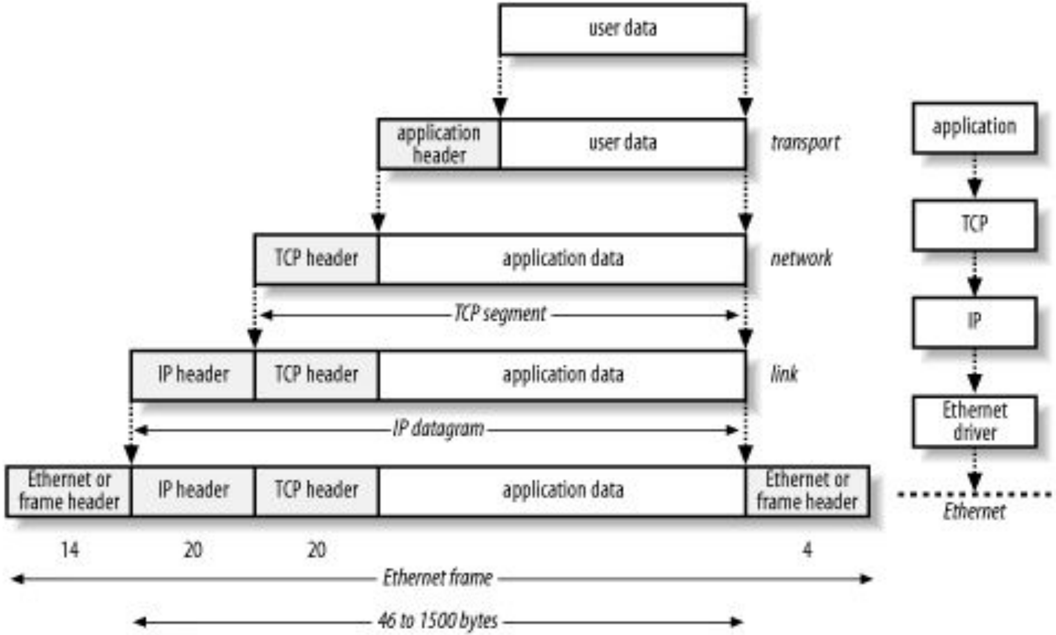
Network measurements

CS-335a 15/10/2021

TA: Eva Perontsi, evaperon@csd.uoc.gr

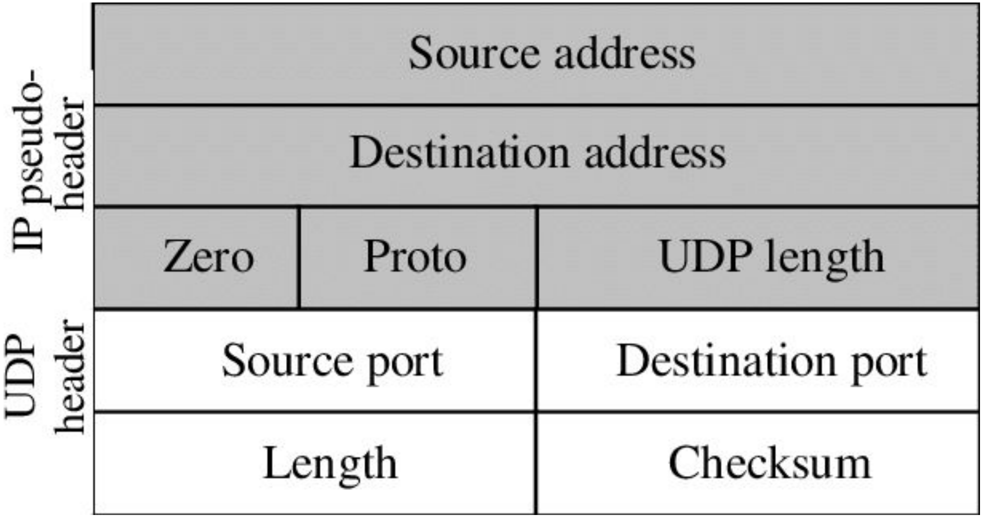
Fundamentals-definitions

Network packet



Packet: a formatted unit of data carried by a packet-switched network. A packet consists of both control information and user data.

Header



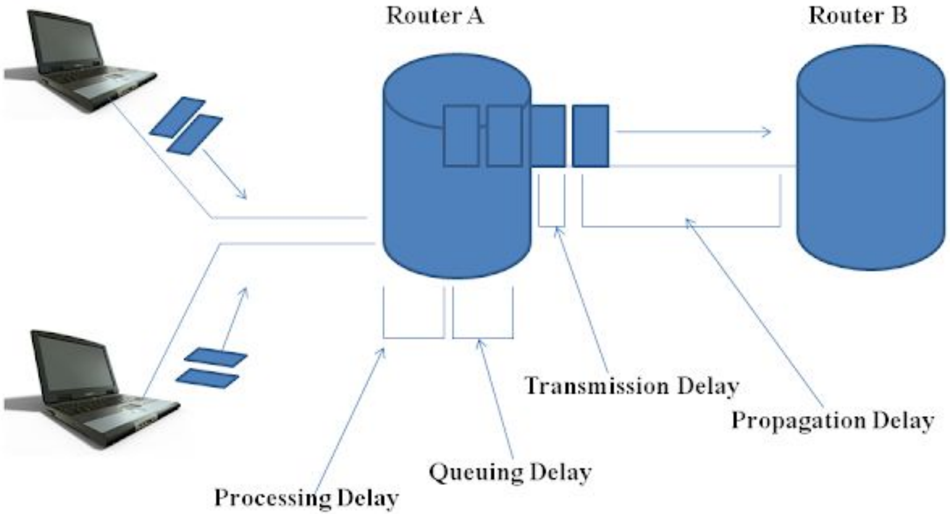
IP pseudo-header	Source address		
	Destination address		
	Zero	Proto	UDP length
UDP header	Source port		Destination port
	Length		Checksum

Header: supplemental data placed at the beginning of a block of data being stored or transmitted

Delay: It specifies the latency for a bit of data to travel across the network from one communication endpoint to another.

- Processing delay – time it takes a router to process the packet header
- Queuing delay – time the packet spends in routing queues
- Transmission delay – time required for the router to push out the packet
- Propagation delay – time it takes a bit to propagate from one router to the next

Network delay

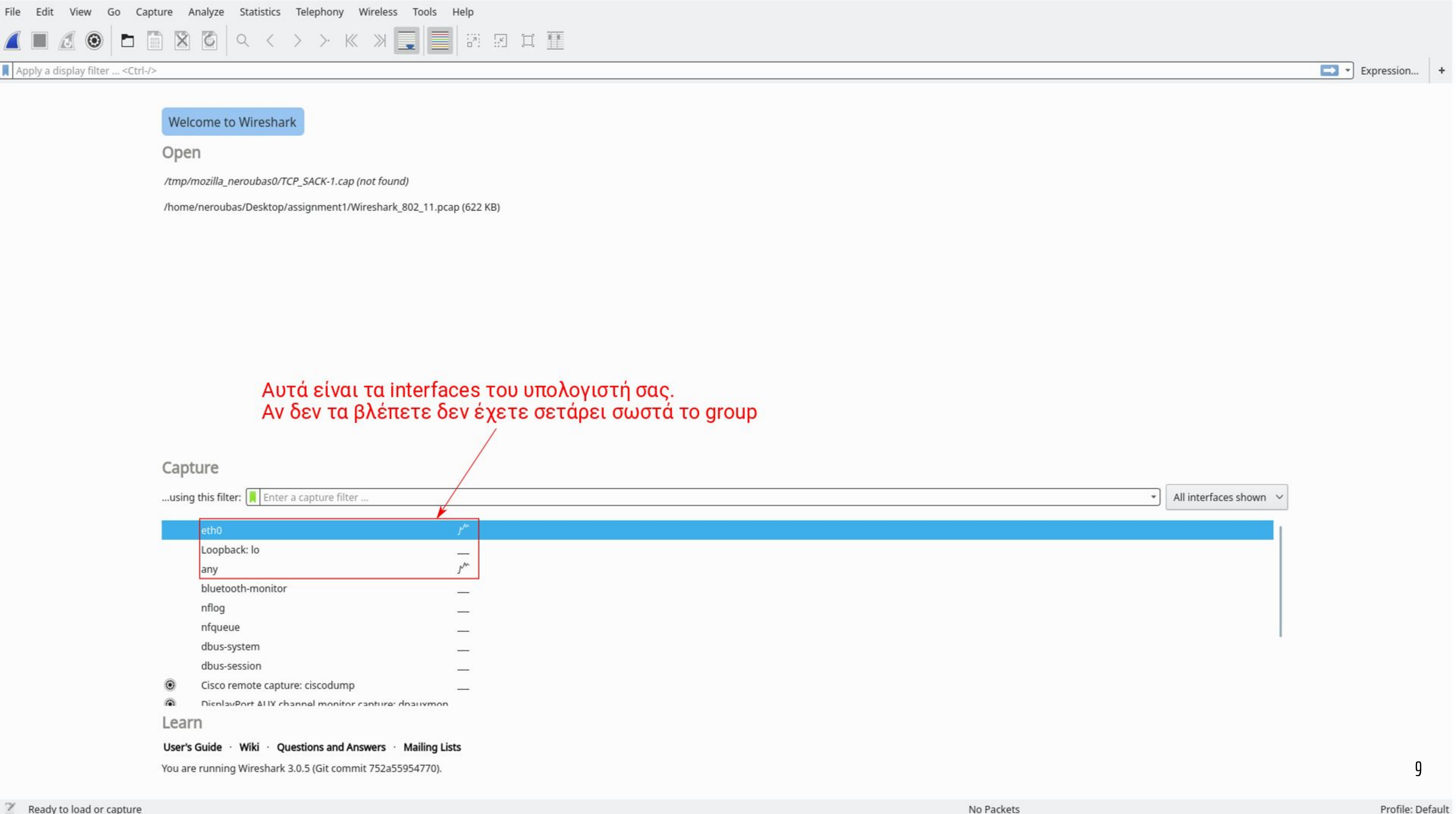


Wireshark

Wireshark setup

Wireshark is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

To install and setup wireshark follow the instructions on the hy335a website.



Welcome to Wireshark

Open

/tmp/mozilla_neroubas0/TCP_SACK-1.cap (not found)

/home/neroubas/Desktop/assignment1/Wireshark_802_11.pcap (622 KB)

Αυτά είναι τα interfaces του υπολογιστή σας.
Αν δεν τα βλέπετε δεν έχετε σετάρει σωστά το group

Capture

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

eth0
Loopback: lo
any
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon

Learn

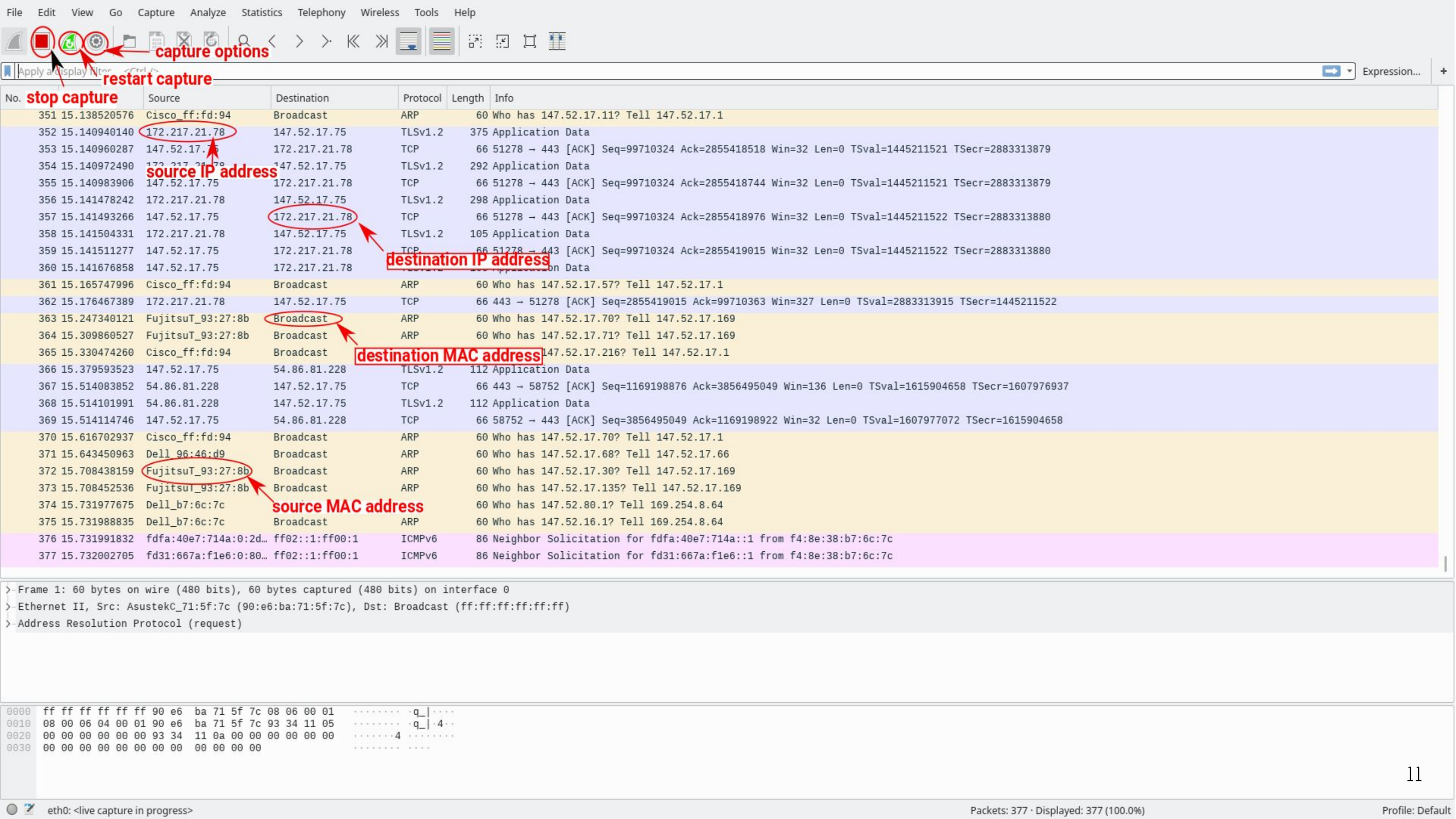
User's Guide · Wiki · Questions and Answers · Mailing Lists

You are running Wireshark 3.0.5 (Git commit 752a55954770).

Ready to load or capture   No Packets   Profile: Default

Wireshark capturing

- In order to start seeing packets, choose an interface and click “Start capturing”
- The packets you see, are the packets that go through your network in real time
- To stop capturing click “stop”
- You can click on a packet to see its details, like source and destination addresses, ports, etc.
- After you stop capturing, you can filter the capture packets, and change the Capture options



```
No.  Time         Source                Destination      Protocol  Length Info
351 15.138520576 Cisco_ff:fd:94        Broadcast         ARP        60 Who has 147.52.17.11? Tell 147.52.17.1
352 15.140940140 172.217.21.78         147.52.17.75      TLSv1.2   375 Application Data
353 15.140960287 147.52.17.75          172.217.21.78     TCP        66 51278 → 443 [ACK] Seq=99710324 Ack=2855418518 Win=32 Len=0 TSval=1445211521 TSecr=2883313879
354 15.140972490 172.217.21.78         147.52.17.75      TLSv1.2   292 Application Data
355 15.140983906 147.52.17.75          172.217.21.78     TCP        66 51278 → 443 [ACK] Seq=99710324 Ack=2855418744 Win=32 Len=0 TSval=1445211521 TSecr=2883313879
356 15.141478242 172.217.21.78         147.52.17.75      TLSv1.2   298 Application Data
357 15.141493266 147.52.17.75          172.217.21.78     TCP        66 51278 → 443 [ACK] Seq=99710324 Ack=2855418976 Win=32 Len=0 TSval=1445211522 TSecr=2883313880
358 15.141504331 172.217.21.78         147.52.17.75      TLSv1.2   105 Application Data
359 15.141511277 147.52.17.75          172.217.21.78     TCP        66 51278 → 443 [ACK] Seq=99710324 Ack=2855419015 Win=32 Len=0 TSval=1445211522 TSecr=2883313880
360 15.141676858 147.52.17.75          172.217.21.78     TLSv1.2   105 Application Data
361 15.165747996 Cisco_ff:fd:94        Broadcast         ARP        60 Who has 147.52.17.57? Tell 147.52.17.1
362 15.176467389 172.217.21.78         147.52.17.75      TCP        66 443 → 51278 [ACK] Seq=2855419015 Ack=99710363 Win=327 Len=0 TSval=2883313915 TSecr=1445211522
363 15.247340121 FujitsuT_93:27:8b     Broadcast         ARP        60 Who has 147.52.17.70? Tell 147.52.17.169
364 15.309860527 FujitsuT_93:27:8b     Broadcast         ARP        60 Who has 147.52.17.71? Tell 147.52.17.169
365 15.330474260 Cisco_ff:fd:94        Broadcast         ARP        60 Who has 147.52.17.216? Tell 147.52.17.1
366 15.379593523 147.52.17.75          54.86.81.228      TLSv1.2   112 Application Data
367 15.514083852 54.86.81.228          147.52.17.75      TCP        66 443 → 58752 [ACK] Seq=1169198876 Ack=3856495049 Win=136 Len=0 TSval=1615904658 TSecr=1607976937
368 15.514101991 54.86.81.228          147.52.17.75      TLSv1.2   112 Application Data
369 15.514114746 147.52.17.75          54.86.81.228      TCP        66 58752 → 443 [ACK] Seq=3856495049 Ack=1169198922 Win=32 Len=0 TSval=1607977072 TSecr=1615904658
370 15.616702937 Cisco_ff:fd:94        Broadcast         ARP        60 Who has 147.52.17.70? Tell 147.52.17.1
371 15.643450963 Dell_96:46:d9         Broadcast         ARP        60 Who has 147.52.17.68? Tell 147.52.17.66
372 15.708438159 FujitsuT_93:27:8b     Broadcast         ARP        60 Who has 147.52.17.30? Tell 147.52.17.169
373 15.708452536 FujitsuT_93:27:8b     Broadcast         ARP        60 Who has 147.52.17.135? Tell 147.52.17.169
374 15.731977675 Dell_b7:6c:7c         Broadcast         ARP        60 Who has 147.52.80.1? Tell 169.254.8.64
375 15.731988835 Dell_b7:6c:7c         Broadcast         ARP        60 Who has 147.52.16.1? Tell 169.254.8.64
376 15.731991832 fdfa:40e7:714a:0:2d… ff02::1:ff00:1    ICMPv6     86 Neighbor Solicitation for fdfa:40e7:714a::1 from f4:8e:38:b7:6c:7c
377 15.732002705 fd31:667a:f1e6:0:80… ff02::1:ff00:1    ICMPv6     86 Neighbor Solicitation for fd31:667a:f1e6::1 from f4:8e:38:b7:6c:7c
```

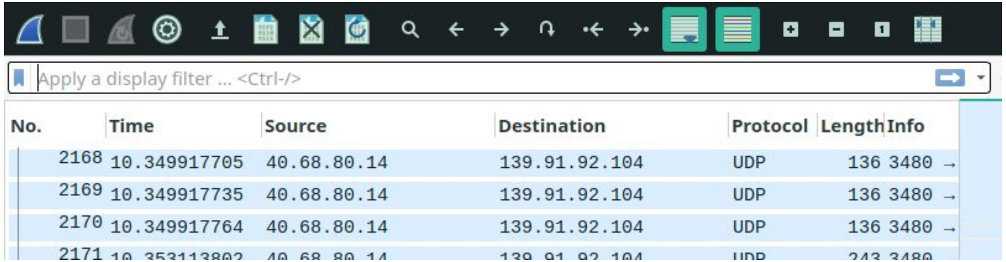
```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: AsustekC_71:5f:7c (90:e6:ba:71:5f:7c), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)
```

```
0000  ff ff ff ff ff ff 90 e6  ba 71 5f 7c 08 06 00 01   ........ .q_|....
0010  08 00 06 04 00 01 90 e6  ba 71 5f 7c 93 34 11 05   ........ .q_|.4..
0020  00 00 00 00 00 00 93 34  11 0a 00 00 00 00 00 00   .......4 ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

eth0: <live capture in progress>   Packets: 377 · Displayed: 377 (100.0%)   Profile: Default

Wireshark filtering

On the top bar:

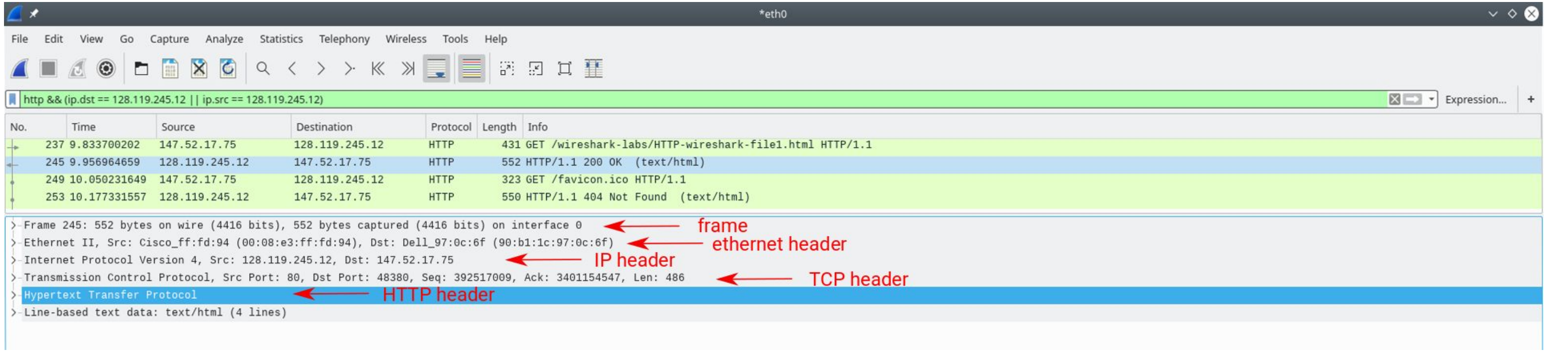


No.	Time	Source	Destination	Protocol	Length	Info
2168	10.349917705	40.68.80.14	139.91.92.104	UDP	136	3480 →
2169	10.349917735	40.68.80.14	139.91.92.104	UDP	136	3480 →
2170	10.349917764	40.68.80.14	139.91.92.104	UDP	136	3480 →
2171	10.353113802	40.68.80.14	139.91.92.104	UDP	243	3480

You can apply filters to the captured packets. To see just TCP packets, just write “tcp”. You can also combine filters, using logical AND (&&), logical OR (||) and logical NOT (!).

Wireshark encapsulation

- Encapsulation allows us to use different protocols in all levels of the TCP/IP stack.
- Wireshark shows us the headers of all these levels (e.g. an HTTP packet)



http && (ip.dst == 128.119.245.12 || ip.src == 128.119.245.12)

No.	Time	Source	Destination	Protocol	Length	Info
237	9.833700202	147.52.17.75	128.119.245.12	HTTP	431	GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1
245	9.956964659	128.119.245.12	147.52.17.75	HTTP	552	HTTP/1.1 200 OK (text/html)
249	10.050231649	147.52.17.75	128.119.245.12	HTTP	323	GET /favicon.ico HTTP/1.1
253	10.177331557	128.119.245.12	147.52.17.75	HTTP	550	HTTP/1.1 404 Not Found (text/html)

```
> Frame 245: 552 bytes on wire (4416 bits), 552 bytes captured (4416 bits) on interface 0
> Ethernet II, Src: Cisco_ff:fd:94 (00:08:e3:ff:fd:94), Dst: Dell_97:0c:6f (90:b1:1c:97:0c:6f)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 147.52.17.75
> Transmission Control Protocol, Src Port: 80, Dst Port: 48380, Seq: 392517009, Ack: 3401154547, Len: 486
> Hypertext Transfer Protocol
> Line-based text data: text/html (4 lines)
```

frame

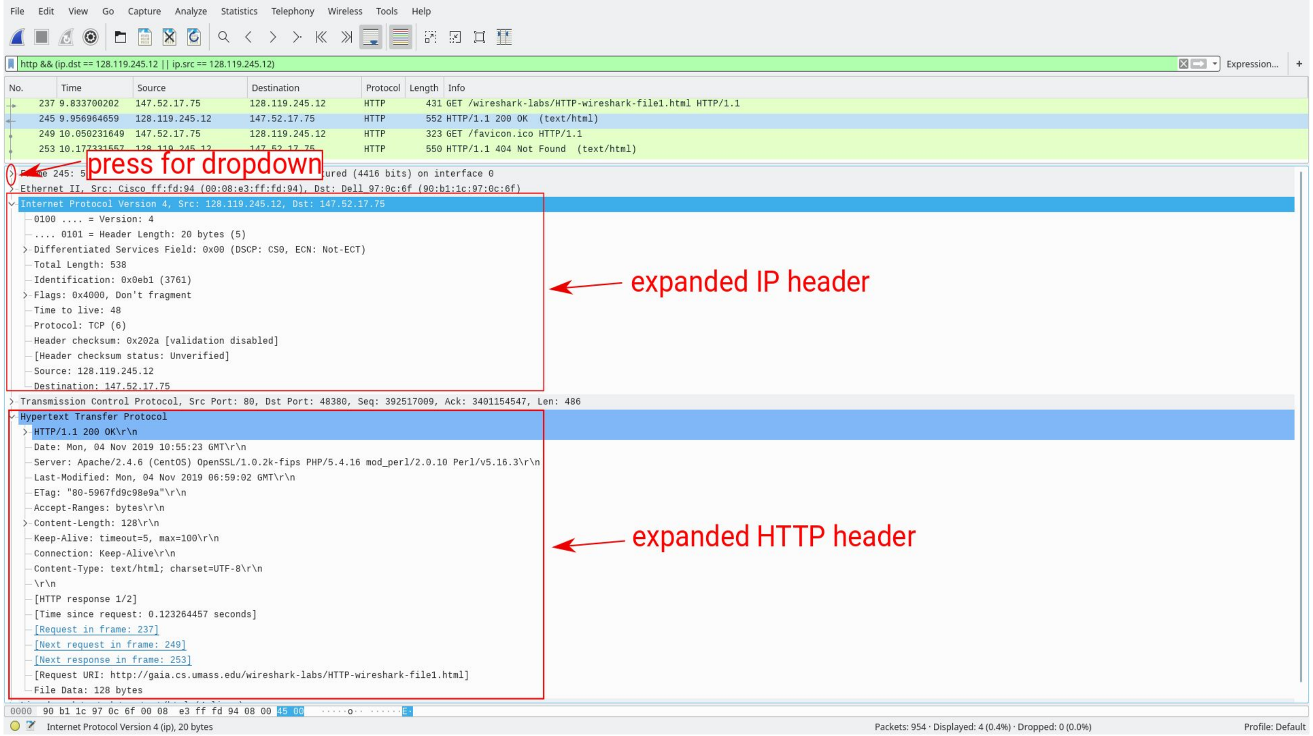
ethernet header

IP header

TCP header

HTTP header

Wireshark encapsulation



File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
+

```
http && (ip.dst == 128.119.245.12 || ip.src == 128.119.245.12)
```

```
No.   Time          Source           Destination      Protocol Length Info
  237 9.833700202   147.52.17.75     128.119.245.12   HTTP        431 GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1
  245 9.956964659   128.119.245.12   147.52.17.75     HTTP        552 HTTP/1.1 200 OK  (text/html)
  249 10.050231649  147.52.17.75     128.119.245.12   HTTP        323 GET /favicon.ico HTTP/1.1
  253 10.177331557  128.119.245.12   147.52.17.75     HTTP        550 HTTP/1.1 404 Not Found  (text/html)
```

press for dropdown

```
> F  me 245: 5                              ured (4416 bits) on interface 0
> Ethernet II, Src: Cisco_ff:fd:94 (00:08:e3:ff:fd:94), Dst: Dell_97:0c:6f (90:b1:1c:97:0c:6f)
v Internet Protocol Version 4, Src: 128.119.245.12, Dst: 147.52.17.75
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 538
    Identification: 0x0eb1 (3761)
  > Flags: 0x4000, Don't fragment
    Time to live: 48
    Protocol: TCP (6)
    Header checksum: 0x202a [validation disabled]
    [Header checksum status: Unverified]
    Source: 128.119.245.12
    Destination: 147.52.17.75
> Transmission Control Protocol, Src Port: 80, Dst Port: 48380, Seq: 392517009, Ack: 3401154547, Len: 486
v Hypertext Transfer Protocol
  > HTTP/1.1 200 OK\r\n
    Date: Mon, 04 Nov 2019 10:55:23 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16 mod_perl/2.0.10 Perl/v5.16.3\r\n
    Last-Modified: Mon, 04 Nov 2019 06:59:02 GMT\r\n
    ETag: "80-5967fd9c98e9a"\r\n
    Accept-Ranges: bytes\r\n
  > Content-Length: 128\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [HTTP response 1/2]
    [Time since request: 0.123264457 seconds]
    [Request in frame: 237]
    [Next request in frame: 249]
    [Next response in frame: 253]
    [Request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
    File Data: 128 bytes
```

expanded IP header

expanded HTTP header

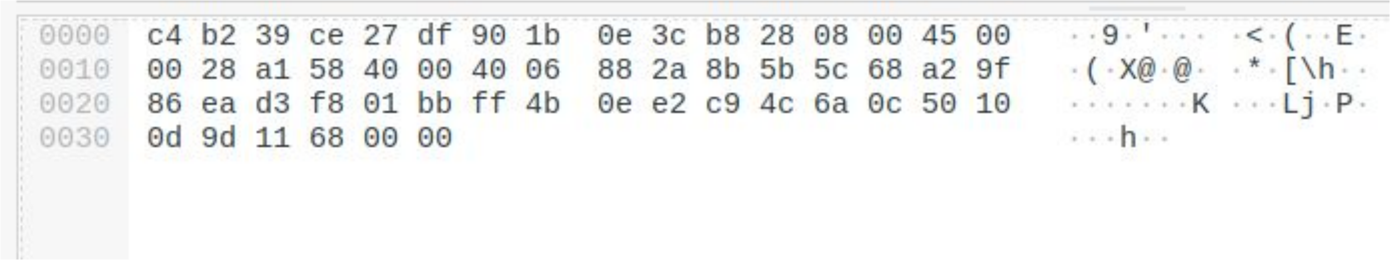
```
0000  90 b1 1c 97 0c 6f 00 08  e3 ff fd 94 08 00 45 00   ·····o·· ······E·
```

```
Internet Protocol Version 4 (ip), 20 bytes          Packets: 954 · Displayed: 4 (0.4%) · Dropped: 0 (0.0%)          Profile: Default
```

Packet details

As mentioned, you can see the details of each package by clicking on it. You can double-click to open it in a new window. There you can see the packet’s details, as they’re shown on the previous slide.

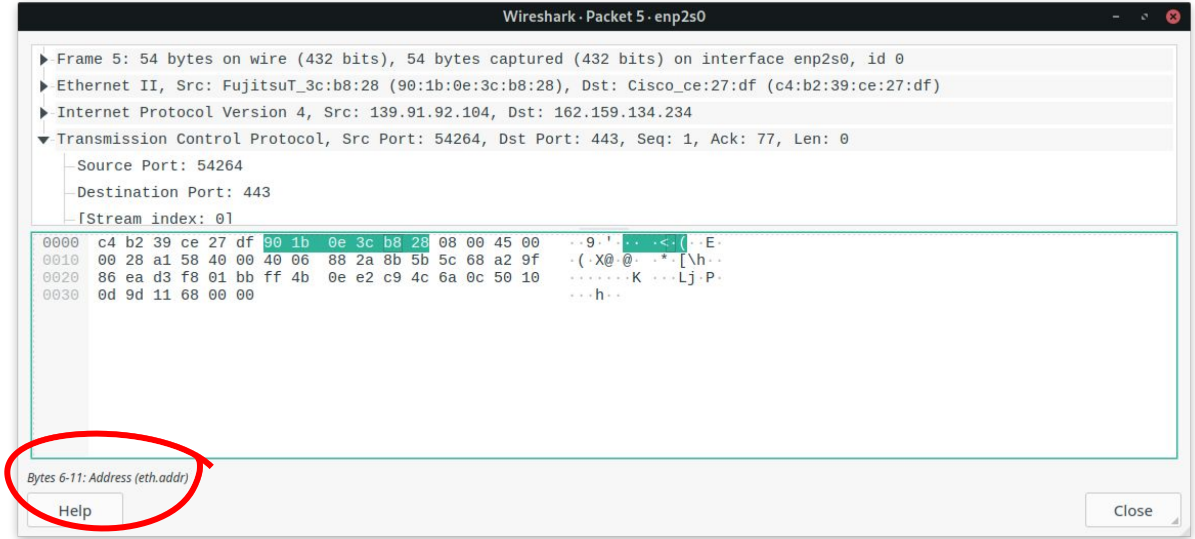
You can also see the hex format of the packet:



```
0000  c4 b2 39 ce 27 df 90 1b  0e 3c b8 28 08 00 45 00   ··9·'···  ·<·(··E·
0010  00 28 a1 58 40 00 40 06  88 2a 8b 5b 5c 68 a2 9f   ·(·X@·@·  ·*·[\h··
0020  86 ea d3 f8 01 bb ff 4b  0e e2 c9 4c 6a 0c 50 10   ·······K  ···Lj·P·
0030  0d 9d 11 68 00 00                                  ···h··
```

Packet details

You can hover over the bytes and see what they represent (see bottom left corner):



```
▶ Frame 5: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface enp2s0, id 0
▶ Ethernet II, Src: FujitsuT_3c:b8:28 (90:1b:0e:3c:b8:28), Dst: Cisco_ce:27:df (c4:b2:39:ce:27:df)
▶ Internet Protocol Version 4, Src: 139.91.92.104, Dst: 162.159.134.234
▼ Transmission Control Protocol, Src Port: 54264, Dst Port: 443, Seq: 1, Ack: 77, Len: 0
    Source Port: 54264
    Destination Port: 443
    [Stream index: 0]
```

```
0000  c4 b2 39 ce 27 df 90 1b  0e 3c b8 28 08 00 45 00   ··9·'···  ·<·(··E·
0010  00 28 a1 58 40 00 40 06  88 2a 8b 5b 5c 68 a2 9f   ·(·X@·@·  ·*·[\h··
0020  86 ea d3 f8 01 bb ff 4b  0e e2 c9 4c 6a 0c 50 10   ·······K  ···Lj·P·
0030  0d 9d 11 68 00 00                                  ···h··
```

Bytes 6-11: Address (eth.addr)

Traceroute & Ping

Traceroute

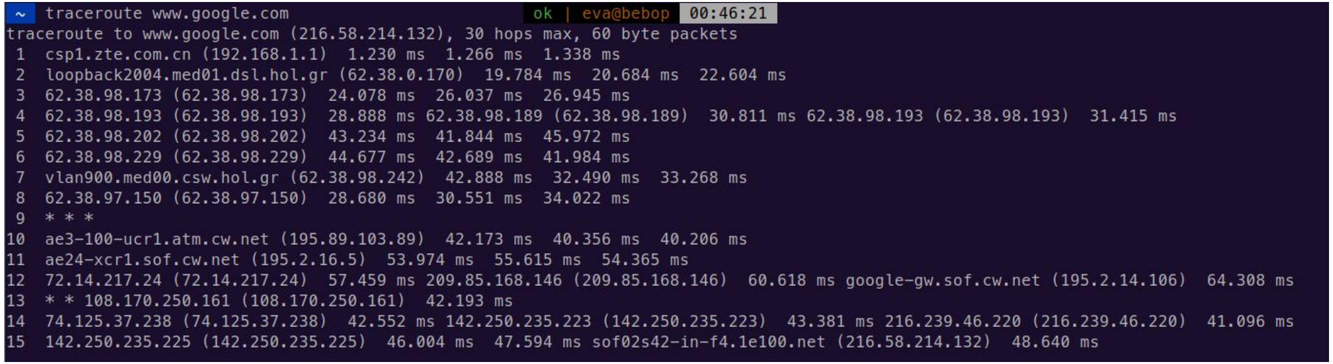
In computing, traceroute (tracert) are computer network diagnostic commands for displaying possible routes (paths) and measuring transit delays of packets across an Internet Protocol (IP) network.

To run traceroute on linux to see the path from your device to www.google.com, run *traceroute* *www.google.com*.

To run traceroute on windows to see the path from your device to www.google.com, run *tracert* *www.google.com*.

Traceroute

From left to right, you can see a) the sequence number of the hop, b) the name/ip address and c), d), e) are the RTTs of the probes we sent to www.google.com



```
~  traceroute www.google.com                                        ok | eva@bebop  00:46:21
traceroute to www.google.com (216.58.214.132), 30 hops max, 60 byte packets
 1  csp1.zte.com.cn (192.168.1.1)  1.230 ms  1.266 ms  1.338 ms
 2  loopback2004.med01.dsl.hol.gr (62.38.0.170)  19.784 ms  20.684 ms  22.604 ms
 3  62.38.98.173 (62.38.98.173)  24.078 ms  26.037 ms  26.945 ms
 4  62.38.98.193 (62.38.98.193)  28.888 ms 62.38.98.189 (62.38.98.189)  30.811 ms 62.38.98.193 (62.38.98.193)  31.415 ms
 5  62.38.98.202 (62.38.98.202)  43.234 ms  41.844 ms  45.972 ms
 6  62.38.98.229 (62.38.98.229)  44.677 ms  42.689 ms  41.984 ms
 7  vlan900.med00.csw.hol.gr (62.38.98.242)  42.888 ms  32.490 ms  33.268 ms
 8  62.38.97.150 (62.38.97.150)  28.680 ms  30.551 ms  34.022 ms
 9  * * *
10  ae3-100-ucr1.atm.cw.net (195.89.103.89)  42.173 ms  40.356 ms  40.206 ms
11  ae24-xcr1.sof.cw.net (195.2.16.5)  53.974 ms  55.615 ms  54.365 ms
12  72.14.217.24 (72.14.217.24)  57.459 ms 209.85.168.146 (209.85.168.146)  60.618 ms google-gw.sof.cw.net (195.2.14.106)  64.308 ms
13  * * 108.170.250.161 (108.170.250.161)  42.193 ms
14  74.125.37.238 (74.125.37.238)  42.552 ms 142.250.235.223 (142.250.235.223)  43.381 ms 216.239.46.220 (216.239.46.220)  41.096 ms
15  142.250.235.225 (142.250.235.225)  46.004 ms  47.594 ms sof02s42-in-f4.1e100.net (216.58.214.132)  48.640 ms
```

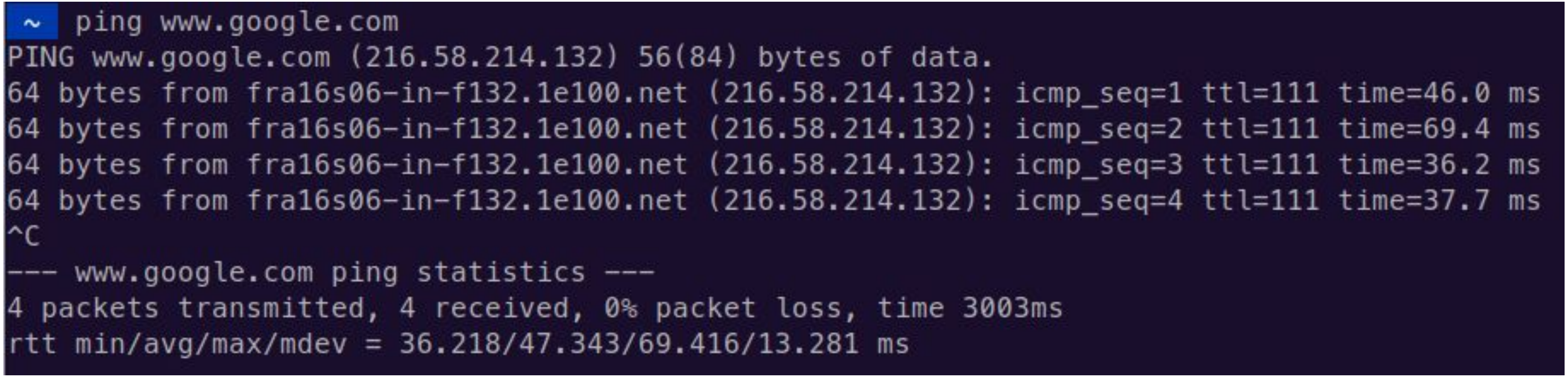
The ‘*’ means the request timed out.
The flag -T means I sent a TCP SYN package as a probe.

Ping

Ping is a computer network administration software utility used to test the reachability of a host on an Internet Protocol network. It is available for virtually all operating systems that have networking capability, including most embedded network administration software.

When you ping a domain from linux, you keep pinging until you stop it with Ctrl+C. When you ping a domain from windows, the default is that it pings for 4 times. If you want to ping for, let's say, 33 times, use *ping -c 33 www.google.com* for linux, and *ping -n 33 www.google.com*

Ping



```
 ~  ping www.google.com
PING www.google.com (216.58.214.132) 56(84) bytes of data.
64 bytes from fra16s06-in-f132.1e100.net (216.58.214.132): icmp_seq=1 ttl=111 time=46.0 ms
64 bytes from fra16s06-in-f132.1e100.net (216.58.214.132): icmp_seq=2 ttl=111 time=69.4 ms
64 bytes from fra16s06-in-f132.1e100.net (216.58.214.132): icmp_seq=3 ttl=111 time=36.2 ms
64 bytes from fra16s06-in-f132.1e100.net (216.58.214.132): icmp_seq=4 ttl=111 time=37.7 ms
^C
--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 36.218/47.343/69.416/13.281 ms
```

You can find additional info on the hy335a webpage

If you have any questions, contact hy335a-list@csd.uoc.gr

Thank you