Nexus 7000 Ethanalyzer 案例分析

目录

Ethanalyzer简介

对于传统的路由交换设备, 当我们在排查控制平面的问题, 比如High CPU或路由协议等相关的问题时, 我们经常需要通过SPAN或其它复杂的debug命令, 把相关的报文截取下来进行更多的分析 在Cisco全新一代的Nexus7000平台,我们可以通过一个内置的工具Ethanalyzer,轻松的在线实现这一功能。
Ethanalyzer是基于开源的Wireshark(Ethereal)代码,集成在NX-OS软件中的,基于命令行的协议分析软件, 不但可以抓取控制平面的协议报文交互, 而且可以进行在线的解码分析。

基本语法

Command	Purpose
ethanalyzer local interface	Captures packets sent or received by the supervisor and provides detail protocol information.
ethanalyzer local interface inband	Captures packets sent or received by the supervisor and provides detail protocol information in the inband and outband interfaces.
ethanalyzer local interface mgmt	Captures packets sent or received by the supervisor and provides detail protocol information in the management interfaces.
ethanalyzer local interface {inband | mgmt} brief	Captures packets sent or received by the supervisor and provides a sun of protocol information.
ethanalyzer local interface {inband | mgmt} limit-captured-frames	Limits the number of frames to capture.
ethanalyzer local interface {inband | mgmt} limit-frame-size	Limits the length of the frame to capture.
ethanalyzer local interface {inband | mgmt} capture-filter	Filters the types of packets to capture.
ethanalyzer local interface {inband | mgmt} display-filter	Filters the types of captured packets to display.
ethanalyzer local interface {inband | mgmt} decode-internal	Decodes the internal frame header for Cisco NX-OS. **Note** Do not use this option if you plan to analyze the data using Wiresh instead of Ethanalyzer.
ethanalyzer local interface {inband | mgmt} write	Saves the captured data to a file.
ethanalyzer local read	Opens the captured data file and analyzes it.

案例介绍

案例1

检查Nexus7000是否收到从192.1.37.37 发来的PING请求。

```
N7k-1# ethanalyzer local interface inband capture-filter  "host 192.1.37.37 and icmp"

Capturing on inband
2011-09-23 10:21:22.709072   192.1.37.37 -> 192.1.37.1    ICMP Echo (ping) request
2011-09-23 10:21:22.709314    192.1.37.1 -> 192.1.37.37  ICMP  Echo (ping) reply
```

从这里,我们能够清晰地看到,Nexus7000(192.1.37.1)收到了来自192.1.37.37的PING请求, 并且进行了回应。

案例2

详细的解码从192.1.37.37 发来的PING请求的报文格式。

```
N7k-1# ethanalyzer local interface inband capture-filter  "host 192.1.37.37 and icmp[icmptype] =
icmp-echo" detail

Capturing on inband
Frame 1 (146 bytes on wire, 114 bytes captured)
  Arrival Time: Sep 23,  2011 10:31:43.765224000
  [Time delta from  previous captured frame: 0.000000000 seconds]
  [Time delta from  previous displayed frame: 0.000000000 seconds]
  [Time since reference  or first frame: 0.000000000 seconds]
  Frame Number: 1
  Frame Length: 146  bytes
  Capture Length: 114  bytes
  [Frame is marked:  False]
  [Protocols in frame:  eth:ip:icmp:data]
Ethernet II, Src: c4:7d:4f:62:81:45 (c4:7d:4f:62:81:45), Dst:  00:26:98:09:1f:c1
(00:26:98:09:1f:c1)
  Destination:  00:26:98:09:1f:c1 (00:26:98:09:1f:c1)
    Address:  00:26:98:09:1f:c1 (00:26:98:09:1f:c1)
    .... ...0 ....  .... .... .... = IG bit: Individual address (unicast)
    .... ..0. ....  .... .... .... = LG bit: Globally unique address (factory default)
  Source:  c4:7d:4f:62:81:45 (c4:7d:4f:62:81:45)
    Address:  c4:7d:4f:62:81:45 (c4:7d:4f:62:81:45)
    .... ...0 ....  .... .... .... = IG bit: Individual address (unicast)
    .... ..0. ....  .... .... .... = LG bit: Globally unique address (factory default)
  Type: IP (0x0800)
Internet Protocol, Src: 192.1.37.37 (192.1.37.37), Dst:  192.1.37.1 (192.1.37.1)
  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: 100
  Identification: 0x0231 (561)
  Flags: 0x00
    0.. = Reserved  bit: Not Set
    .0. = Don't  fragment: Not Set
    ..0 = More  fragments: Not Set
  Fragment offset: 0
  Time to live: 255
  Protocol: ICMP (0x01)
  Header checksum: 0xef3e  [correct]
    [Good: True]
    [Bad : False]
  Source: 192.1.37.37  (192.1.37.37)
  Destination:  192.1.37.1 (192.1.37.1)
```

```
Internet Control Message Protocol
  Type: 8 (Echo (ping)  request)
Code: 0 ()
Checksum: 0xd31b  [correct]
Identifier: 0x007c
Sequence number: 0  (0x0000)
Data (72 bytes)

0000  00 00 00 00 b7 a3 f3  0e ab cd ab cd ab cd ab cd    ................
0010  ab cd ab cd ab cd ab  cd ab cd ab cd ab cd ab cd    ................
0020  ab cd ab cd ab cd ab  cd ab cd ab cd ab cd ab cd    ................
0030  ab cd ab cd ab cd ab  cd ab cd ab cd ab cd ab cd    ................
0040  ab cd ab cd ab cd ab  cd                           ........
    Data:  00000000B7A3F30EABCDABCDABCDABCDABCDABCDABCDABCD...
    [Length: 72]


N7k-1# 1 packet captured
```

案例3

把Ethanalyzer捕获的报文存储在本地。

```
N7k-1# ethanalyzer local interface inband limit-captured-frames  100 write bootflash:zixu.cap

Capturing on inband
100
Program exited with status 0.
```

通过这个命令,我们将Ethanalyzer捕获到的报文存储在Bootflash上的zixu.cap文件中。

案例4

搜寻捕获的文件中,是否有从192.1.37.37 发来的PING包。

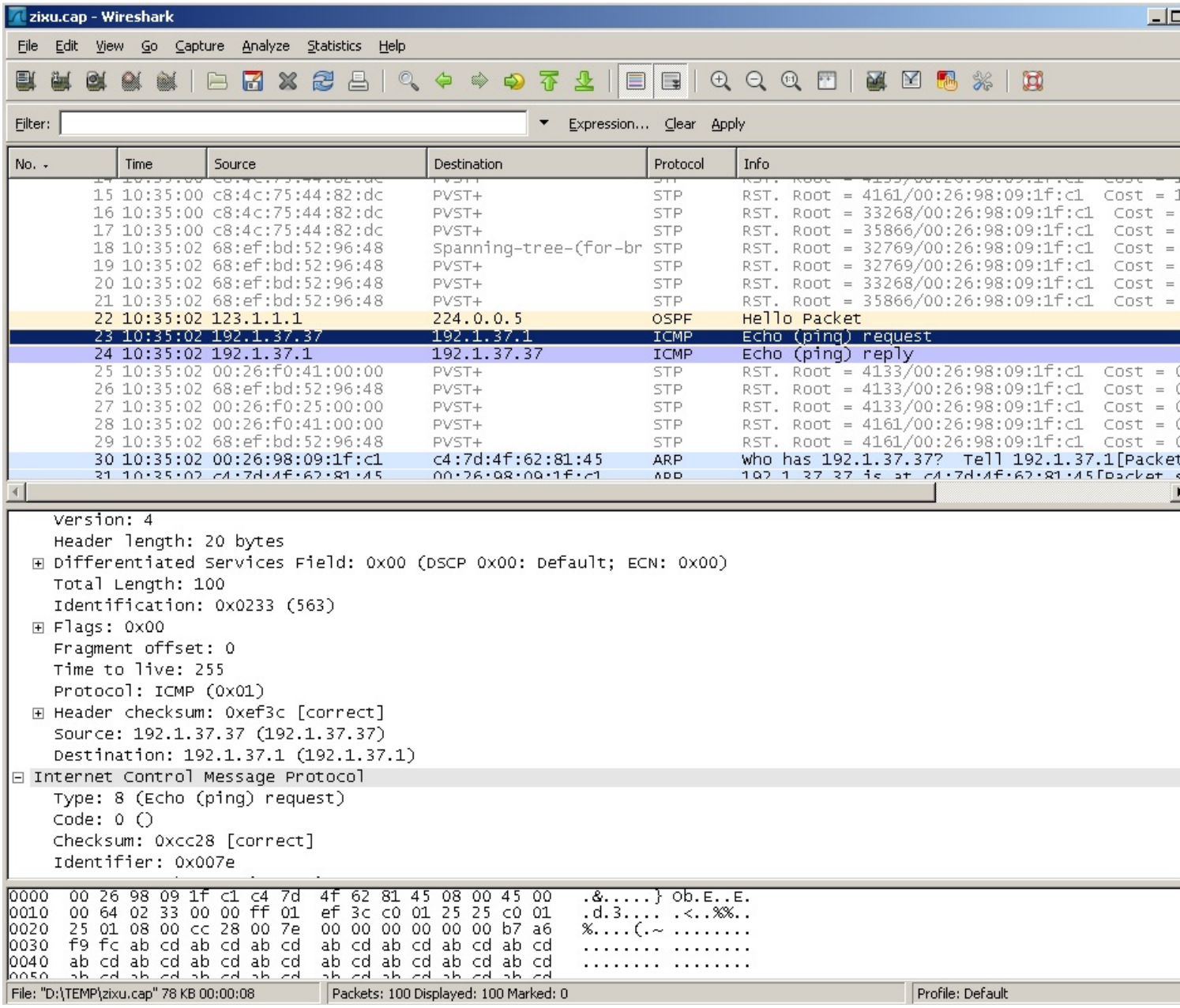
```
N7k-1# ethanalyzer local read bootflash:zixu.cap display-filter  "ip.src==192.1.37.37 &&
icmp.type==8

2011-09-23 10:35:02.136354   192.1.37.37 -> 192.1.37.1   ICMP  Echo (ping) request
2011-09-23 10:35:08.336327   192.1.37.37 -> 192.1.37.1    ICMP Echo (ping) request
Program exited with status 0.
```

案例5

把捕获的文件,上传到电脑,通过图形化的Wireshark进行分析.

```
N7k-1# copy bootflash:zixu.cap tftp://zixu-wxp vrf management
Trying to connect to tftp server......
Connection to Server Established.
TFTP put operation was successful
Copy complete, now saving to disk (please wait)...
```

然后,我们就可以在电脑上用Wireshark打开这个文件进行分析了。



相关知识

从上面的案例介绍中我们可以看到,对于Ethanalyzer,捕获报文的语法和显示报文的语法是不同的, 这一点和Wireshark是完全一致的。
举例来说, 假如我们只想看到从某台主机发来的PING请求,
对应的捕获语法为:

```
ethanalyzer  local interface inband capture-filter "host x.x.x.x and icmp[icmptype] =  icmp-echo"
```

显示语法为:

```
ethanalyzer  local read bootflash:xxx.cap display-filter "ip.src==x.x.x.x &&  icmp.type==8"
```

这是因为Ethanalyzer的捕获语法是基于TCPDUMP, 而显示语法是基于Wireshark, 从下面的链接, 可以找到更多的关于这两种语法的详细信息。

TCPDUMP:
http://www.tcpdump.org/tcpdump_man.html

Wireshark:
http://wiki.wireshark.org/DisplayFilters