

Nexus 7000 M シリーズ モジュールの ELAM 手順

目次

[概要](#)

[トポロジ](#)

[入力フォワーディング エンジンの決定](#)

[トリガーの設定](#)

[キャプチャの開始](#)

[結果の解釈](#)

[その他の検証](#)

概要

このドキュメントでは、Cisco Nexus 7000 (N7K) M シリーズ モジュールで ELAN を実行するために使用する手順、最も重要な出力、結果の解釈方法について説明します。

ヒント： ELAM の概要については、「[ELAM の概要](#)」を参照してください。

トポロジ



この例では、VLAN 2500 (10.0.5.101) 上のホストのポート Eth4/1 から VLAN 55 (10.0.3.101) 上のホストのポート Eth3/5 に Internet Control Message Protocol (ICMP) 要求を送信します。ELAM は、10.0.5.101 から 10.0.3.101 へのこの単一パケットをキャプチャするために使用されます。重要な点として、ELAN を使用してキャプチャできるのは単一のフレームです。

N7K で ELAM を実行するには、最初に適切なモジュールに接続する必要があります (このためネットワーク管理者権限が必要です) 。

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

入力フォーディングエンジンの決定

トラフィックはポート Eth4/1 でスイッチに着信することになっています。システム内のモジュールを確認すると、**モジュール 4** は M シリーズ モジュールであることがわかります。N7K は完全分散型であり、データプレーントラフィックの転送に関する決定はスーパーバイザではなくモジュールが行うことに留意してください。

```
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 N7K-M148GT-11       ok
5     0      Supervisor module-1X      N7K-SUP1             active *
6     0      Supervisor module-1X      N7K-SUP1             ha-standby
```

M シリーズ モジュールの場合、**Eureka** という内部コード名が設定されたレイヤ 2 (L2) フォワーディングエンジン (FE) で ELAN を実行します。L2 FE データバス (DBUS) に、L2 およびレイヤ 3 (L3) ルックアップ前の元のヘッダー情報が含まれており、結果バス (RBUS) に L3 および L2 の両方のルックアップ後の結果が含まれていることに注意してください。L3 ルックアップは、**Lamira** という内部コード名が設定された L3/レイヤ 4 (L4) FE によって行われます。このプロセスは、Supervisor Engine 2T を実行する Cisco Catalyst 6500 シリーズ スイッチプラットフォームで使用されるプロセスと同じです。

N7K M シリーズ モジュールでは、モジュールごとに複数の FE を使用できるため、ポート **Eth4/1** に使用される **Eureka** ASIC を判別する必要があります。それには、次のコマンドを実行します。

```
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
```

コマンドの出力を見ると、ポート **Eth4/1** は **Eureka (L2LKP)** インスタンス 0 上にあることがわかります。

注: M シリーズ モジュールの場合、ELAM 構文では 1 を基準とした値を使用するため、インスタンス 0 は、ELAN を設定するときにはインスタンス 1 になります。このことは、F シリーズ モジュールには当てはまりません。

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

トリガーの設定

Eureka ASIC は、IPv4、IPv6、およびその他の ELAM トリガーをサポートします。ELAM トリガーは、フレームタイプに対応している必要があります。フレームが IPv4 フレームである場合、トリガーも IPv4 である必要があります。IPv4 フレームはその他のトリガーではキャプチャされません。IPv6 にも同じ論理が適用されます。

Nexus オペレーティング システム (NX-OS) では、ELAM トリガーの区切り文字として疑問符を使用できます。

```
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
```

送信元 IPv4 アドレスと宛先 IPv4 アドレスに応じてフレームをキャプチャするため、上記の例では送信元 IPv4 アドレスと宛先 IPv4 アドレスの値だけが指定されています。

Eureka では、DBUS および RBUS に対してトリガーが設定されることを要件としています。RBUS データを入れられるパケット バッファ (PB) には、2 つのタイプがあります。どちらのタイプの PB インスタンスが適切であるかは、モジュールタイプと入力ポートによって決まります。一般に、最初に PB1 を設定し、それで RBUS が起動しない場合は PB2 を使用して設定を繰り返すことを推奨します。

次に DBUS トリガーを示します。

```
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-correlate
```

次に RBUS トリガーを示します。

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

注: DBUS トリガーの末尾にある `rbi-correlate` キーワードは、RBUS を cap2 ビットで正しく起動させるために必要です。

キャプチャの開始

入力 FE が選択され、トリガーを設定したら、キャプチャを開始できます。

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

ELAM のステータスを確認するには、**status** コマンドを入力します。

```
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 pbl ip if cap2 1
```

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

```
No configuration
```

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

```
No configuration
```

トリガーに一致するフレームを FE が受信すると、ELM ステータスは **Triggered** と示されます。

```
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 pbl ip if cap2 1
```

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

```
No configuration
```

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

```
No configuration
```

結果の解釈

ELAM の結果を表示するには、**show dbus** コマンドと **show rbus** コマンドを入力します。同じトリガーに大量のトラフィックが一致する場合、DBUS と RBUS が異なるフレームで起動する可能性があります。したがって、DBUS データと RBUS データで内部シーケンス番号を調べて、番号が一致することを確認することが重要となります。

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

```
seq = 0x05
```

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

```
seq = 0x05
```

次に、この例に最も関連性のある ELAM データ出力部分の抜粋を示します (一部の出力は省略されています)。

```
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
```

DBUS データで、送信元 MAC アドレスが d0d0.fdb7.3dc2、宛先 MAC アドレスが 0000.0c07.ac65 となっているフレームが VLAN 2500 で受信されていることを確認できます。また、これは送信元が 10.0.5.101、宛先が 10.0.3.101 の IPv4 フレームであることもわかります。

ヒント： この出力に示されていない有用なその他のフィールド (タイプ オブ サービス (ToS) 値、IP フラグ、IP 長、L2 フレーム長など) があります。

フレームが受信されたポートを検証するには、SRC_INDEX コマンド (送信元の Local Target Logic (LTL)) を入力します。N7K で 1 つのポートまたはポート グループに LTL をマッピングするには、次のコマンドを入力します。

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

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

上記の出力に、0xa21 の SRC_INDEX がポート Eth4/1 にマップすることが示されています。これにより、フレームがポート Eth4/1 で受信されていることを確認できます。

RBUS データで、フレームが VLAN 55 にルーティングされていること、TTL が DBUS データでの 0xff から RBUS データでの 0xfe に減少していることを確認できます。送信元および宛先 MAC アドレスが、それぞれ 8478.ac0e.4741 と 0005.73a9.5541 に書き換えられていることも確認できます。さらに、DEST_INDEX (宛先 LTL) からの出力ポートも確認できます。

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

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

上記の出力には、0x9ed の DEST_INDEX がポート Eth3/5 にマップすることが示されています。これにより、フレームがポート Eth3/5 から送信されていることを確認できます。

その他の検証

スイッチにより LTL プールがどのように割り当てられているかを検証するには、show system internal pixm info ltl-region コマンドを入力します。このコマンドの出力は、LTL が物理ポートに一致しない場合に LTL の目的を理解する上で役立ちます。Drop LTL がその良い例です。

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