MC833A - Programação de Redes de Computadores

Professor Nelson Fonseca

http://www.lrc.ic.unicamp.br/mc833/

Agenda

- Ferramentas de Redes
- Wireshark (sniffers)
- TCPDump (sniffers)

- Exercício 1 – Ferramentas e Sniffers

Ferramentas Redes

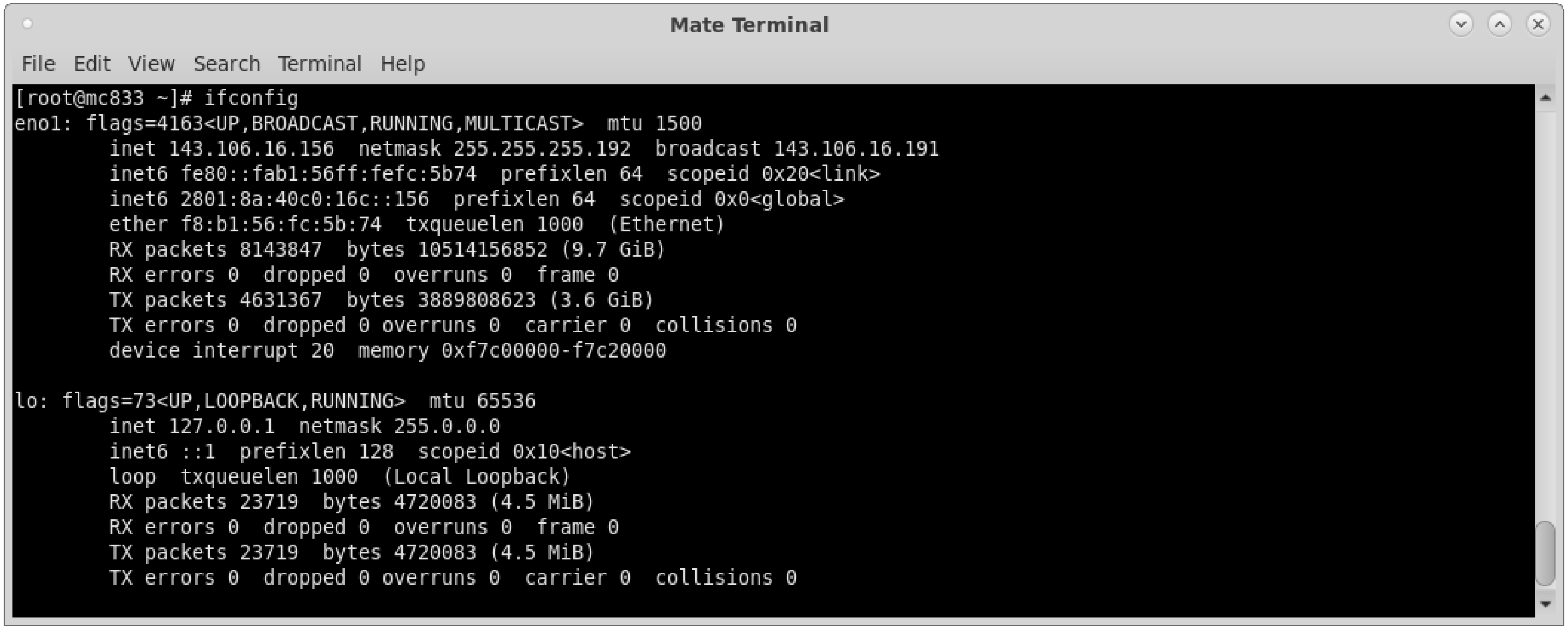
- Configuração e informação de interfaces.
- Resolução de nomes para endereço IP (DNS)
- Realizando uma conexão
- Resolução de endereço IP para MAC (ARP)
- Performance e Estatistica

ifconfig

Função:
Configurar e obter informações sobre as interfaces de redes (link e camada de rede)

- Alteração do endereço IP, mascara de rede, MTU, MAC address, etc...
- Informações sobre as estatisticas da interface: envio e recebimento de bytes/pacotes, erros, colisões, etc...

Exemplo da saida do Ifconfig



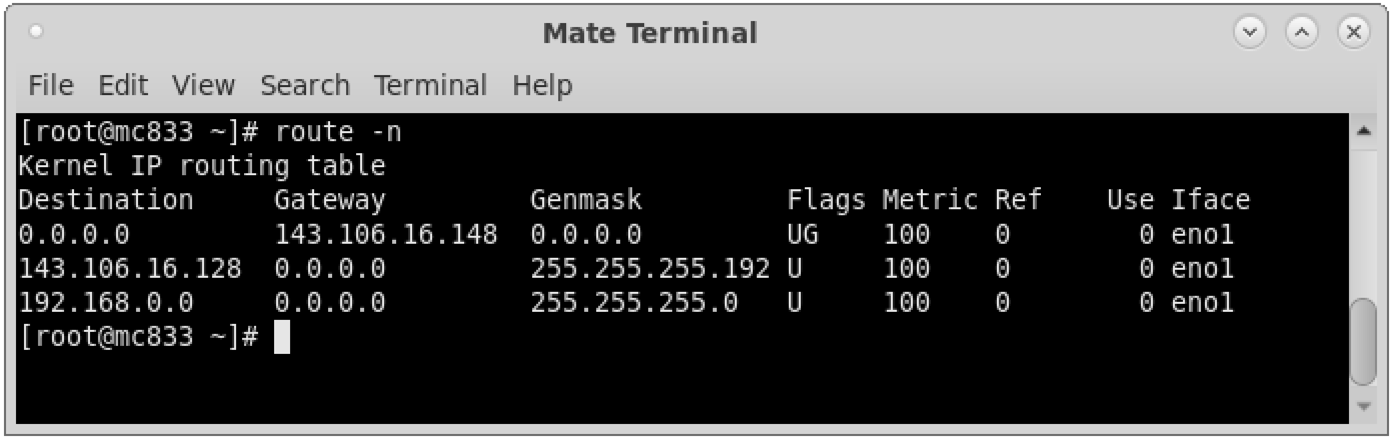
```
[root@mc833 ~]# ifconfig
eno1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 143.106.16.156  netmask 255.255.255.192  broadcast 143.106.16.191
        inet6 fe80::fab1:56ff:fefc:5b74  prefixlen 64  scopeid 0x20<link>
        inet6 2801:8a:40c0:16c::156  prefixlen 64  scopeid 0x0<global>
        ether f8:b1:56:fc:5b:74  txqueuelen 1000  (Ethernet)
        RX packets 8143847  bytes 10514156852 (9.7 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4631367  bytes 3889808623 (3.6 GiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 20  memory 0xf7c00000-f7c20000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 23719  bytes 4720083 (4.5 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 23719  bytes 4720083 (4.5 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

route

Função:
Configurar e obter informações sobre as tabelas de roteamento IP

- Alteração da tabela de roteamento (IP address e interfaces)
- Geralmente utilizado para alteração do default gateway

Exemplo da saida do route



```
[root@mc833 ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         143.106.16.148  0.0.0.0         UG    100    0        0 eno1
143.106.16.128  0.0.0.0         255.255.255.192 U     100    0        0 eno1
192.168.0.0     0.0.0.0         255.255.255.0   U     100    0        0 eno1
[root@mc833 ~]#
```

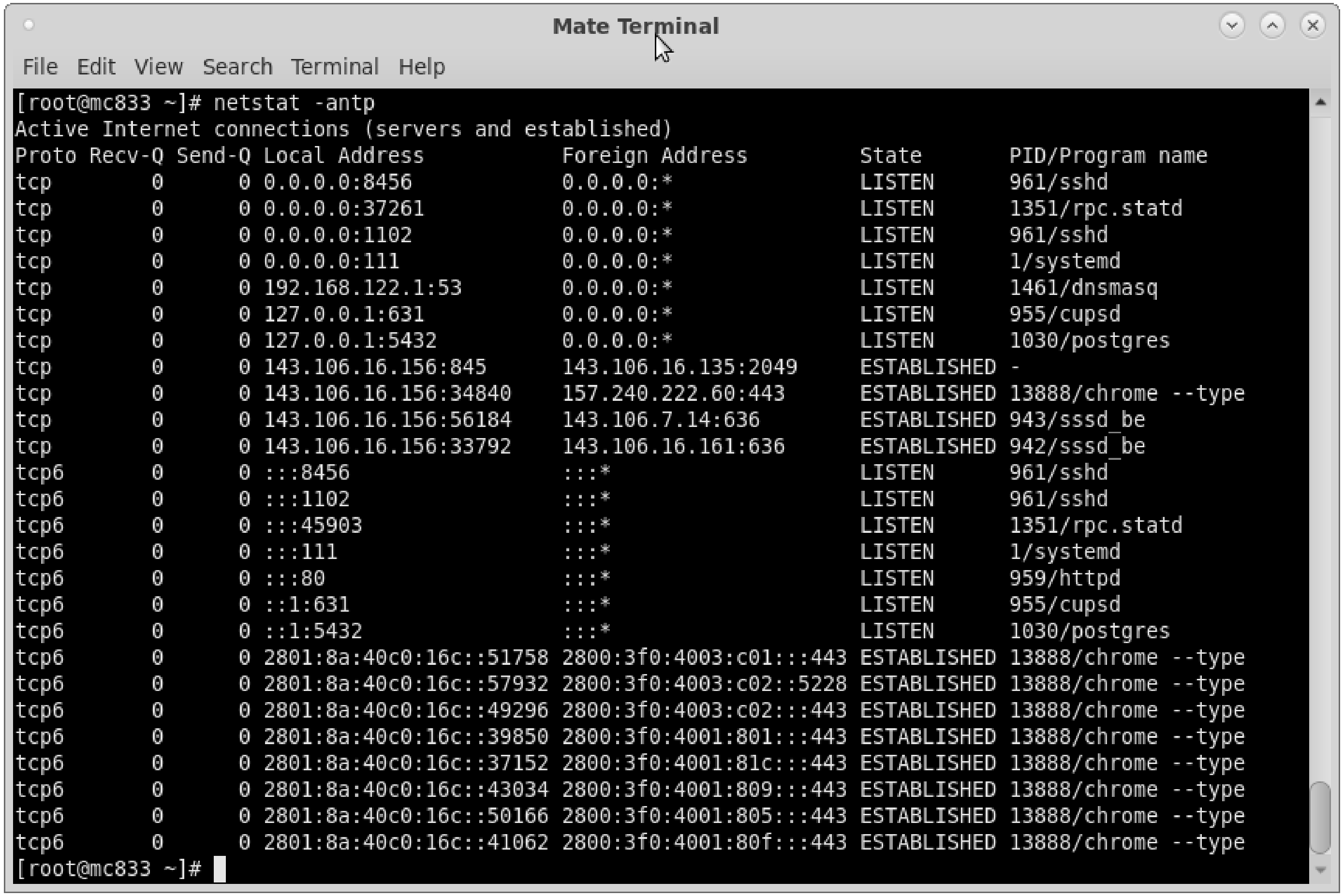
netstat

Função:
Obter informações sobre as conexões TCP e UDP, tabela de roteamento e estatisticas dos protocolos de rede.

- Geralmente utilizado para mostrar informações sobre as conexões ativas da maquina.
- Protocolo de transporte, IP de origem:Porta e IP de destino:Porta para cada conexão ativa.

```
$ netstat -antp ; netstat -anup
```

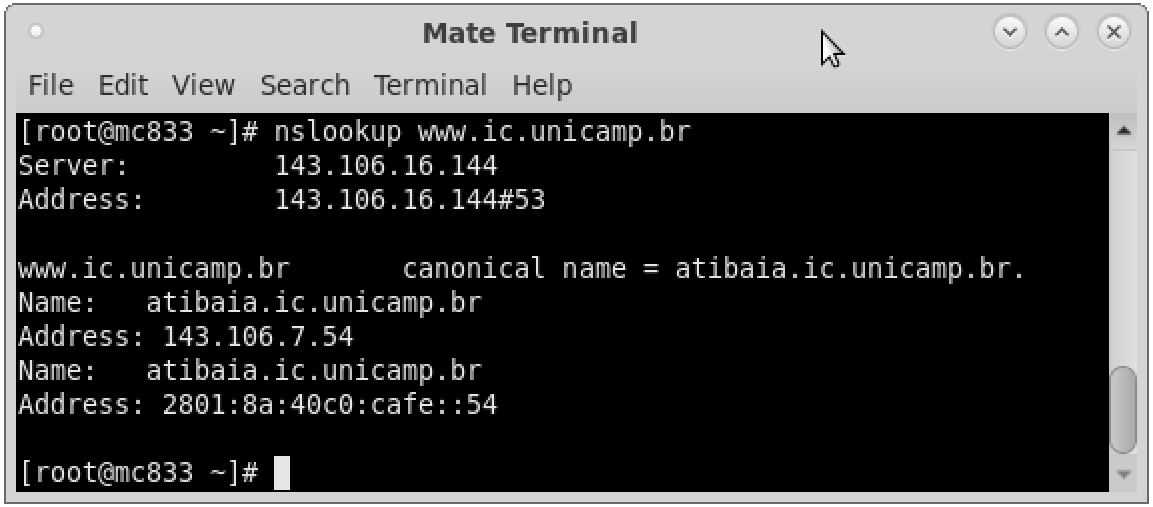
netstat



```
[root@mc833 ~]# netstat -antp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:8456            0.0.0.0:*               LISTEN      961/sshd
tcp        0      0 0.0.0.0:37261           0.0.0.0:*               LISTEN      1351/rpc.statd
tcp        0      0 0.0.0.0:1102            0.0.0.0:*               LISTEN      961/sshd
tcp        0      0 0.0.0.0:111             0.0.0.0:*               LISTEN      1/systemd
tcp        0      0 192.168.122.1:53        0.0.0.0:*               LISTEN      1461/dnsmasq
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      955/cupsd
tcp        0      0 127.0.0.1:5432          0.0.0.0:*               LISTEN      1030/postgres
tcp        0      0 143.106.16.156:845      143.106.16.135:2049     ESTABLISHED -
tcp        0      0 143.106.16.156:34840    157.240.222.60:443      ESTABLISHED 13888/chrome --type
tcp        0      0 143.106.16.156:56184    143.106.7.14:636        ESTABLISHED 943/sssd_be
tcp        0      0 143.106.16.156:33792    143.106.16.161:636      ESTABLISHED 942/sssd_be
tcp6       0      0 :::8456                 :::*                    LISTEN      961/sshd
tcp6       0      0 :::1102                 :::*                    LISTEN      961/sshd
tcp6       0      0 :::45903                :::*                    LISTEN      1351/rpc.statd
tcp6       0      0 :::111                  :::*                    LISTEN      1/systemd
tcp6       0      0 :::80                   :::*                    LISTEN      959/httpd
tcp6       0      0 ::1:631                 :::*                    LISTEN      955/cupsd
tcp6       0      0 ::1:5432                :::*                    LISTEN      1030/postgres
tcp6       0      0 2801:8a:40c0:16c::51758 2800:3f0:4003:c01:::443 ESTABLISHED 13888/chrome --type
tcp6       0      0 2801:8a:40c0:16c::57932 2800:3f0:4003:c02::5228 ESTABLISHED 13888/chrome --type
tcp6       0      0 2801:8a:40c0:16c::49296 2800:3f0:4003:c02:::443 ESTABLISHED 13888/chrome --type
tcp6       0      0 2801:8a:40c0:16c::39850 2800:3f0:4001:801:::443 ESTABLISHED 13888/chrome --type
tcp6       0      0 2801:8a:40c0:16c::37152 2800:3f0:4001:81c:::443 ESTABLISHED 13888/chrome --type
tcp6       0      0 2801:8a:40c0:16c::43034 2800:3f0:4001:809:::443 ESTABLISHED 13888/chrome --type
tcp6       0      0 2801:8a:40c0:16c::50166 2800:3f0:4001:805:::443 ESTABLISHED 13888/chrome --type
tcp6       0      0 2801:8a:40c0:16c::41062 2800:3f0:4001:80f:::443 ESTABLISHED 13888/chrome --type
[root@mc833 ~]#
```

nslookup

Função:
Resolver o endereço IP de cada host via protocolo DNS. Também é possivel realizar a resolução inversa IP para host.



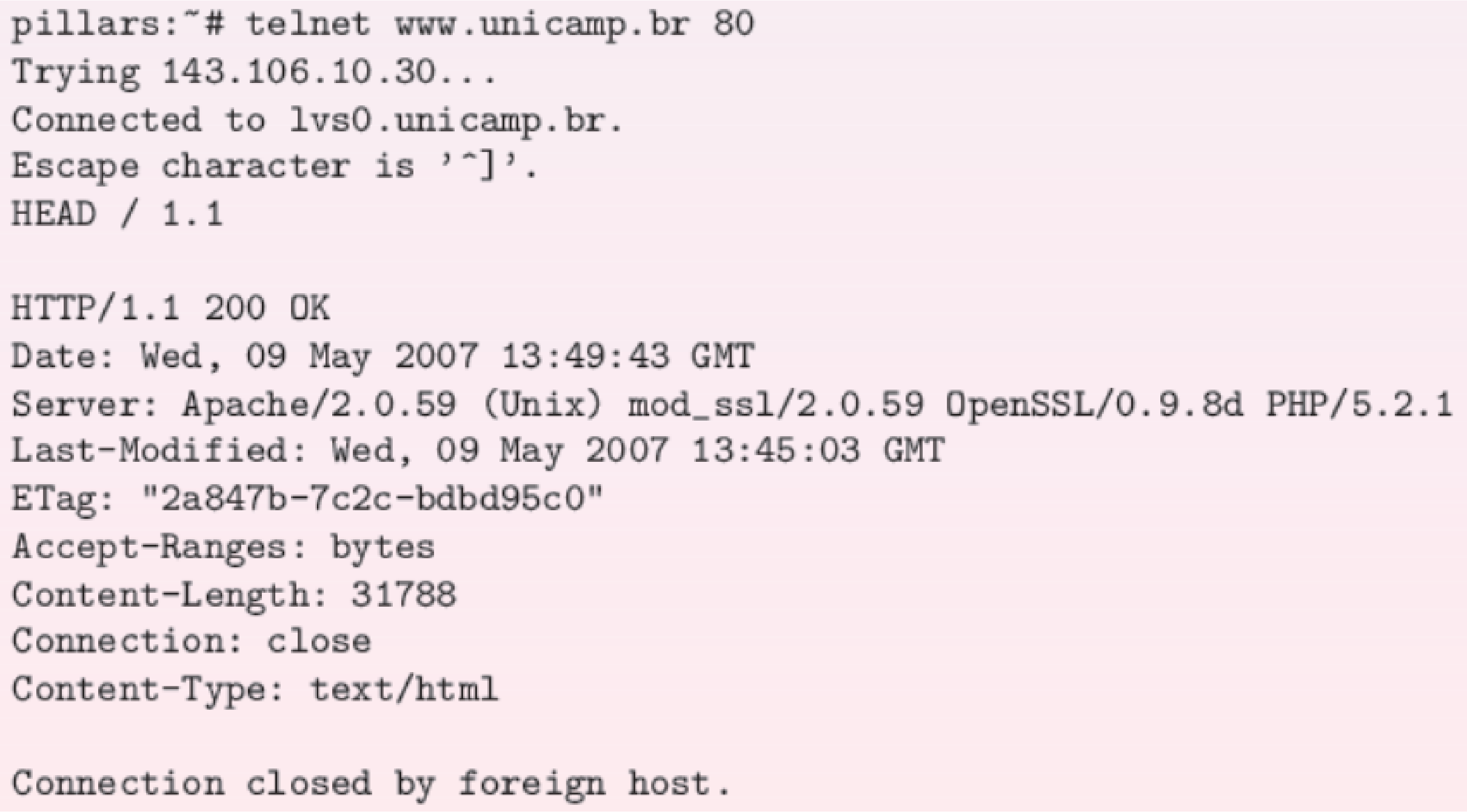
```
[root@mc833 ~]# nslookup www.ic.unicamp.br
Server:         143.106.16.144
Address:        143.106.16.144#53

www.ic.unicamp.br       canonical name = atibaia.ic.unicamp.br.
Name:   atibaia.ic.unicamp.br
Address: 143.106.7.54
Name:   atibaia.ic.unicamp.br
Address: 2801:8a:40c0:cafe::54

[root@mc833 ~]# 
```

telnet

Função:
Utilizado para realizar testes para descobrir bloqueios na rede e comunicação de portas.

- Em sua utilização é necessário conhecer alguns comandos especificos da camada de aplicação (comandos HTTP, comandos SMTP, IMAP, etc...)

Exemplo de uma conexão telnet

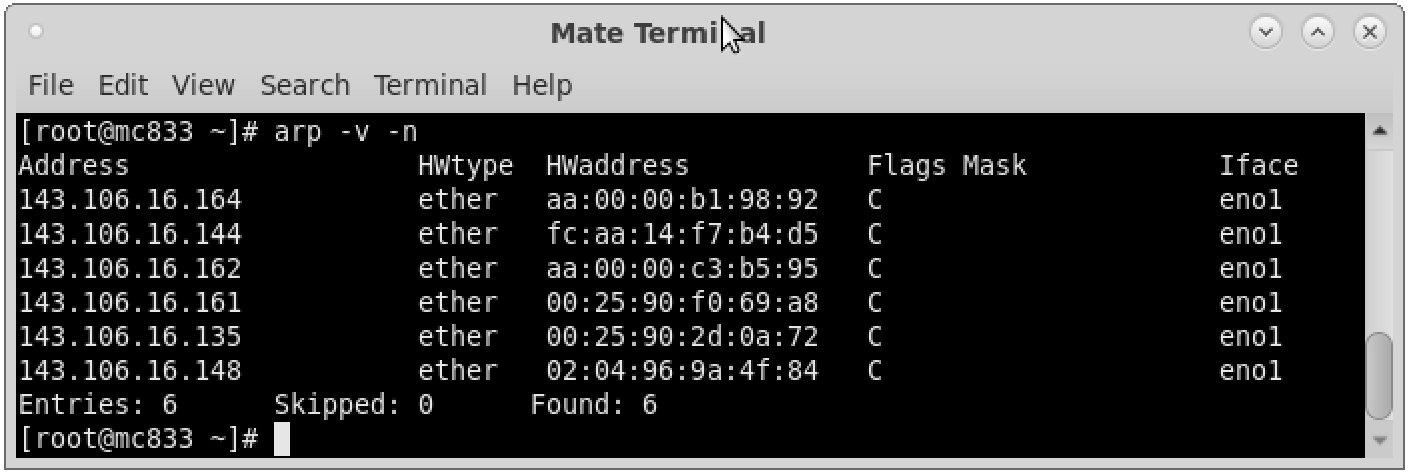


```
pillars:~# telnet www.unicamp.br 80
Trying 143.106.10.30...
Connected to lvs0.unicamp.br.
Escape character is ’^]’.
HEAD / 1.1

HTTP/1.1 200 OK
Date: Wed, 09 May 2007 13:49:43 GMT
Server: Apache/2.0.59 (Unix) mod_ssl/2.0.59 OpenSSL/0.9.8d PHP/5.2.1
Last-Modified: Wed, 09 May 2007 13:45:03 GMT
ETag: "2a847b-7c2c-bdbd95c0"
Accept-Ranges: bytes
Content-Length: 31788
Connection: close
Content-Type: text/html

Connection closed by foreign host.
```

arp

Função:
Usado para resolução de endereços IP para a camada de enlace.

- Geralmente utilizado pra mostrar a tabela ARP



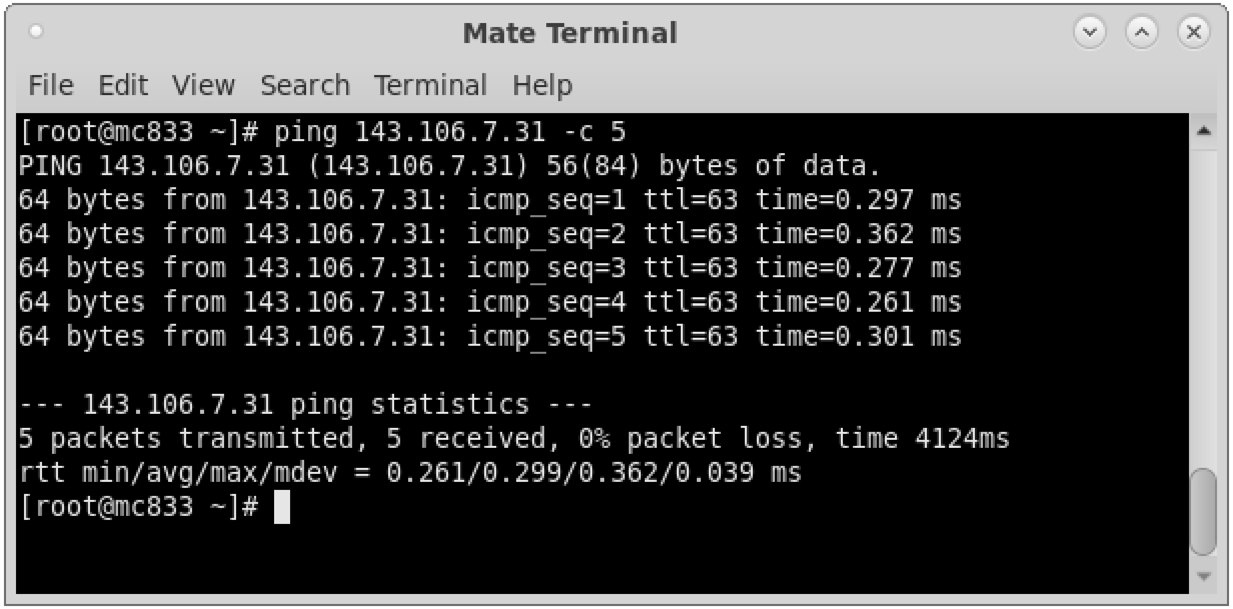
```
[root@mc833 ~]# arp -v -n
Address                  HWtype  HWaddress           Flags Mask            Iface
143.106.16.164           ether   aa:00:00:b1:98:92   C                     eno1
143.106.16.144           ether   fc:aa:14:f7:b4:d5   C                     eno1
143.106.16.162           ether   aa:00:00:c3:b5:95   C                     eno1
143.106.16.161           ether   00:25:90:f0:69:a8   C                     eno1
143.106.16.135           ether   00:25:90:2d:0a:72   C                     eno1
143.106.16.148           ether   02:04:96:9a:4f:84   C                     eno1
Entries: 6      Skipped: 0      Found: 6
[root@mc833 ~]# 
```

ping

Função:
Utilizado para testar conectividade entre dois hosts e medir o RTT de envio dos pacotes.

- Bastante utilizado para testar se um host esta “vivo” na rede.
- Envia datagramas ICMP com mensagens de ECHO_REQUEST e recebe como resposta um datagrama ICMP ECHO_REPLY
- O valor do RTT é utilizado para checar a integridade da rota.

Exemplo de um ping



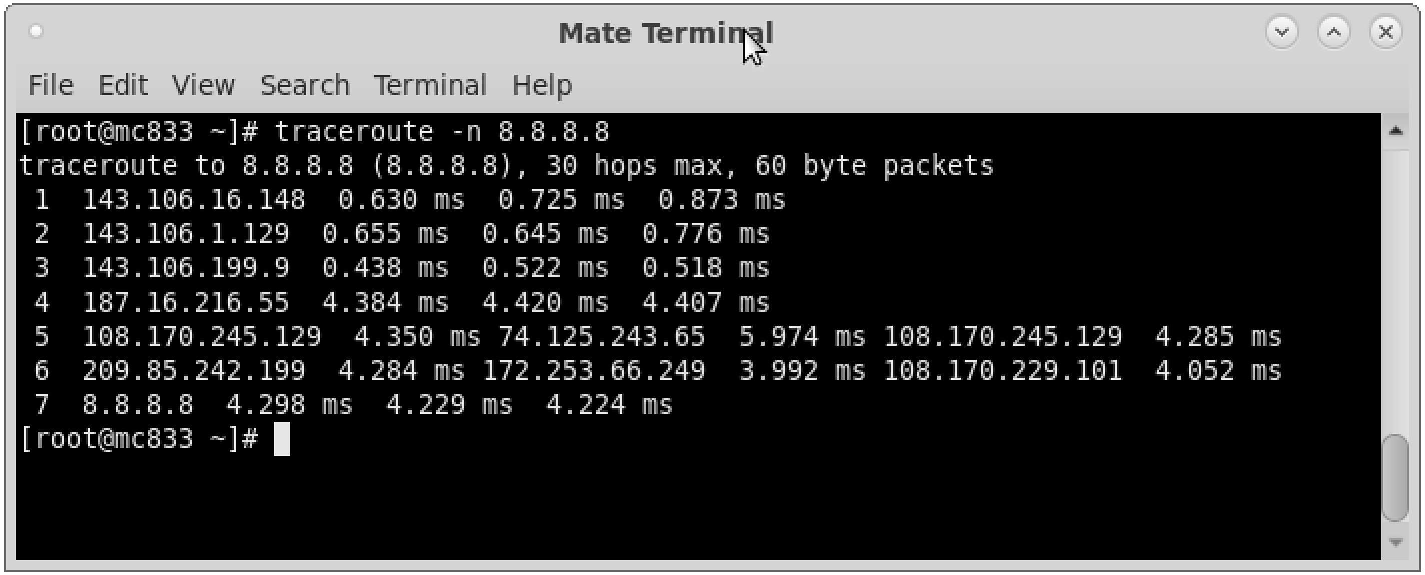
```
[root@mc833 ~]# ping 143.106.7.31 -c 5
PING 143.106.7.31 (143.106.7.31) 56(84) bytes of data.
64 bytes from 143.106.7.31: icmp_seq=1 ttl=63 time=0.297 ms
64 bytes from 143.106.7.31: icmp_seq=2 ttl=63 time=0.362 ms
64 bytes from 143.106.7.31: icmp_seq=3 ttl=63 time=0.277 ms
64 bytes from 143.106.7.31: icmp_seq=4 ttl=63 time=0.261 ms
64 bytes from 143.106.7.31: icmp_seq=5 ttl=63 time=0.301 ms

--- 143.106.7.31 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4124ms
rtt min/avg/max/mdev = 0.261/0.299/0.362/0.039 ms
[root@mc833 ~]# 
```

traceroute

Função:
Envia pacotes para um determinado host afim de medir o RTT de cada hop passado pelo roteador.

- Envia 3 pacotes UDP com TTL igual a 1. Apos isso envia novamente 3 pacotes UDP com TTL igual a 2, e assim continua até alcançar o host de destino.
- Para evitar que pacotes fiquem infinitamente na rede, cada roteador decrementa o TTL a cada pacote que passar por ele. Quando o TTL chega a 0, o roteador destroi o pacote e envia um ICMP para o host de origem informando que o pacote excedeu o limite (TIME_EXCEEDED)

Exemplo do traceroute



```
[root@mc833 ~]# traceroute -n 8.8.8.8
traceroute to 8.8.8.8 (8.8.8.8), 30 hops max, 60 byte packets
 1  143.106.16.148  0.630 ms  0.725 ms  0.873 ms
 2  143.106.1.129  0.655 ms  0.645 ms  0.776 ms
 3  143.106.199.9  0.438 ms  0.522 ms  0.518 ms
 4  187.16.216.55  4.384 ms  4.420 ms  4.407 ms
 5  108.170.245.129  4.350 ms 74.125.243.65  5.974 ms 108.170.245.129  4.285 ms
 6  209.85.242.199  4.284 ms 172.253.66.249  3.992 ms 108.170.229.101  4.052 ms
 7  8.8.8.8  4.298 ms  4.229 ms  4.224 ms
[root@mc833 ~]#
```

iPerf

iPerf é uma ferramenta para medições de largura de banda em redes IP.

Suporta o ajuste de vários parâmetros relacionados a temporização, buffers e protocolos (TCP, UDP, SCTP com IPv4 e IPv6).

```
server$ iperf -s

client$ iperf -c 143.106.16.156
```

Para cada teste, ele informa a largura de banda, a perda e outros parâmetros

netcat

O netcat é um utilitário de rede de computadores para ler e gravar em conexões de rede usando TCP ou UDP. O comando é projetado para ser um back-end confiável que pode ser usado diretamente por outros programas e scripts.

```
server$ nc -l 9000

client$ echo teste | nc 127.0.0.1 9000
```

Wireshark

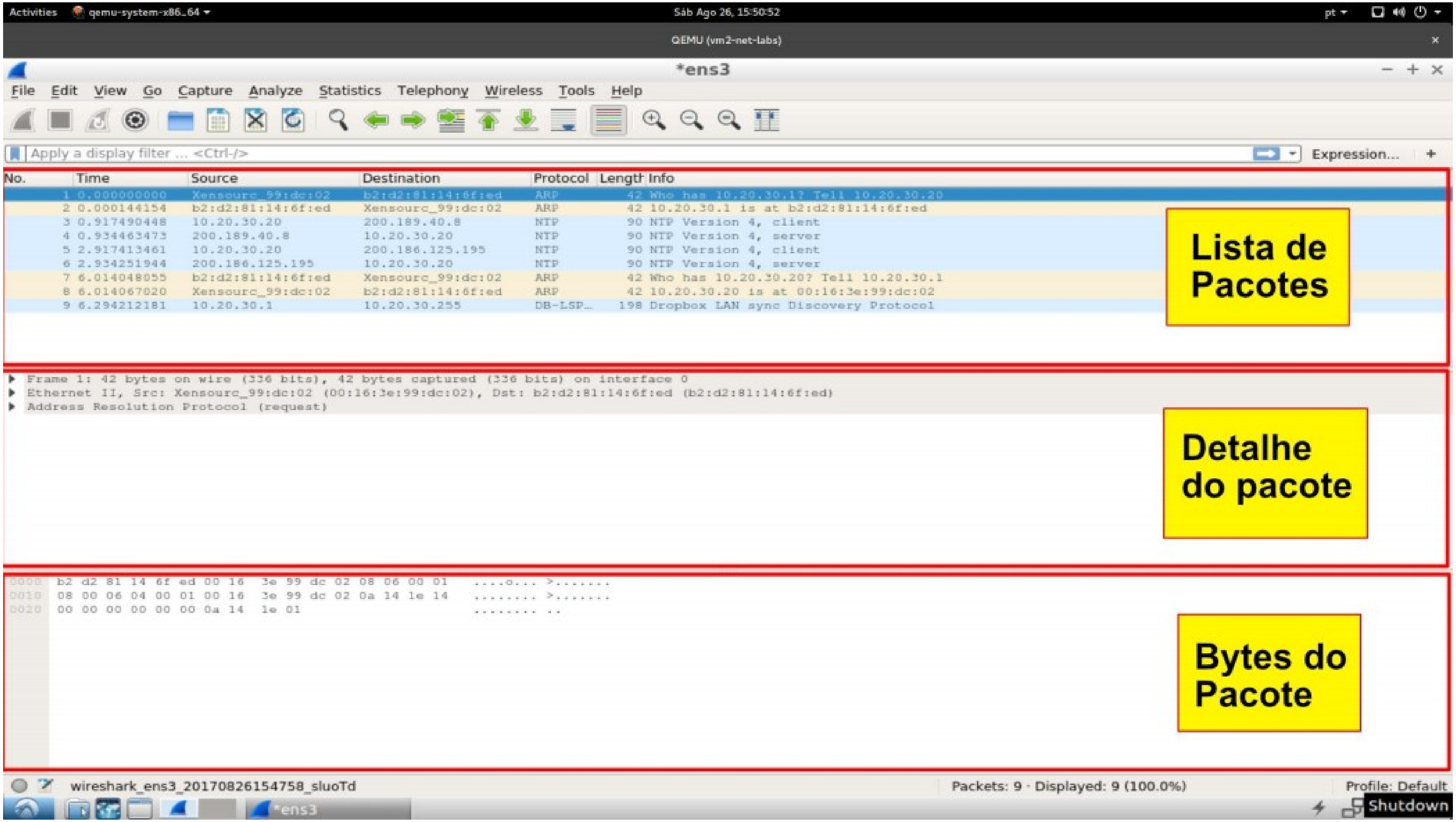
- Introdução ao Wireshark
- Interfaces
- Capturando pacotes
- Analisando pacotes
- Filtrando pacotes
- Atividade

Wireshark

É o analisador mais utilizado de redes.

- Multiplataforma: Windows, Linux, OS X, Solaris, FreeBSD.
- Open source (GPL)
- Inspeção de milhares de protocolos / Captura em tempo-real e análise offline
- Dados de rede capturados podem ser navegados via interface gráfica ou por terminal através da ferramenta Tshark
- Dados em tempo real podem ser lidos da Ethernet, IEEE 802.11, PPP/HDLC, ATM,
- Bluetooth, USB, Token Ring, Frame Relay, FDDI, entre outros

Interfaces



```
No.  Time         Source               Destination          Protocol Length Info
   1 0.000000000  Xensourc_99:dc:02    b2:d2:81:14:6f:ed    ARP        42 Who has 10.20.30.1? Tell 10.20.30.20
   2 0.000144154  b2:d2:81:14:6f:ed    Xensourc_99:dc:02    ARP        42 10.20.30.1 is at b2:d2:81:14:6f:ed
   3 0.917490448  10.20.30.20          200.189.40.8         NTP        90 NTP Version 4, client
   4 0.934463473  200.189.40.8         10.20.30.20          NTP        90 NTP Version 4, server
   5 2.917413461  10.20.30.20          200.186.125.195      NTP        90 NTP Version 4, client
   6 2.934251944  200.186.125.195      10.20.30.20          NTP        90 NTP Version 4, server
   7 6.014048055  b2:d2:81:14:6f:ed    Xensourc_99:dc:02    ARP        42 Who has 10.20.30.20? Tell 10.20.30.1
   8 6.014067020  Xensourc_99:dc:02    b2:d2:81:14:6f:ed    ARP        42 10.20.30.20 is at 00:16:3e:99:dc:02
   9 6.294212181  10.20.30.1           10.20.30.255         DB-LSP…   198 Dropbox LAN sync Discovery Protocol
```

Lista de Pacotes

```
▶ Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
▶ Ethernet II, Src: Xensourc_99:dc:02 (00:16:3e:99:dc:02), Dst: b2:d2:81:14:6f:ed (b2:d2:81:14:6f:ed)
▶ Address Resolution Protocol (request)
```

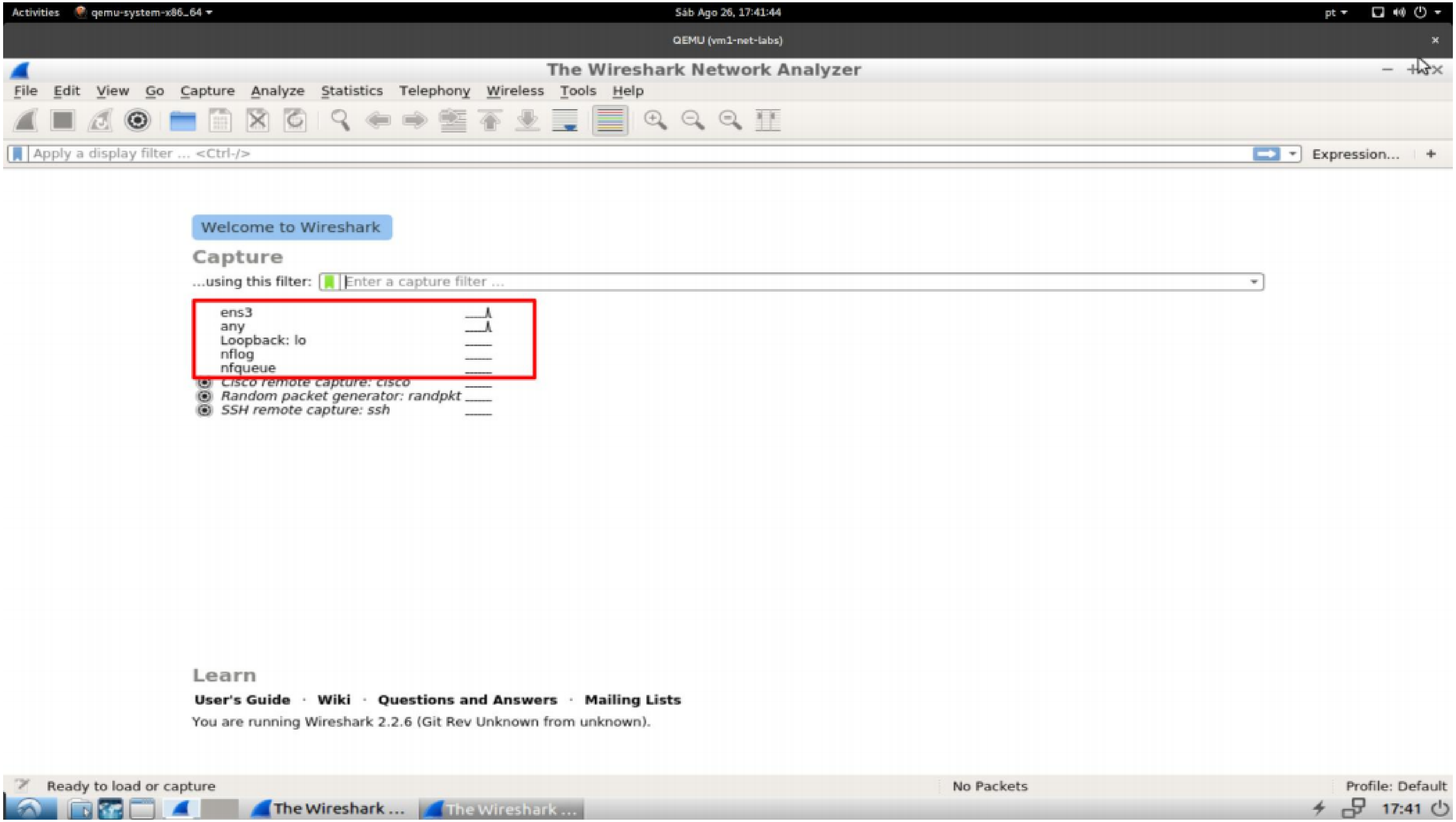
Detalhe do pacote

```
0000  b2 d2 81 14 6f ed 00 16  3e 99 dc 02 08 06 00 01   ....o... >.......
0010  08 00 06 04 00 01 00 16  3e 99 dc 02 0a 14 1e 14   ........ >.......
0020  00 00 00 00 00 00 0a 14  1e 01                     ........ ..
```

Bytes do Pacote

wireshark_ens3_20170826154758_sluoTd   Packets: 9 · Displayed: 9 (100.0%)   Profile: Default

Capturando pacotes (tela inicial)



Activities
qemu-system-x86_64
Sáb Ago 26, 17:41:44
pt
QEMU (vm1-net-labs)
The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

Welcome to Wireshark

Capture

...using this filter: Enter a capture filter ...

ens3
any
Loopback: lo
nflog
nfqueue
Cisco remote capture: cisco
Random packet generator: randpkt
SSH remote capture: ssh

Learn

User's Guide · Wiki · Questions and Answers · Mailing Lists

You are running Wireshark 2.2.6 (Git Rev Unknown from unknown).

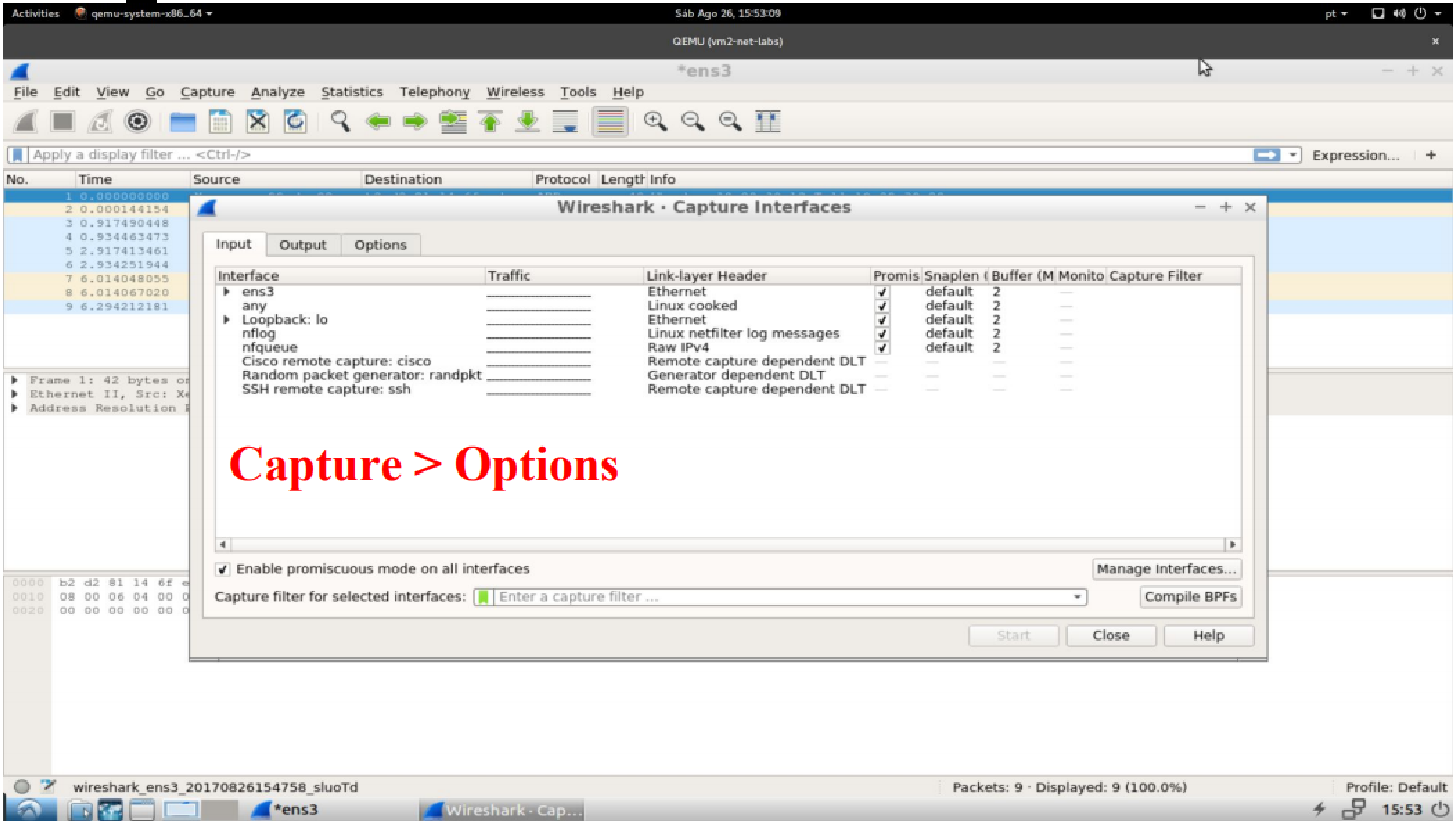
Ready to load or capture

No Packets

Profile: Default

The Wireshark ...
The Wireshark ...
17:41

Capturando pacotes (trocando interfaces)



Activities
qemu-system-x86_64
Sáb Ago 26, 15:53:09
pt
QEMU (vm2-net-labs)
*ens3
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Expression…
No. Time Source Destination Protocol Length Info
Wireshark · Capture Interfaces
Input Output Options
Manage Interfaces…
Compile BPFs
Start Close Help
*ens3
Wireshark · Cap…
15:53

```
1 0.000000000
2 0.000144154
3 0.917490448
4 0.934463473
5 2.917413461
6 2.934251944
7 6.014048055
8 6.014067020
9 6.294212181
```

```
▸ Frame 1: 42 bytes o
▸ Ethernet II, Src: X
▸ Address Resolution P
```

```
0000  b2 d2 81 14 6f e
0010  08 00 06 04 00 0
0020  00 00 00 00 00 0
```

Interface	Traffic	Link-layer Header	Promis	Snaplen (	Buffer (M	Monito	Capture Filter
▸ ens3		Ethernet	☑	default	2	—	
any		Linux cooked	☑	default	2	—	
▸ Loopback: lo		Ethernet	☑	default	2	—	
nflog		Linux netfilter log messages	☑	default	2	—	
nfqueue		Raw IPv4	☑	default	2	—	
Cisco remote capture: cisco		Remote capture dependent DLT	—	—	—	—	
Random packet generator: randpkt		Generator dependent DLT	—	—	—	—	
SSH remote capture: ssh		Remote capture dependent DLT	—	—	—	—	

Capture > Options

☑ Enable promiscuous mode on all interfaces

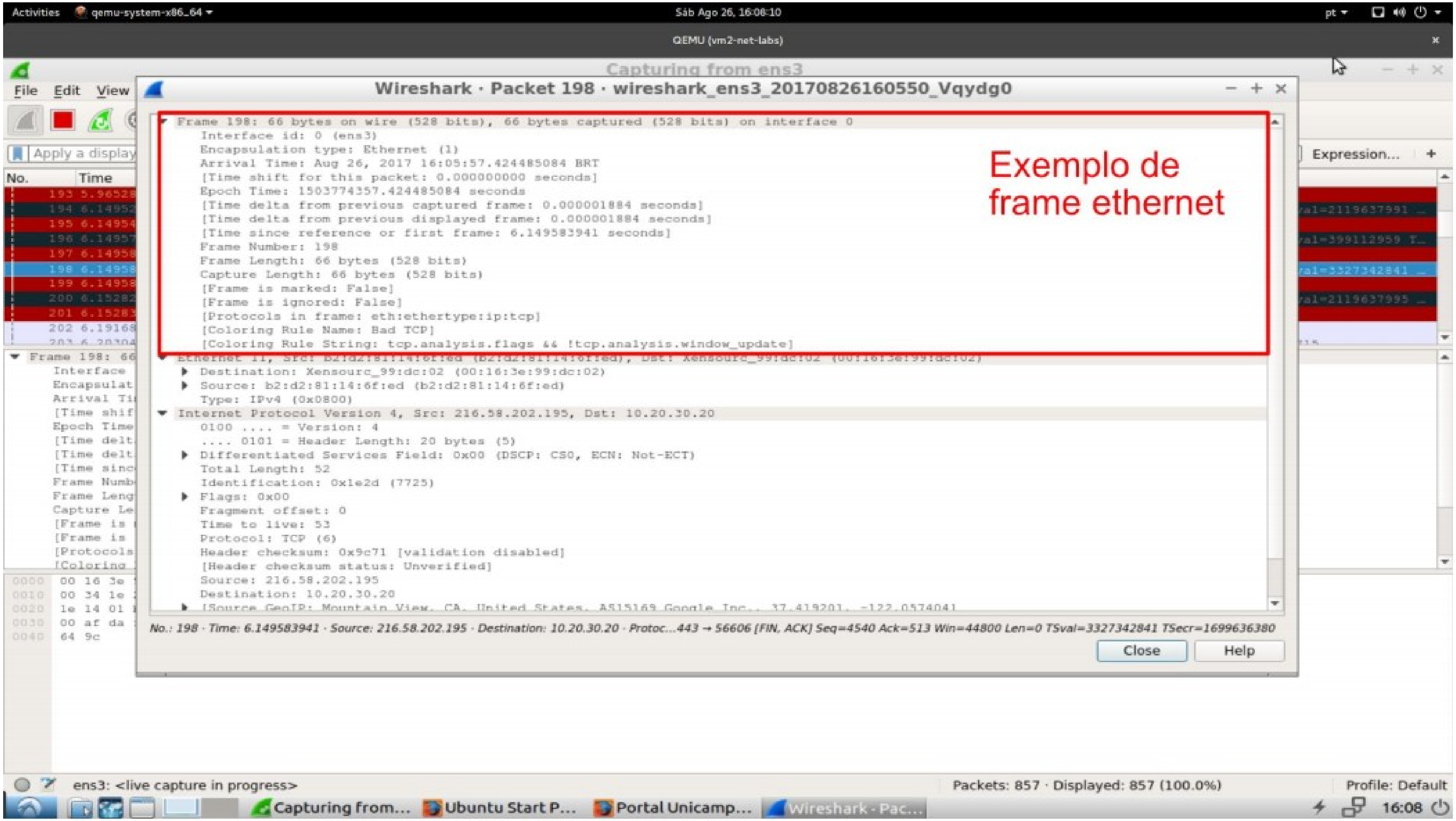
Capture filter for selected interfaces: Enter a capture filter …

wireshark_ens3_20170826154758_sluoTd

Packets: 9 · Displayed: 9 (100.0%)

Profile: Default

Analisando pacotes



Wireshark · Packet 198 · wireshark_ens3_20170826160550_Vqydg0

```
▾ Frame 198: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
    Interface id: 0 (ens3)
    Encapsulation type: Ethernet (1)
    Arrival Time: Aug 26, 2017 16:05:57.424485084 BRT
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1503774357.424485084 seconds
    [Time delta from previous captured frame: 0.000001884 seconds]
    [Time delta from previous displayed frame: 0.000001884 seconds]
    [Time since reference or first frame: 6.149583941 seconds]
    Frame Number: 198
    Frame Length: 66 bytes (528 bits)
    Capture Length: 66 bytes (528 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ip:tcp]
    [Coloring Rule Name: Bad TCP]
    [Coloring Rule String: tcp.analysis.flags && !tcp.analysis.window_update]
▾ Ethernet II, Src: b2:d2:81:14:6f:ed (b2:d2:81:14:6f:ed), Dst: Xensourc_99:dc:02 (00:16:3e:99:dc:02)
  ▸ Destination: Xensourc_99:dc:02 (00:16:3e:99:dc:02)
  ▸ Source: b2:d2:81:14:6f:ed (b2:d2:81:14:6f:ed)
    Type: IPv4 (0x0800)
▾ Internet Protocol Version 4, Src: 216.58.202.195, Dst: 10.20.30.20
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 52
    Identification: 0x1e2d (7725)
  ▸ Flags: 0x00
    Fragment offset: 0
    Time to live: 53
    Protocol: TCP (6)
    Header checksum: 0x9c71 [validation disabled]
    [Header checksum status: Unverified]
    Source: 216.58.202.195
    Destination: 10.20.30.20
  ▸ [Source GeoIP: Mountain View, CA, United States, AS15169 Google Inc., 37.419201, -122.057404]
```

No.: 198 · Time: 6.149583941 · Source: 216.58.202.195 · Destination: 10.20.30.20 · Protoc...443 → 56606 [FIN, ACK] Seq=4540 Ack=513 Win=44800 Len=0 TSval=3327342841 TSecr=1699636380

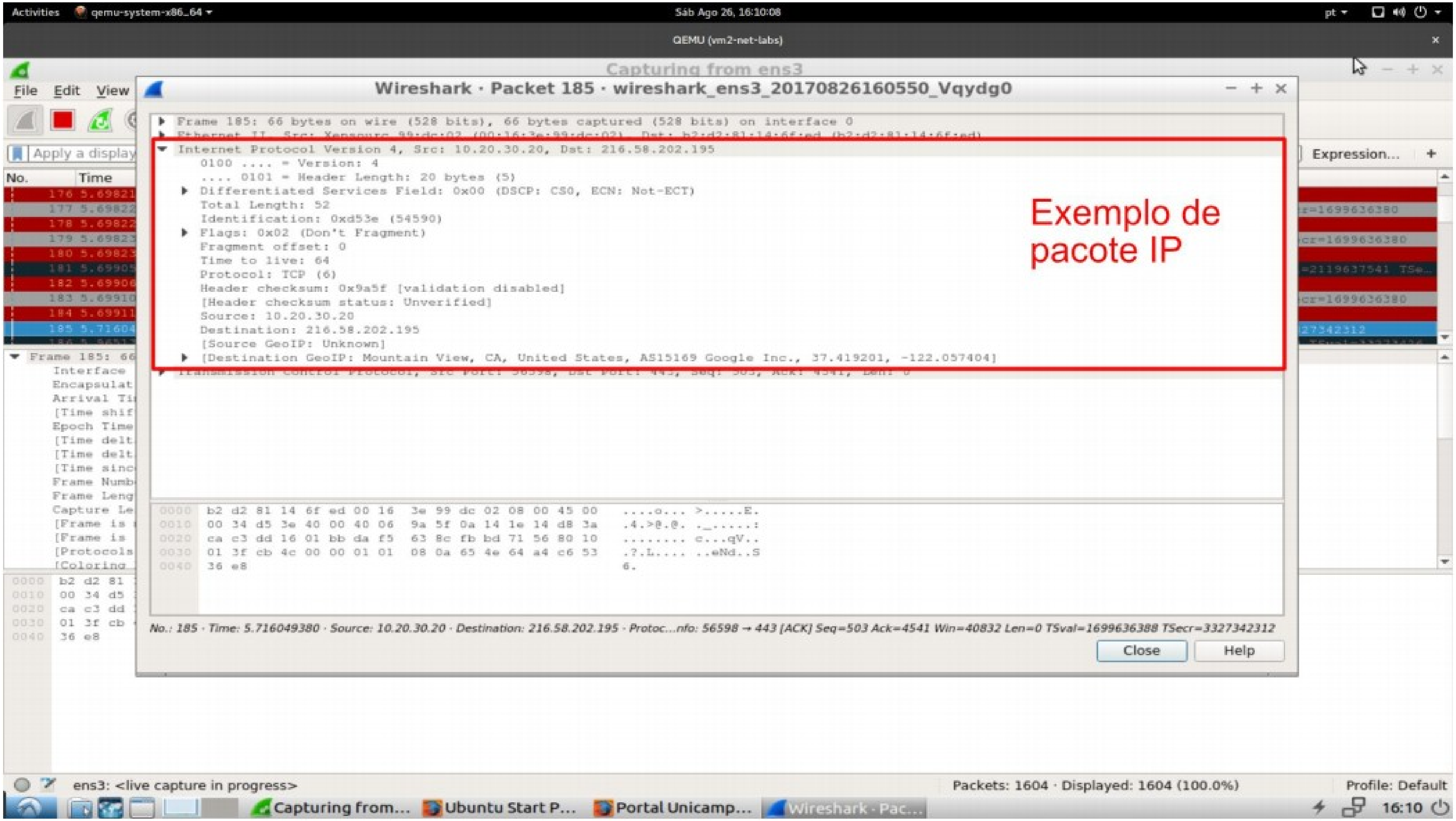
Exemplo de frame ethernet

ens3: <live capture in progress>

Packets: 857 · Displayed: 857 (100.0%)

Profile: Default

Analisando pacotes



Wireshark · Packet 185 · wireshark_ens3_20170826160550_Vqydg0

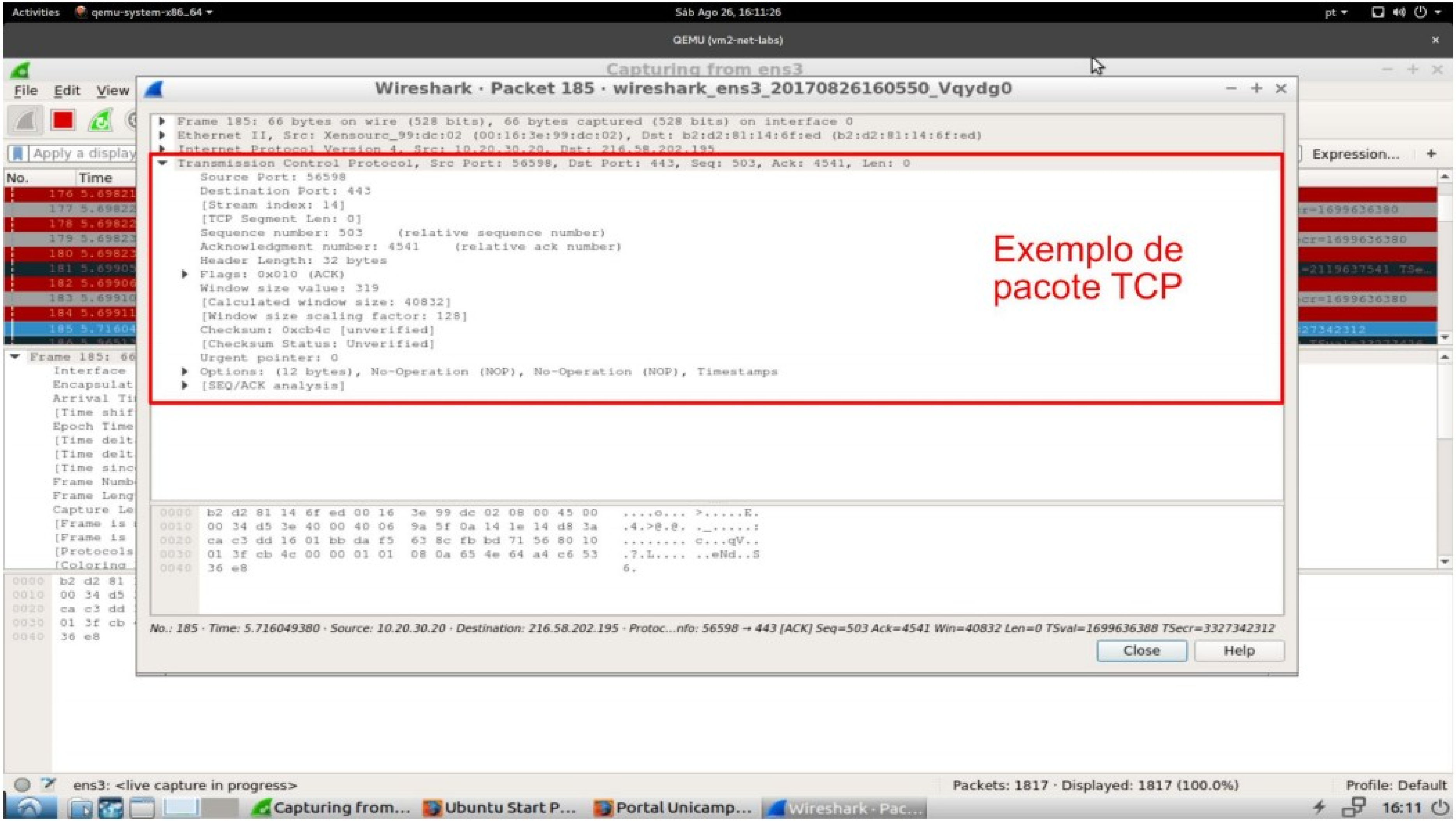
```
▶ Frame 185: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▶ Ethernet II, Src: Xensourc_99:dc:02 (00:16:3e:99:dc:02), Dst: b2:d2:81:14:6f:ed (b2:d2:81:14:6f:ed)
▼ Internet Protocol Version 4, Src: 10.20.30.20, Dst: 216.58.202.195
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▶ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 52
    Identification: 0xd53e (54590)
  ▶ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
    Header checksum: 0x9a5f [validation disabled]
    [Header checksum status: Unverified]
    Source: 10.20.30.20
    Destination: 216.58.202.195
    [Source GeoIP: Unknown]
  ▶ [Destination GeoIP: Mountain View, CA, United States, AS15169 Google Inc., 37.419201, -122.057404]
▶ Transmission Control Protocol, Src Port: 56598, Dst Port: 443, Seq: 503, Ack: 4541, Len: 0
```

Exemplo de pacote IP

```
0000  b2 d2 81 14 6f ed 00 16  3e 99 dc 02 08 00 45 00   ....o... >.....E.
0010  00 34 d5 3e 40 00 40 06  9a 5f 0a 14 1e 14 d8 3a   .4.>@.@. ._.....:
0020  ca c3 dd 16 01 bb da f5  63 8c fb bd 71 56 80 10   ........ c...qV..
0030  01 3f cb 4c 00 00 01 01  08 0a 65 4e 64 a4 c6 53   .?.L.... ..eNd..S
0040  36 e8                                              6.
```

No.: 185 · Time: 5.716049380 · Source: 10.20.30.20 · Destination: 216.58.202.195 · Protoc...nfo: 56598 → 443 [ACK] Seq=503 Ack=4541 Win=40832 Len=0 TSval=1699636388 TSecr=3327342312

Analisando pacotes



Wireshark · Packet 185 · wireshark_ens3_20170826160550_Vqydg0

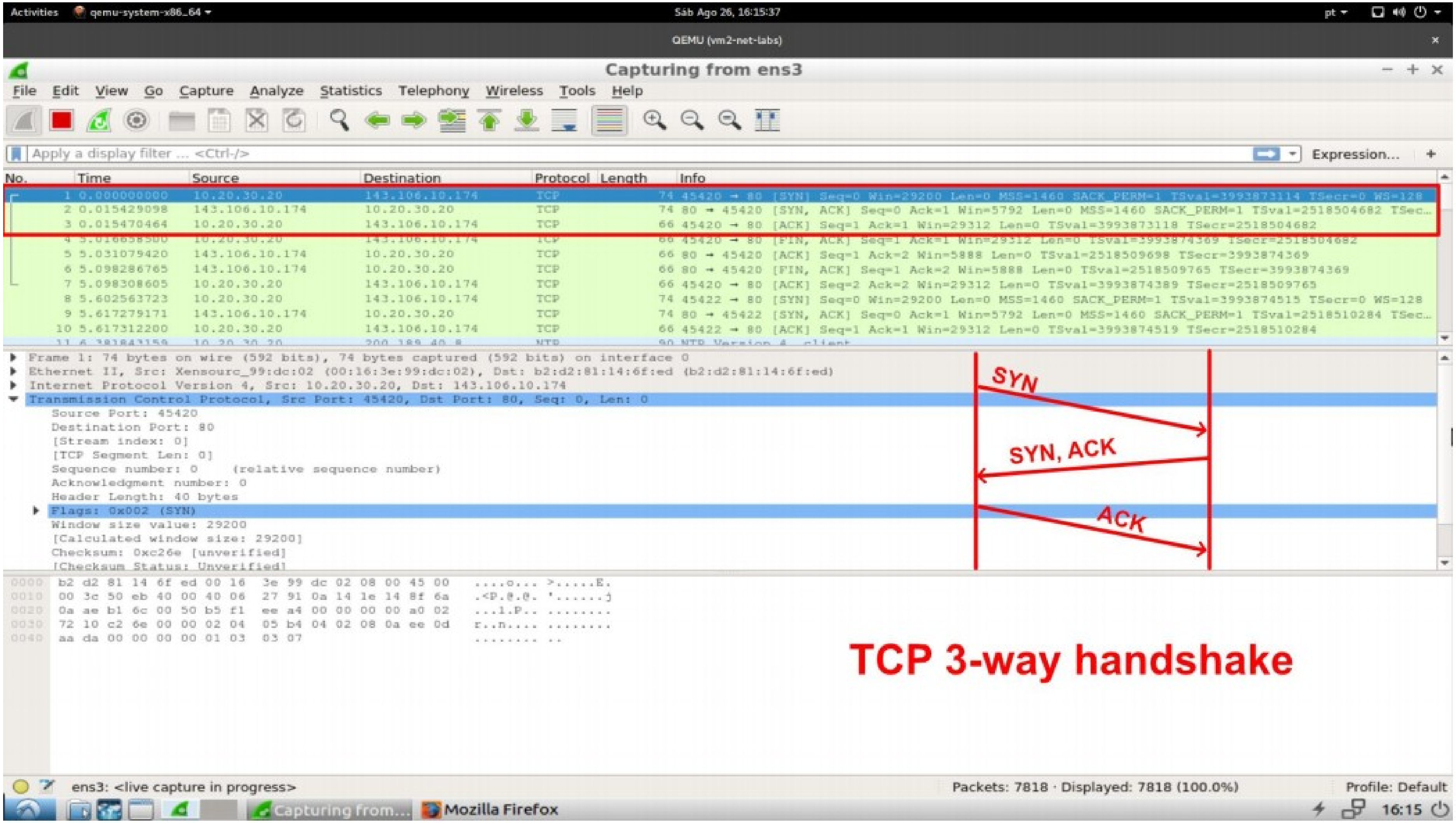
```
▶ Frame 185: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
▶ Ethernet II, Src: Xensourc_99:dc:02 (00:16:3e:99:dc:02), Dst: b2:d2:81:14:6f:ed (b2:d2:81:14:6f:ed)
▶ Internet Protocol Version 4, Src: 10.20.30.20, Dst: 216.58.202.195
▼ Transmission Control Protocol, Src Port: 56598, Dst Port: 443, Seq: 503, Ack: 4541, Len: 0
    Source Port: 56598
    Destination Port: 443
    [Stream index: 14]
    [TCP Segment Len: 0]
    Sequence number: 503    (relative sequence number)
    Acknowledgment number: 4541    (relative ack number)
    Header Length: 32 bytes
  ▶ Flags: 0x010 (ACK)
    Window size value: 319
    [Calculated window size: 40832]
    [Window size scaling factor: 128]
    Checksum: 0xcb4c [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  ▶ Options: (12 bytes), No-Operation (NOP), No-Operation (NOP), Timestamps
  ▶ [SEQ/ACK analysis]
```

Exemplo de pacote TCP

```
0000  b2 d2 81 14 6f ed 00 16  3e 99 dc 02 08 00 45 00   ....o... >.....E.
0010  00 34 d5 3e 40 00 40 06  9a 5f 0a 14 1e 14 d8 3a   .4.>@.@. ._.....:
0020  ca c3 dd 16 01 bb da f5  63 8c fb bd 71 56 80 10   ........ c...qV..
0030  01 3f cb 4c 00 00 01 01  08 0a 65 4e 64 a4 c6 53   .?.L.... ..eNd..S
0040  36 e8                                              6.
```

No.: 185 · Time: 5.716049380 · Source: 10.20.30.20 · Destination: 216.58.202.195 · Protoc...nfo: 56598 → 443 [ACK] Seq=503 Ack=4541 Win=40832 Len=0 TSval=1699636388 TSecr=3327342312

Analisando pacotes



```
No.  Time         Source          Destination     Protocol Length Info
   1 0.000000000  10.20.30.20     143.106.10.174  TCP      74 45420 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3993873114 TSecr=0 WS=128
   2 0.015429098  143.106.10.174  10.20.30.20     TCP      74 80 → 45420 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=2518504682 TSec…
   3 0.015470464  10.20.30.20     143.106.10.174  TCP      66 45420 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3993873118 TSecr=2518504682
   4 5.016658500  10.20.30.20     143.106.10.174  TCP      66 45420 → 80 [FIN, ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3993874369 TSecr=2518504682
   5 5.031079420  143.106.10.174  10.20.30.20     TCP      66 80 → 45420 [ACK] Seq=1 Ack=2 Win=5888 Len=0 TSval=2518509698 TSecr=3993874369
   6 5.098286765  143.106.10.174  10.20.30.20     TCP      66 80 → 45420 [FIN, ACK] Seq=1 Ack=2 Win=5888 Len=0 TSval=2518509765 TSecr=3993874369
   7 5.098308605  10.20.30.20     143.106.10.174  TCP      66 45420 → 80 [ACK] Seq=2 Ack=2 Win=29312 Len=0 TSval=3993874389 TSecr=2518509765
   8 5.602563723  10.20.30.20     143.106.10.174  TCP      74 45422 → 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3993874515 TSecr=0 WS=128
   9 5.617279171  143.106.10.174  10.20.30.20     TCP      74 80 → 45422 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=2518510284 TSec…
  10 5.617312200  10.20.30.20     143.106.10.174  TCP      66 45422 → 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3993874519 TSecr=2518510284
  11 6.381843159  10.20.30.20     200.189.40.8    NTP      90 NTP Version 4, client
```

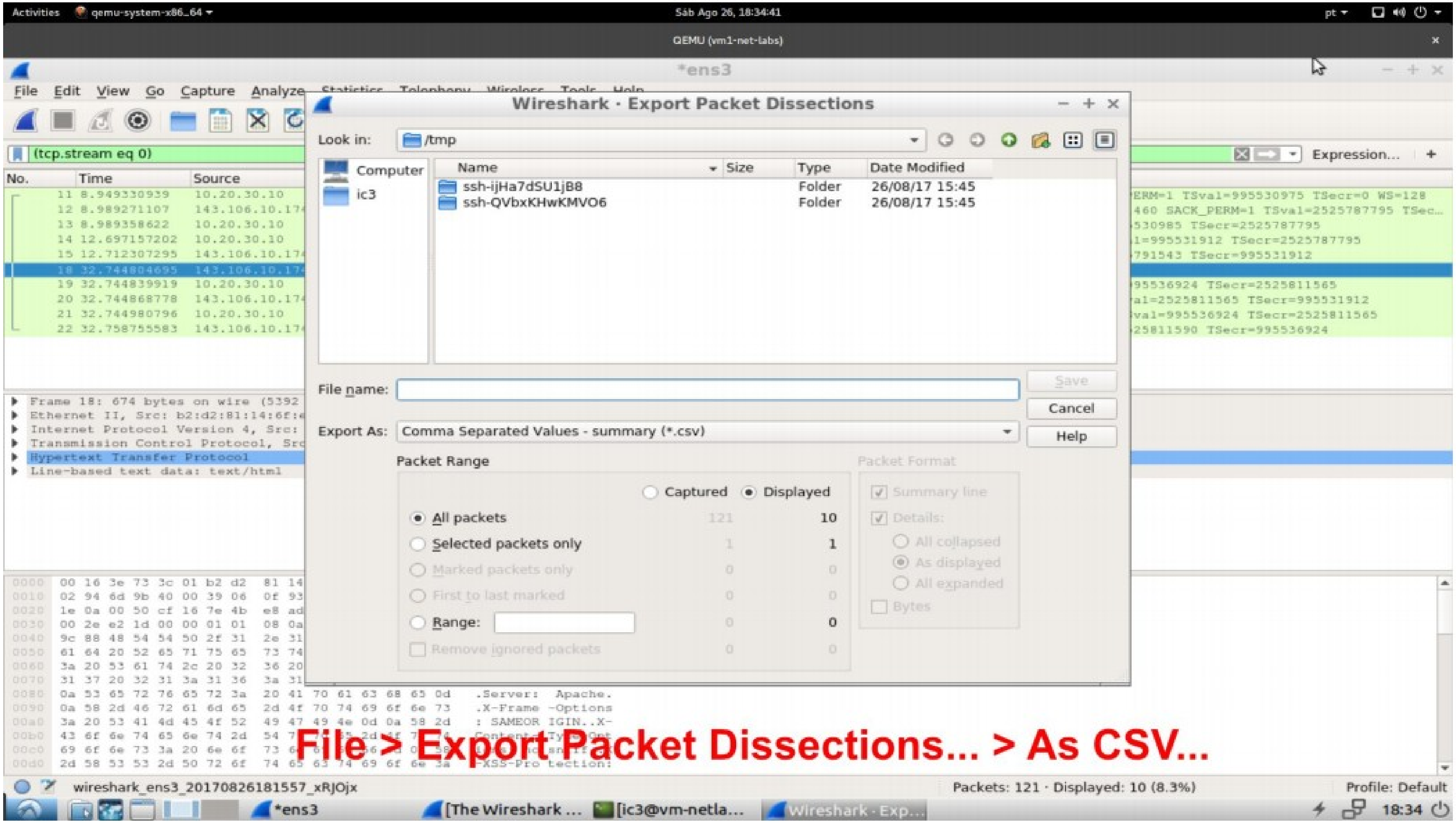
```
▸ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
▸ Ethernet II, Src: Xensourc_99:dc:02 (00:16:3e:99:dc:02), Dst: b2:d2:81:14:6f:ed (b2:d2:81:14:6f:ed)
▸ Internet Protocol Version 4, Src: 10.20.30.20, Dst: 143.106.10.174
▾ Transmission Control Protocol, Src Port: 45420, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 45420
    Destination Port: 80
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Acknowledgment number: 0
    Header Length: 40 bytes
  ▸ Flags: 0x002 (SYN)
    Window size value: 29200
    [Calculated window size: 29200]
    Checksum: 0xc26e [unverified]
    [Checksum Status: Unverified]
```

```
0000  b2 d2 81 14 6f ed 00 16  3e 99 dc 02 08 00 45 00   ....o... >.....E.
0010  00 3c 50 eb 40 00 40 06  27 91 0a 14 1e 14 8f 6a   .<P.@.@. '......j
0020  0a ae b1 6c 00 50 b5 f1  ee a4 00 00 00 00 a0 02   ...l.P.. ........
0030  72 10 c2 6e 00 00 02 04  05 b4 04 02 08 0a ee 0d   r..n.... ........
0040  aa da 00 00 00 00 01 03  03 07                     ........ ..
```

ens3: <live capture in progress>   Packets: 7818 · Displayed: 7818 (100.0%)   Profile: Default

TCP 3-way handshake

Salvando e manipulando pacotes



(tcp.stream eq 0)

```
No.   Time          Source
   11 8.949330939   10.20.30.10        ERM=1 TSval=995530975 TSecr=0 WS=128
   12 8.989271107   143.106.10.174     460 SACK_PERM=1 TSval=2525787795 TSec…
   13 8.989358622   10.20.30.10        530985 TSecr=2525787795
   14 12.697157202  10.20.30.10        l=995531912 TSecr=2525787795
   15 12.712307295  143.106.10.174     791543 TSecr=995531912
   18 32.744804695  143.106.10.174
   19 32.744839919  10.20.30.10        95536924 TSecr=2525811565
   20 32.744868778  143.106.10.174     al=2525811565 TSecr=995531912
   21 32.744980796  10.20.30.10        val=995536924 TSecr=2525811565
   22 32.758755583  143.106.10.174     25811590 TSecr=995536924
```

```
▶ Frame 18: 674 bytes on wire (5392
▶ Ethernet II, Src: b2:d2:81:14:6f:e
▶ Internet Protocol Version 4, Src:
▶ Transmission Control Protocol, Src
▶ Hypertext Transfer Protocol
▶ Line-based text data: text/html
```

```
0000  00 16 3e 73 3c 01 b2 d2  81 14
0010  02 94 6d 9b 40 00 39 06  0f 93
0020  1e 0a 00 50 cf 16 7e 4b  e8 ad
0030  00 2e e2 1d 00 00 01 01  08 0a
0040  9c 88 48 54 54 50 2f 31  2e 31
0050  61 64 20 52 65 71 75 65  73 74
0060  3a 20 53 61 74 2c 20 32  36 20
0070  31 37 20 32 31 3a 31 36  3a 31
0080  0a 53 65 72 76 65 72 3a  20 41 70 61 63 68 65 0d   .Server:  Apache.
0090  0a 58 2d 46 72 61 6d 65  2d 4f 70 74 69 6f 6e 73   .X-Frame -Options
00a0  3a 20 53 41 4d 45 4f 52  49 47 49 4e 0d 0a 58 2d   : SAMEOR IGIN..X-
00b0  43 6f 6e 74 65 6e 74 2d  54 7                      Content-Type-Opt
00c0  69 6f 6e 73 3a 20 6e 6f  73 6
00d0  2d 58 53 53 2d 50 72 6f  74 65 63 74 69 6f 6e 3a   -XSS-Pro tection:
```

Wireshark · Export Packet Dissections

Look in: /tmp

Name	Size	Type	Date Modified
ssh-ijHa7dSU1jB8		Folder	26/08/17 15:45
ssh-QVbxKHwKMVO6		Folder	26/08/17 15:45

File name:

Export As: Comma Separated Values - summary (*.csv)

Packet Range

	○ Captured	● Displayed
● All packets	121	10
○ Selected packets only	1	1
○ Marked packets only	0	0
○ First to last marked	0	0
○ Range:	0	0
□ Remove ignored packets	0	0

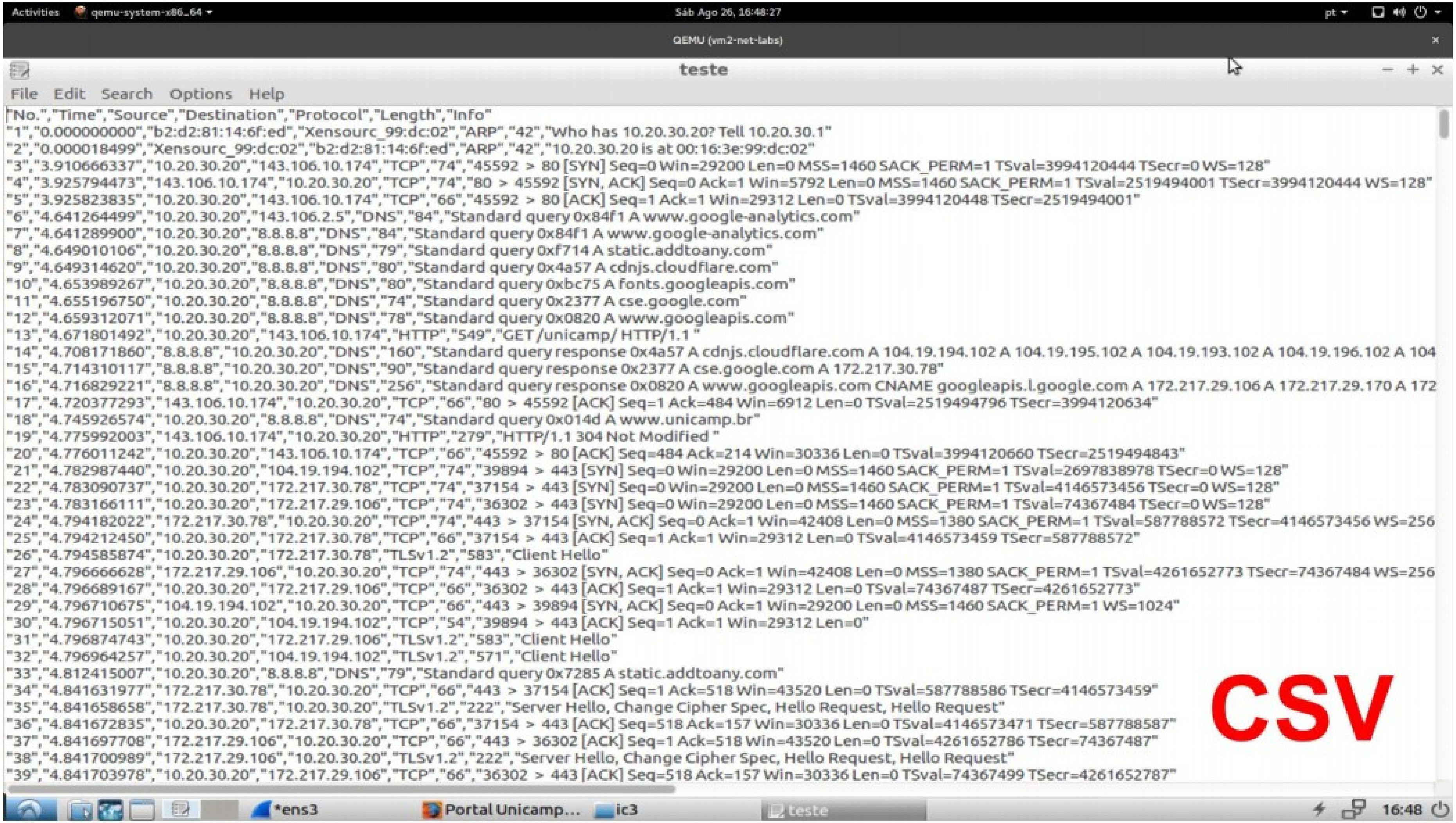
Packet Format

☑ Summary line
☑ Details:
○ All collapsed
● As displayed
○ All expanded
□ Bytes

wireshark_ens3_20170826181557_xRJOjx   Packets: 121 · Displayed: 10 (8.3%)   Profile: Default

File > Export Packet Dissections... > As CSV...

Salvando e manipulando pacotes



```
"No.","Time","Source","Destination","Protocol","Length","Info"
"1","0.000000000","b2:d2:81:14:6f:ed","Xensourc_99:dc:02","ARP","42","Who has 10.20.30.20? Tell 10.20.30.1"
"2","0.000018499","Xensourc_99:dc:02","b2:d2:81:14:6f:ed","ARP","42","10.20.30.20 is at 00:16:3e:99:dc:02"
"3","3.910666337","10.20.30.20","143.106.10.174","TCP","74","45592 > 80 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=3994120444 TSecr=0 WS=128"
"4","3.925794473","143.106.10.174","10.20.30.20","TCP","74","80 > 45592 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=2519494001 TSecr=3994120444 WS=128"
"5","3.925823835","10.20.30.20","143.106.10.174","TCP","66","45592 > 80 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=3994120448 TSecr=2519494001"
"6","4.641264499","10.20.30.20","143.106.2.5","DNS","84","Standard query 0x84f1 A www.google-analytics.com"
"7","4.641289900","10.20.30.20","8.8.8.8","DNS","84","Standard query 0x84f1 A www.google-analytics.com"
"8","4.649010106","10.20.30.20","8.8.8.8","DNS","79","Standard query 0xf714 A static.addtoany.com"
"9","4.649314620","10.20.30.20","8.8.8.8","DNS","80","Standard query 0x4a57 A cdnjs.cloudflare.com"
"10","4.653989267","10.20.30.20","8.8.8.8","DNS","80","Standard query 0xbc75 A fonts.googleapis.com"
"11","4.655196750","10.20.30.20","8.8.8.8","DNS","74","Standard query 0x2377 A cse.google.com"
"12","4.659312071","10.20.30.20","8.8.8.8","DNS","78","Standard query 0x0820 A www.googleapis.com"
"13","4.671801492","10.20.30.20","143.106.10.174","HTTP","549","GET /unicamp/ HTTP/1.1 "
"14","4.708171860","8.8.8.8","10.20.30.20","DNS","160","Standard query response 0x4a57 A cdnjs.cloudflare.com A 104.19.194.102 A 104.19.195.102 A 104.19.193.102 A 104.19.196.102 A 104
"15","4.714310117","8.8.8.8","10.20.30.20","DNS","90","Standard query response 0x2377 A cse.google.com A 172.217.30.78"
"16","4.716829221","8.8.8.8","10.20.30.20","DNS","256","Standard query response 0x0820 A www.googleapis.com CNAME googleapis.l.google.com A 172.217.29.106 A 172.217.29.170 A 172
"17","4.720377293","143.106.10.174","10.20.30.20","TCP","66","80 > 45592 [ACK] Seq=1 Ack=484 Win=6912 Len=0 TSval=2519494796 TSecr=3994120634"
"18","4.745926574","10.20.30.20","8.8.8.8","DNS","74","Standard query 0x014d A www.unicamp.br"
"19","4.775992003","143.106.10.174","10.20.30.20","HTTP","279","HTTP/1.1 304 Not Modified "
"20","4.776011242","10.20.30.20","143.106.10.174","TCP","66","45592 > 80 [ACK] Seq=484 Ack=214 Win=30336 Len=0 TSval=3994120660 TSecr=2519494843"
"21","4.782987440","10.20.30.20","104.19.194.102","TCP","74","39894 > 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2697838978 TSecr=0 WS=128"
"22","4.783090737","10.20.30.20","172.217.30.78","TCP","74","37154 > 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=4146573456 TSecr=0 WS=128"
"23","4.783166111","10.20.30.20","172.217.29.106","TCP","74","36302 > 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=74367484 TSecr=0 WS=128"
"24","4.794182022","172.217.30.78","10.20.30.20","TCP","74","443 > 37154 [SYN, ACK] Seq=0 Ack=1 Win=42408 Len=0 MSS=1380 SACK_PERM=1 TSval=587788572 TSecr=4146573456 WS=256
"25","4.794212450","10.20.30.20","172.217.30.78","TCP","66","37154 > 443 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=4146573459 TSecr=587788572"
"26","4.794585874","10.20.30.20","172.217.30.78","TLSv1.2","583","Client Hello"
"27","4.796666628","172.217.29.106","10.20.30.20","TCP","74","443 > 36302 [SYN, ACK] Seq=0 Ack=1 Win=42408 Len=0 MSS=1380 SACK_PERM=1 TSval=4261652773 TSecr=74367484 WS=256
"28","4.796689167","10.20.30.20","172.217.29.106","TCP","66","36302 > 443 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=74367487 TSecr=4261652773"
"29","4.796710675","104.19.194.102","10.20.30.20","TCP","66","443 > 39894 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=1024"
"30","4.796715051","10.20.30.20","104.19.194.102","TCP","54","39894 > 443 [ACK] Seq=1 Ack=1 Win=29312 Len=0"
"31","4.796874743","10.20.30.20","172.217.29.106","TLSv1.2","583","Client Hello"
"32","4.796964257","10.20.30.20","104.19.194.102","TLSv1.2","571","Client Hello"
"33","4.812415007","10.20.30.20","8.8.8.8","DNS","79","Standard query 0x7285 A static.addtoany.com"
"34","4.841631977","172.217.30.78","10.20.30.20","TCP","66","443 > 37154 [ACK] Seq=1 Ack=518 Win=43520 Len=0 TSval=587788586 TSecr=4146573459"
"35","4.841658658","172.217.30.78","10.20.30.20","TLSv1.2","222","Server Hello, Change Cipher Spec, Hello Request, Hello Request"
"36","4.841672835","10.20.30.20","172.217.30.78","TCP","66","37154 > 443 [ACK] Seq=518 Ack=157 Win=30336 Len=0 TSval=4146573471 TSecr=587788587"
"37","4.841697708","172.217.29.106","10.20.30.20","TCP","66","443 > 36302 [ACK] Seq=1 Ack=518 Win=43520 Len=0 TSval=4261652786 TSecr=74367487"
"38","4.841700989","172.217.29.106","10.20.30.20","TLSv1.2","222","Server Hello, Change Cipher Spec, Hello Request, Hello Request"
"39","4.841703978","10.20.30.20","172.217.29.106","TCP","66","36302 > 443 [ACK] Seq=518 Ack=157 Win=30336 Len=0 TSval=74367499 TSecr=4261652787"
```

CSV

Filtrando pacotes

Exemplos:

- Capturar apenas tráfego de/para o IP 172.18.5.4
  - host 172.168.5.4
- Capturar tráfego de/para uma faixa de IPs
  - net 192.168.0.0./24
  - net 192.168.0.0 mask 255.255.255.0
- Capturar tráfego vindo de uma faixa de IPs
  - src net 192.168.0.0/24
  - src net 192.168.0.0 mask 255.255.255.0
- Dica: use Expression...

Filtrando pacotes

Exemplos:

- Capturar tráfego para uma faixa de IPs

  dst net 192.168.0.0/24

  dst net 192.168.0.0 mask 255.255.255.0
- Capturar apenas tráfego DNS (porta 53)

  port 53
- Capturar tráfego não-HTTP e não-SMTP no seu servidor

  host www.example.com and not (port 80 or port 25)

  host www.example.com and not port 80 and not port 25
- Capturar tudo menos tráfego DNS e ARP

  port not 53 and not arp

Filtrando pacotes

Exemplos:

- Capturar tráfego dentro de uma faixa de portas

  (tcp[2:2] > 1500 and tcp[2:2] < 1550) or (tcp[4:2] > 1500 and tcp[4:2] < 1550)

  tcp portrange 1501-1549
- Capturar apenas o tipo EAPOL de Ethernet

  ether proto 0x888e
- Capturar apenas tráfego IP

  ip
- Capturar apenas tráfego unicast

  not broadcast and not multicast

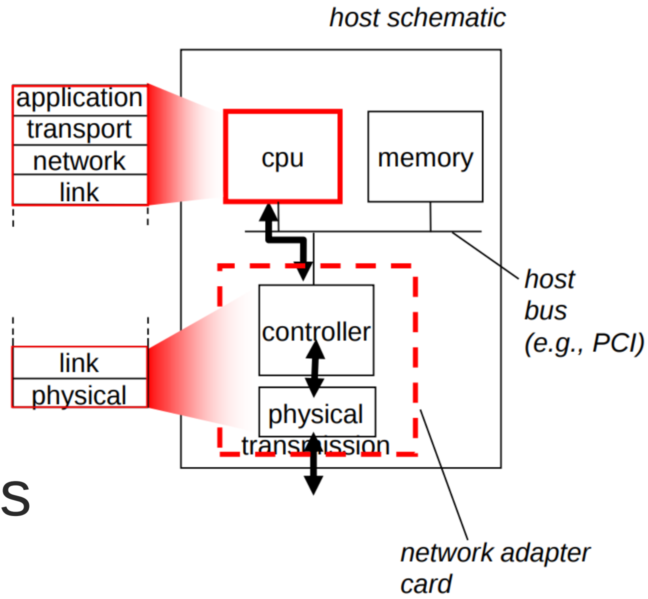
  (útil para limpar o ruído da rede se você quer apenas visualizar o tráfego de e para sua máquina)

TCPDump

TCPDump

Sniffer – analisador de tráfego

- Baseado na API libpcap
- Disponível para Unix-like,
- WinDump versão para Windows
- Outro sniffer: Wireshark



TCPDump

Filtros

- tcpdump host 10.90.100.1
- tcpdump src host 10.90.100.1
- tcpdump dst host 10.90.100.1
- tcpdump port <port number>
- tcpdump src port 80
- tcpdump dst port 80

- Filtering on a tcp flag
  - tcpdump 'tcp[tcpflags] & (tcp-syn) != 0'
  - tcpdump 'tcp[tcpflags] & (tcp-rst) != 0'

TCPDump

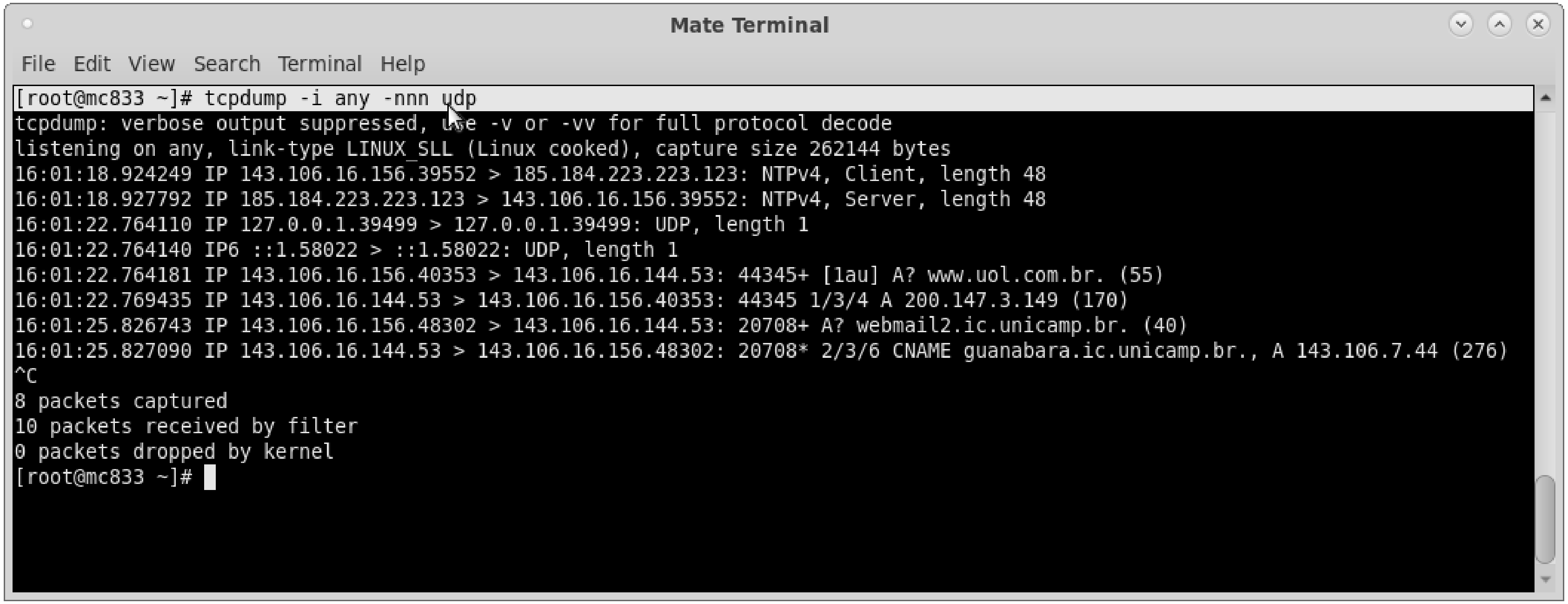
Combinando Expressões

- Operador AND
  - tcpdump host 10.90.100.1 and port 80
  - tcpdump src host 172.16.101.20 and dst port 80
  - tcpdump src host 172.16.101.20 and dst host 10.90.100.1

- Salvando saida em arquivo (pode ser vizualidado no wireshark)
  - tcpdump -w /tmp/saida.pcap

- Escolhendo a interface para sniffer
  - tcpdump -i any
  - tcpdump -i eth0

TCPDump



```
[root@mc833 ~]# tcpdump -i any -nnn udp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
16:01:18.924249 IP 143.106.16.156.39552 > 185.184.223.223.123: NTPv4, Client, length 48
16:01:18.927792 IP 185.184.223.223.123 > 143.106.16.156.39552: NTPv4, Server, length 48
16:01:22.764110 IP 127.0.0.1.39499 > 127.0.0.1.39499: UDP, length 1
16:01:22.764140 IP6 ::1.58022 > ::1.58022: UDP, length 1
16:01:22.764181 IP 143.106.16.156.40353 > 143.106.16.144.53: 44345+ [1au] A? www.uol.com.br. (55)
16:01:22.769435 IP 143.106.16.144.53 > 143.106.16.156.40353: 44345 1/3/4 A 200.147.3.149 (170)
16:01:25.826743 IP 143.106.16.156.48302 > 143.106.16.144.53: 20708+ A? webmail2.ic.unicamp.br. (40)
16:01:25.827090 IP 143.106.16.144.53 > 143.106.16.156.48302: 20708* 2/3/6 CNAME guanabara.ic.unicamp.br., A 143.106.7.44 (276)
^C
8 packets captured
10 packets received by filter
0 packets dropped by kernel
[root@mc833 ~]# 
```