

ASIC (Tahoe) NX-OS ELAM évolutif pour le cloud Nexus 9000

Contenu

[Introduction](#)

[Matériel applicable](#)

[Procédure Nexus Tahoe ASIC ELAM](#)

[Topologie](#)

[Étape 1 : vérification de l'ASIC, de la tranche et de l'ID source du port](#)

[Étape 2 : fixation au module](#)

[Étape 3 - Passez en mode de configuration ELAM et spécifiez l'ASIC approprié à partir de l'étape 1](#)

[Étape 4 - Configurez le déclencheur ELAM](#)

[Étape 5 - Définissez les déclencheurs ELAM à l'aide des adresses IP SRC et DEST](#)

[Étape 6 - Démarrez le module ELAM](#)

[Étape 7 - Vérifiez si votre ELAM est déclenché, puis examinez les résultats](#)

[Affichage des résultats ELAM pour les versions de NX-OS antérieures à la version 7.0\(3\)I5\(2\)](#)

[ELAM encapsulé VXLAN :](#)

[Configuration ELAM après NX-OS 7.0\(3\)I7\(2\)](#)

Introduction

Ce document décrit les étapes utilisées pour effectuer un ELAM (Embedded Logic Analyzer Module) sur une série de modules Cisco Nexus 9000 CloudScale ASIC, couvre les sorties les plus pertinentes et décrit comment interpréter les résultats.

Astuce : Reportez-vous au document [Présentation d'ELAM](#) pour une présentation d'ELAM.

Matériel applicable

La procédure décrite dans ce document s'applique uniquement au matériel suivant :

N9K-C93180YC-EX	N9K-C92304QC
N9K-X9736C-EX	N9K-C92300YC
N9K-C93108TC-EX	N9K-X9788TC-FX
N9K-X9732C-EX	N9K-X97284YC-FX
N9K-X97160YC-EX	N9K-C93180YC-FX

N9K-C93180LC-EX	N9K-C93108TC-FX
N9K-C92160YC-X	N9K-C9348GC-FXP
N9K-C9272Q	N9K-X9732C-FX
N9K-C9236C	N9K-C9336C-FX2
N9K-C93240YC-FX2	N9K-C93300YC-FX2
N9K-C9364C	N9K-C932C

Procédure Nexus Tahoe ASIC ELAM

Topologie



Étape 1 : vérification de l'ASIC, de la tranche et de l'ID source du port

```
N9K-C92160YC-X-2# show hardware internal tah interface e1/49
IfIndex: 436232192
DstIndex: 5952
IfType: 26
Asic: 0 <<<<<<<<<< Asic: 0
AsicPort: 56
SrcId: 48 <<<<<<<<<< Slice: 1 <<<<<<<<<< PortOnSlice: 24
```

Attention : ELAM ne doit être utilisé que sur une fenêtre de terminal, car vous gérez le contenu global pour chaque tranche, lu-a2d, etc.

Par exemple, un port-channel (PO) peut avoir deux liaisons, Eth 1/53 qui correspond à la tranche 0 et Eth 1/54 qui correspond à la tranche 1. L'installation d'ELAM sur deux fenêtres de terminal séparées à la fois pour les différentes tranches ne sera pas utile car cette dernière tranche (disons tranche 1) écrasera la première (tranche 0), finissant par obtenir le même résultat sur les deux fenêtres de terminal.

Vous pouvez vérifier ces informations via :

```
N9K-C92160YC-X-2# show system internal ethpm info interface e1/49 | i i src
IF_STATIC_INFO: port_name=Ethernet1/49,if_index:0x1a006000,ltl=5952,slot=0,
nxos_port=192,dmod=1,dpid=56,
unit=0,queue=65535,xbar_unitbmp=0x0,ns_pid=255,slice_num=1,port_on_slice=24,src_id=48
```

Étape 2 : fixation au module

```
N9K-C92160YC-X-2# attach mod 1
```

Étape 3 - Passez en mode de configuration ELAM et spécifiez l'ASIC approprié à partir de l'étape 1

```
module-1# debug platform internal tah elam asic 0
```

Étape 4 - Configurez le déclencheur ELAM

Note: Il existe de nombreuses options que vous pouvez spécifier ici en fonction du paquet/flux capturé

```
module-1(TAH-elam)# trigger init asic 0 slice 1 lu-a2d 1 in-select 6 out-select 0 use-src-id 48
```

Astuce :

- Si les ports d'entrée et de sortie se trouvent sur des tranches différentes sur le même ASIC, ELAM sur la tranche de sortie ne capturera pas le paquet sortant, car le paquet ne passera pas par les blocs LUX sur la tranche de sortie et contournera donc ELAM.
- lu-a2d 0 est utilisé pour l'ELAM inverse où le déclencheur est basé sur le résultat et lu-a2d 1 est utilisé pour l'ELAM où le déclencheur est basé sur les attributs de paquet
- Utilisez toujours 6 pour la sélection entrante et 0 pour la désélection

Avertissement : N'utilisez pas 0 après lu-a2d car cela pourrait provoquer une panne du commutateur - voir [CSCvd64106](#) pour plus de détails

Étape 5 - Définissez les déclencheurs ELAM à l'aide des adresses IP SRC et DEST

Dans cet exemple, l'adresse IP source est 192.0.2.2 et l'adresse IP de destination est 192.0.2.1, comme indiqué ci-dessous :

```
module-1(TAH-elam-insel6)# reset
module-1(TAH-elam-insel6)# set outer ipv4 dst_ip 192.0.2.1 src_ip 192.0.2.2
```

Note: Assurez-vous de réinitialiser car la commande "set" prévaut dans tous les ELAM et peut l'empêcher de se déclencher sur des champs inattendus.

hardware internal tah interface ex/y | i niv"

- Les **dmod** et **dpid** correspondent à l'interface de sortie. Ceci peut être vérifié avec "**show system internal ethpm info int ex/y | i dpid**"
- En outre, "**show interface hardware-mapping**" peut être utilisé pour valider le dmod/dpid.

REMARQUES :

1. Lorsqu'il s'agit de confirmer si le paquet est réellement abandonné, le champ "**Abandons finaux**" est le **SEUL** à prendre en compte. En d'autres termes, bien que vous puissiez voir des exceptions lancées dans d'autres champs comme LUA/B/C/D, cela *ne signifie pas nécessairement* que le paquet est abandonné. Veuillez examiner attentivement ce résultat (en discuter avec le TAC si nécessaire).

2. L'indicateur sup_hit sera défini pour le trafic envoyé au processeur (**détail du rapport**) | **grep sup_hit**).

- Vous pouvez décoder la raison en utilisant « **show system internal access-list sup-redirect-stats all** » et en faisant correspondre l'index sup
- Assurez-vous que le '**mode de routage système**' correct est configuré (**show system routing mode**) Selon les directives et les limites documentées dans [Considérations relatives au déploiement VXLAN](#) Le « Mode de routage du système : template-vxlan-scale » ne s'applique pas aux versions 7.0(3)I5(2) et ultérieures de Cisco NX-OS. Lors de l'utilisation de VXLAN BGP EVPN en combinaison avec Cisco NX-OS version 7.0(3)I4(x) ou NX-OS version 7.0(3)I5(1), le « Mode de routage système : template-vxlan-scale » est requis sur les plates-formes matérielles suivantes : Commutateurs Cisco Nexus 9300-EX Commutateurs Cisco Nexus 9500 avec cartes de ligne X9700-EX Le changement du « mode de routage du système » nécessite un rechargement du commutateur.

Exemple de trafic avec CPU punt :

```
module-1(TAH-elam-insel6)# report
```

```
SUGARBOWL ELAM REPORT SUMMARY  
=====
```

```
Incoming Interface: Eth1/3  
Src Idx : 0x9, Src BD : 23  
Outgoing Interface Info: dmod 1, dpid 72  
Dst Idx : 0x601, Dst BD : 802
```

```
Packet Type: IPv4
```

```
Dst MAC address: B0:8B:CF:A3:D0:4B  
Src MAC address: 00:10:DB:FF:10:00  
.1q Tag0 VLAN: 23, cos = 0x0  
  
Dst IPv4 address: 192.0.2.1  
Src IPv4 address: 192.0.2.2  
Ver      = 4, DSCP      = 2, Don't Fragment = 1  
Proto    = 6, TTL      = 49, More Fragments = 0  
Hdr len  = 20, Pkt len = 60, Checksum      = 0x63c3
```

```
L4 Protocol : 6  
TCP Dst Port : 80  
TCP Src Port : 46340
```

```
Sup hit: 1, Sup Idx : 2720 <<---- CPU punt, use below CLI to resolve the meaning of Sup Idx
```

```
Drop Info:
```

LUA:
LUB:
LUC:
LUD:
Final Drops:

```
# show system internal access-list sup-redirect-stats all | grep 2720 2720 copp-system-p-acl-  
http 63
```

Affichage des résultats ELAM pour les versions de NX-OS antérieures à la version 7.0(3)I5(2)

+ Est-ce qu'il y a un en-tête dot1q ?

```
module-1(TAH-elam-insel6)# report | grep pr_lu_vec_l2v.qtag0 GBL_C++: [MSG] pr_lu_vec_l2v.qtag0_vld:  
0x1 << dot1q yes? 0x1 GBL_C++: [MSG] pr_lu_vec_l2v.qtag0_cos: 0x0 GBL_C++: [MSG]  
pr_lu_vec_l2v.qtag0_de: 0x0 GBL_C++: [MSG] pr_lu_vec_l2v.qtag0_vlan: 0xA << VL 10
```

+ Vérifier le VLAN :

```
module-1(TAH-elam-insel6)# report | grep -1 fpx_lookup_vec.lkup.macsakey.key.fid  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.vld: 0x1  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.fid_type: 0x0  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.fid_vld: 0x0  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.fid: 0xA << dec 0xA = VL 10  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.mac: 0xFEC80E2715
```

+ Vérifiez SRC MAC (vous pouvez voir ceci dans l'étape précédente) :

```
module-1(TAH-elam-insel6)# report | grep -i fpx_lookup_vec.lkup.macsakey.key.mac  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.mac: 0xFEC80E2715 << 00fe.c80e.2715  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.mac: 0xFEC80E2715  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.mac: 0xFEC80E2715  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.mac: 0xFEC80E2715  
GBL_C++: [MSG] fpx_lookup_vec.lkup.macsakey.key.mac: 0xFEC80E2715
```

+ Est-ce un nouvel apprentissage ?

```
module-1(TAH-elam-insel6)# report | grep -1 fpx_lookup_vec.sa_notify_info  
GBL_C++: [MSG] fpx_lookup_vec.lkup.ptvec.misc1.tcp_flags: 0x0  
GBL_C++: [MSG] fpx_lookup_vec.sa_notify_info: 0x5200000C060  
GBL_C++: [MSG] fpx_lookup_vec.sa_notify_info.enable: 0x0 << This will be set to 0x1 for learning  
to happen  
GBL_C++: [MSG] fpx_lookup_vec.sa_notify_info.conv_learn_only: 0x0
```

+ Vérifier SRC & DST IP :

```
module-1(TAH-elam-insel6)# report | grep vec_l3v.ip.*a  
GBL_C++: [MSG] pr_lu_vec_l3v.ip.da: 0x0000000000000000c0000201 << DST IP: 192.0.2.1  
GBL_C++: [MSG] pr_lu_vec_l3v.ip.sa: 0x0000000000000000c0000202 << SRC IP: 192.0.2.2
```

+ Vérifiez votre SRC_ID d'entrée :

```
module-1(TAH-elam-insel6)# report | egrep SRC  
GBL_C++: [MSG] SRCID: 0x30
```

```
module-1(TAH-elam-insel6)# report | grep vec.ihdr.ieth.hdr.src_idx  
GBL_C++: [MSG] lurw_vec.ihdr.ieth.hdr.src_idx: 0xA9 << sh hardware internal tah int e1/49 | i i niv_idx
```

+ Si ELAM ne se déclenche pas, il se présente comme suit :

```
module-1(TAH-elam-insel6)# report  
GBL_C++: [MSG] tahusd_elam_wrapper_report:27d:asic type 5 inst 0 slice 1 a_to_d 1 insel 6  
outsel 0  
GBL_C++: [MSG] Inside tahusd_elam_wrapper_dav_report  
GBL_C++: [MSG] ELAM not yet triggered <<<<<<
```

ELAM encapsulé VXLAN :

Étant donné que les paquets VXLAN seraient encapsulés, l'ELAM doit être déclenché sur l'en-tête INNER plutôt que sur l'en-tête OUTER. Voir l'exemple ci-dessous pour une trame ARP :

```
module-1# debug platform internal tah elam asic 0  
module-1(TAH-elam)# trigger init asic 0 slice 1 in-select 7 out-select 0 use-src-id 48  
module-1(TAH-elam-insel7)# reset  
module-1(TAH-elam-insel7)# set inner arp source-ip-addr 192.0.2.2 target-ip-addr 192.0.2.1  
module-1(TAH-elam-insel7)# start  
module-1(TAH-elam-insel7)# report
```

Configuration ELAM après NX-OS 7.0(3)I7(2)

Après NX-OS 7.0(3)I7(2), ELAM peut désormais être déclenché globalement sans spécifier le numéro ASIC ou le numéro de tranche pour plus de facilité. Voir l'exemple ci-dessous :

```
Nexus-9K# debug platform internal tah elam  
Nexus-9K(TAH-elam)# trigger init  
Nexus-9K(TAH-elam-insel6)# reset  
Nexus-9K(TAH-elam-insel6)# set outer ipv4 dst_ip 192.0.2.1 src_ip 192.0.2.2  
Nexus-9K(TAH-elam-insel6)# start  
Nexus-9K(TAH-elam-insel6)# report
```

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.