

# 目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[Ethanalyzer](#)

## 简介

本文描述如何使用内置的数据包捕获工具，Ethanalyzer，在连结3000/5000/7000交换机。

## 先决条件

### 要求

本文档没有任何特定的要求。

### 使用的组件

本文档中的信息根据连结3000，连结5000和连结7000交换机。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

## Ethanalyzer

Ethanalyzer是排除故障控制层面和流量的有用工具被注定交换CPU。Mgmt是排除故障押mgmt0接口的数据包接口。入站LOW (eth3)是为低优先级的(ping、telnet，Secure Shell) CPU限定的数据流，和入站喂(eth4)是为高优先权(生成树协议，网桥协议数据单元，FIP) CPU限定的数据流。

**注意：**您能使用显示过滤器或捕获过滤器作为选项。显示过滤器选项在连结5000更喜欢，并且捕获过滤器在连结3000和连结7000更喜欢。

常用的显示过滤器可以在[Wireshark](#)找到

常用的捕获过滤器可以在[Wireshark](#)找到

**注意：**因为连结5000使用内部VLAN传送帧，Ethanalyzer有内部VLAN。连结5000传送根据内部VLAN和Ethanalyzer的帧显示内部VLAN。当您排除故障与Ethanalyzer时，VLAN ID能造成

困难。然而，您能使用show system命令内部fcfwd fwcvidmap cvid为了确定映射。下面是一个示例。

```
Nexus# ethanalyzer local interface inbound-low detail display-filter icmp
Capturing on eth3
Frame 16 (102 bytes on wire, 102 bytes captured)
  Arrival Time: Sep 7, 2011 15:42:37.081178000
  [Time delta from previous captured frame: 0.642560000 seconds]
  [Time delta from previous displayed frame: 1315424557.081178000 seconds]
  [Time since reference or first frame: 1315424557.081178000 seconds]
  Frame Number: 16
  Frame Length: 102 bytes
  Capture Length: 102 bytes
  [Frame is marked: False]
  [Protocols in frame: eth:vlan:ip:icmp:data]
Ethernet II, Src: 00:0d:ec:a3:81:bc (00:0d:ec:a3:81:bc),
Dst: 00:05:73:ce:3c:7c (00:05:73:ce:3c:7c)
  Destination: 00:05:73:ce:3c:7c (00:05:73:ce:3c:7c)
    Address: 00:05:73:ce:3c:7c (00:05:73:ce:3c:7c)
      .... .0 .... = IG bit: Individual address (unicast)
      .... .0 .... = LG bit: Globally unique address(factory default)
  Source: 00:0d:ec:a3:81:bc (00:0d:ec:a3:81:bc)
    Address: 00:0d:ec:a3:81:bc (00:0d:ec:a3:81:bc)
      .... .0 .... = IG bit: Individual address (unicast)
      .... .0 .... = LG bit: Globally unique address(factory default)
  Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN
  000. .... = Priority: 0
  ...0 .... = CFI: 0
  ... 0000 0011 1001 = ID: 57 <<-----
  Type: IP (0x0800)
Internet Protocol, Src: 144.1.1.63 (144.1.1.63), Dst: 144.1.1.41 (144.1.1.41)
  Version: 4
  Header length: 20 bytes
  Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    0000 00.. = Differentiated Services Codepoint: Default (0x00)
      .... .0. = ECN-Capable Transport (ECT): 0
      .... ..0 = ECN-CE: 0
  Total Length: 84
  Identification: 0x1118 (4376)
<snip>
```

正如你看到的Ethanalyzer表明数据包在VLAN 57接收，是内部VLAN。然而，因为57不在十六进制，VLAN 57不是实际VLAN。57在十六进制是0x0039。此命令确定在十六进制的实际VLAN。

```
Nexus# show system internal fcfwd fwcvidmap cvid | grep 0x0039
0x0039 enet 0x01 0x0090 0100.0000.080a 0100.0000.0809
0x0039 fc 0x01 0x0090 0100.0000.0007 0100.0000.0006
```

0x0090是在十六进制的实际VLAN。您必须然后转换编号到十进制，是144。此计算说明在上一个帧的实际VLAN是VLAN 144，虽然Ethanalyzer指示是57。

这是捕获FIP帧用显示过滤器VLAN.(etype==0x8914)的示例

```
Nexus# ethanalyzer local interface inbound-hi display-filter vlan.etype==0x8914
Capturing on eth4
2011-10-18 13:36:47.047492 00:c0:dd:15:d4:41 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:48.313531 00:c0:dd:15:d0:95 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:49.373483 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
```

```

2011-10-18 13:36:49.373868 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:49.374131 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:49.374378 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:49.374618 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:49.374859 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:49.375098 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
2011-10-18 13:36:49.375338 00:0d:ec:a3:81:80 -> 01:10:18:01:00:01 0x8914
PRI: 3 CFI: 0 ID: 56
10 packets captured
Program exited with status 0.

```

Nexus#

这是捕获从一特定的CNA的示例(vFC1311的FKA帧附加对Po1311)。此配置造成Ethalyzer发现从主机的FKA每八秒，是FKA计时器。

Nexus# **show flogi database**

```

-----
INTERFACE VSAN FCID PORT NAME NODE NAME
-----
vfc15 200 0x1e0000 50:0a:09:81:89:4b:84:32 50:0a:09:80:89:4b:84:32
vfc16 200 0x1e0003 50:0a:09:81:99:4b:84:32 50:0a:09:80:89:4b:84:32
vfc17 200 0x1e0002 21:00:00:c0:dd:12:b9:b7 20:00:00:c0:dd:12:b9:b7
vfc18 200 0x1e0006 21:00:00:c0:dd:14:6a:73 20:00:00:c0:dd:14:6a:73
vfc19 200 0x1e0001 21:00:00:c0:dd:11:00:49 20:00:00:c0:dd:11:00:49
vfc20 200 0x1e0007 21:00:00:c0:dd:12:0e:37 20:00:00:c0:dd:12:0e:37
vfc23 200 0x1e0004 10:00:00:00:c9:85:2d:e5 20:00:00:00:c9:85:2d:e5
vfc1311 200 0x1e0008 10:00:00:00:c9:9d:23:73 20:00:00:00:c9:9d:23:73

```

Total number of flogi = 8.

Nexus# **ethalyzer local interface inbound-hi display-filter "eth.addr==00:00:c9:9d:23:73 && vlan.etype==0x8914 && frame.len==60"limit-captured-frames 0**  
**Capturing on eth4**

```

2011-10-22 11:06:11.352329 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:06:19.352116 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:06:27.351897 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:06:35.351674 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:06:43.351455 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:06:51.351238 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:06:59.351016 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:07:07.350790 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:07:15.350571 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:07:23.350345 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:07:31.350116 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:07:39.349899 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:07:47.349674 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914

```

```
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:07:55.349481 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:08:03.349181 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:08:11.348965 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:08:19.348706 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:08:27.348451 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
2011-10-22 11:08:35.348188 00:00:c9:9d:23:73 -> 00:0d:ec:a3:81:80 0x8914
PRI: 3 CFI: 0 ID: 24
52 packets dropped
```

Nexus# 19 packets captured

上一个捕获只显示报头。您可能也打印详细信息数据包;但是,当您使用详细信息选项时,写捕获到文件然后打开有Wireshark的文件是最佳的。

```
Nexus# ethanalyzer local interface inbound-hi detail display-filter
vlan.etype==0x8914 write bootflash:flogi.pcap ?
<CR>
>Redirect it to a file
>>Redirect it to a file in append mode
display Display packets even when writing to a file
| Pipe command output to filter
```

这是捕获LACP帧的示例:

```
Nexus# ethanalyzer local interface inbound-hi display-filter slow
Capturing on eth42011-12-05 12:00:08.472289 00:0d:ec:a3:81:92 -> 01:80:c2:00:00:02 LACP Link
Aggregation Control ProtocolVersion 1. Actor Port = 16651 Partner Port = 283
2011-12-05 12:00:16.944912 00:1d:a2:00:02:99 -> 01:80:c2:00:00:02 LACP Link
Aggregation Control ProtocolVersion 1. Actor Port = 283 Partner Port = 16651
2011-12-05 12:00:25.038588 00:22:55:77:e3:ad -> 01:80:c2:00:00:02 LACP Link
Aggregation Control ProtocolVersion 1. Actor Port = 16666 Partner Port = 16643
2011-12-05 12:00:25.394222 00:1b:54:c1:94:99 -> 01:80:c2:00:00:02 LACP Link
Aggregation Control ProtocolVersion 1. Actor Port = 282 Partner Port = 16644
2011-12-05 12:00:26.613525 00:0d:ec:8f:c9:ee -> 01:80:c2:00:00:02 LACP Link
Aggregation Control ProtocolVersion 1. Actor Port = 295 Partner Port = 295
2011-12-05 12:00:26.613623 00:0d:ec:8f:c9:ef -> 01:80:c2:00:00:02 LACP Link
Aggregation Control ProtocolVersion 1. Actor Port = 296 Partner Port = 296
```

这是捕获所有帧的示例来源与MAC地址00:26:f0 (通配符过滤器)。

```
Nexus# ethanalyzer local interface inbound-hi display-filter
"eth.src[0:3]==00:26:f0" limit-captured-frames 0
Capturing on eth4
2012-06-20 16:37:22.721291 00:26:f0:05:00:00 -> 01:80:c2:00:00:00 STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
2012-06-20 16:37:22.721340 00:26:f0:05:00:00 -> 01:00:0c:cc:cc:cd STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
2012-06-20 16:37:22.721344 00:26:f0:05:00:00 -> 01:00:0c:cc:cc:cd STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
2012-06-20 16:37:22.721348 00:26:f0:05:00:00 -> 01:00:0c:cc:cc:cd STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
19 packets dropped
Nexus# 4 packets captured
```

**注意:** 在上一个输出中,您看到"19被丢弃的数据包"。这些数据包没有由Ethanalyzer实际上丢弃,然而没有捕获。

确保您选择适当的CPU队列(入站喂,入站lo或者mgmt)。

这是普通的流量类型和队列：

- 入站LOW -监听SUP LOW (eth3) (在交换机虚拟接口的地址解析服务(ARP) /IP的互联网组管理协议)
- 入站喂- SUP海伊(eth4) (STP , FIP , 在以太网(FCoE)的光纤信道 , FC , Cisco发现协议 , 链路层发现协议/数据中心桥接功能开关协议、链路汇聚控制协议 , 单向链路检测)
- Mgmt -带外(任何通过mgmt0接口)
- FIP (结构洛金 , 清楚虚链路 , FKA) : VLAN.etype==0x8914
- FCoE (波尔特洛金 , 域名系统) : VLAN.etype==0x8906

这是捕获FIP和FCoE的示例：

```
Nexus# ethanalyzer local interface inbound-hi display-filter
"eth.src[0:3]==00:26:f0" limit-captured-frames 0
Capturing on eth4
2012-06-20 16:37:22.721291 00:26:f0:05:00:00 -> 01:80:c2:00:00:00 STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
2012-06-20 16:37:22.721340 00:26:f0:05:00:00 -> 01:00:0c:cc:cc:cd STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
2012-06-20 16:37:22.721344 00:26:f0:05:00:00 -> 01:00:0c:cc:cc:cd STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
2012-06-20 16:37:22.721348 00:26:f0:05:00:00 -> 01:00:0c:cc:cc:cd STP Conf.
Root = 8192/d0:57:4c:b7:dc:00 Cost = 200 Port = 0x9004
19 packets dropped
Nexus# 4 packets captured
```

这是一些ARP过滤器：

```
Nexus# ethanalyzer local interface inbound-low display-filter
arp.src.hw_mac==0013.8066.8ac2
Capturing on eth3
2012-07-12 21:23:54.643346 00:13:80:66:8a:c2 ->
ff:ff:ff:ff:ff:ff ARP Who has 172.18.121.59? Tell 172.18.121.1
```

NexusF340.24.10-5548-2# 1 packets captured

```
Nexus# ethanalyzer local interface inbound-low display-filter
arp.src.proto_ipv4==172.18.121.4
Capturing on eth3
2012-07-12 21:25:38.767772 00:05:73:ab:29:fc ->
ff:ff:ff:ff:ff:ff ARP Who has 172.18.121.1? Tell 172.18.121.4
```