Use Troubleshoot Guide for Ethanalyzer on Nexus 7000

Contents

Introduction

This document describes the Ethanalyzer, a Cisco NX-OS integrated packet capture tool for control packets based upon Wireshark.

Background Information

Wireshark is an open-source, network protocol analyzer widely used across many industries and educational institutions. It decodes packets captured by libpcap, the packet capture library. Cisco NX-OS runs on top of the Linux kernel, which uses the libpcap library in order to support packet capture.

With Ethanalyzer, you can:

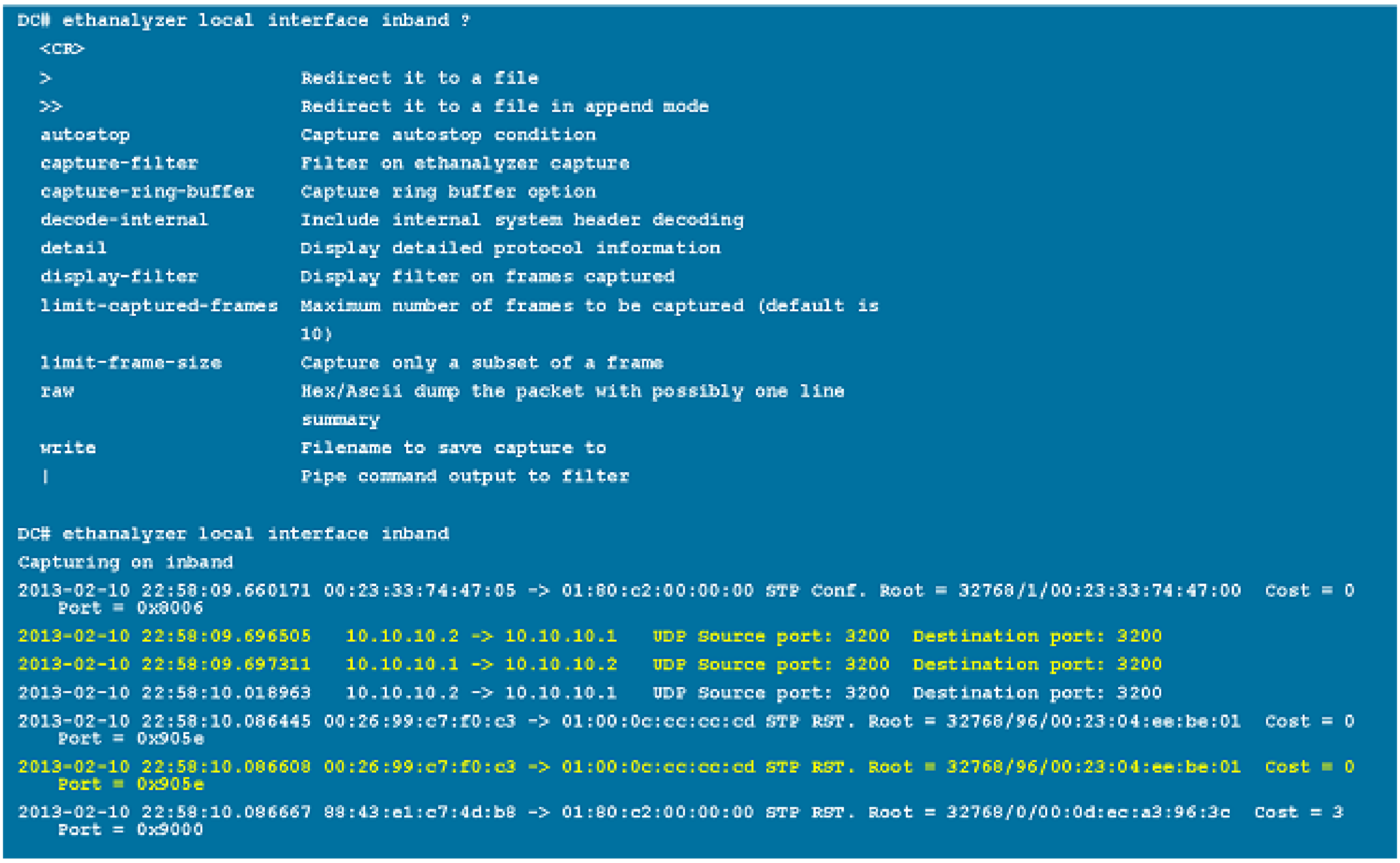
- Capture packets sent or received by the Supervisor.
- Set the number of packets to be captured.
- Set the length of the packets to be captured.
- Display packets with summary or detailed protocol information.
- Open and save packet data captured.
- Filter packets captured on many criteria.
- Filter packets to be displayed on many criteria.
- Decode the internal 7000 header of the control packet.

Ethanalyzer cannot:

- Warn you when your network experiences problems. However, Ethanalyzer can help you determine the cause of the problem.
- Capture data plane traffic that is forwarded in hardware.
- Support interface-specific capture.

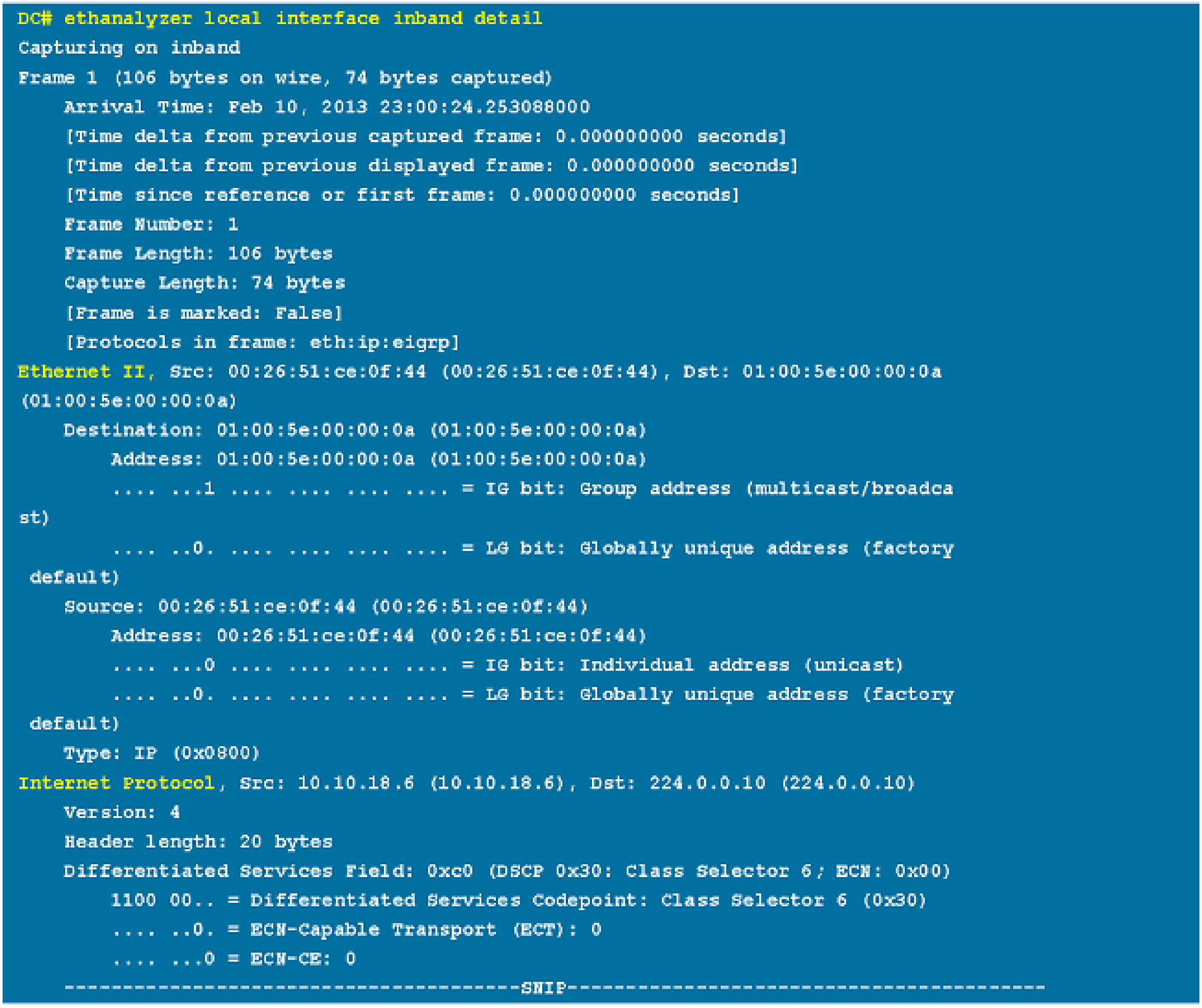
Output Options

This is a summary view of output from the **ethanalyzer local interface inband** command. The ? option displays help.



```
DC# ethanalyzer local interface inband ?
  <CR>
  >                       Redirect it to a file
  >>                      Redirect it to a file in append mode
  autostop                Capture autostop condition
  capture-filter          Filter on ethanalyzer capture
  capture-ring-buffer     Capture ring buffer option
  decode-internal         Include internal system header decoding
  detail                  Display detailed protocol information
  display-filter          Display filter on frames captured
  limit-captured-frames   Maximum number of frames to be captured (default is
                          10)
  limit-frame-size        Capture only a subset of a frame
  raw                     Hex/Ascii dump the packet with possibly one line
                          summary
  write                   Filename to save capture to
  |                       Pipe command output to filter

DC# ethanalyzer local interface inband
Capturing on inband
2013-02-10 22:58:09.660171 00:23:33:74:47:05 -> 01:80:c2:00:00:00 STP Conf. Root = 32768/1/00:23:33:74:47:00  Cost = 0
   Port = 0x8006
2013-02-10 22:58:09.696505   10.10.10.2 -> 10.10.10.1   UDP Source port: 3200  Destination port: 3200
2013-02-10 22:58:09.697311   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200
2013-02-10 22:58:10.018963   10.10.10.2 -> 10.10.10.1   UDP Source port: 3200  Destination port: 3200
2013-02-10 22:58:10.086445 00:26:99:c7:f0:c3 -> 01:00:0c:cc:cc:cd STP RST. Root = 32768/96/00:23:04:ee:be:01  Cost = 0
   Port = 0x905e
2013-02-10 22:58:10.086608 00:26:99:c7:f0:c3 -> 01:00:0c:cc:cc:cd STP RST. Root = 32768/96/00:23:04:ee:be:01  Cost = 0
   Port = 0x905e
2013-02-10 22:58:10.086667 88:43:e1:c7:4d:b8 -> 01:80:c2:00:00:00 STP RST. Root = 32768/0/00:0d:ec:a3:96:3c  Cost = 3
   Port = 0x9000
```

Use the detail option for detailed protocol information. ^C can be used to abort and get the switch prompt back in the middle of a capture if required.



```
DC# ethanalyzer local interface inband detail
Capturing on inband
Frame 1 (106 bytes on wire, 74 bytes captured)
    Arrival Time: Feb 10, 2013 23:00:24.253088000
    [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: 106 bytes
    Capture Length: 74 bytes
    [Frame is marked: False]
    [Protocols in frame: eth:ip:eigrp]
Ethernet II, Src: 00:26:51:ce:0f:44 (00:26:51:ce:0f:44), Dst: 01:00:5e:00:00:0a
(01:00:5e:00:00:0a)
    Destination: 01:00:5e:00:00:0a (01:00:5e:00:00:0a)
        Address: 01:00:5e:00:00:0a (01:00:5e:00:00:0a)
        .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadca
st)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory
 default)
    Source: 00:26:51:ce:0f:44 (00:26:51:ce:0f:44)
        Address: 00:26:51:ce:0f:44 (00:26:51:ce:0f:44)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory
 default)
    Type: IP (0x0800)
Internet Protocol, Src: 10.10.18.6 (10.10.18.6), Dst: 224.0.0.10 (224.0.0.10)
    Version: 4
    Header length: 20 bytes
    Differentiated Services Field: 0xc0 (DSCP 0x30: Class Selector 6; ECN: 0x00)
        1100 00.. = Differentiated Services Codepoint: Class Selector 6 (0x30)
        .... ..0. = ECN-Capable Transport (ECT): 0
        .... ...0 = ECN-CE: 0
    ---------------------------------------SNIP---------------------------------------
```

Filter Options

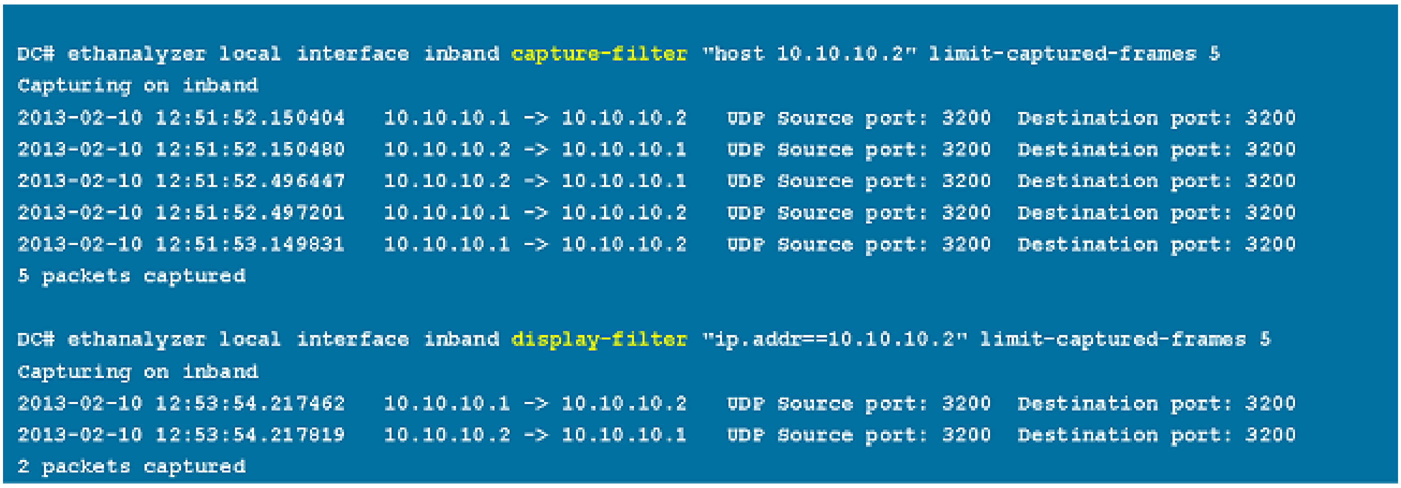
Capture-Filter

Use the capture-filter option in order to select which packets to display or save to disk during capture. A capture filter maintains a high rate of capture while it filters. Because full dissection has not been done on the packets, the filter fields are predefined and limited.

Display-Filter

Use the display-filter option in order to change the view of a capture file (tmp file). A display filter uses fully dissected packets, so you can do very complex and advanced filtering when you analyze a network tracefile. However, the tmp file can fill quickly since it first captures all packets and then displays only the desired packets.

In this example, limit-captured-frames is set to 5. With the capture-filter option, Ethanalyzer shows you five packets which match the filter host 10.10.10.2. With the display-filter option, Ethanalyzer first captures five packets then displays only the packets that match the filter ip.addr==10.10.10.2.



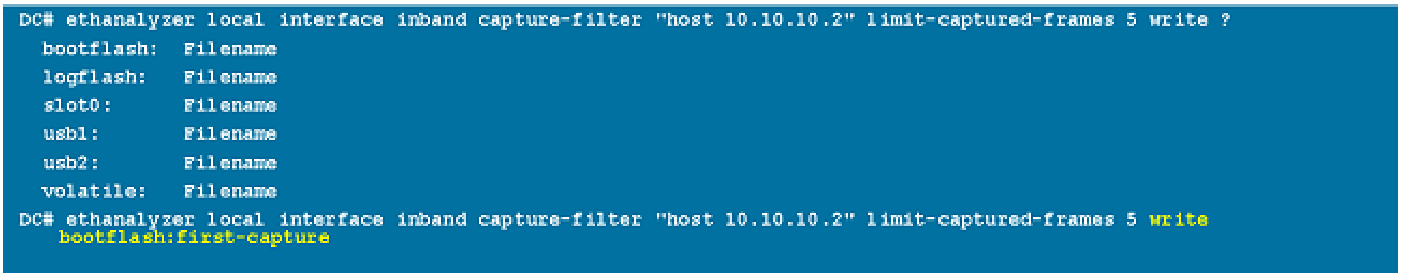
```
DC# ethanalyzer local interface inband capture-filter "host 10.10.10.2" limit-captured-frames 5
Capturing on inband
2013-02-10 12:51:52.150404   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200
2013-02-10 12:51:52.150480   10.10.10.2 -> 10.10.10.1   UDP Source port: 3200  Destination port: 3200
2013-02-10 12:51:52.496447   10.10.10.2 -> 10.10.10.1   UDP Source port: 3200  Destination port: 3200
2013-02-10 12:51:52.497201   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200
2013-02-10 12:51:53.149831   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200
5 packets captured

DC# ethanalyzer local interface inband display-filter "ip.addr==10.10.10.2" limit-captured-frames 5
Capturing on inband
2013-02-10 12:53:54.217462   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200
2013-02-10 12:53:54.217819   10.10.10.2 -> 10.10.10.1   UDP Source port: 3200  Destination port: 3200
2 packets captured
```

Write Options

Write

The write option lets you write the capture data to a file in one of the storage devices (such as bootflash or logflash) on the Cisco Nexus 7000 Series Switch for later analysis. The capture file size is limited to 10 MB.

An example Ethanalyzer command with a write option is **ethanalyzer local interface inband write bootflash:** capture_file_name. An example of a write option with capture-filter and an output file name of first-capture is:

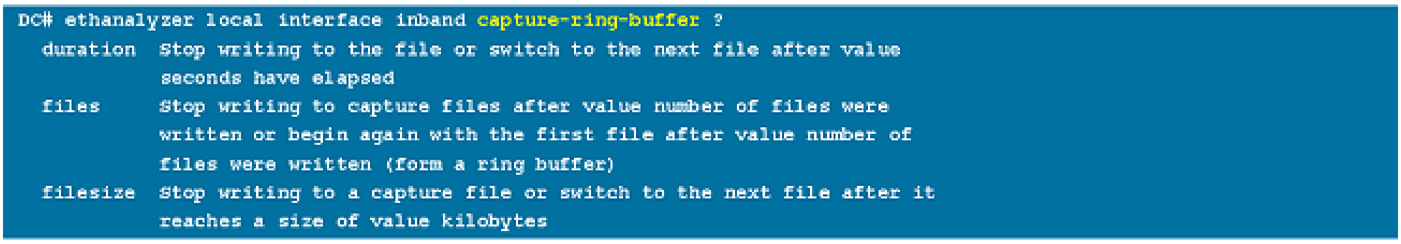


```
DC# ethanalyzer local interface inband capture-filter "host 10.10.10.2" limit-captured-frames 5 write ?
  bootflash:   Filename
  logflash:    Filename
  slot0:       Filename
  usb1:        Filename
  usb2:        Filename
  volatile:    Filename
DC# ethanalyzer local interface inband capture-filter "host 10.10.10.2" limit-captured-frames 5 write
   bootflash:first-capture
```

When the capture data is saved to a file, the captured packets are, by default, not displayed in the terminal window. The display option forces Cisco NX-OS to display the packets while it saves the capture data to a file.

Capture-Ring-Buffer

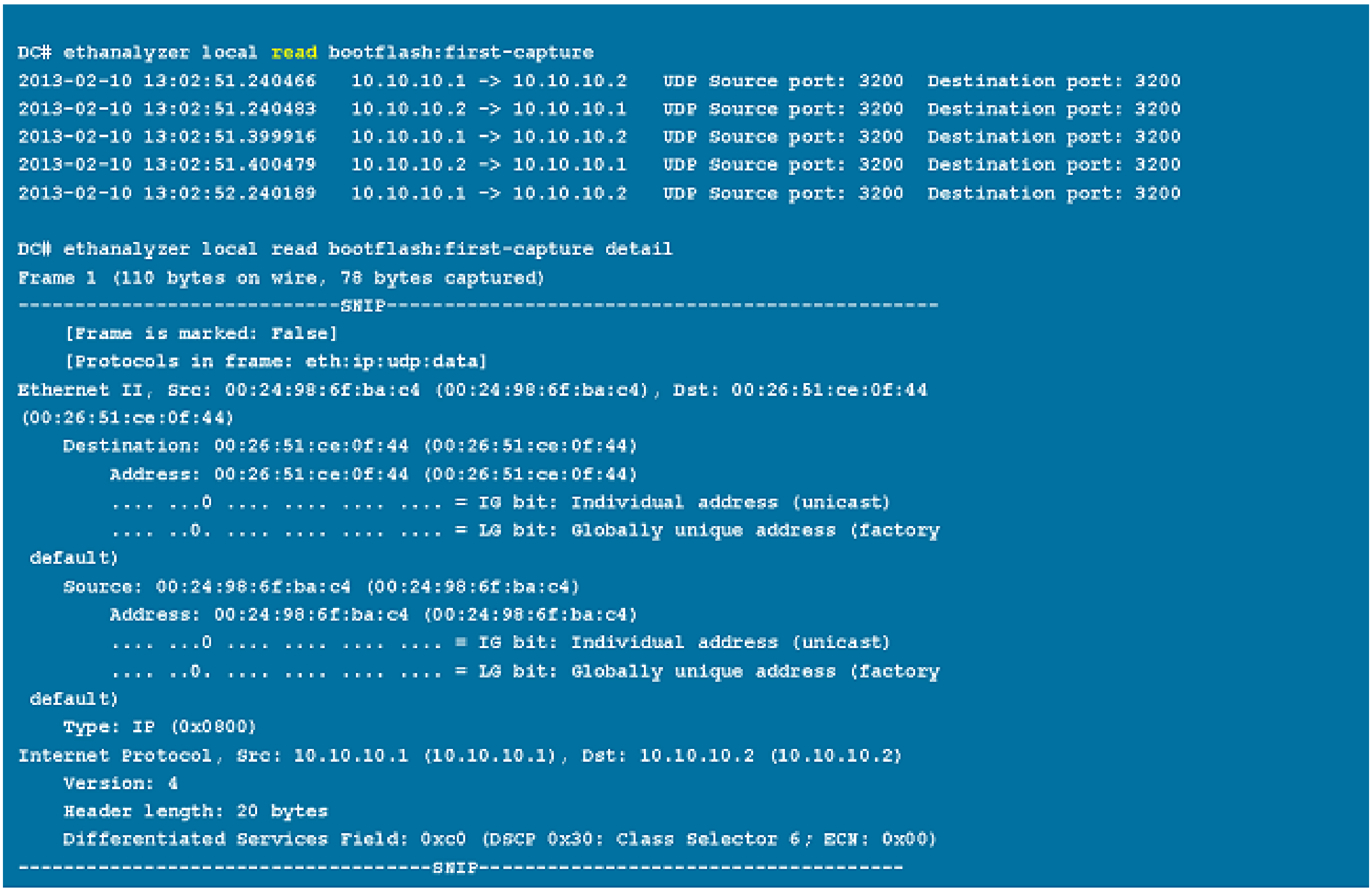
The capture-ring-buffer option creates multiple files after a specified number of seconds, a specified number of files, or a specified file size. Definitions of those options are in this screen shot:



```
DC# ethanalyzer local interface inband capture-ring-buffer ?
  duration  Stop writing to the file or switch to the next file after value
            seconds have elapsed
  files     Stop writing to capture files after value number of files were
            written or begin again with the first file after value number of
            files were written (form a ring buffer)
  filesize  Stop writing to a capture file or switch to the next file after it
            reaches a size of value kilobytes
```

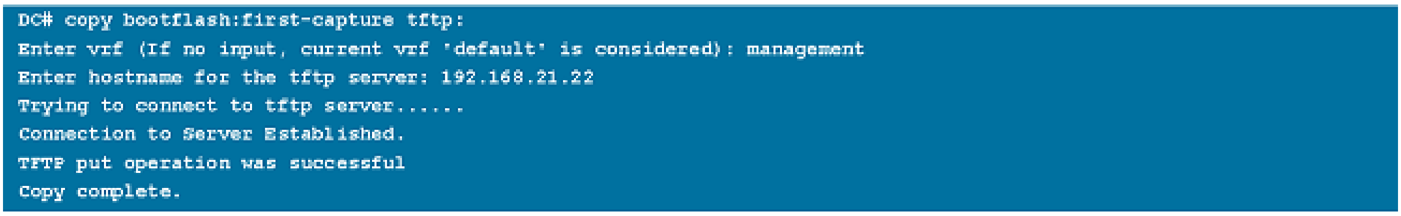
Read Options

The read option lets you read the saved file on the device itself.

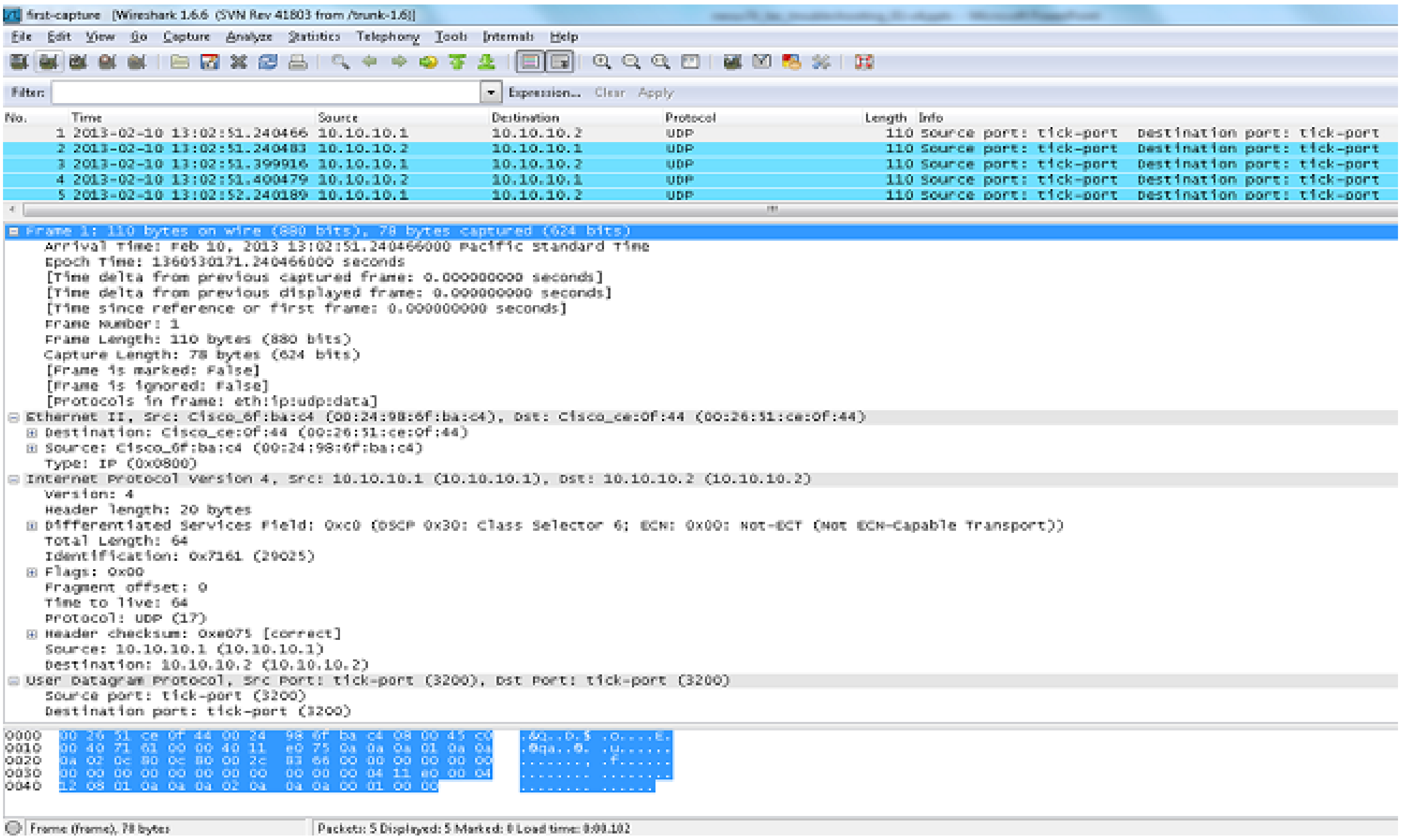


```
DC# ethanalyzer local read bootflash:first-capture
2013-02-10 13:02:51.240466   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200
2013-02-10 13:02:51.240483   10.10.10.2 -> 10.10.10.1   UDP Source port: 3200  Destination port: 3200
2013-02-10 13:02:51.399916   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200
2013-02-10 13:02:51.400479   10.10.10.2 -> 10.10.10.1   UDP Source port: 3200  Destination port: 3200
2013-02-10 13:02:52.240189   10.10.10.1 -> 10.10.10.2   UDP Source port: 3200  Destination port: 3200

DC# ethanalyzer local read bootflash:first-capture detail
Frame 1 (110 bytes on wire, 78 bytes captured)
----------------------------SNIP------------------------------------------------
    [Frame is marked: False]
    [Protocols in frame: eth:ip:udp:data]
Ethernet II, Src: 00:24:98:6f:ba:c4 (00:24:98:6f:ba:c4), Dst: 00:26:51:ce:0f:44
(00:26:51:ce:0f:44)
    Destination: 00:26:51:ce:0f:44 (00:26:51:ce:0f:44)
        Address: 00:26:51:ce:0f:44 (00:26:51:ce:0f:44)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory
 default)
    Source: 00:24:98:6f:ba:c4 (00:24:98:6f:ba:c4)
        Address: 00:24:98:6f:ba:c4 (00:24:98:6f:ba:c4)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory
 default)
    Type: IP (0x0800)
Internet Protocol, Src: 10.10.10.1 (10.10.10.1), Dst: 10.10.10.2 (10.10.10.2)
    Version: 4
    Header length: 20 bytes
    Differentiated Services Field: 0xc0 (DSCP 0x30: Class Selector 6; ECN: 0x00)
------------------------------------SNIP--------------------------------------
```

You can also transfer the file to a server or a PC and read it with Wireshark or any other application that can read cap or pcap files.



```
DC# copy bootflash:first-capture tftp:
Enter vrf (If no input, current vrf 'default' is considered): management
Enter hostname for the tftp server: 192.168.21.22
Trying to connect to tftp server......
Connection to Server Established.
TFTP put operation was successful
Copy complete.
```


first-capture [Wireshark 1.6.6 (SVN Rev 41803 from /trunk-1.6)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression... Clear Apply

```
No.  Time                         Source       Destination  Protocol  Length Info
   1 2013-02-10 13:02:51.240466 10.10.10.1   10.10.10.2   UDP          110 Source port: tick-port  Destination port: tick-port
   2 2013-02-10 13:02:51.240483 10.10.10.2   10.10.10.1   UDP          110 Source port: tick-port  Destination port: tick-port
   3 2013-02-10 13:02:51.399916 10.10.10.1   10.10.10.2   UDP          110 Source port: tick-port  Destination port: tick-port
   4 2013-02-10 13:02:51.400479 10.10.10.2   10.10.10.1   UDP          110 Source port: tick-port  Destination port: tick-port
   5 2013-02-10 13:02:52.240189 10.10.10.1   10.10.10.2   UDP          110 Source port: tick-port  Destination port: tick-port
```

```
⊟ Frame 1: 110 bytes on wire (880 bits), 78 bytes captured (624 bits)
    Arrival Time: Feb 10, 2013 13:02:51.240466000 Pacific Standard Time
    Epoch Time: 1360530171.240466000 seconds
    [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: 110 bytes (880 bits)
    Capture Length: 78 bytes (624 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ip:udp:data]
⊟ Ethernet II, Src: Cisco_6f:ba:c4 (00:24:98:6f:ba:c4), Dst: Cisco_ce:0f:44 (00:26:51:ce:0f:44)
  ⊞ Destination: Cisco_ce:0f:44 (00:26:51:ce:0f:44)
  ⊞ Source: Cisco_6f:ba:c4 (00:24:98:6f:ba:c4)
    Type: IP (0x0800)
⊟ Internet Protocol Version 4, Src: 10.10.10.1 (10.10.10.1), Dst: 10.10.10.2 (10.10.10.2)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0xc0 (DSCP 0x30: Class Selector 6; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 64
    Identification: 0x7161 (29025)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 64
    Protocol: UDP (17)
  ⊞ Header checksum: 0xe075 [correct]
    Source: 10.10.10.1 (10.10.10.1)
    Destination: 10.10.10.2 (10.10.10.2)
⊟ User Datagram Protocol, Src Port: tick-port (3200), Dst Port: tick-port (3200)
    Source port: tick-port (3200)
    Destination port: tick-port (3200)
```

```
0000   00 26 51 ce 0f 44 00 24  98 6f ba c4 08 00 45 c0   .&Q..D.$ .o....E.
0010   00 40 71 61 00 00 40 11  e0 75 0a 0a 0a 01 0a 0a   .@qa..@. .u......
0020   0a 02 0c 80 0c 80 00 2c  83 66 00 00 00 00 00 00   .......,  .f......
0030   00 00 00 00 00 00 00 00  00 00 00 04 11 e0 00 04   ........ ........
0040   12 08 01 0a 0a 0a 02 0a  0a 0a 00 01 00 00         ........ ......
```

Frame (frame), 78 bytes | Packets: 5 Displayed: 5 Marked: 0 Load time: 0:00.102

Decode-Internal with Detail Option

The decode-internal option reports internal information on how the Nexus 7000 forwards the packet. This information helps you understand and troubleshoot the flow of packets through the CPU.

```
DC# ethanalyzer local interface inband decode-internal capture-filter "host 10.10.10.2" limit-captured-frames 5
detail
Capturing on inband
NXOS Protocol
    NXOS VLAN: 0====================➔VLAN in decimal=0=L3 interface
    NXOS SOURCE INDEX: 1024 ========================➔PIXM LTL source index in decimal=400=SUP inband
    NXOS DEST INDEX: 2569========================➔PIXM LTL destination index in decimal=0xa09=e1/25
Frame 1 (78 bytes on wire, 78 bytes captured)
    Arrival Time: Feb 10, 2013 22:40:02.216492000
    [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: 78 bytes
    Capture Length: 78 bytes
    [Frame is marked: False]
    [Protocols in frame: eth:ip:udp:data]
Ethernet II, Src: 00:26:51:ce:0f:43 (00:26:51:ce:0f:43), Dst: 00:24:98:6f:ba:c3
(00:24:98:6f:ba:c3)
    Destination: 00:24:98:6f:ba:c3 (00:24:98:6f:ba:c3)
        Address: 00:24:98:6f:ba:c3 (00:24:98:6f:ba:c3)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory
 default)
    Source: 00:26:51:ce:0f:43 (00:26:51:ce:0f:43)
-----------------------------------------------SNIP----------------------------------------
```

Convert the NX-OS index to hexadecimal, then use the **show system internal pixm info ltl x** command in order to map the local target logic (LTL) index to a physical or logical interface.

Examples of Capture-Filter Values

Capture Traffic to or from an IP Host

```
host 10.1.1.1
```

Capture Traffic to or from a Range of IP Addresses

```
net 172.16.7.0/24
net 172.16.7.0 mask 255.255.255.0
```

Capture Traffic from a Range of IP Addresses

```
src net 172.16.7.0/24
src net 172.16.7.0 mask 255.255.255.0
```

Capture Traffic to a Range of IP Addresses

```
dst net 172.16.7.0/24
dst net 172.16.7.0 mask 255.255.255.0
```

Capture Traffic Only on a Certain Protocol - Capture Only DNS Traffic

DNS is the Domain Name System Protocol.

```
port 53
```

Capture Traffic Only on a Certain Protocol - Capture Only DHCP Traffic

DHCP is the Dynamic Host Configuration Protocol.

```
port 67 or port 68
```

Capture Traffic Not on a Certain Protocol - Exclude HTTP or SMTP Traffic

SMTP is the Simple Mail Transfer Protocol.

```
host 172.16.7.3 and not port 80 and not port 25
```

Capture Traffic Not on a Certain Protocol - Exclude ARP and DNS Traffic

ARP is the Address Resolution Protocol.

```
port not 53 and not arp
```

Capture Only IP Traffic - Exclude Lower Layer Protocols like ARP and STP

STP is the Spanning Tree Protocol.

```
ip
```

Capture Only Unicast Traffic - Exclude Broadcast and Multicast Announcements

```
not broadcast and not multicast
```

Capture Traffic Within a Range of Layer 4 Ports

```
tcp portrange 1501-1549
```

Capture Traffic Based on Ethernet Type - Capture EAPOL Traffic

EAPOL is the Extensible Authentication Protocol over LAN.

```
ether proto 0x888e
```

IPv6 Capture Workaround

```
ether proto 0x86dd
```

Capture Traffic Based on IP Protocol Type

```
ip proto 89
```

Reject Ethernet Frames Based on MAC Address - Exclude Traffic That Belongs to the LLDP Multicast Group

LLDP is the Link Layer Discovery Protocol.

```
not ether dst 01:80:c2:00:00:0e
```

Capture UDLD, VTP, or CDP Traffic

UDLD is Unidirectional Link Detection, VTP is the VLAN Trunking Protocol, and CDP is the Cisco Discovery Protocol.

```
ether host 01:00:0c:cc:cc:cc
```

Capture Traffic to or from a MAC Address

```
ether host 00:01:02:03:04:05
```

Note:
and = &&
or = ||
not = !
MAC address format : xx:xx:xx:xx:xx:xx

Common Control Plane Protocols

- UDLD: Destination Media Access Controller (DMAC) = 01-00-0C-CC-CC-CC and EthType = 0x0111
- LACP: DMAC = 01:80:C2:00:00:02 and EthType = 0x8809. LACP stands for Link Aggregation Control Protocol.
- STP: DMAC = 01:80:C2:00:00:00 and EthType = 0x4242 - or - DMAC = 01:00:0C:CC:CC:CD and

EthType = 0x010B
- CDP: DMAC = 01-00-0C-CC-CC-CC and EthType = 0x2000
- LLDP: DMAC = 01:80:C2:00:00:0E or 01:80:C2:00:00:03 or 01:80:C2:00:00:00 and EthType = 0x88CC
- DOT1X: DMAC = 01:80:C2:00:00:03 and EthType = 0x888E. DOT1X stands for IEEE 802.1x.
- IPv6: EthType = 0x86DD
- List of UDP and TCP port numbers

Known Issues

Cisco bug ID CSCue48854: Ethanalyzer capture-filter does not capture traffic from CPU on SUP2.

Cisco bug ID CSCtx79409: Cannot use capture filter with decode-internal.

Cisco bug ID CSCvi02546: SUP3 generated packet can have FCS, this is expected behaviour.

Related Information

- Wireshark: CaptureFilters
- Wireshark: DisplayFilters
- Technical Support & Documentation - Cisco Systems