

Nexus 9000 Cloud Scale ASIC(Tahoe) NX-OS ELAM

목차

[소개](#)

[적용 가능한 하드웨어](#)

[Nexus Tahoe ASIC ELAM 절차](#)

[토폴로지](#)

[1단계 - 포트의 ASIC, 슬라이스 및 SrcId 확인](#)

[2단계 - 모듈에 연결](#)

[3단계 - ELAM 컨피그레이션 모드로 들어가서 1단계에서 적절한 ASIC를 지정합니다](#)

[4단계 - ELAM 트리거 구성](#)

[5단계 - SRC 및 DEST IP를 사용하여 ELAM 트리거 설정](#)

-

[6단계 - ELAM 시작](#)

[7단계 - ELAM이 트리거되었는지 확인한 후 결과를 검토합니다.](#)

[7.0\(3\)I5\(2\) 이전 버전의 NX-OS에 대한 ELAM 결과 보기](#)

[VXLAN 캡슐화 ELAM:](#)

[NX-OS 7.0\(3\)I7\(2\) 이후 ELAM 구성](#)

소개

ELAM(Embedded Logic AnCisco Nexus 9000 CloudScale ASIC)은

팁: ELAM에 [대한 개요](#)는 ELAM 개요 문서를 참조하십시오.

적용 가능한 하드웨어

이 문서에서 다루는 절차는 다음 하드웨어에만 적용됩니다.

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-C9332C

Nexus Tahoe ASIC ELAM 절차

토폴로지



1단계 - 포트의 ASIC, 슬라이스 및 SrcId 확인

```
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
```

주의: 각 슬라이스, lu-a2d 등에 대한 글로벌 콘텐츠를 유지 관리하므로 ELAM은 하나의 터미널 창에서만 사용해야 합니다.

예를 들어 포트 채널(PO)에는 슬라이스 0에 해당하는 Eth 1/53과 슬라이스 1에 해당하는 Eth 1/54라는 두 개의 링크가 있을 수 있습니다. 서로 다른 슬라이스에 대해 서로 다른 두 개의 터미널 창에 ELAM을 설정하는 것은 후자의 슬라이스(즉 슬라이스 1)가 첫 번째 슬라이스(슬라이스 0)를 덮어쓰므로 도움이 되지 않으며 결국 두 터미널 창에서 동일한 결과를 얻을 수 있습니다.

다음은 통해 이 정보를 다시 확인할 수 있습니다.

```
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
```

2단계 - 모듈에 연결

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

3단계 - ELAM 컨피그레이션 모드로 들어가서 1단계에서 적절한 ASIC를 지정합니다

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

4단계 - ELAM 트리거 구성

참고: 캡처되는 패킷/플로우에 따라 여기에서 지정할 수 있는 여러 옵션이 있습니다

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

팁:

- 인그레스 및 이그레스 포트가 동일한 ASIC의 서로 다른 슬라이스에 있는 경우, 패킷이 이그레스 슬라이스의 LUX 블록을 통과하지 않으므로 ELAM을 우회하므로 이그레스 슬라이스의 ELAM이 발신 패킷을 캡처하지 않습니다.
- lu-a2d 0은 트리거가 결과를 기반으로 하는 역방향 ELAM에 사용되고 lu-a2d 1은 트리거가 패킷 특성을 기반으로 하는 ELAM에 사용됩니다
- 인-선택에는 항상 6, 아웃-선택에는 0 사용

경고: lu-a2d 후 0을 사용하지 마십시오. 스위치에 충돌이 발생할 수 있습니다. 자세한 내용은 [CSCvd64106](#)을 참조하십시오.

5단계 - SRC 및 DEST IP를 사용하여 ELAM 트리거 설정

이 예에서 소스 IP는 192.0.2.2이고 대상 IP는 192.0.2.1입니다.

```
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
```

참고: "set" 명령이 ELAM 전체에서 우선하며 예기치 않은 필드에서 트리거되거나 트리거되지 않도록 할 수 있으므로 "reset"을 확인하십시오.

6단계 - ELAM 시작

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

```
GBL_C++: [MSG] tahusd_elam_wrapper_init:36:asic type 5 inst 0 slice 1 a_to_d 1 insel 6 outsel 0
```


1. 패킷이 실제로 삭제되고 있는지를 확인하는 경우에는 "최종 삭제" 필드가 유일한 항목입니다. 즉, LUA/B/C/D와 같은 다른 필드에 예외가 throw되는 것을 볼 수 있지만, 반드시 패킷이 삭제되는 것은 아닙니다. 이 결과를 자세히 검토하십시오 (필요에 따라 TAC와 논의).

2. CPU로 전송된 트래픽에는 sup_hit 플래그가 설정됩니다(보고서 세부 정보). | grep sup_hit).

- 'show system internal access-list sup-redirect-stats all'을 사용하고 sup 인덱스와 일치시켜 이 유를 디코딩할 수 있습니다
- 올바른 '시스템 라우팅 모드'가 구성되었는지 확인합니다(시스템 라우팅 모드 표시) VXLAN 구축 시 고려 사항에 설명된 [지침 및 제한 사항](#) "시스템 라우팅 모드: template-vxlan-scale"은 Cisco NX-OS 릴리스 7.0(3)I5(2) 이상에는 적용되지 않습니다.VXLAN BGP EVPN을 Cisco NX-OS Release 7.0(3)I4(x) 또는 NX-OS Release 7.0(3)I5(1)과 함께 사용할 경우 "시스템 라우팅 모드: template-vxlan-scale"은 다음 하드웨어 플랫폼에 필요합니다.Cisco Nexus 9300-EX SwitchX9700-EX 라인 카드가 장착된 Cisco Nexus 9500 스위치"시스템 라우팅 모드"를 변경하려면 스위치를 다시 로드해야 합니다.

CPU 펀트를 경험하는 트래픽의 예:

```
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
```

7.0(3)I5(2) 이전 버전의 NX-OS에 대한 ELAM 결과 보기

+ 여기에 dot1q 헤더가 있습니까?

```
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
+ VLAN 확인:
```

```
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
+ SRC MAC 선택(이전 단계에서도 실제로 확인 가능):
```

```
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
+ 새로운 학습 내용입니까?
```

```
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
```

+ SRC 및 DST IP 확인:

```
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
```

+ 인그레스 SRC_ID 확인:

```
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
```

+ ELAM이 트리거되지 않으면 다음과 같이 표시됩니다.

```
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 <<<<<<
```

VXLAN 캡슐화 ELAM:

VXLAN 패킷은 캡슐화되므로 ELAM은 OUTER 헤더가 아닌 INNER 헤더에서 트리거되어야 합니다. ARP 프레임은 아래 예를 참조하십시오.

```
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
```

NX-OS 7.0(3)I7(2) 이후 ELAM 구성

NX-OS 7.0(3)I7(2) 이후, ELAM은 이제 손쉽게 ASIC 또는 슬라이스 번호를 지정하지 않고도 전체적으로 트리거될 수 있습니다. 아래 예를 참조하십시오.

```
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
```

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.