

# 连结7000 M系列模块伊拉姆步骤

## 目录

[简介](#)

[拓扑](#)

[确定入口转发引擎](#)

[配置触发](#)

[开始捕获](#)

[解释结果](#)

[另外的验证](#)

## 简介

本文描述用于的步骤为了执行思科连结的伊拉姆7000个(N7K) M系列模块，解释最相关的输出，并且描述如何解释结果。

**提示：**参考一概述的[伊拉姆概述文件](#)在伊拉姆。

## 拓扑



在本例中，在VLAN 2500的一台主机(10.0.5.101)，端口Eth4/1发送互联网控制消息协议(ICMP)请求到在VLAN 55的一台主机(10.0.3.101)，端口Eth3/5。伊拉姆用于为了获取从10.0.5.101的此单个数据包到10.0.3.101。请记住伊拉姆允许您捕获单个帧。

为了执行N7K的伊拉姆，您必须首先连接到适当的模块(这要求网络Admin权限)：

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

## 确定入口转发引擎

流量预计对入口在端口Eth4/1的交换机。当您检查在系统时的模块，您看到**模块4**是M系列模块。请记住N7K是充分分布式，并且模块，不是Supervisor，做出dataplane流量的转发决策。

```
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
```

对于M系列模块，请执行Layer2 (L2)转发引擎的(FE)伊拉姆与内部共同的书目分类编号**尤里卡**。注意L2 FE数据总线(DBUS)在L2和第3层(L3)查找前包含原始报头题头信息，并且结果总线(RBUS)在L3和L2以后包含结果查找。L3查找由L3/Layer 4 (L4) FE执行与内部共同的书目分类编号**Lamira**，是在Cisco Catalyst 6500系列交换机平台使用的同一进程运行Supervisor引擎2T。

N7K M系列模块能使用多个FEs每个模块，因此您必须确定使用在端口Eth4/1的FE的**尤里卡**ASIC。输入此命令为了验证此：

```
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
```

在输出中，您能看到端口Eth4/1在**0尤里卡(L2LKP)**实例。

**注意：**对于M系列模块，当您配置伊拉姆时，伊拉姆语法用途基于1值，因此实例0变为实例1。这不是F系列模块的盒。

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

## 配置触发

**尤里卡**ASIC支持IPv4,IPv6的伊拉姆触发和其他。伊拉姆触发必须与帧类型对齐。如果帧是IPv4帧，则触发必须也是IPv4。IPv4帧没有用**其他**触发捕获。同样逻辑适用于IPv6。

使用连结操作系统(NX-OS)，您能使用问号字符为了分离伊拉姆触发：

```
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

对于此示例，帧根据源和目的IPv4地址捕获，那么仅那些值指定。

尤里卡要求触发为DBUS和RBUS设置。有两不同数据包缓冲区(PB) RBUS数据能驻留。正确PB实例的确定取决于确切的模块类型和入站端口。一般，推荐您配置PB1，并且，如果RBUS不触发，然后请重复与PB2的配置。

这是DBUS触发：

```
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-corelate
```

这是RBUS触发：

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

**注意：**rbi相互关系关键字在DBUS触发结束时在cap2位要求为了RBUS能正确地触发。

## 开始捕获

即然入口FE选择，并且您配置触发，您能开始捕获：

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

为了检查ELAM的状态，请输入status命令：

```
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 pb1 ip if cap2 1
```

LM-DBUS: Dis-Armed

No configuration

LM-RBUS: Dis-Armed

No configuration

一旦匹配触发的帧由FE接收，伊拉姆状态显示如被触发：

```
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 pb1 ip if cap2 1
```

LM-DBUS: Dis-Armed

No configuration

LM-RBUS: Dis-Armed

No configuration

## 解释结果

为了显示伊拉姆结果，请输入**显示d总线并且显示rbus**命令。如果有该大容积的流量匹配同样触发，DBUS，并且RBUS在不同的帧也许触发。所以，检查在DBUS和RBUS数据的内部序号为了保证是重要的他们配比：

```
module-4(eureka-elam)# show dbus | i seq
seq = 0x05
module-4(eureka-elam)# show rbus | i seq
seq = 0x05
```

这是摘自与此示例是最相关的伊拉姆数据的部分(若干输出省略)：

```
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
```

有DBUS数据，您能验证帧在与d0d0.fdb7.3dc2源MAC地址和0000.0c07.ac65目标MAC地址的VLAN 2500接收。您能也看到这是从10.0.5.101来源的IPv4帧和被注定对10.0.3.101。

**提示：**有在此输出中没有包括，例如服务类型(ToS)值、IP标志、IP长度和L2帧长度的几个其他有用的字段。

为了验证在哪个端口帧接收，请输入**SRC\_INDEX**命令(来源Local Target Logic (LTL))。输入此命令为了映射LTL对端口或端口组N7K的：

```
N7K# show system internal pixm info ltl 0xa21
Member info
-----
Type                LTL
-----
PHY_PORT            Eth4/1
FLOOD_W_FPOE        0x8014
```

输出显示那0xa21地图SRC\_INDEX到端口Eth4/1。这确认帧在端口Eth4/1接收。

有RBUS数据，您能验证帧路由对VLAN 55，并且TTL从在DBUS数据的0xff减少对在RBUS数据的0xfe。您能分别看到源及目的地MAC地址重写对8478.ac0e.4741和0005.73a9.5541。另外，您能确认从DEST\_INDEX(目的地LTL)的输出端口：

```
N7K# show system internal pixm info ltl 0x9ed
Member info
-----
Type                LTL
-----
PHY_PORT            Eth3/5
```

```
FLOOD_W_FPOE 0x8017
FLOOD_W_FPOE 0x8016
```

输出显示那0x9ed地图DEST\_INDEX到端口Eth3/5。这确认帧从端口Eth3/5发送。

## 另外的验证

为了验证交换机如何分配LTL池，请输入show system内部pixm信息LTL赤道区命令。如果没有匹配到物理端口，从此命令的输出是有用的为了了解LTL的目的。此的好的实例是丢弃LTL：

```
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
<b>Drop DI</b>	<b>2</b>	<b>0x119f to 0x11a0</b>
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