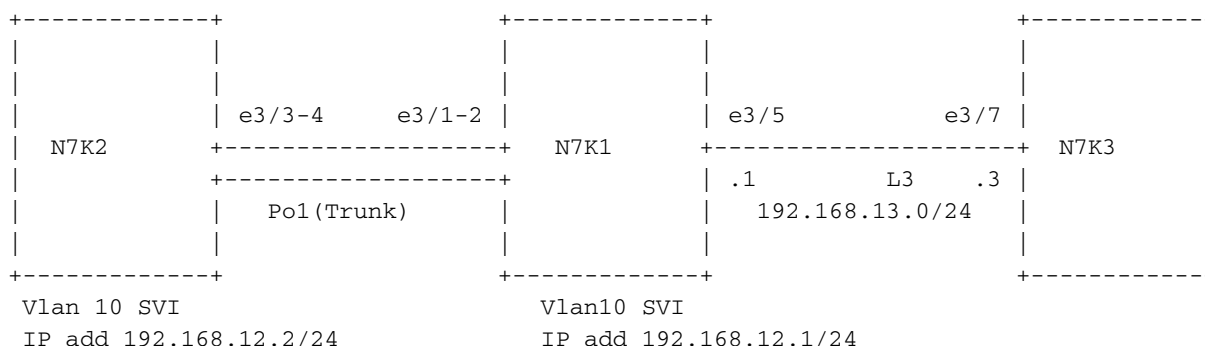
L'ELAM est un outil puissant, granulaire et non intrusif qui est le plus souvent utilisé par les ingénieurs du centre d'assistance technique Cisco (TAC) qui travaillent sur des plates-formes de commutation matérielle. Cependant, il est important de savoir que l'outil ELAM capture un seul paquet à la fois. En d'autres termes, le premier paquet reçu après le déclenchement de l'ELAM.

Dépannage

Cette section décrit comment dépanner ELAM sur un module de la gamme F3 dans les déploiements qui n'impliquent pas l'utilisation d'un câble de dérivation, ainsi que dans les déploiements qui utilisent des câbles de dérivation.

Dépannage d'ELAM sur les modules de la gamme F3 sans câbles de dérivation

Voici la topologie utilisée pour les exemples de cette section :



Voici quelques notes sur cette topologie :

- Les N7Ks exécutent NX-OS Version 6.2.8a.
- Les requêtes ping sont envoyées de l'interface VLAN 10 N7K2 à une adresse IP distante 192.168.12.1.

- L'ELAM capture les paquets sur le N7K1.
- Un N7K-F312FQ-25 est utilisé, qui est un module 12 ports 10/40 Gigabit Ethernet inséré dans le logement 3.

Avant de commencer à dépanner votre système, vous devez confirmer la connectivité de base :

```
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
```

Vous devez également vérifier l'apprentissage des adresses MAC (Media Access Control) sur le moteur de supervision (Sup) et le module pour le saut suivant :

```
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 | Pol |

Ce résultat montre l'apprentissage MAC sur le module/le matériel ; cependant, pour connaître l'interface, vous devez convertir l'index :

```
N7K2# show hardware mac address-table 3 address e4c7.2210.a142
FE | Valid| PI|  BD |      MAC      | Index | Stat| SW  | Modi| Age| Tmr| GM| Sec|
  |     |  |   |             |       |    |   |   | fied|Byte| Sel|   | ure|
```

```

---|-----|---|-----|-----|-----|-----|-----|-----|---|---|
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---

Entrez ces commandes afin d'obtenir le numéro VDC (Virtual Device Context) (dans cet exemple, il est 3) et vérifiez l'adresse MAC directement sur le module :

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 |

Déterminez la liaison sur le canal de port 1 qui est utilisée pour transférer le trafic sur le Sup à partir de N7K2, ainsi que la liaison qui est utilisée pour envoyer une réponse de N7K3 lorsque le canal de port 1 est utilisé de N7K1 à 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

Envoyez une requête ping à partir de N7K2 (adresse IP 192.168.12.2) et capturez les paquets sur N7K1 dans la direction d'entrée afin de confirmer que les paquets sont transférés vers N7K3 (adresse IP 192.168.13.3).

Avant d'envoyer la requête ping, vous devez connaître l'accumulation de matériel. Complétez ces étapes afin de comprendre l'accumulation :

1. Fixez le module :

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

2. Identifiez l'instance *de flanker*. Le module flanker est un ASIC de commutation sur puce (SOC) pour le module de la gamme F3. Chaque module de liaison est mappé à deux ports externes sur le module (les informations changent par type de module et sont spécifiques au N7K-F312FQ-25).

Il y a 12 ports sur le module, et chaque ASIC correspond à deux ports sur la façade, ce qui signifie qu'il y a 6 (0-5) instances de flanker disponibles sur le module (le nombre d'instances est basé sur zéro). **Note:** Vérifiez que vous disposez des privilèges d'administration réseau avant de commencer. Lorsque vous capturez le paquet qui arrive de N7K2 via le canal de port 1 sur N7K1, recherchez les ports (e3/1 et e3/2) qui sont mappés à chaque instance :

```
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 |

!--- The L2KLP for both ports is 0, so both belong to instance 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 |
| 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. Sélectionnez l'instance, définissez le déclencheur et commencez la capture. Il est toutefois important de comprendre que de nombreuses options peuvent être utilisées avec le déclencheur 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-----

Ces deux options sont importantes si vous voulez inclure le DBUS dans la capture (le paquet reçu par le commutateur). Il s'agit du paquet brut qui n'est pas soumis à une recherche. Le RBUS affiche les résultats de recherche dans le matériel d'un DBUS. Pour une analyse et un ELAM complets, vous devez capturer à la fois le RBUS et le DBUS.

Le résultat suivant montre les types de paquets que vous pouvez capturer avec l'option DBUS. Dans cet exemple, le paquet IPv4 (Internet Protocol Version 4) est sélectionné :

```

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

```

Voici quelques options supplémentaires que vous pouvez choisir d'utiliser :

```

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

```

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

```

if                If Trigger Condition

```


| | |
|---------------------------------|---------------------------------|
| 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 |
| vl | Vl |
| vlan | Vlan |
| vn-p | Vn p |
| vn-valid | Vn-valid |
| vqi | Vqi |
| vqi-valid | Vqi-valid |
| vsl-num | Vsl-num |

Cette sortie montre l'option de déclenchement final :

```

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

```

Note: La configuration RBUS n'est généralement pas complexe et reste simple.

4. Afin de vérifier le déclencheur, entrez la commande **status**, lancez le processus de capture et lancez une requête ping de N7K2 vers N7K3 (192.168.12.1 à 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. Si l'état indique **Déclenché**, vérifiez si le RBUS et le DBUS ont tous deux le même numéro de séquence afin de confirmer qu'ils sont pour le même paquet. Dans cet exemple, **0x55** est utilisé, mais la colonne qui montre le numéro de séquence est différente :

```
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. Entrez les commandes **show dbus** et **show rbus** afin de vérifier le DBUS et le RBUS. Recherchez l'*index source* dans le résultat de la commande DBUS et l'*index de destination* dans le résultat de la commande RBUS :

```
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-lbl          : 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              13-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-1q       : 0x0
status-is-1q         : 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 | di1-v5-delta-length | : 0x0 |
| di1-v5-delta-length-plus | : 0x0 | di1-v4-delta-length | : 0x0 |
| di1-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 | di1-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. Vérifiez l'index de destination et source sur le 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

```

Ce résultat confirme que le paquet a été reçu sur le canal de port 1 (**Po1**) et transféré via **Eth3/5**.

8. Vérifiez la logique de cible locale (LTL) sur le module pour une programmation correcte :

```

module-3# show system internal pixm info ltl-cb ltl 0xa2c
  ltl |ltl_type|if_index|lc_type| vdc |v4_fpoe|v5_fpoe| base_fpoe_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. Capturez le paquet ELAM en sortie. Afin de capturer le paquet, envoyez une réponse ping de l'adresse IP 192.168.13.3 à 192.168.12.2. Vous devez définir la capture avec le mot clé **de sortie** sur les interfaces du canal de port 1 (e3/1-2). Les interfaces appartiennent à l'instance **0**, comme décrit précédemment.

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-12-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-12-elam)# start
module-3(fln-12-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-12-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-12-elam)#
```

```
module-3(fln-12-elam)# show dbus | in seq
sequence-number      : 0x8d          vl          : 0x3
```

!--- The sequence number is the same.

```
module-3(fln-12-elam)# show rbus | in seq
vl                   : 0x0          sequence-number    : 0x8d
```

```
module-3(fln-12-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          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                  : 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-1q  : 0x0
status-is-1q : 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

```

Comme indiqué, les index source et de destination font tous deux partie du DBUS (contrairement à ce qui est indiqué dans la capture d'entrée).

```
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          vs1-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
```

Les adresses IP source et de destination sont correctes, comme décodées après la capture de paquets ELAM d'entrée ; cependant, la direction est incontestablement opposée par rapport à l'ELAM d'entrée, car le trafic de retour est capturé.

10. Vérifiez la logique basée sur les couleurs (CBL) pour le module 3 du canal de port 1 sur N7K1 afin de déterminer si VLAN 10 transfère le trafic à travers celui-ci. La CBL est une logique basée sur l'interface physique. Vous devez donc saisir le numéro d'interface membre du canal de port 1 sur N7K1, et non le numéro de canal de port. Dans le résultat suivant, vous pouvez voir que le VLAN 10 le transmet comme prévu.

La CBL est utilisée afin de déterminer l'état STP (Spanning Tree Protocol) d'un port au sein du matériel. Il est possible que l'interface affiche le transfert lorsque vous vérifiez le STP pour un VLAN sur le Sup, mais le module bloque le trafic. **Note:** Vous devez vérifier la liste de contrôle de trame individuellement pour les deux interfaces membres (e3/1 et 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  |                          |  
-----
```

Note: La commande précédente concerne le port channel 1 (le module 3 est sur 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 |
|-----

```

Note: De même, cette commande vérifie la CBL pour le canal de port 2 (e3/2).

Dépannage d'ELAM sur les modules de la gamme F3 avec câbles séparés

La procédure ELAM pour un module de la gamme F3 lorsqu'un câble de dérivation est connecté ne diffère pas des procédures ELAM sur un port de module normal. Cependant, des modifications sont apportées à la vérification du gestionnaire d'index de port (PIXM) lors des tentatives de conversion de l'index en numéro de panneau avant, auquel cas les interfaces sont reçues à partir du câble de dérivation.

Voici la topologie utilisée pour les exemples de cette section :

```

+-----+ BreakOut Cable +-----+
|         | e3/8/1-4 te1/1 |         |
|         | +-----+         |
| N7K3    | |e3/8      | 1/2 | 4500-X |
|         | +-----+         |
|         | Po2        | 1/3 |         |
|         | +-----+         |
|         |          1/4 |         |
+-----+ +-----+
Vlan20 SVI                               Vlan20 SVI
IP add 192.168.20.3                       IP add 192.168.20.1

```

Dans cet exemple, un câble de dérivation est connecté à l'interface Ethernet 3/8, qui divise le port 40 Gigabit en quatre ports 10 Gigabit. La configuration requise est fournie dans cette section à titre de référence.

```
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
-----
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

```

Dans la sortie précédente, vous pouvez voir que l'interface Ethernet 3/7 est toujours un port 40 Gigabit ; cependant, l'interface Ethernet 3/8 est maintenant divisée en quatre ports 10 Gigabit, qui peuvent être configurés individuellement :

```
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
```

Commencez la capture de paquets de l'adresse IP 20 de l'interface virtuelle commutée N7K3 (SVI) (192.168.20.3) vers l'adresse IP 4500 SVI 20 (192.168.20.1). Le paquet sera capturé sur N7K3 en sortie vers 4500 et la réponse sera envoyée de 4500 à N7K3.

Comme décrit dans la section précédente, vous devez connaître l'instance de flanker pour appliquer le déclencheur. Ce résultat montre la pièce jointe du module 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 |

Dans cette sortie, le port 8 se trouve sur l'instance flanker 3. Maintenant que vous connaissez l'instance, vous pouvez placer le déclencheur via les adresses IP source et de destination. Comme vous allez capturer la requête ping de N7K3 à 4500, il s'agira d'un ELAM de sortie.

```

module-3# elam asic flanker instance 3
module-3(fln-elam)# layer2
module-3(fln-l2-elam)# trigger dbus ipv4 egress if source-ipv4-address 192.168.20.3
destination-ipv4-address 192.168.20.1
module-3(fln-l2-elam)# trigger rbus egress if trig

```

```

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: Configured
ELAM Slot 3 instance 3: 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 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: Armed
ELAM Slot 3 instance 3: L2 RBUS Configuration: trigger rbus egress if trig
L2 RBUS: Armed

```

La requête ping est lancée de N7K3 à 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

Voici le statut ELAM :

```
module-3(fln-12-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
```

Vérifiez que les numéros de séquence sont identiques :

```
module-3(fln-12-elam)# show dbus | in seq
```

```
sequence-number      : 0x27          vl          : 0x3
```

```
module-3(fln-12-elam)# show rbus | in seq
```

```
vl                   : 0x0          sequence-number : 0x27
```

Les numéros de séquence sont identiques. Vous pouvez maintenant vérifier les informations DBUS et RBUS :

```
module-3(fln-12-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          vs1-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-1q        : 0x0
status-is-1q         : 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          copp                 : 0x0
segment-id-valid     : 0x0          segment-id           : 0x0
dti-type-vpnid       : 0x0          mlh-type             : 0x5
ib-length-bundle     : 0x0
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-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        : 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

```

Convertissez les index de destination et source en ports de la façade afin de confirmer le flux :

```

N7K3# show system internal pixm info lt1 0x400
0x0400 is in SUP In-band LTL range

```

Ce résultat montre l'index source. Vous savez qu'il est correct à cause des requêtes ping qui arrivent à N7K3 à partir du Sup. La sortie suivante montre l'interface de sortie (e3/8/1), qui est l'une des deux interfaces sur le N7K qui autorise VLAN 20. L'autre interface est e3/8/4, qui est bloquée sur 4500 en raison du protocole STP.

```

N7K3# show system internal pixm info lt1 0x82
0x0082 is in DCE/FC pool

```

Member info

```

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

```

Vérifiez la CBL pour les ports qui ont été créés avec le câble de dérivation sur le N7K. Pour vérifier la liste de contrôle de trame, vous devez disposer des numéros de port matériel pour tous les ports nouvellement formés.

Note: L'interface e3/8 n'existe pas sur le commutateur. Seuls les ports nouvellement formés apparaissent.

```

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

```

Comme le câble de dérivation est utilisé et que l'interface e3/8 n'existe pas sur le commutateur, le calcul utilisé pour obtenir les modifications du numéro de port matériel est modifié. Pour tout module prenant en charge le découpage, la numérotation des ports matériels est différente. Vous

devez d'abord vérifier si un port prend en charge le découpage :

```
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
```

Comme indiqué, le port e3/7 prend en charge le découpage, ce qui signifie que sa bande passante peut être divisée en quatre ports 10 Gigabit. De même, d'autres modules de la gamme F3 dotés de ports 100 Gigabit peuvent être divisés en dix ports chacun avec 10 Gigabits, ou trois ports avec 40 Gigabits avec surabonnement. Cela dépend du module.

Puisque le module de la gamme F3 dans cet exemple comporte 40 ports Gigabit et que chaque port peut être divisé en quatre ports chacun, les numéros de port matériel pour chaque port sont 0-3, 4-7, 8-11...40-43, 44-47 sur une échelle zéro. Si vous avez un câble de dérivation sur un port pour le premier exemple, sa numérotation de port matériel sera 0, 1, 2 et 3. Si vous ne disposez pas d'un câble de dérivation, sa numérotation de port matériel sera 0 (1, 2 et 3 ne seront pas actifs).

Comme le port parent est e3/8, son numéro de port matériel sera 28 s'il est utilisé sans le câble de dérivation et 28, 29, 30 et 31 s'il est utilisé avec le câble de dérivation. Cette sortie de commande affiche les ports matériels actifs (basés sur zéro) :

```
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
```

Le numéro de port matériel du port défectueux est **28**, qui est maintenant divisé en quatre (28-32). Maintenant, vous pouvez connecter le module 3 et vérifier la CBL dans le matériel :

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

Le module de la gamme F3 s'attend à ce que le numéro de port soit formaté selon une échelle unique. Pour cette raison, saisissez 29, 30, 31 et 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.

Voici la configuration en cours pour l'interface Ethernet 3/8/1 afin de vérifier et de confirmer l'état

de transmission VLAN :

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

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

Voici la configuration en cours pour l'interface Ethernet 3/8/2 afin de vérifier et de confirmer l'état de transmission VLAN :

```
interface Ethernet3/8/2
switchport
switchport mode trunk
switchport trunk allowed vlan 30,40
no shutdown
```

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

Voici la configuration en cours pour l'interface Ethernet 3/8/3 afin de vérifier et de confirmer l'état de transfert VLAN :

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

```
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 | |

Voici la configuration en cours pour l'interface Ethernet 3/8/4 afin de vérifier et de confirmer l'état de transfert VLAN (tous les VLAN configurés sont autorisés) :

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

```
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 | |

La liste CBL indique que les VLAN corrects sont transférés.

Vous pouvez utiliser la commande **show hardware internal error module <module number>** afin d'obtenir le numéro de port matériel. Cette commande est utile lorsque vous devez vérifier les pertes internes qui n'apparaissent pas dans la sortie de commande **show interface x/y**. Voici un exemple :

```
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 | 00000000000000745 | 26 - |
| 455 | igr in upm: pkts with symbol/sequence error rcvd | 00000000000000745 | 25 - |
| 487 | igr in upm: pkts with symbol/sequence error rcvd | 00000000000000745 | 27 - |
| 519 | igr in upm: pkts with symbol/sequence error rcvd | 00000000000000745 | 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 - |