

部署ELAM以捕獲Nexus 7000系列交換機上的VXLAN封裝資料包

目錄

[簡介](#)

[背景資訊](#)

[拓撲](#)

[配置觸發器](#)

[解釋結果](#)

[相關資訊](#)

簡介

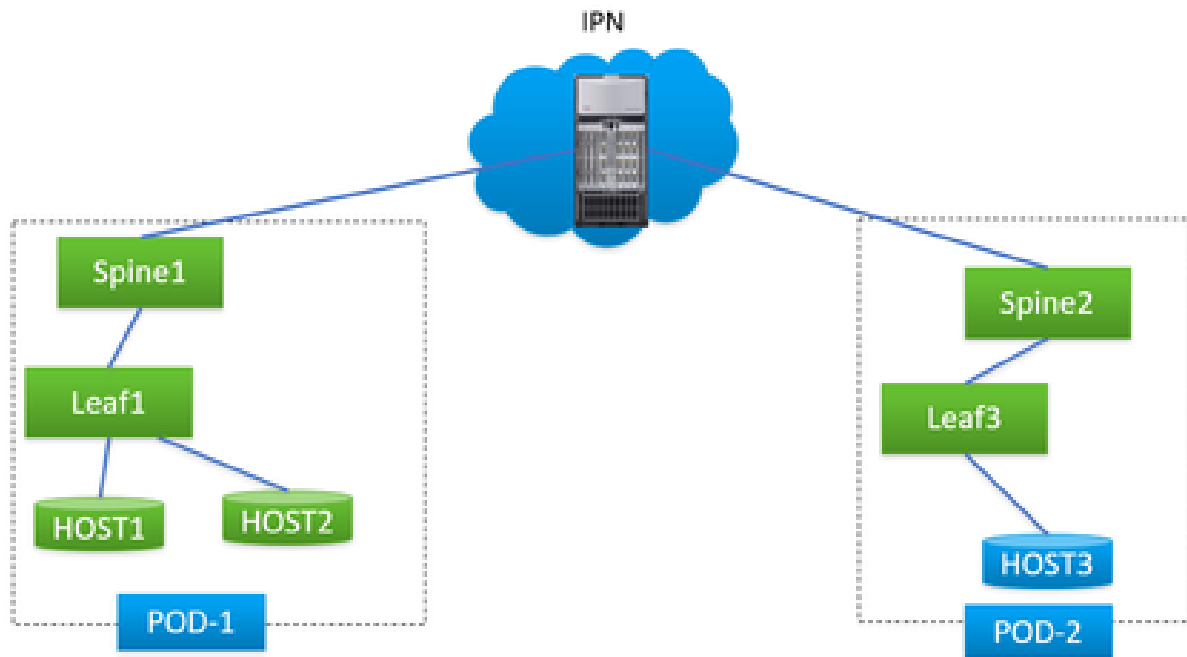
本檔案介紹如何部署嵌入式邏輯分析器模組(ELAM)以擷取Nexus 7000系列交換器上的VXLAN封裝封包。

 提示：有關ELAM的概觀，請參閱ELAM概述文檔。

背景資訊

許多使用者目前將N7K用作其ACI MPOD/MSITE部署的IPN/ISN傳輸裝置。然而，與N9K相比，N7K缺乏基於豐富的外層(l2(vntag)|l3|l4)-inner(l2|l3|l4)-8組合來設定ELAM觸發器的強大功能。因此，從ELAM的角度確定特定VXLAN封裝的資料包是否在IPN邊緣觸及N7K變得非常困難。本文檔概述了解決此挑戰的方法。

拓撲



在此場景中，顯示了簡單的ACI MPOD拓撲，其中IPN是帶F3卡的N7K。HOST1和HOST2位於pod1中，HOST3位於pod2中。HOST1可以與HOST3通訊，但HOST2無法通訊。ACI工程師進行故障排除後，確定從HOST2到HOST3的資料包從pod1中的spine1傳送到N7K，但pod2中的spine2從未收到這些資料包。這通過ACI主幹上的ELAM驗證，導致懷疑資料包被丟棄在N7K上。

是否可以僅根據ACI主幹上的ELAM結果確定問題歸因於N7K?當然不是。輸出spine1上的ELAM指示它已將資料包傳送到N7K，但這並不保證資料包實際到達N7K，因為由於較低層的問題，資料包在ELAM週期之後仍可能被丟棄。但是，當您在N7K端對這些特定資料包進行ELAM時，它可以幫助我們準確地識別問題中涉及的正确裝置。

配置觸發器

「l3-packet-length」是N7K平台上幾乎所有不同層代LC的有效ELAM觸發器。因此，讓我們利用它來建立IPN N7K上的ELAM條件。該任務涉及控制HOST以傳輸具有指定資料包長度的測試資料包，如下所示：

```
<#root>
```

```
#
```

```
ping 172.28.1.20 packet-size 777
```

```
PING 172.28.1.20 (172.28.1.20): 777 data bytes
```

```
785 bytes from 172.28.1.20: icmp_seq=0 ttl=252 time=1.246 ms
```

```
785 bytes from 172.28.1.20: icmp_seq=1 ttl=252 time=0.846 ms
```

```
785 bytes from 172.28.1.20: icmp_seq=2 ttl=252 time=0.84 ms
```

```
785 bytes from 172.28.1.20: icmp_seq=3 ttl=252 time=0.814 ms
```

```
785 bytes from 172.28.1.20: icmp_seq=4 ttl=252 time=0.817 ms
```

```
--- 172.28.1.20 ping statistics ---
```

```
5 packets transmitted, 5 packets received, 0.00% packet loss
```

```
round-trip min/avg/max = 0.814/0.912/1.246 ms
```

PING實用程式整合到任何型別的作業系統中，但引數僅根據您使用的作業系統略有變化。要強調的關鍵一點是，注意在作業系統中啟動PING時指定的資料包大小。在本範例中，777B代表純資料長度，需要額外8B（ICMP標頭）和20B（IP標頭）才能取得最終的IP長度805B。進行VXLAN封裝後（增加額外50B額外負荷），可以預期到達N7K的資料包為855B。讓我們在ELAM中配置它。

在本示例中，連線到spine1的介面是E7/1，而E7/4連線到spine2。

```
<#root>
```

```
#
```

```
show module 7
```

Mod	Ports	Module-Type	Model	Status
7	12	10/40 Gbps Ethernet Module	N7K-F312FQ-25	ok

```
module-7#
```

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

```

```

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

```

因此，需要在例項0中設定它。

```
<#root>
```

```
module-7# elam asic flanker instance 0
```

```
module-7(fln-elam)# layer2
```

```
module-7(f1n-12-elam)#
trigger dbus ipv4 ingress if 13-packet-length 855

module-7(f1n-12-elam)#
trigger rbus ingress if trig

module-7(f1n-12-elam)# start
module-7(f1n-12-elam)# status

ELAM Slot 7 instance 0: L2 DBUS Configuration: trigger dbus ipv4 ingress if 13-packet-length 855
L2 DBUS: Triggered

ELAM Slot 7 instance 0: L2 RBUS Configuration: trigger rbus ingress if trig
L2 RBUS: Triggered
```

解釋結果

```
<#root>
```

```
module-7(f1n-12-elam)#
```

```
show dbus
```

```
cp = 0x10084d00, buf = 0x10084d00, end = 0x10091050
```

```
-----
Flanker Instance 00 - Capture Buffer On L2 DBUS:
```

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

```
is_l2_egress: 0x0000, data_size: 0x023
```

```
[000]: 14f4a000 08010000 00000000 6d200800 00006000 00000000 01800100 00000000 00000000 00000000 00003000
```

```
0590 00990000 00000000 00000000 00000005 88405000 00000000 00000000 00000000 00000000 00000000 00000000
```

```
a4 2dbeef00
```

```
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	1b10-vld	: 0x0
1b10-eos	: 0x0	1b10-1b1	: 0x0
1b10-exp	: 0x0	1b10-ttl	: 0x0
1b11-exp	: 0x0	1b11-ttl	: 0x0
ipv4	: 0x0	ipv6	: 0x0
14-protocol	: 0x11		
df	: 0x0		
mf	: 0x0	frag	: 0x0
t11	: 0x1f	13-packet-length	: 0x357
option	: 0x0	tos	: 0x0
sup-eid	: 0x0	header-type	: 0x1
error	: 0x0	redirect	: 0x0
port-id	: 0x0	last-ethertype	: 0x800
12-frame-type	: 0x0	da-type	: 0x0
packet-type	: 0x0	12-length-check	: 0x0
ip-da-multicast	: 0x0	ip-multicast	: 0x0
ip-multicast-control	: 0x0	ids-check-fail	: 0x0
tr	: 0x0	outer-cos	: 0x0
inner-cos	: 0x0	vqi-valid	: 0x0
vqi	: 0x0	packet-length	: 0x369
vlan	: 0x4	destination-index	: 0x0
source-index	: 0x30		
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 : 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	: 0x60	v1 : 0x0
inner-de-valid	: 0x0	de-cfi : 0x0
second-inner-cos	: 0x0	tunnel-type : 0x2

UDP OTV/LISP TUNNEL BNDL

vlan-tag-valid: 0x0	segment-id-valid: 0x0
v1: 0x0	de: 0x0
sgt-valid: 0x0	inner-ip-ttl: 0x0
ip-da-multicast: 0x0	
lisp-inst-id: 0x2c8004	
lisp-flags: 0xc8	isis-mac-da-valid: 0x0

```
type: 0x0
shim-valid      : 0x0
segment-id-valid : 0x0          copp          : 0x0
dti-type-vpnid  : 0x0          segment-id   : 0x0
ib-length-bundle : 0x58840     m1h-type     : 0x5
ulh-type        : 0x4
```

```
source-ipv4-address: 10.0.200.64
```

```
destination-ipv4-address: 10.1.224.67
```

```
mim-destination-mac-address : 0000.0000.0000
```

```
mim-source-mac-address : 0000.0000.0000
```

```
destination-mac-address : 00c1.b1c9.c2c4
```

```
source-mac-address : 000d.0d0d.0d0d
```

由於第3層封包長度用作觸發器，因此可能會無意中擷取背景封包觸發ELAM。因此，必須利用捕獲中的其他欄位對捕獲結果進行雙重交叉檢查。這可確保捕獲的資料包與我們的預期標準相一致，包括源IP(sip)、目標IP(dip)、生存時間(ttl)、源索引等欄位。有趣的是，雖然N7K不支援使用VXLAN VNID作為觸發器，但在輸出直譯器中，欄位「lisp-inst-id: 0x2c8004」對應於VXLAN報頭中的VNID。

```
<#root>
```

```
module-7(fln-l2-elam)# dec
```

```
0x2c8004
```

```
2916356
```

```
Leaf3#
```

```
show system internal epm endpoint ip 172.28.1.20
```

```
MAC : 0000.2222.1202 ::: Num IPs : 1
```


IP# 0 : 172.28.1.20 ::: IP# 0 flags : host-tracked| ::: l3-sw-hit: Yes ::: flags2 :

Vlan id : 186 ::: Vlan vnid : 11494 ::: VRF name : zixu:vrf

BD vnid : 16482209 :::

VRF vnid : 2916356

/* Confirming the VNID from ACI LEAF side */

Phy If : 0x1a00b000 ::: Tunnel If : 0

Interface : Ethernet1/12

Flags : 0x80005c04 ::: sclass : 16388 ::: Ref count : 5

EP Create Timestamp : 01/22/2021 15:42:49.243582

EP Update Timestamp : 02/08/2021 11:26:52.882308

EP Flags : local|IP|MAC|host-tracked|sclass|timer|

module-7(fln-12-elam)#

show rbus

cp = 0x100a96fc, buf = 0x100a96fc, end = 0x100b5a4c

Flanker Instance 00 - Capture Buffer On L2 RBUS:

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

is_l2_egress: 0x0000, data_size: 0x018

[000]: 0015cb30 0000006d 20000000 03000000 00000000 00000000 00000014 2d8000a0 3c3c0000 00000000 020000

0000 00000400 00008000 005d0000 001e0002 2bd7c0cf f96002a0 000000ba

Printing packet 0

L2 RBUS INGRESS CONTENT

pad : 0x572c valid : 0x1
l2-rbus-trigger : 0x1 sequence-number : 0x60
rit-ipv4-id : 0x0 ipv4-tunnel-encap : 0x0

rit-mp1s-rw	: 0x0	m12-ptr	: 0x0
m13-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	m12-delta-length	: 0x0
m12-delta-length-plus	: 0x0	m13-delta-length	: 0x0
m13-delta-length-plus	: 0x0	s-vector	: 0x0
1cpu-ff-valid	: 0x0	sup-di-vqi	: 0x0
erspan-term-index-dir	: 0x0	erspan-buffer-check	: 0x0
12-tunnel-decapped	: 0x0	13-delta-length	: 0x0
rit-crc16-valid	: 0x1	rit-crc16	: 0x42d8
vntag-p	: 0x0	frr-recirc	: 0x0
ingress-lif	: 0x5	ear1-proxy-vld	: 0x0
md-di-vld	: 0x0	rc	: 0x0
segment-id-valid	: 0x0	t11-out	: 0x1e
t11-mid	: 0x1e	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
tr	: 0x0	de	: 0x0
cos	: 0x0	inner-drop-eligibility	: 0x0
inner-cos	: 0x0	acos	: 0x0
di-1t1-index	: 0x3c		
13-multicast-di	: 0x3c		
source-index	: 0x30	vlan	: 0x4
index-direct	: 0x0	di1-valid	: 0x1
vqi	: 0x4a	di2-valid	: 0x0
v5-fpoe-idx	: 0x0	di2-fpoe-idx	: 0x0

```

13-multicast-v5      : 0x0          dft                : 0x0
dfst                 : 0x0          13-learning-ff    : 0x0
result-rbh          : 0x40          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
12-copy-cause       : 0x0          13-rit-ptr        : 0x5d
sg-tag              : 0x0          trill-nh-id       : 0x0
ttl-in              : 0x1e          fc-up             : 0x0
up-did              : 0x0          did               : 0x22bd
up-sid              : 0x0          sid               : 0xf819ff
shim-12-tunnel-encap: 0x0          shim-ls-hash      : 0xb
shim-rc             : 0x0          shim-lif          : 0x5
shim-replication-pkt: 0x0          shim-router-mac   : 0x1
shim-mark-enable    : 0x0          shim-qos-group-id : 0x0
shim-destination-table-index: 0x5d          shim-acos-preserve : 0x0
mim-destination-mac-address : 0000.0000.0000
mim-source-mac-address : 0000.0000.0000

```

```
module-7(f1n-12-elam)#
```

```
show system internal pixmc info 1t1-cb 1t1 0x30
```

```

1t1 | 1t1_type | if_index | 1c_type | vdc | v4_fpoe | v5_fpoe | base_fpoe_idx | flag
0x0030 | 5 | |
Eth7/1
| 2 | 4 | 0x00 | 0x00 | 0x0000 | 0x0

```

```
module-7(fln-12-elam)#
```

```
show system internal pixmc info lt1-cb lt1 0x3c
```

```
lt1 | lt1_type | if_index | lc_type | vdc | v4_fpoe | v5_fpoe | base_fpoe_idx | flag  
0x003c | 5 |  
Eth7/4  
| 2 | 4 | 0x00 | 0x00 | 0x0000 | 0x0
```

如果您嘗試在ACI泛洪BD內擷取廣播、未知單點傳播和多播(BUM)封包，則只需設定靜態ARP並指向主機上不存在的MAC，然後以相同方式啟動PING，此方法也會起作用。

相關資訊

有關如何在不同的N7K LC上使用ELAM的詳細資訊，請參閱以下連結：

- [ELAM概述](#)
- [N7K M系列模組ELAM過程](#)
- [N7K F1模組ELAM過程](#)
- [N7K F2模組ELAM過程](#)
- [N7K M3模組ELAM過程](#)
- [思科技術支援與下載](#)

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。