

Procédure du module ELAM de M-gamme de Nexus 7000

Contenu

[Introduction](#)

[Topologie](#)

[Déterminez l'engine d'expédition d'entrée](#)

[Configurez le déclencheur](#)

[Commencez la capture](#)

[Interprétez les résultats](#)

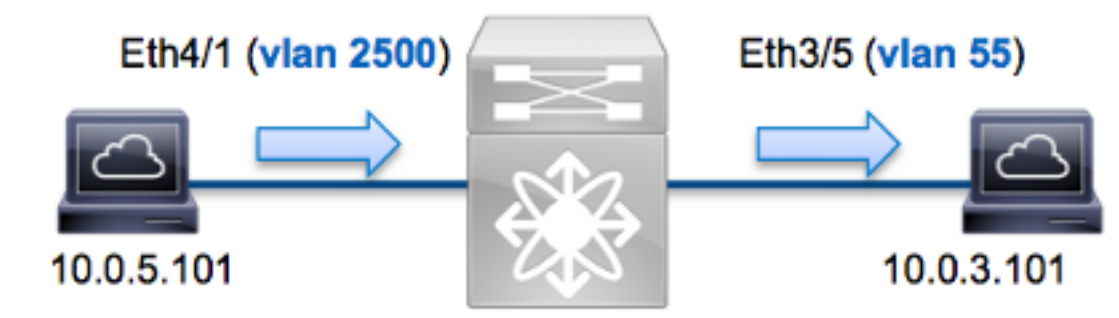
[Vérification supplémentaire](#)

Introduction

Ce document décrit les étapes utilisées afin d'exécuter un ELAM sur Cisco Nexus 7000 modules de la M-gamme (N7K), explique les sorties les plus appropriées, et décrit comment interpréter les résultats.

Conseil : Référez-vous au document d'[aperçu ELAM](#) pour un aperçu sur ELAM.

Topologie



Dans cet exemple, un hôte sur VLAN 2500 (**10.0.5.101**), le port **Eth4/1** envoie une demande de Protocole ICMP (Internet Control Message Protocol) à un hôte sur VLAN 55 (**10.0.3.101**), mettent en communication **Eth3/5**. ELAM est utilisé afin de capturer ce paquet simple de **10.0.5.101** à **10.0.3.101**. Il est important de se souvenir qu'ELAM te permet pour capturer une trame simple.

Afin d'exécuter un ELAM sur le N7K, vous devez d'abord se connecter au module approprié (ceci exige le privilège de réseau-admin) :

```
N7K# attach module 4
Attaching to module 4 ...
To exit type 'exit', to abort type '$.'
module-4#
```

Déterminez l'engine d'expédition d'entrée

Le trafic est prévu au d'entrée le commutateur sur le port **Eth4/1**. Quand vous vérifiez les modules dans le système, vous voyez que le **module 4** est un module de M-gamme. Il est important de se souvenir que le N7K plein-est distribué, et que les modules, pas le superviseur, prennent les décisions d'expédition pour le trafic de dataplane.

```
N7K# show module
Mod  Ports  Module-Type                Model                Status
---  -
3    32     10 Gbps Ethernet Module    N7K-M132XP-12       ok
4    48     10/100/1000 Mbps Ethernet Module N7K-M148GT-11      ok
5     0     Supervisor module-1X       N7K-SUP1             active *
6     0     Supervisor module-1X       N7K-SUP1             ha-standby
```

Pour des modules de M-gamme, exécutez l'ELAM sur la couche 2 (L2) expédiant l'engine (technicien) avec l'**Eureka** interne de nom de code. Notez que le bus de données technicien L2 (D-BUS) contient les informations d'en-tête d'origine avant que le L2 et pose 3 consultations (L3), et le bus de résultat (RBUS) contient les résultats après les consultations L3 et L2. La consultation L3 est exécutée par le technicien L3/Layer 4 (L4) avec le nom de code interne **Lamira**, qui est le même processus utilisé sur la plate-forme de commutateur de gamme Cisco Catalyst 6500 qui exécute l'engine 2T de superviseur.

Les modules de M-gamme N7K peuvent utiliser plusieurs FEs pour chaque module, ainsi vous devez déterminer l'**Eureka** ASIC qui est utilisé pour le technicien sur le port **Eth4/1**. Sélectionnez cette commande afin de vérifier ceci :

```
module-4# show hardware internal dev-port-map
(some output omitted)
-----
CARD_TYPE:          48 port 1G
>Front Panel ports:48
-----
Device name          Dev role          Abbr num_inst:
-----
>Eureka             DEV_LAYER_2_LOOKUP L2LKP 1
-----+
+-----+++FRONT PANEL PORT TO ASIC INSTANCE MAP+++-----+
+-----+
FP port|PHYS |SECUR |MAC_0 |RWR_0 |L2LKP |L3LKP |QUEUE |SWICHF
1    0    0    0    0    0    0    0    0
2     0    0    0    0    0    0    0    0
```

Dans la sortie, vous pouvez voir que le port **Eth4/1** est sur l'exemple **0** de l'**Eureka** (**L2LKP**).

Remarque: Pour des modules de M-gamme, les valeurs des utilisations 1-based de syntaxe ELAM, ainsi l'exemple **0** devient l'exemple **1** quand vous configurez l'ELAM. Ce n'est pas la caisse pour des modules de F-gamme.

```
module-4# elam asic eureka instance 1
module-4(eureka-elam)#
```

Configurez le déclencheur

L'Eureka ASIC prend en charge des déclencheurs ELAM pour l'ipv4, l'IPv6, et d'autres. Le déclencheur ELAM doit aligner avec le type de trame. Si la trame est une trame d'ipv4, alors le déclencheur doit également être ipv4. Une trame d'ipv4 n'est pas capturée avec un *autre* déclencheur. La même logique s'applique à l'IPv6.

Avec les systèmes d'exploitation de Nexus (NX-OS), vous pouvez employer le caractère de point d'interrogation afin de séparer le déclencheur ELAM :

```
module-4(eureka-elam)# trigger dbus dbi ingress ipv4 if ?
(some output omitted)
destination-flood          Destination Flood
destination-index          Destination Index
destination-ipv4-address   Destination IP Address
destination-mac-address    Destination MAC Address
ip-tos                     IP TOS
ip-total-len               IP Total Length
ip-ttl                     IP TTL
source-mac-address         Source MAC Address
vlan-id                    Vlan ID Number
```

Pour cet exemple, la trame est capturée selon la source et des adresses d'ipv4 de destination, ainsi seulement ces valeurs sont spécifiées.

L'Eureka exige que des déclencheurs sont placés pour le D-BUS et le RBUS. Il y a deux tampons de paquets différents (PB) en lesquels les données RBUS peuvent résider. La détermination de l'exemple correct de PB dépend du type et du port d'entrée précis de module. Typiquement, il est recommandé que vous configurez PB1, et si le RBUS ne déclenche pas, alors répétez la configuration avec PB2.

Voici le déclencheur de D-BUS :

```
module-4(eureka-elam)# trigger dbus dbi ingress ipv4 if source-ipv4-address
10.0.5.101 destination-ipv4-address 10.0.3.101 rbi-corelate
```

Voici le déclencheur RBUS :

```
module-4(eureka-elam)# trigger rbus rbi pb1 ip if cap2 1
```

Remarque: Le mot clé de **rbi-corrélation à la fin** du déclencheur de D-BUS est exigé pour que le RBUS déclenche correctement sur le bit **cap2**.

Commencez la capture

Maintenant que le technicien d'entrée est sélectionné et vous avez configuré le déclencheur, vous pouvez commencer la capture :

```
module-4(eureka-elam)# start
```

Afin de vérifier l'état de l'ELAM, sélectionnez la commande d'**état** :

```
module-4(eureka-elam)# status
```

Instance: 1

EU-DBUS: Armed

```
trigger dbus dbi ingress ipv4 if source-ipv4-address 10.0.5.101
```

```
destination-ipv4-address 10.0.3.101 rbi-corelate
```

```
EU-RBUS: Armed
```

```
trigger rbus rbi pb1 ip if cap2 1
```

```
LM-DBUS: Dis-Armed
```

```
No configuration
```

```
LM-RBUS: Dis-Armed
```

```
No configuration
```

Une fois la trame qui apparie le déclencheur est reçue par le technicien, les expositions d'état ELAM comme **déclenchée** :

```
module-4(eureka-elam)# status
```

```
Instance: 1
```

```
EU-DBUS: Triggered
```

```
trigger dbus dbi ingress ipv4 if source-ipv4-address 10.0.5.101
```

```
destination-ipv4-address 10.0.3.101 rbi-corelate
```

```
EU-RBUS: Triggered
```

```
trigger rbus rbi pb1 ip if cap2 1
```

```
LM-DBUS: Dis-Armed
```

```
No configuration
```

```
LM-RBUS: Dis-Armed
```

```
No configuration
```

Interprétez les résultats

Afin d'afficher les résultats ELAM, entrez dans le **d-bus d'exposition** et **affichez les** commandes de **rbus**. S'il y a un grand volume du trafic qui apparie les mêmes déclencheurs, le D-BUS et le RBUS pourraient déclencher sur différentes trames. Par conséquent, il est important de vérifier les numéros de séquence internes sur les données de D-BUS et RBUS afin de s'assurer qu'ils s'assortissent :

```
module-4(eureka-elam)# show dbus | i seq
```

```
seq = 0x05
```

```
module-4(eureka-elam)# show rbus | i seq
```

```
seq = 0x05
```

Voici l'extrait des données ELAM qui sont les plus appropriées à cet exemple (une certaine sortie est omise) :

```
module-4(eureka-elam)# show dbus
```

```
seq = 0x05
```

```
vlan = 2500
```

```
source_index = 0x00a21
```

```
l3_protocol = 0x0 (0:IPv4, 6:IPv6)
```

```
l3_protocol_type = 0x01, (1:ICMP, 2:IGMP, 4:IP, 6:TCP, 17:UDP)
```

```
dmac = 00.00.0c.07.ac.65
```

```
smac = d0.d0.fd.b7.3d.c2
```

```
ip_ttl = 0xff
```

```
ip_source = 010.000.005.101
```

```
ip_destination = 010.000.003.101
```

```
module-4(eureka-elam)# show rbus
```

```
seq = 0x05
```

```
flood = 0x0
```

```
dest_index = 0x009ed
```

```
vlan = 55
```

```
ttl = 0xfe
```

```
data(rit/dmac/recir) = 00.05.73.a9.55.41
```

```
data(rit/smac/recir) = 84.78.ac.0e.47.41
```

Avec les données de **D-BUS**, vous pouvez vérifier que la trame est reçue sur VLAN 2500 avec une adresse MAC source de **d0d0.fdb7.3dc2** et une adresse MAC de destination de

0000.0c07.ac65. Vous pouvez également voir que c'est une trame d'ipv4 qui est originaire de 10.0.5.101, et est destinée à 10.0.3.101.

Conseil : Il y a plusieurs autres champs utiles qui ne sont pas inclus dans cette sortie, telle que la valeur de Type de service (ToS), les indicateurs IP, la longueur IP, et la longueur de la trame L2.

Afin de vérifier sur quel port la trame est reçue, sélectionnez la commande **SRC_INDEX** (la logique locale de cible de source (le LTL)). Sélectionnez cette commande afin de tracer un LTL à un port ou à un groupe de ports pour le N7K :

```
N7K# show system internal pixm info ltl 0xa21
```

```
Member info
```

```
-----  
Type                LTL  
-----  
PHY_PORT          Eth4/1  
FLOOD_W_FPOE       0x8014
```

La sortie prouve que le **SRC_INDEX** de **0xa21** trace pour mettre en communication **Eth4/1**. Ceci confirme que la trame est reçue sur le port **Eth4/1**.

Avec les données **RBUS**, vous pouvez vérifier que la trame est conduite à VLAN 55, et que le TTL est décrémenté de **0xff** dans les données de **D-BUS** à **0xfe** dans les données **RBUS**. Vous pouvez voir que la source et les adresses de MAC de destination sont réécrites à **8478.ac0e.4741** et à **0005.73a9.5541**, respectivement. Supplémentaire, vous pouvez confirmer le port de sortie du **DEST_INDEX** (LTL de destination) :

```
N7K# show system internal pixm info ltl 0x9ed
```

```
Member info
```

```
-----  
Type                LTL  
-----  
PHY_PORT          Eth3/5  
FLOOD_W_FPOE       0x8017  
FLOOD_W_FPOE       0x8016
```

La sortie prouve que le **DEST_INDEX** de **0x9ed** trace pour mettre en communication **Eth3/5**. Ceci confirme que la trame est envoyée du port **Eth3/5**.

Vérification supplémentaire

Afin de vérifier comment le commutateur alloue le groupe LTL, sélectionnez la commande **interne de LTL-région de l'information de pixm de show system**. La sortie de cette commande est utile afin de comprendre le but d'un LTL si elle n'est pas appariée à un port physique. Un bon exemple de ceci est un LTL de **baisse** :

```
N7K# show system internal pixm info ltl 0x11a0
```

```
0x11a0 is not configured
```

```
N7K# show system internal pixm info ltl-region
```

```
LTL POOL TYPE                SIZE                RANGE  
=====
```

DCE/FC Pool	1024	0x0000 to 0x03ff
SUP Inband LTL	32	0x0400 to 0x041f
MD Flood LTL	1	0x0420

Central R/W		1	0x0421
UCAST Pool		1536	0x0422 to 0x0a21
PC Pool		1720	0x0a22 to 0x10d9
LC CPU Pool		32	0x1152 to 0x1171
EARL Pool		72	0x10da to 0x1121
SPAN Pool		48	0x1122 to 0x1151
UCAST VDC Use Pool		16	0x1172 to 0x1181
UCAST Generic Pool		30	0x1182 to 0x119f
LISP Pool		4	0x1198 to 0x119b
Invalid SI		1	0x119c to 0x119c
ESPAN SI		1	0x119d to 0x119d
Recirc SI		1	0x119e to 0x119e
Drop DI		2	0x119f to 0x11a0
UCAST (L3_SVI_SI) Region		31	0x11a1 to 0x11bf
UCAST (Fex/GPC/SVI-ES)	3648		0x11c0 to 0x1fff
UCAST Reserved for Future Use Region		2048	0x2000 to 0x27ff
=====> UCAST MCAST BOUNDARY <=====			
VDC OMF Pool		32	0x2800 to 0x281f