

Nexus 7000 F1 モジュールの ELAM 手順

目次

[概要](#)

[トポロジ](#)

[入カフォワーディング 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) F1 モジュールで ELAM を実行する手順、最も関連性のある出力、結果の解釈方法を説明します。

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 10 上のホスト (10.1.1.101、MAC アドレス 0050.56a1.1a01) がポート Eth3/18 から Internet Control Message Protocol (ICMP) 要求をやはり VLAN 10 上のホスト (10.1.1.102、MAC アドレス 0050.56a1.1aef) のポート Eth3/26 に送信します。10.1.1.101 から 10.1.1.102 への 1 つのフレームをキャプチャするため ELAM が使用されます。ELAM により、キャプチャできるフレームが 1 つだけであることに留意してください。

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 3
Attaching to module 3 ...
To exit type 'exit', to abort type '$.'
module-3#
```

入カフォワーディング エンジンの決定

トラフィックはポート Eth3/18 からスイッチに入ると想定されます。システム内のモジュールを確認すると、Module 3 が F1 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 3
Mod  Ports  Module-Type                Model                Status
-----
3    32      1/10 Gbps Ethernet Module  N7K-F132XP-15      ok
```

F1 モジュールで、内部コード名 **Orion** のレイヤ 2 (L2) フォワーディング エンジン (FE) で ELAM を実行します。N7K F1 にはモジュールあたり 16 個の FE があるため、ポート **Eth3/18** でこの FE に対して使用する **Orion ASIC** を決定する必要があります。確認するには、次のコマンドを入力します。

```
module-3# show hardware internal dev-port-map
(some output omitted)
-----
CARD_TYPE:          DCE 32 port 10G
>Front Panel ports:32
-----
Device name          Dev role              Abbr num_inst:
-----
>Orion Fwding Driver  DEV_LAYER_2_LOOKUP   L2LKP  16
+-----+
+-----+++FRONT PANEL PORT TO ASIC INSTANCE MAP+++-----+
+-----+
FP port |  PHYS |  MAC_0 |  L2LKP |  QUEUE | SWICHF
...
   18   8     8     8     8     1
```

出力から、ポート **Eth3/18** は **Orion (L2LKP)** インスタンス **8** にあることを確認できます。

```
module-3# elam ASIC orion instance 8
module-3(orion-elam)#
```

トリガーの設定

N7K プラットフォーム上の他の FE と比べて、**Orion ASIC** の ELAM トリガーは非常に限られています。これは、この F1 が L2 専用モジュールであるためです。したがって、スイッチング決定は、MAC アドレス情報 (または FabricPath 環境の SwitchID) に基づいて行われます。

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-3(orion-elam)# trigger di field ?
da          Destination mac-address
mim_da     Destination mac-in-mac-address
mim_sa     Source mac-in-mac-address
sa         Source mac-address
vlan
```

この例では、入力判断ブロックの送信元 MAC アドレスと宛先 MAC アドレスに基づいてフレームがキャプチャされます。

注: この F1 モジュールに別個の DBus トリガーと RBus トリガーは必要ありません。

ここで、トリガーを示します。

```
module-3(orion-elam)# trigger di field sa 0050.56a1.1a01 da 0050.56a1.1aef
```

キャプチャの開始

この F1 モジュールは、トリガーが設定された直後に ELAM が開始するので、他の N7K モジュールとは異なります。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-3(orion-elam)# status
Armed
```

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-3(orion-elam)# status
Triggered
```

結果の解釈

ELAM の結果を表示するには、**show capture** コマンドを入力します。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-3(orion-elam)# show capture
dc3v4_si[11:0]      :                17
vlanx              :                a
di                 :                1e or 1f
res_eth_da         :                5056a11aef
res_eth_sa         :                5056a11a01
```

注: F1 モジュールでは、転送の決定に使用される ELAM データと、転送結果を含むデータが同じ出力にバインドされます。また、ELAM 出力の MAC アドレス形式には、先頭に追加されるゼロが含まれていないことに注意してください。

```
Destination MAC (res_eth_da) 5056a11aef = 0050.56a1.1aef
Source MAC      (res_eth_sa) 5056a11a01 = 0050.56a1.1a01
```

この出力では、送信元 Local Target Logic (LTL) (**dc3v4_si**)、宛先 LTL (**DI**)、VLAN (**vlanx**)、および送信元 MAC アドレスと宛先 MAC アドレス (それぞれ、**5056a11a01** および **5056a11aef**) を確認できます。

送信元 LTL (**dc3v4_si**) はフレームを受信したポートを表します。F1 ELAM は、宛先 LTL の結果が 2 つ表示されています (**1e** または **1f**)。このようなあいまいな結果が生じるのは、ELAM パーサーが ELAM データの最下位ビットを読み取ることができないためです。このため、宛先アドレスのハードウェア MAC アドレス エントリを確認し、ELAM の宛先 LTL を使って確認することをお勧めします。

```
N7K# show system internal pixm info ltl 0x17
Type                LTL
```

```
-----
PHY_PORT            Eth3/18
```

上記の出力には、**0x17** の送信元 LTL がポート **Eth3/18** にマップすることが表示されています。

これにより、フレームがポート **Eth3/18** で受信されたことを確認できます。

```
module-3# show hardware mac address-table fe 8
address 0050.56a1.1aef vlan 10 vdc 1
```

(some output omitted)

```
FE | Valid| PI|  BD |      MAC      | Index|
  |      |  |   |              |      |
-----+-----+-----+-----+-----+-----+
8   1    0  34   0050.56a1.1aef  0x0001f
```

```
N7K# show system internal pixm info ltl 0x1f
Type                LTL
-----
```

```
PHY_PORT            Eth3/26
```

この出力で、**Orion**インスタンス **8** (**Eth3/18** の転送に関する決定を下す FE) に、宛先 MAC アドレス **0050.56a1.1aef** に対するハードウェア MAC アドレス エントリ **0x1f** が設定されていることを確認できます。このインデックスは、F1 ELAM データ内の宛先 LTL (di) でもあります。

また、LTL **0x1f** がポート **Eth3/26** にマップすることも確認できます。これにより、フレームがポート **Eth3/26** 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 2048          0x2000 to 0x27ff
=====> UCAST MCAST BOUNDARY <=====
```

