Wireshark-Part 1

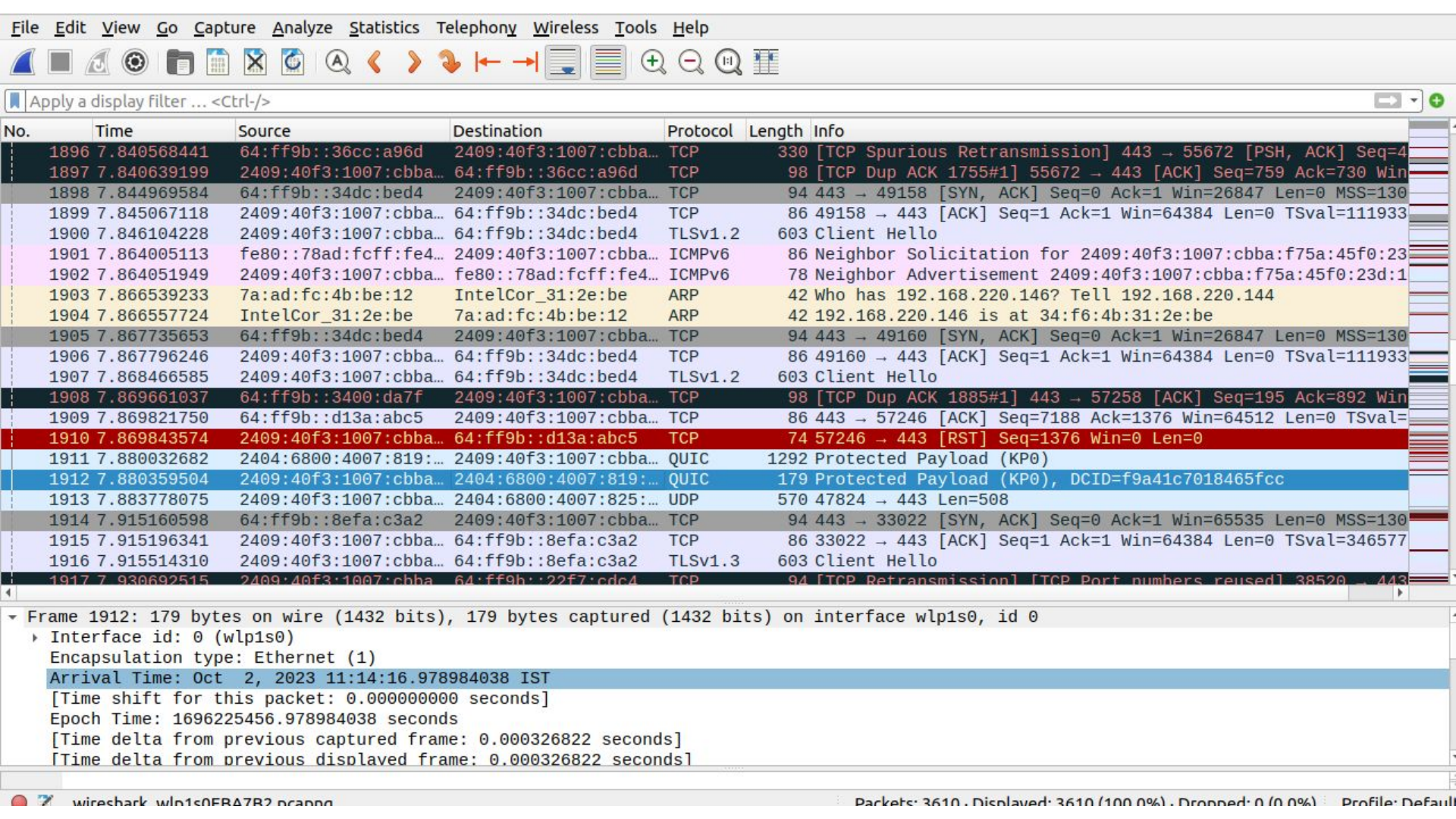
Contents

- Wireshark Introduction
- ARP
- ICMP

Wireshark

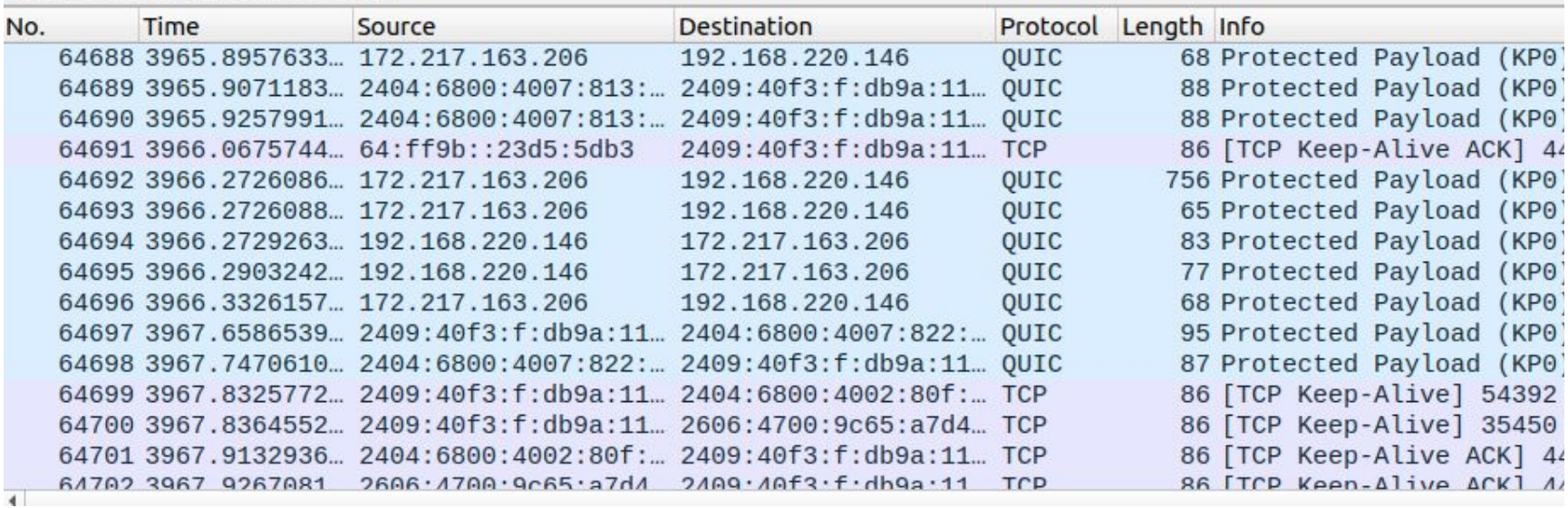
Introduction

- Wireshark is a network protocol analyzer, or an application that captures packets from a network connection, such as from your computer to your home, office or the internet.
- Wireshark does three things:
  - Packet Capture: Wireshark listens to a network connection in real time and then grabs entire streams of traffic – quite possibly tens of thousands of packets at a time.
  - Filtering: Wireshark is capable of slicing and dicing all of this random live data using filters. By applying a filter, you can obtain just the information you need to see.
  - Visualization: Wireshark, like any good packet sniffer, allows you to dive right into the very middle of a network packet. It also allows you to visualize entire conversations and network streams.
- Wireshark has many uses, including troubleshooting networks that have performance issues.
- Cybersecurity professionals often use Wireshark to trace connections, view the contents of suspect network transactions and identify bursts of network traffic.
- Wireshark can be used as a learning tool.
  - Those new to information security can use Wireshark as a tool to understand network traffic analysis, how communication takes place when particular protocols are involved and where it goes wrong when certain issues occur.


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
1896 7.840568441 64:ff9b::36cc:a96d 2409:40f3:1007:cbba... TCP 330 [TCP Spurious Retransmission] 443 → 55672 [PSH, ACK] Seq=4
1897 7.840639199 2409:40f3:1007:cbba... 64:ff9b::36cc:a96d TCP 98 [TCP Dup ACK 1755#1] 55672 → 443 [ACK] Seq=759 Ack=730 Win
1898 7.844969584 64:ff9b::34dc:bed4 2409:40f3:1007:cbba... TCP 94 443 → 49158 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=0 MSS=130
1899 7.845067118 2409:40f3:1007:cbba... 64:ff9b::34dc:bed4 TCP 86 49158 → 443 [ACK] Seq=1 Ack=1 Win=64384 Len=0 TSval=111933
1900 7.846104228 2409:40f3:1007:cbba... 64:ff9b::34dc:bed4 TLSv1.2 603 Client Hello
1901 7.864005113 fe80::78ad:fcff:fe4... 2409:40f3:1007:cbba... ICMPv6 86 Neighbor Solicitation for 2409:40f3:1007:cbba:f75a:45f0:23
1902 7.864051949 2409:40f3:1007:cbba... fe80::78ad:fcff:fe4... ICMPv6 78 Neighbor Advertisement 2409:40f3:1007:cbba:f75a:45f0:23d:1
1903 7.866539233 7a:ad:fc:4b:be:12 IntelCor_31:2e:be ARP 42 Who has 192.168.220.146? Tell 192.168.220.144
1904 7.866557724 IntelCor_31:2e:be 7a:ad:fc:4b:be:12 ARP 42 192.168.220.146 is at 34:f6:4b:31:2e:be
1905 7.867735653 64:ff9b::34dc:bed4 2409:40f3:1007:cbba... TCP 94 443 → 49160 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=0 MSS=130
1906 7.867796246 2409:40f3:1007:cbba... 64:ff9b::34dc:bed4 TCP 86 49160 → 443 [ACK] Seq=1 Ack=1 Win=64384 Len=0 TSval=111933
1907 7.868466585 2409:40f3:1007:cbba... 64:ff9b::34dc:bed4 TLSv1.2 603 Client Hello
1908 7.869661037 64:ff9b::3400:da7f 2409:40f3:1007:cbba... TCP 98 [TCP Dup ACK 1885#1] 443 → 57258 [ACK] Seq=195 Ack=892 Win
1909 7.869821750 64:ff9b::d13a:abc5 2409:40f3:1007:cbba... TCP 86 443 → 57246 [ACK] Seq=7188 Ack=1376 Win=64512 Len=0 TSval=
1910 7.869843574 2409:40f3:1007:cbba... 64:ff9b::d13a:abc5 TCP 74 57246 → 443 [RST] Seq=1376 Win=0 Len=0
1911 7.880032682 2404:6800:4007:819:... 2409:40f3:1007:cbba... QUIC 1292 Protected Payload (KP0)
1912 7.880359504 2409:40f3:1007:cbba... 2404:6800:4007:819:... QUIC 179 Protected Payload (KP0), DCID=f9a41c7018465fcc
1913 7.883778075 2409:40f3:1007:cbba... 2404:6800:4007:825:... UDP 570 47824 → 443 Len=508
1914 7.915160598 64:ff9b::8efa:c3a2 2409:40f3:1007:cbba... TCP 94 443 → 33022 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=130
1915 7.915196341 2409:40f3:1007:cbba... 64:ff9b::8efa:c3a2 TCP 86 33022 → 443 [ACK] Seq=1 Ack=1 Win=64384 Len=0 TSval=346577
1916 7.915514310 2409:40f3:1007:cbba... 64:ff9b::8efa:c3a2 TLSv1.3 603 Client Hello
Frame 1912: 179 bytes on wire (1432 bits), 179 bytes captured (1432 bits) on interface wlp1s0, id 0
Interface id: 0 (wlp1s0)
Encapsulation type: Ethernet (1)
Arrival Time: Oct 2, 2023 11:14:16.978984038 IST
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1696225456.978984038 seconds
[Time delta from previous captured frame: 0.000326822 seconds]
wireshark_wlp1s0FBA7B2.pcapng
Packets: 3610 · Displayed: 3610 (100.0%) · Dropped: 0 (0.0%)
Profile: Default

Packet Diagram Pane Functions in Wireshark

- **Traffic Pane:**
  - The traffic pane just shows the traffic which is on your network .



No.	Time	Source	Destination	Protocol	Length	Info
64688	3965.8957633…	172.217.163.206	192.168.220.146	QUIC	68	Protected Payload (KP0)
64689	3965.9071183…	2404:6800:4007:813:…	2409:40f3:f:db9a:11…	QUIC	88	Protected Payload (KP0)
64690	3965.9257991…	2404:6800:4007:813:…	2409:40f3:f:db9a:11…	QUIC	88	Protected Payload (KP0)
64691	3966.0675744…	64:ff9b::23d5:5db3	2409:40f3:f:db9a:11…	TCP	86	[TCP Keep-Alive ACK] 44
64692	3966.2726086…	172.217.163.206	192.168.220.146	QUIC	756	Protected Payload (KP0)
64693	3966.2726088…	172.217.163.206	192.168.220.146	QUIC	65	Protected Payload (KP0)
64694	3966.2729263…	192.168.220.146	172.217.163.206	QUIC	83	Protected Payload (KP0)
64695	3966.2903242…	192.168.220.146	172.217.163.206	QUIC	77	Protected Payload (KP0)
64696	3966.3326157…	172.217.163.206	192.168.220.146	QUIC	68	Protected Payload (KP0)
64697	3967.6586539…	2409:40f3:f:db9a:11…	2404:6800:4007:822:…	QUIC	95	Protected Payload (KP0)
64698	3967.7470610…	2404:6800:4007:822:…	2409:40f3:f:db9a:11…	QUIC	87	Protected Payload (KP0)
64699	3967.8325772…	2409:40f3:f:db9a:11…	2404:6800:4002:80f:…	TCP	86	[TCP Keep-Alive] 54392
64700	3967.8364552…	2409:40f3:f:db9a:11…	2606:4700:9c65:a7d4…	TCP	86	[TCP Keep-Alive] 35450
64701	3967.9132936…	2404:6800:4002:80f:…	2409:40f3:f:db9a:11…	TCP	86	[TCP Keep-Alive ACK] 44
64702	3967.9267081…	2606:4700:9c65:a7d4…	2409:40f3:f:db9a:11…	TCP	86	[TCP Keep-Alive ACK] 44

- **Packet details** panel
    - Packet details panel is such amazing panel where you can also learn about the protocol fields that are contained by each of them, and you can understand it much clearer.

```
Frame 408: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface wlp1s0, id 0
Ethernet II, Src: 3e:9e:a1:04:bd:54 (3e:9e:a1:04:bd:54), Dst: IntelCor_31:2e:be (34:f6:4b:31:2e:be)
Internet Protocol Version 6, Src: fe80::3c9e:a1ff:fe04:bd54, Dst: 2409:40f3:f:db9a:19a5:f412:61b0:a65
Internet Control Message Protocol v6
```

- **Packet Bytes:**
    - The Packet bytes have its own significance which can help in analyzing the data which is based on the Bytes level.

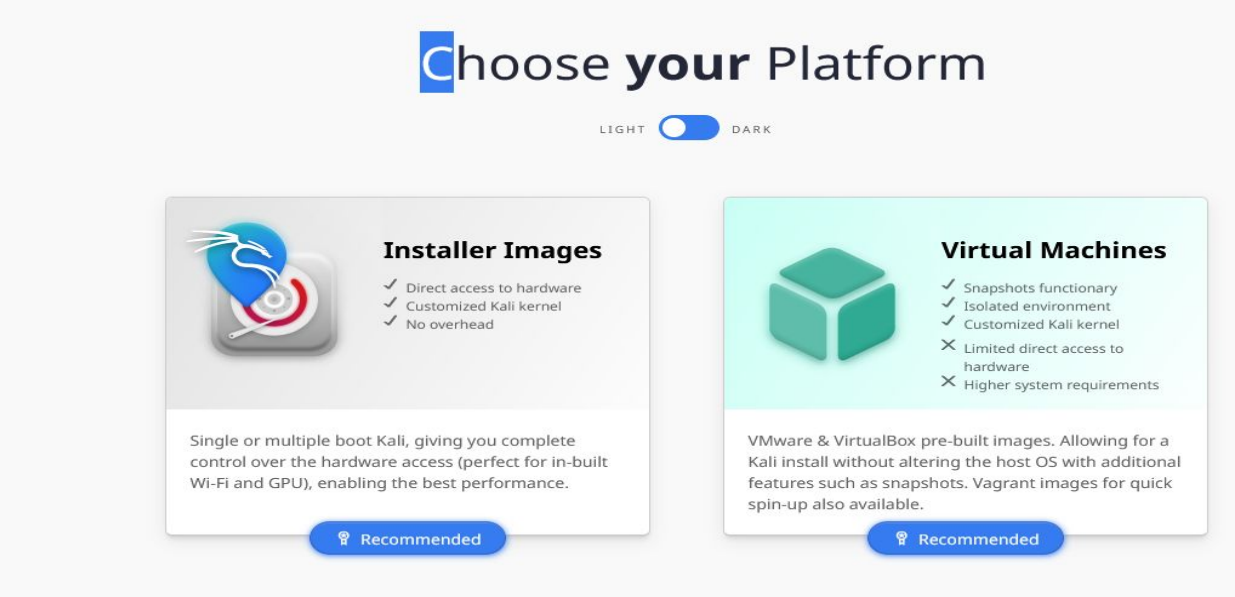
```
0000  34 f6 4b 31 2e be 3e 9e  a1 04 bd 54 86 dd 60 00   4·K1.·>· ···T··`·
0010  00 00 00 20 3a ff fe 80  00 00 00 00 00 00 3c 9e   ··· :··· ······<·
0020  a1 ff fe 04 bd 54 24 09  40 f3 00 0f db 9a 19 a5   ·····T$· @·······
0030  f4 12 61 b0 a6 56 87 00  95 69 00 00 00 00 24 09   ··a··V·· ·i····$·
0040  40 f3 00 0f db 9a 19 a5  f4 12 61 b0 a6 56 01 01   @······ ··a··V··
0050  3e 9e a1 04 bd 54                                  >····T
```

Installation

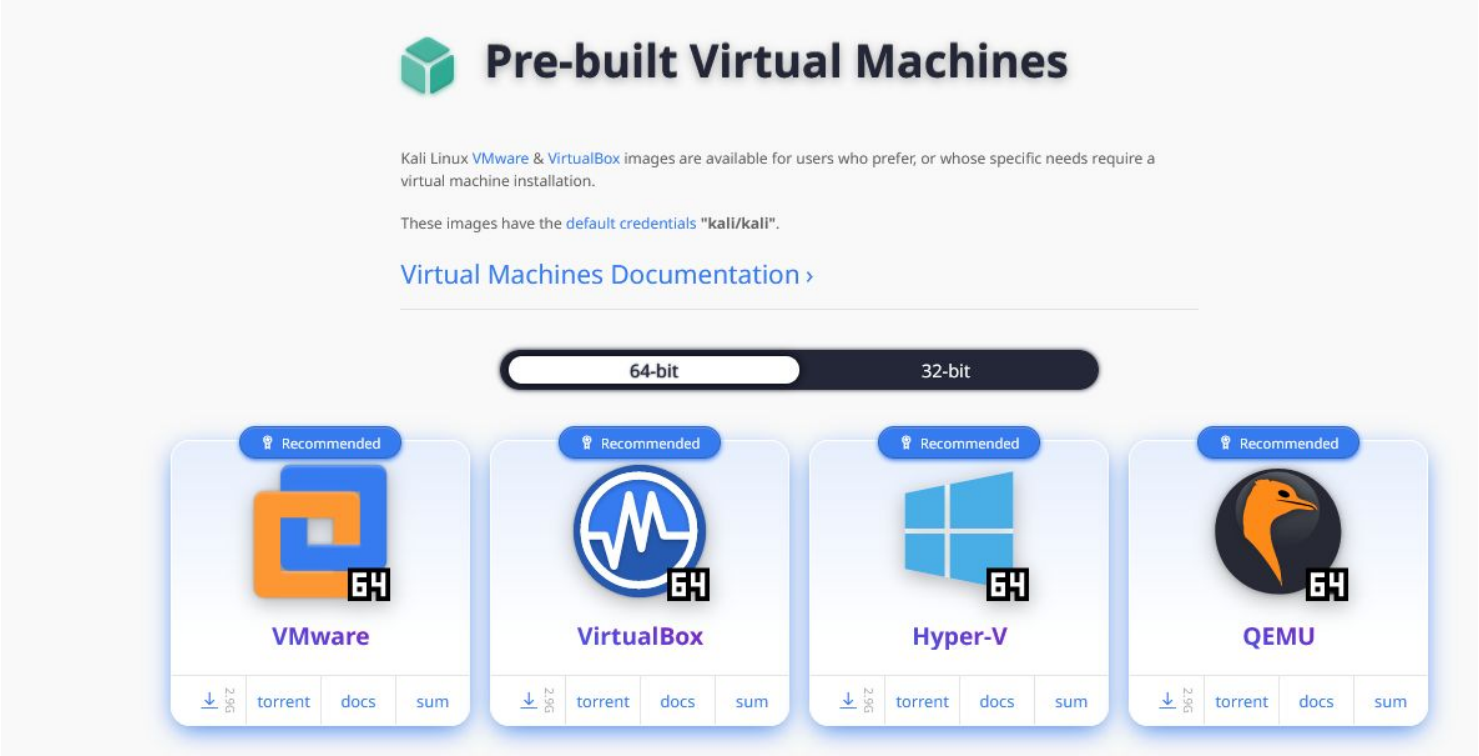
- **How to Install Wireshark on Windows**
  - If you're a Windows operating system user, download the version appropriate for your particular version. Follow the wizard to install.
- **How to Install Wireshark on Linux**
  - sudo apt-get install wireshark
  - sudo dpkg-reconfigure wireshark-common
  - sudo usermod -a -G wireshark $USER
  - Once you have completed the above steps, you then log out and log back in, and then start Wireshark:
  - wireshark &

Installing kali linux

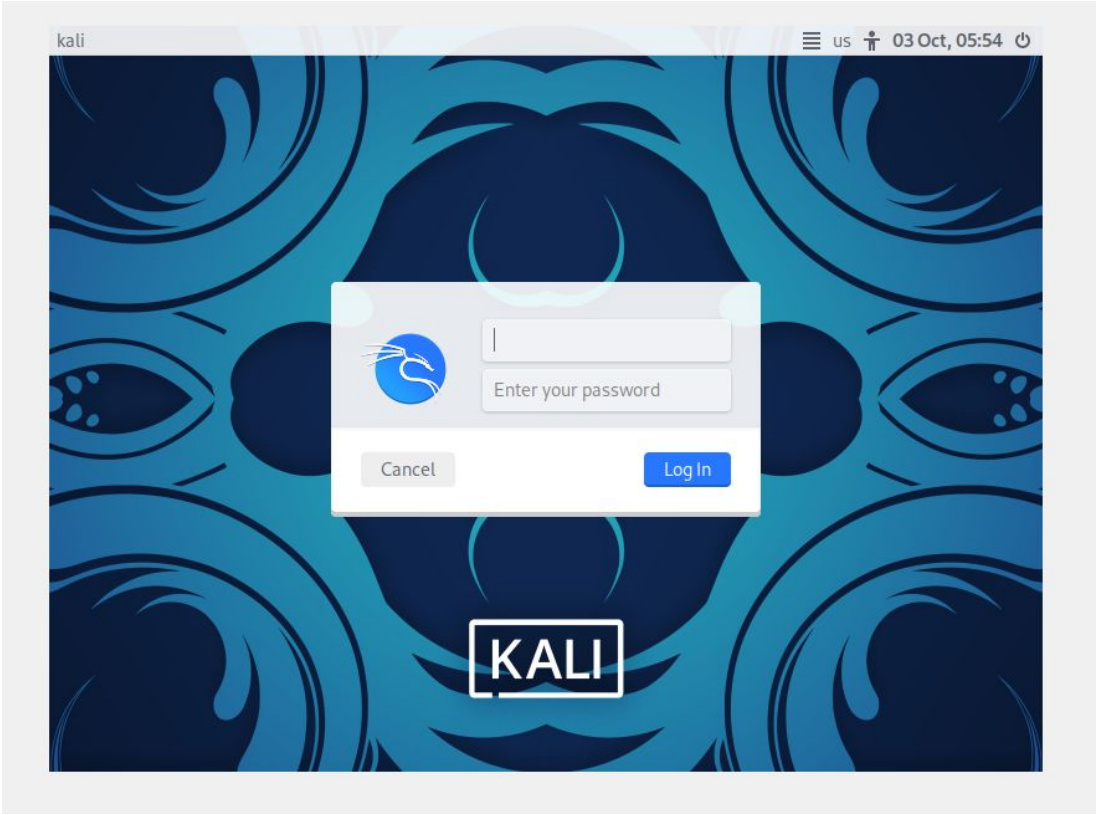
- https://www.kali.org/get-kali/#kali-platforms
-



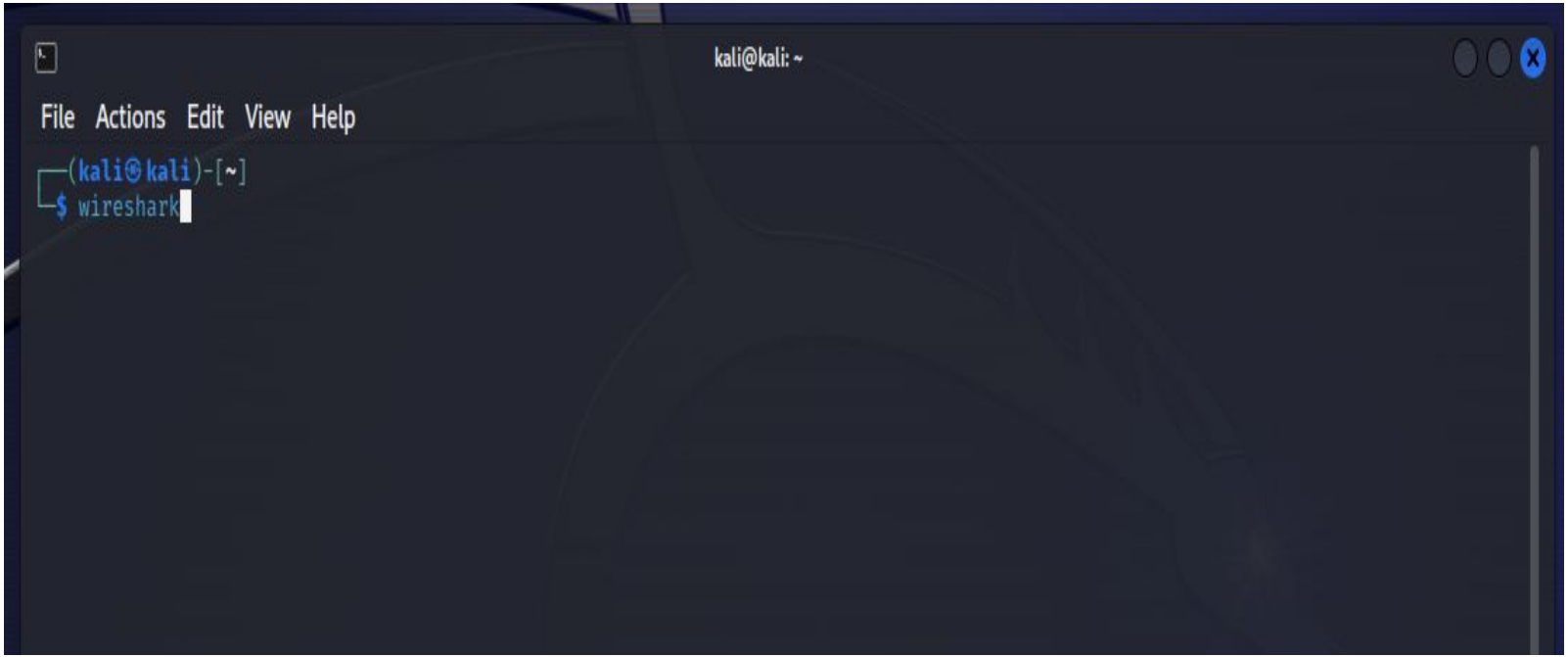
- Virtual Machines

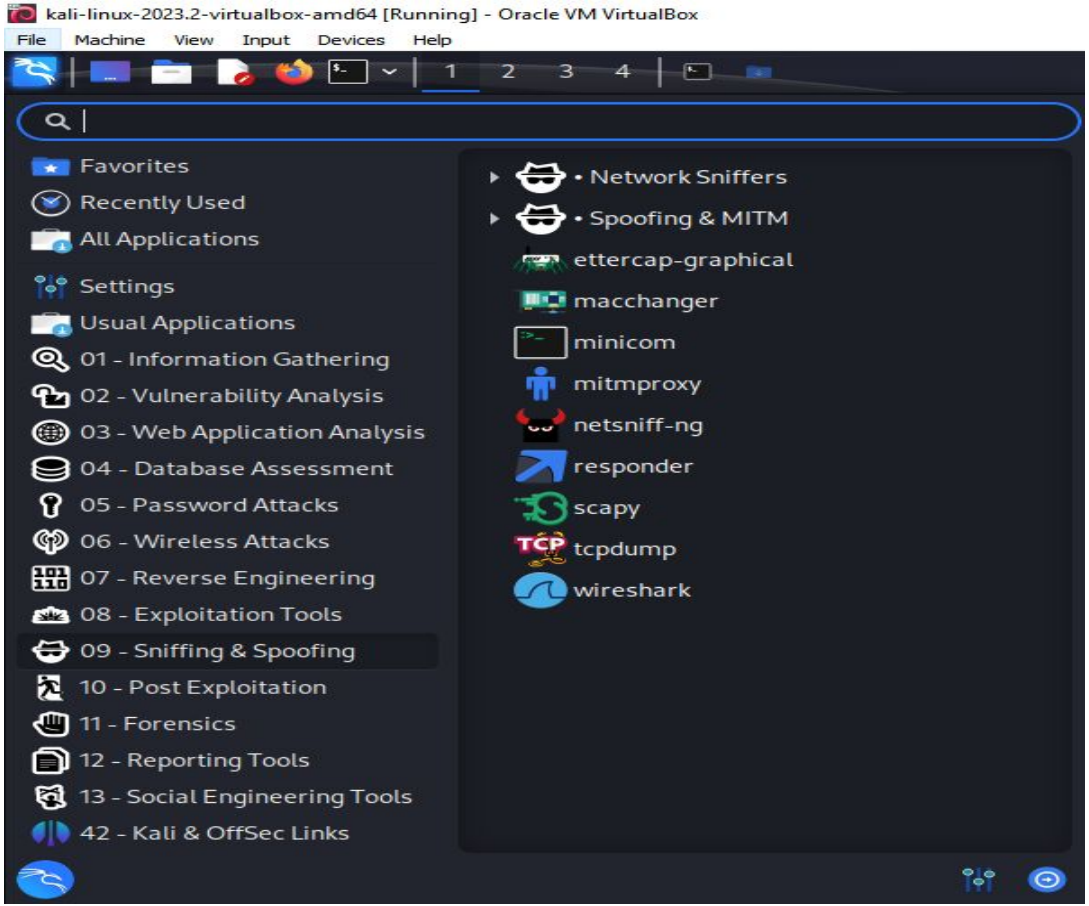


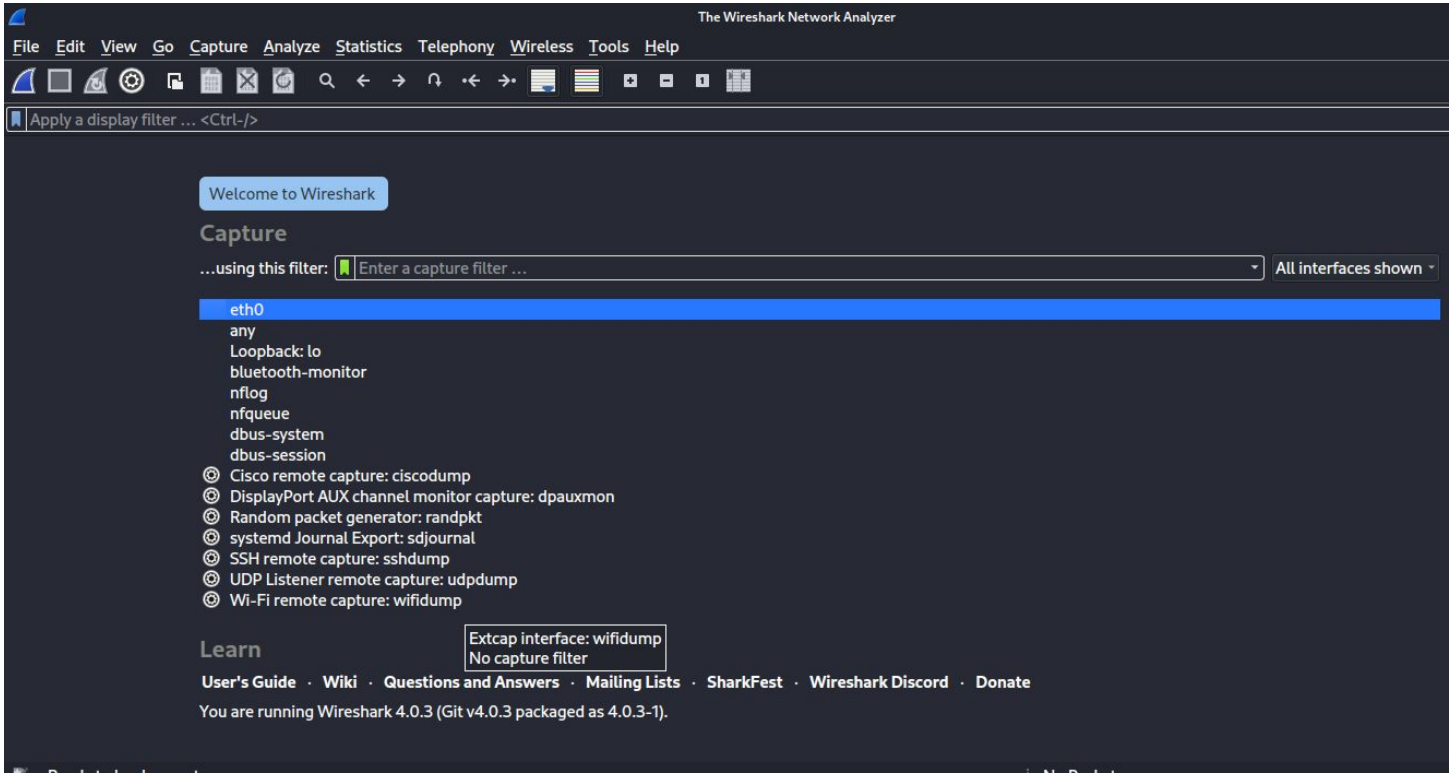
- Add vm
  - Machine —> Add

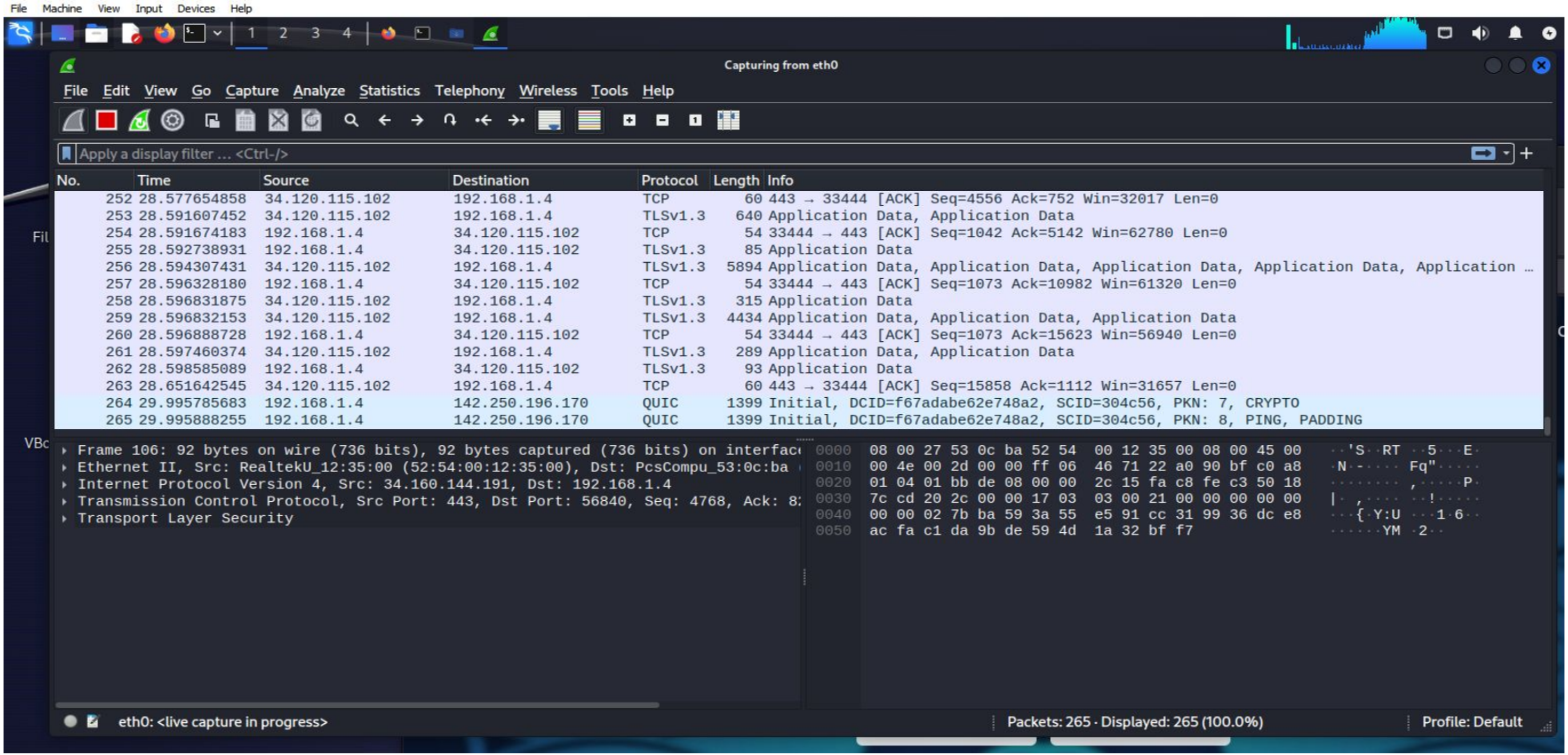


Working with wireshark



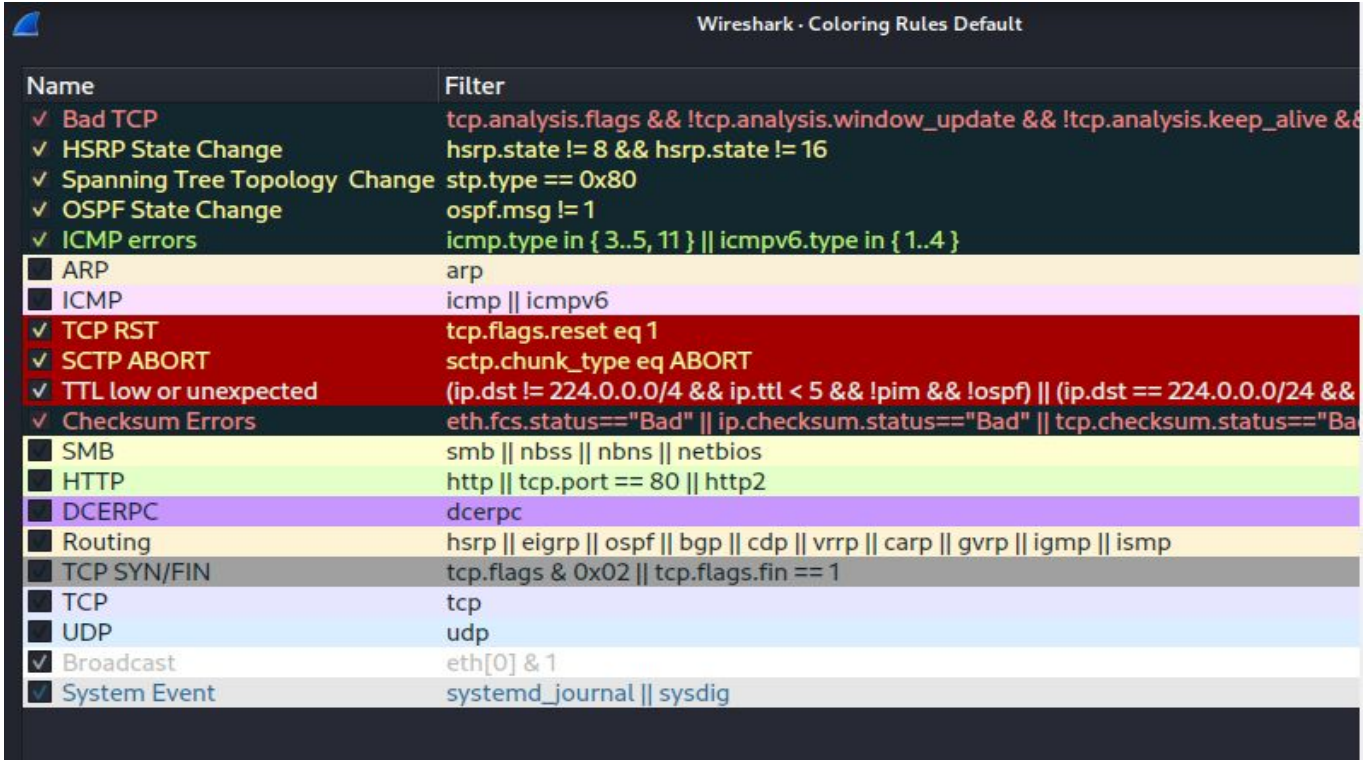

kali-linux-2023.2-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
1
2
3
4
Favorites
Recently Used
All Applications
Settings
Usual Applications
01 - Information Gathering
02 - Vulnerability Analysis
03 - Web Application Analysis
04 - Database Assessment
05 - Password Attacks
06 - Wireless Attacks
07 - Reverse Engineering
08 - Exploitation Tools
09 - Sniffing & Spoofing
10 - Post Exploitation
11 - Forensics
12 - Reporting Tools
13 - Social Engineering Tools
42 - Kali & OffSec Links
• Network Sniffers
• Spoofing & MITM
ettercap-graphical
macchanger
minicom
mitmproxy
netsniff-ng
responder
scapy
tcpdump
wireshark


The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
Welcome to Wireshark
Capture
…using this filter: Enter a capture filter …
All interfaces shown
eth0
any
Loopback: lo
bluetooth-monitor
nflog
nfqueue
dbus-system
dbus-session
Cisco remote capture: ciscodump
DisplayPort AUX channel monitor capture: dpauxmon
Random packet generator: randpkt
systemd Journal Export: sdjournal
SSH remote capture: sshdump
UDP Listener remote capture: udpdump
Wi-Fi remote capture: wifidump
Extcap interface: wifidump
No capture filter
Learn
User's Guide · Wiki · Questions and Answers · Mailing Lists · SharkFest · Wireshark Discord · Donate
You are running Wireshark 4.0.3 (Git v4.0.3 packaged as 4.0.3-1).


File Machine View Input Devices Help
1 2 3 4
Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
No. Time Source Destination Protocol Length Info
252 28.577654858 34.120.115.102 192.168.1.4 TCP 60 443 → 33444 [ACK] Seq=4556 Ack=752 Win=32017 Len=0
253 28.591607452 34.120.115.102 192.168.1.4 TLSv1.3 640 Application Data, Application Data
254 28.591674183 192.168.1.4 34.120.115.102 TCP 54 33444 → 443 [ACK] Seq=1042 Ack=5142 Win=62780 Len=0
255 28.592738931 192.168.1.4 34.120.115.102 TLSv1.3 85 Application Data
256 28.594307431 34.120.115.102 192.168.1.4 TLSv1.3 5894 Application Data, Application Data, Application Data, Application Data, Application …
257 28.596328180 192.168.1.4 34.120.115.102 TCP 54 33444 → 443 [ACK] Seq=1073 Ack=10982 Win=61320 Len=0
258 28.596831875 34.120.115.102 192.168.1.4 TLSv1.3 315 Application Data
259 28.596832153 34.120.115.102 192.168.1.4 TLSv1.3 4434 Application Data, Application Data, Application Data
260 28.596888728 192.168.1.4 34.120.115.102 TCP 54 33444 → 443 [ACK] Seq=1073 Ack=15623 Win=56940 Len=0
261 28.597460374 34.120.115.102 192.168.1.4 TLSv1.3 289 Application Data, Application Data
262 28.598585089 192.168.1.4 34.120.115.102 TLSv1.3 93 Application Data
263 28.651642545 34.120.115.102 192.168.1.4 TCP 60 443 → 33444 [ACK] Seq=15858 Ack=1112 Win=31657 Len=0
264 29.995785683 192.168.1.4 142.250.196.170 QUIC 1399 Initial, DCID=f67adabe62e748a2, SCID=304c56, PKN: 7, CRYPTO
265 29.995888255 192.168.1.4 142.250.196.170 QUIC 1399 Initial, DCID=f67adabe62e748a2, SCID=304c56, PKN: 8, PING, PADDING
Frame 106: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interfac
Ethernet II, Src: RealtekU_12:35:00 (52:54:00:12:35:00), Dst: PcsCompu_53:0c:ba
Internet Protocol Version 4, Src: 34.160.144.191, Dst: 192.168.1.4
Transmission Control Protocol, Src Port: 443, Dst Port: 56840, Seq: 4768, Ack: 8
Transport Layer Security
0000 08 00 27 53 0c ba 52 54 00 12 35 00 08 00 45 00
0010 00 4e 00 2d 00 00 ff 06 46 71 22 a0 90 bf c0 a8
0020 01 04 01 bb de 08 00 00 2c 15 fa c8 fe c3 50 18
0030 7c cd 20 2c 00 00 17 03 03 00 21 00 00 00 00 00
0040 00 00 02 7b ba 59 3a 55 e5 91 cc 31 99 36 dc e8
0050 ac fa c1 da 9b de 59 4d 1a 32 bf f7
eth0: <live capture in progress>
Packets: 265 · Displayed: 265 (100.0%)
Profile: Default

What the Color Coding Means in Wireshark

- The default coloring scheme – You can view this by going to View >> Coloring Rules

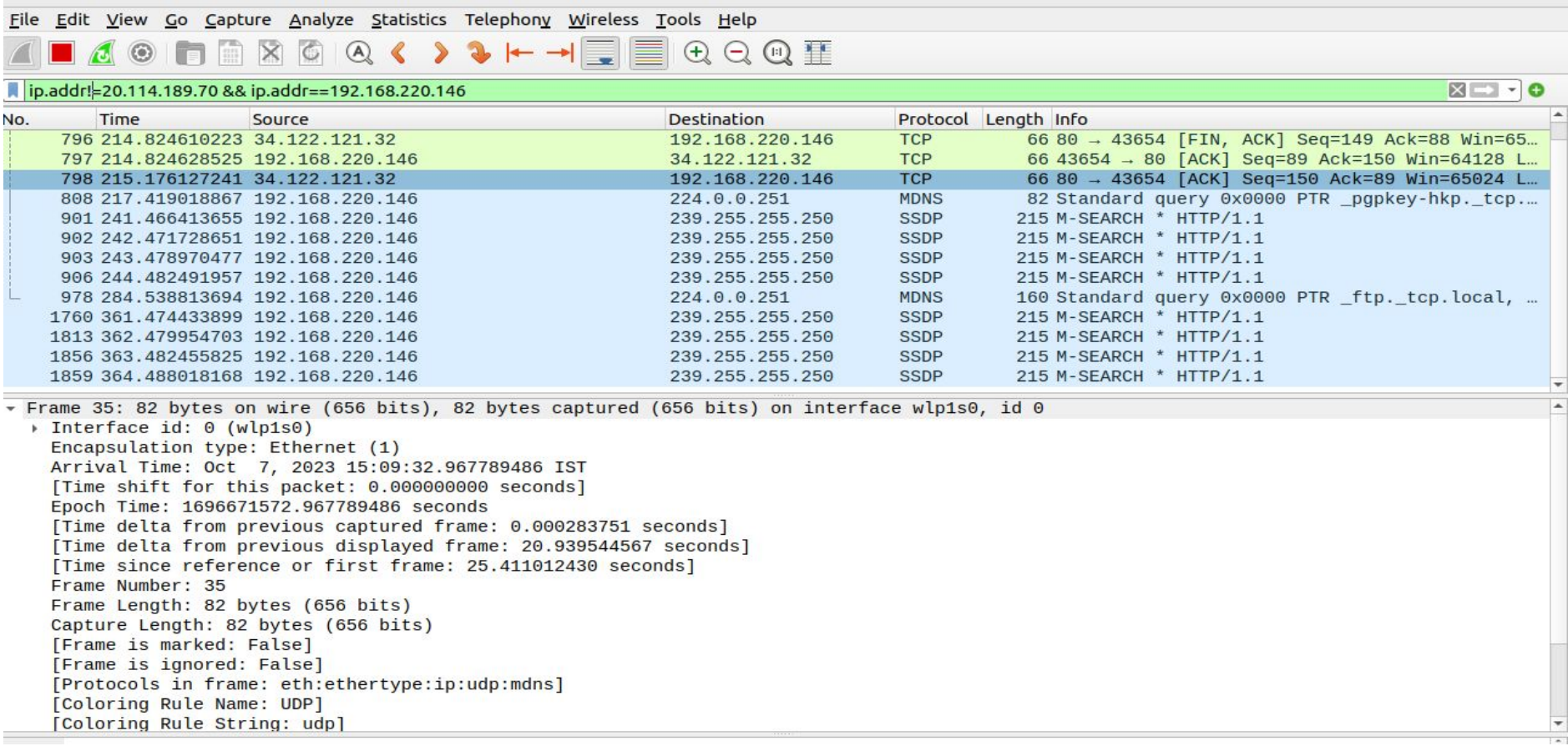


How to Filter and Inspect Packets in Wireshark

Valid filter rules are always colored green. If you make a mistake on a filter rule, the box will turn a vivid pink.

`ip.addr`	Specifies an IPv4 address
`ipv6.addr`	Specifies an IPv6 address
`src`	Source - where the packet came from
`dst`	Destination - where the packet is going

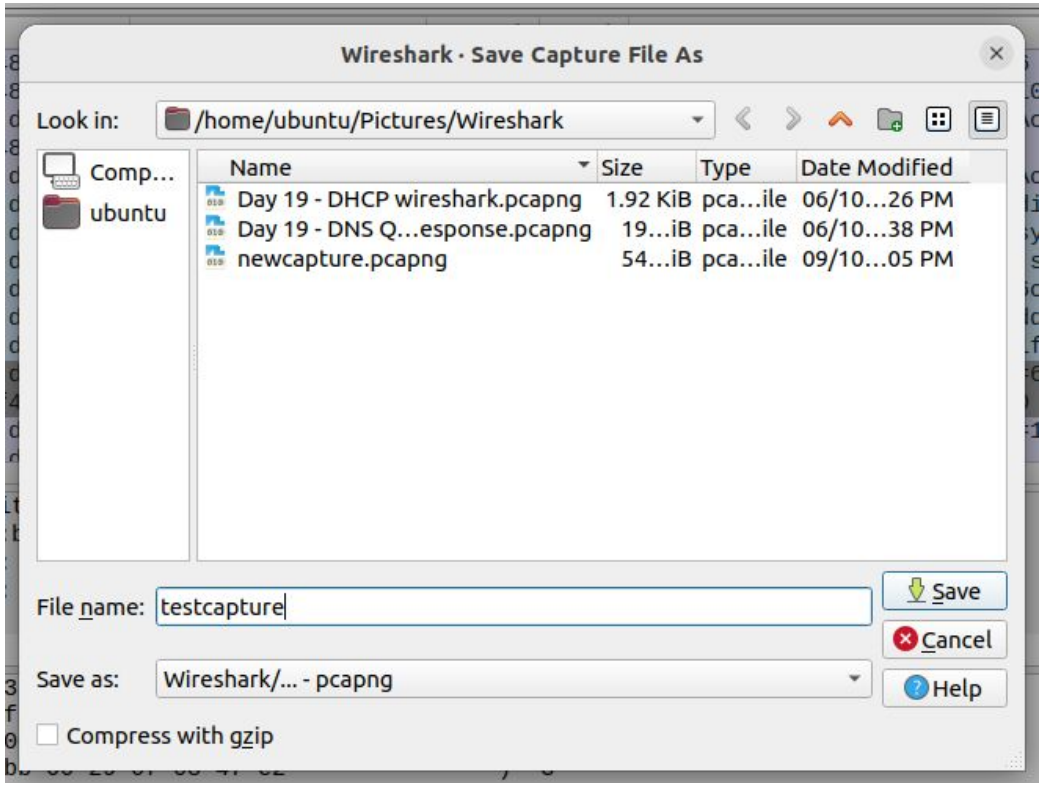
`&&`	Means “and,” as in, “Choose the IP address of 192.168.2.1 and 192.168.2.2”
`==`	Means “equals,” as in “Choose only IP address 192.168.2.1”
`!`	Means “not,” as in, do not show a particular IP address or source port


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ip.addr!=20.114.189.70 && ip.addr==192.168.220.146
No. Time Source Destination Protocol Length Info
796 214.824610223 34.122.121.32 192.168.220.146 TCP 66 80 → 43654 [FIN, ACK] Seq=149 Ack=88 Win=65…
797 214.824628525 192.168.220.146 34.122.121.32 TCP 66 43654 → 80 [ACK] Seq=89 Ack=150 Win=64128 L…
798 215.176127241 34.122.121.32 192.168.220.146 TCP 66 80 → 43654 [ACK] Seq=150 Ack=89 Win=65024 L…
808 217.419018867 192.168.220.146 224.0.0.251 MDNS 82 Standard query 0x0000 PTR _pgpkey-hkp._tcp.…
901 241.466413655 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
902 242.471728651 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
903 243.478970477 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
906 244.482491957 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
978 284.538813694 192.168.220.146 224.0.0.251 MDNS 160 Standard query 0x0000 PTR _ftp._tcp.local, …
1760 361.474433899 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
1813 362.479954703 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
1856 363.482455825 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
1859 364.488018168 192.168.220.146 239.255.255.250 SSDP 215 M-SEARCH * HTTP/1.1
Frame 35: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface wlp1s0, id 0
Interface id: 0 (wlp1s0)
Encapsulation type: Ethernet (1)
Arrival Time: Oct 7, 2023 15:09:32.967789486 IST
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1696671572.967789486 seconds
[Time delta from previous captured frame: 0.000283751 seconds]
[Time delta from previous displayed frame: 20.939544567 seconds]
[Time since reference or first frame: 25.411012430 seconds]
Frame Number: 35
Frame Length: 82 bytes (656 bits)
Capture Length: 82 bytes (656 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: eth:ethertype:ip:udp:mdns]
[Coloring Rule Name: UDP]
[Coloring Rule String: udp]

`tcp.port==8080`	Filters packets to show a port of your own choosing – in this case, port 8080
`!(ip.src == 162.248.16.53)`	Shows all packets except those originating from 162.248.16.53
`!(ipv6.dst == 2607:f8b0:400a:15::b)`	Shows all packets except those going to the IPv6 address of 2607:f8b0:400a:15::b
`ip.addr == 192.168.4.1 && ip.addr == 192.168.4.2`	Shows both 192.168.4.1 and 192.168.4.2
`http.request`	Shows only http requests – useful when troubleshooting or visualizing web traffic

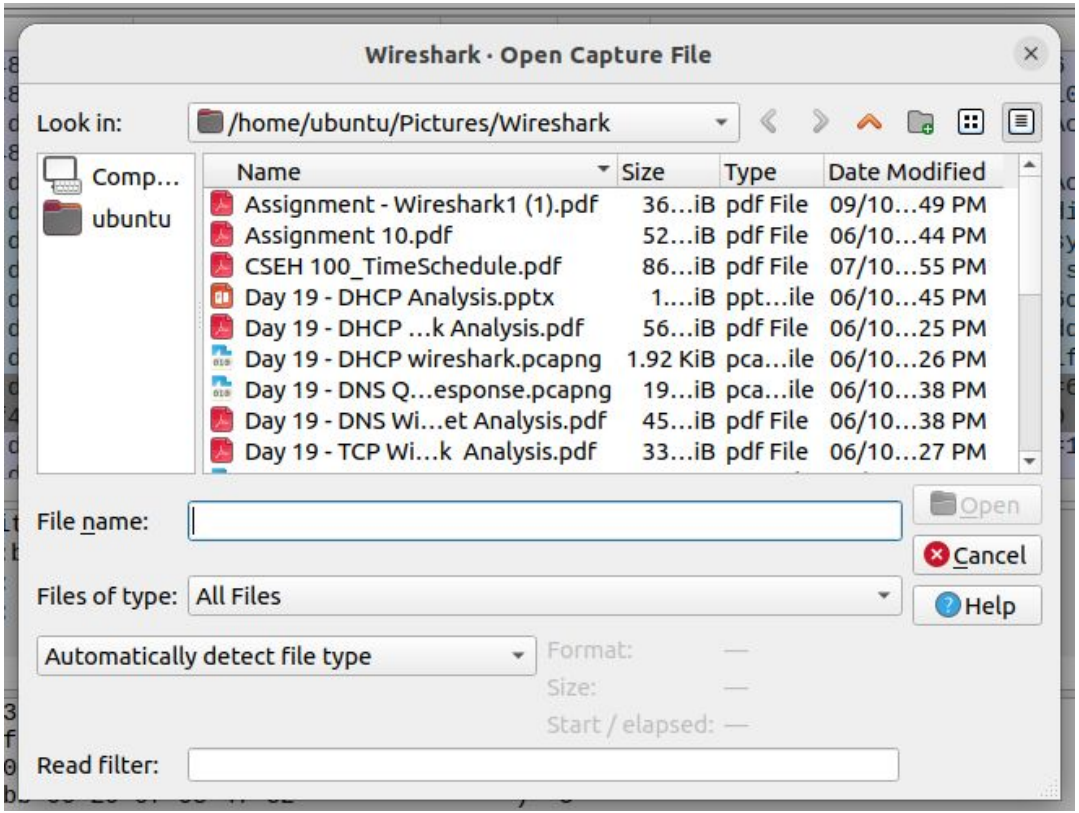
Saving a Capture

- File -> save

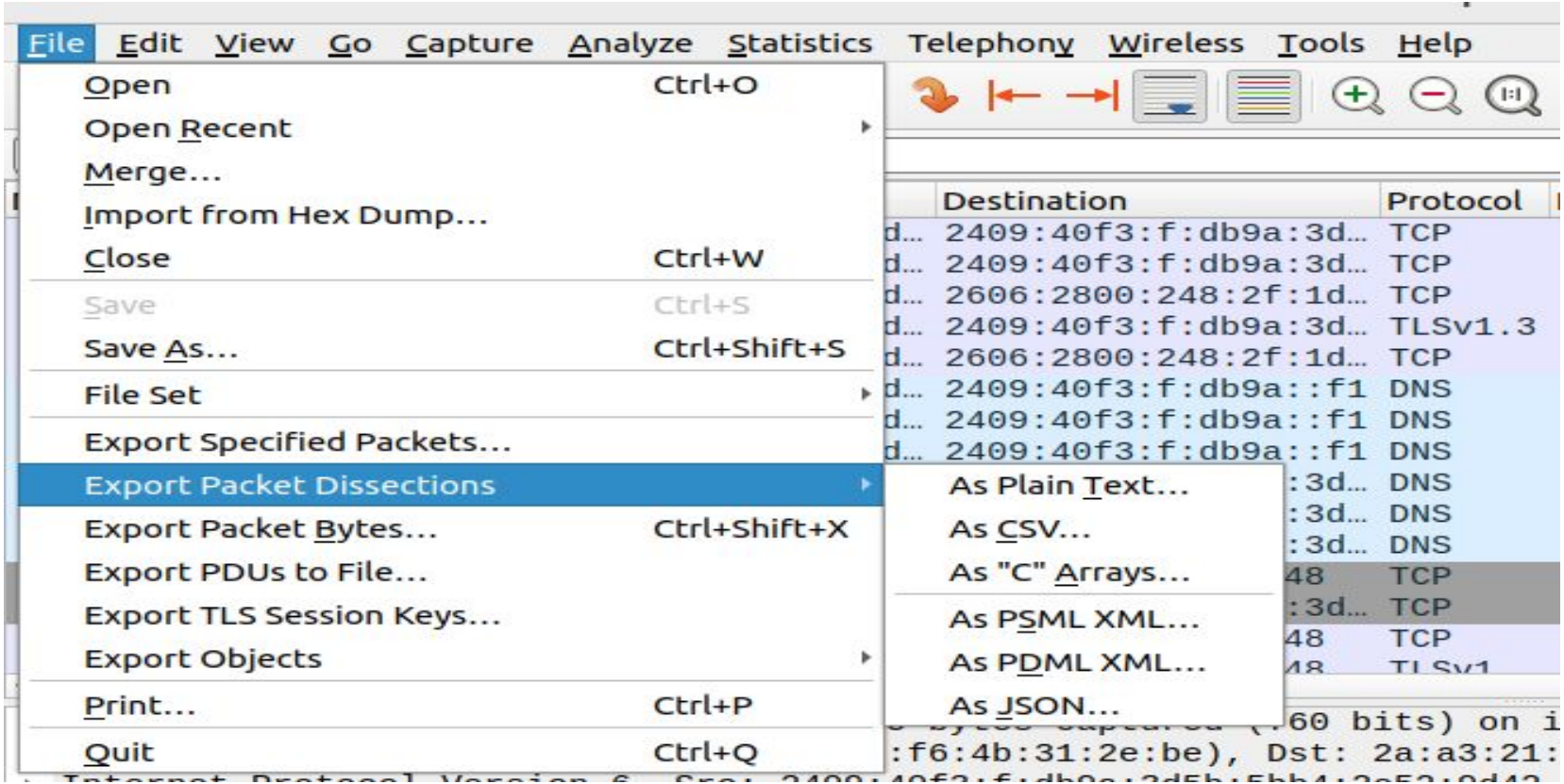


Opening a Capture

- File -> Open



Export as csv



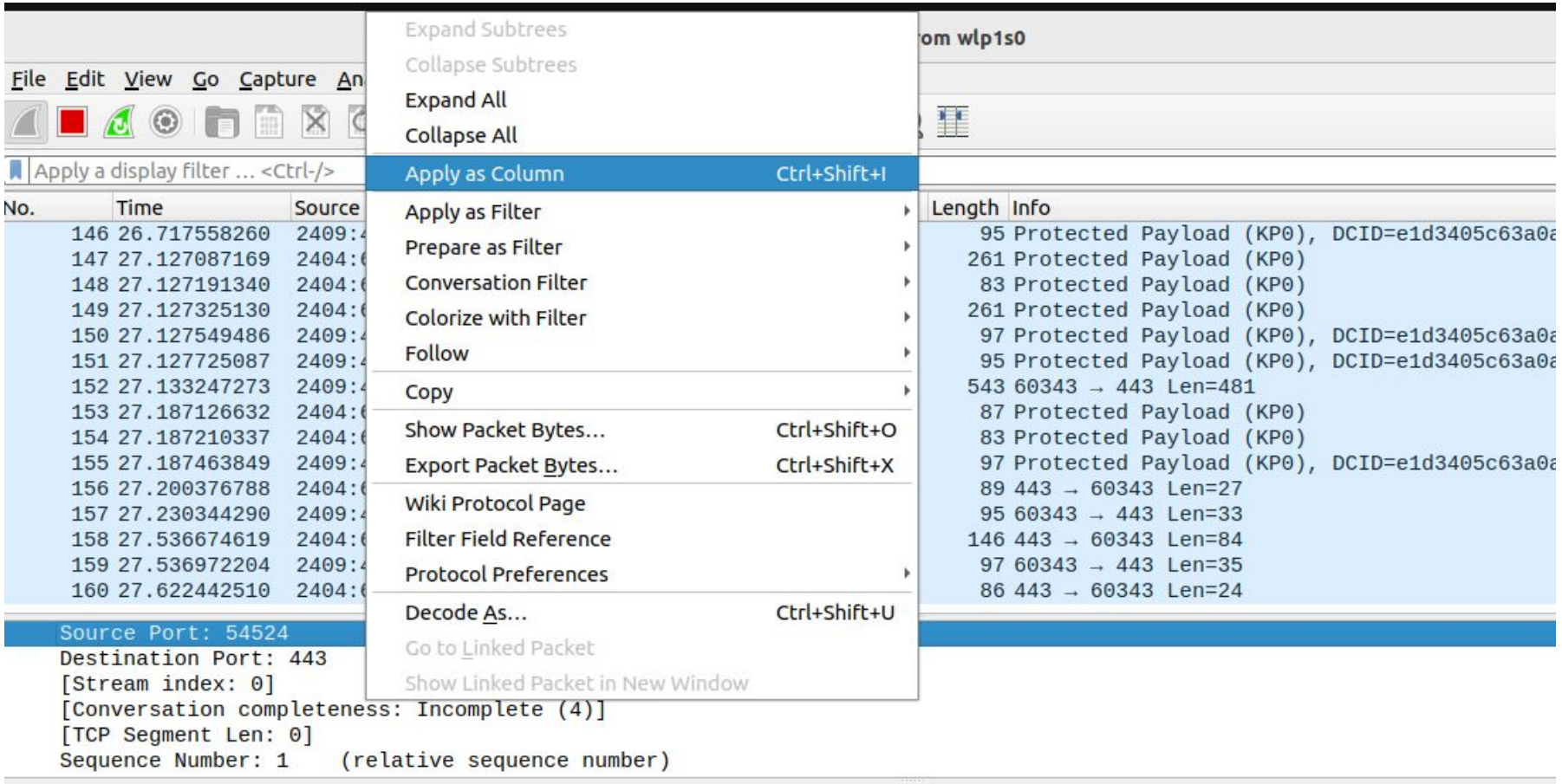
File Edit View Insert Format Styles Sheet Data Tools Window Help

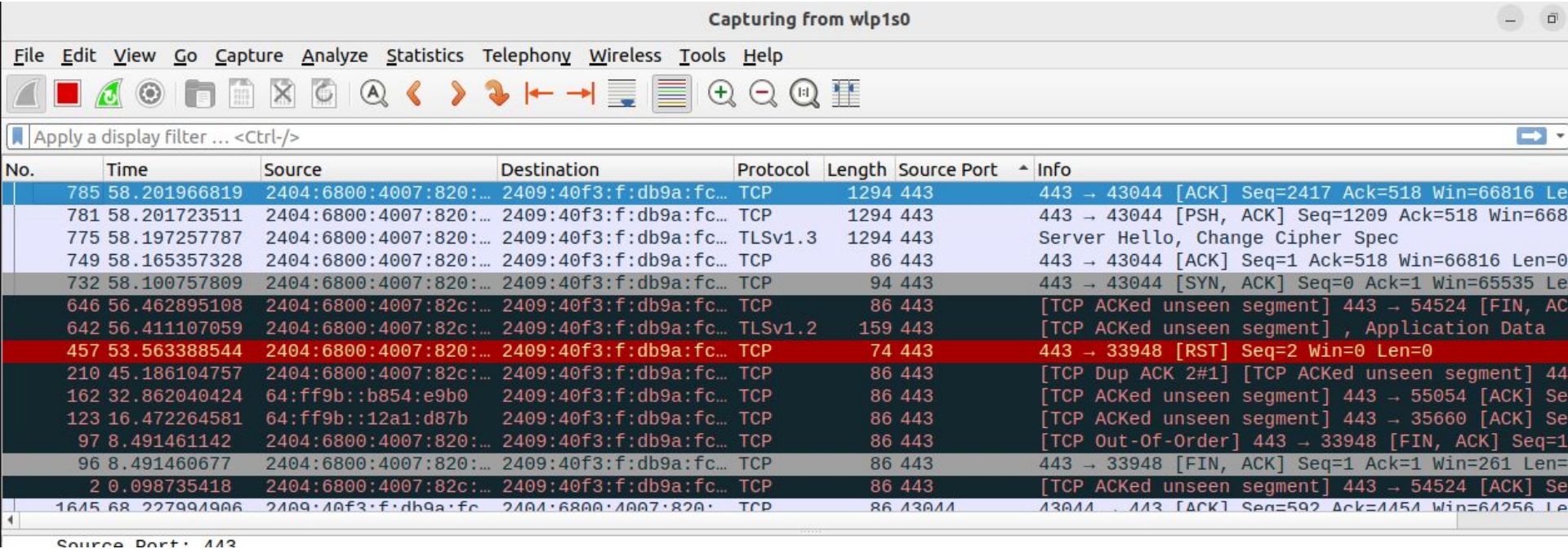
Liberation Sans 10 pt

A1 No.

	A	B	C	D	E	F	
1	No.	Time	Source	Destination	Protocol	Length	Info
2	1	0	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:821::200a	UDP	95	35599 > 443 Len=33
3	2	0.094493804	2404:6800:4007:821::200a	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	UDP	87	443 > 35599 Len=25
4	3	3.173973229	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	86	33892 > 443 [FIN, ACK] Seq=1 Ack=1 Win=502 Len=0 TSval=3297885598 TSec
5	4	3.174144271	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	94	55684 > 443 [SYN] Seq=0 Win=64320 Len=0 MSS=1340 SACK_PERM=1 TSval
6	5	3.258383495	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TCP	86	443 > 33892 [ACK] Seq=1 Ack=2 Win=261 Len=0 TSval=1717573156 TSecr=32
7	6	3.266708959	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TCP	94	443 > 55684 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1300 SACK_PE
8	7	3.266754638	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	86	55684 > 443 [ACK] Seq=1 Ack=1 Win=64384 Len=0 TSval=3297885691 TSecr=
9	8	3.267106912	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TLSv1.3	603	Client Hello
10	9	3.353378225	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TCP	86	443 > 55684 [ACK] Seq=1 Ack=518 Win=66816 Len=0 TSval=853684478 TSecr=
11	10	3.385595436	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TLSv1.3	1294	Server Hello, Change Cipher Spec
12	11	3.385638243	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	86	55684 > 443 [ACK] Seq=518 Ack=1209 Win=63232 Len=0 TSval=3297885810 TS
13	12	3.388944161	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TCP	1294	443 > 55684 [PSH, ACK] Seq=1209 Ack=518 Win=66816 Len=1208 TSval=8536
14	13	3.388961621	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	86	55684 > 443 [ACK] Seq=518 Ack=2417 Win=63104 Len=0 TSval=3297885813 TS
15	14	3.389179838	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TCP	1294	443 > 55684 [ACK] Seq=2417 Ack=518 Win=66816 Len=1208 TSval=853684512
16	15	3.389184607	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	86	55684 > 443 [ACK] Seq=518 Ack=3625 Win=63104 Len=0 TSval=3297885813 TS
17	16	3.389396301	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TLSv1.3	913	Application Data
18	17	3.389412035	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	86	55684 > 443 [ACK] Seq=518 Ack=4452 Win=63104 Len=0 TSval=3297885814 TS
19	18	3.390051536	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TLSv1.3	160	Change Cipher Spec, Application Data
20	19	3.390264337	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TLSv1.3	184	Application Data
21	20	3.39066977	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	1374	55684 > 443 [ACK] Seq=690 Ack=4452 Win=64256 Len=1288 TSval=329788581
22	21	3.390691484	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TLSv1.3	360	Application Data
23	22	3.405193461	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TLSv1.3	159	Application Data
24	23	3.455484217	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TCP	86	443 > 55684 [ACK] Seq=4452 Ack=592 Win=66816 Len=0 TSval=853684583 TS
25	24	3.455484428	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TCP	86	443 > 55684 [ACK] Seq=4452 Ack=690 Win=66816 Len=0 TSval=853684583 TS
26	25	3.455801654	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TLSv1.3	1042	Application Data, Application Data
27	26	3.455801749	2404:6800:4007:82a::2004	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	TLSv1.3	117	Application Data
28	27	3.455901032	2409:40f3:f:db9a:3d5b:5bb4:2e52:9d42	2404:6800:4007:82a::2004	TCP	86	55684 > 443 [ACK] Seq=2325 Ack=5439 Win=64256 Len=0 TSval=3297885880 T

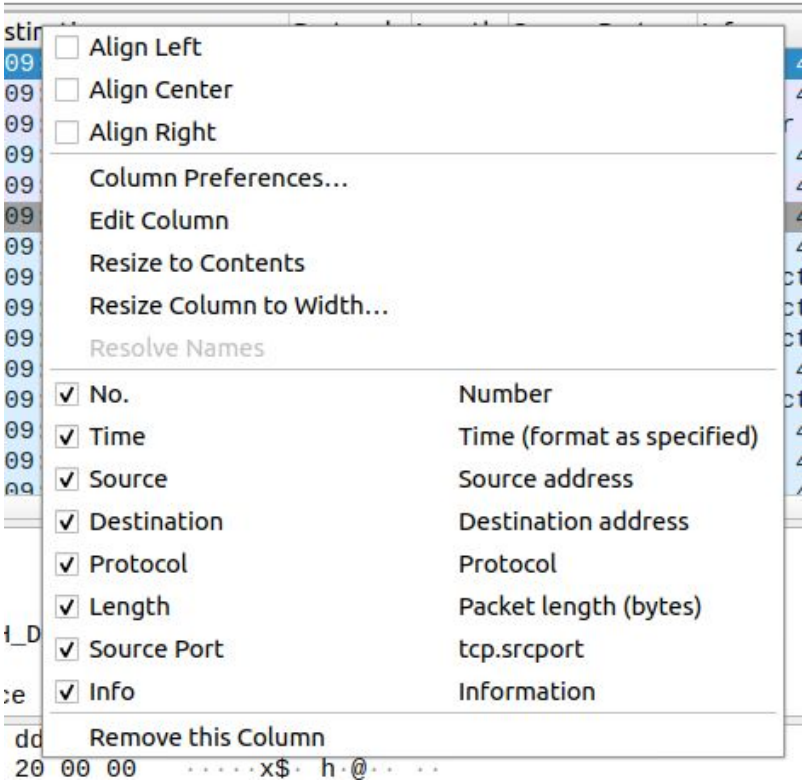
Adding additional column




Capturing from wlp1s0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
No. Time Source Destination Protocol Length Source Port Info
785 58.201966819 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TCP 1294 443 443 → 43044 [ACK] Seq=2417 Ack=518 Win=66816 Le
781 58.201723511 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TCP 1294 443 443 → 43044 [PSH, ACK] Seq=1209 Ack=518 Win=668
775 58.197257787 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TLSv1.3 1294 443 Server Hello, Change Cipher Spec
749 58.165357328 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TCP 86 443 443 → 43044 [ACK] Seq=1 Ack=518 Win=66816 Len=0
732 58.100757809 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TCP 94 443 443 → 43044 [SYN, ACK] Seq=0 Ack=1 Win=65535 Le
646 56.462895108 2404:6800:4007:82c:… 2409:40f3:f:db9a:fc… TCP 86 443 [TCP ACKed unseen segment] 443 → 54524 [FIN, AC
642 56.411107059 2404:6800:4007:82c:… 2409:40f3:f:db9a:fc… TLSv1.2 159 443 [TCP ACKed unseen segment] , Application Data
457 53.563388544 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TCP 74 443 443 → 33948 [RST] Seq=2 Win=0 Len=0
210 45.186104757 2404:6800:4007:82c:… 2409:40f3:f:db9a:fc… TCP 86 443 [TCP Dup ACK 2#1] [TCP ACKed unseen segment] 44
162 32.862040424 64:ff9b::b854:e9b0 2409:40f3:f:db9a:fc… TCP 86 443 [TCP ACKed unseen segment] 443 → 55054 [ACK] Se
123 16.472264581 64:ff9b::12a1:d87b 2409:40f3:f:db9a:fc… TCP 86 443 [TCP ACKed unseen segment] 443 → 35660 [ACK] Se
97 8.491461142 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TCP 86 443 [TCP Out-Of-Order] 443 → 33948 [FIN, ACK] Seq=1
96 8.491460677 2404:6800:4007:820:… 2409:40f3:f:db9a:fc… TCP 86 443 443 → 33948 [FIN, ACK] Seq=1 Ack=1 Win=261 Len=
2 0.098735418 2404:6800:4007:82c:… 2409:40f3:f:db9a:fc… TCP 86 443 [TCP ACKed unseen segment] 443 → 54524 [ACK] Se
Source Port: 443

Removing a column

- Right-click on any column header and un check the column to be removed.



ARP

Address Resolution Protocol

Introduction

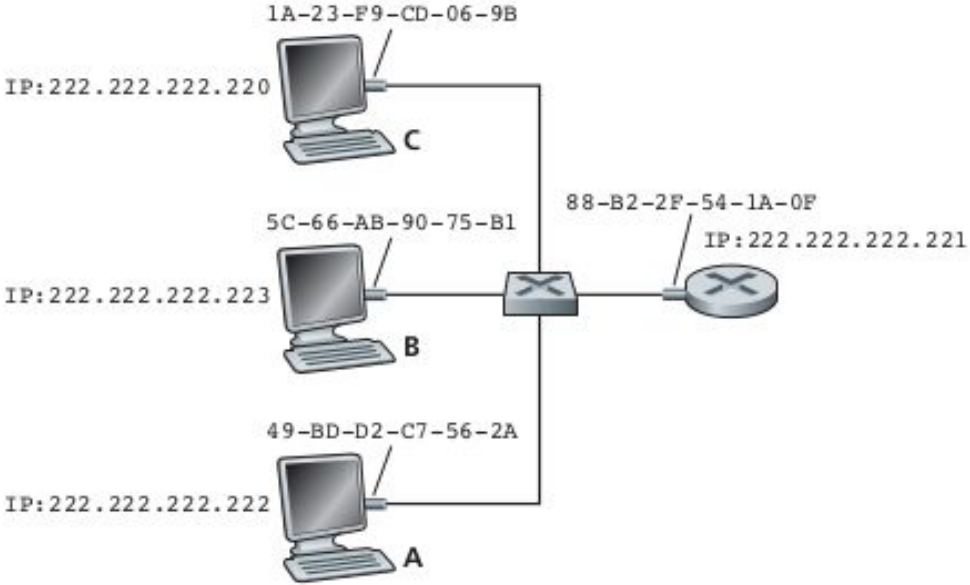
- ARP is used to map an IP address (e.g., 192.168.0.10) to an underlying MAC (Media Access Control) address (e.g., 01:02:03:04:05:06).
- Without MAC address we cannot send any packet.
- **Working of ARP:**
- When a host has to find the MAC address of the destination (using the destination's IP address) the ARP program checks its ARP lookup table to see if IP to MAC address translation is already done.
    - If it is done, the ARP packet is displayed in the form of an **ARP REPLY** (which has the MAC address of the destination) using the ARP lookup table.
    - If not, it'll send **ARP REQUEST** in the form of a broadcast packet in the network to all the devices in the LAN in order to ask who has the destination IP address, and then the destination will send back **ARP REPLY** (by giving the MAC address of the destination) and after giving this reply, it'll store the new MAC address in the ARP lookup table.
- MAC Addresses are a unique 48-bit hardware number of a computer, which is embedded into a network card NIC (known as Network Interface Card) during the time of manufacturing.
- The MAC Address is also known as the Physical Address of a network device.

- Because there are both network-layer addresses (for example, Internet IP addresses) and link-layer addresses (that is, MAC addresses), there is a need to translate between them.
- This is the job of the Address Resolution Protocol (ARP)
- An ARP module in the sending host takes any IP address on the same LAN as input, and returns the corresponding MAC address.
- One important difference between the two resolvers is that DNS resolves host names for hosts anywhere in the Internet, whereas ARP resolves IP addresses only for hosts and router interfaces on the same subnet.
- If a node in California were to try to use ARP to resolve the IP address for a node in Mississippi, ARP would return with an error.
- ARP is plug-and-play; that is, an ARP table gets built automatically—it doesn't have to be configured by a system administrator.
- And if a host becomes disconnected from the subnet, its entry is eventually deleted from the other ARP tables in the subnet.
- In truth, it is not hosts and routers that have link-layer addresses but rather their adapters (that is, network interfaces) that have link-layer addresses.

- Each host and router has an ARP table in its memory, which contains mappings of IP addresses to MAC addresses.
- The ARP table also contains a time-to-live (TTL) value, which indicates when each mapping will be deleted from the table.

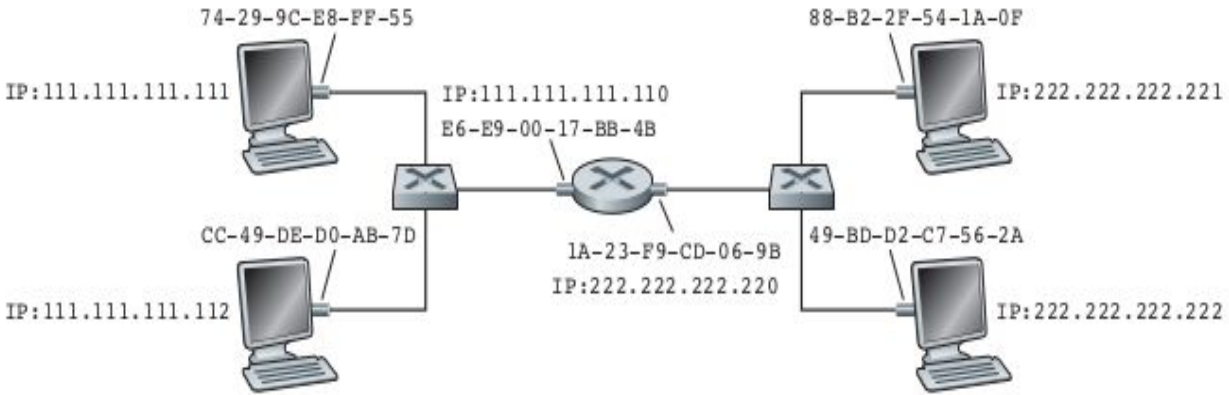
IP Address	MAC Address	TTL
222.222.222.221	88-B2-2F-54-1A-0F	13:45:00
222.222.222.223	5C-66-AB-90-75-B1	13:52:00

- Both ARP query and response packets have the same format.
- The purpose of the ARP query packet is to query all the other hosts and routers on the subnet to determine the MAC address corresponding to the IP address that is being resolved.
- The frame containing the ARP query is received by all the other adapters on the subnet.
- Each of these ARP modules checks to see if its IP address matches the destination IP address in the ARP packet.
- The one with a match sends back to the querying host a response ARP packet with the desired mapping.
- The querying host can then update its ARP table and send its IP datagram


1A-23-F9-CD-06-9B
IP:222.222.222.220
C
88-B2-2F-54-1A-0F
IP:222.222.222.221
5C-66-AB-90-75-B1
IP:222.222.222.223
B
49-BD-D2-C7-56-2A
IP:222.222.222.222
A

Because the router has two interfaces, it has two IP addresses, two ARP modules, and two adapters. Of course, each adapter in the network has its own MAC address.

The router now has to determine the correct interface on which the datagram is to be forwarded. This is done by consulting a forwarding table in the router.



arp

No.	Time	Source	Destination	Protocol	Length	Info
1903	7.866539233	7a:ad:fc:4b:be:12	IntelCor_31:2e:be	ARP	42	Who has 192.168.220.146? Tell 192.168.220.144
1904	7.866557724	IntelCor_31:2e:be	7a:ad:fc:4b:be:12	ARP	42	192.168.220.146 is at 34:f6:4b:31:2e:be

```
▸ Frame 1903: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface wlp1s0, id 0
▸ Ethernet II, Src: 7a:ad:fc:4b:be:12 (7a:ad:fc:4b:be:12), Dst: IntelCor_31:2e:be (34:f6:4b:31:2e:be)
▾ Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 7a:ad:fc:4b:be:12 (7a:ad:fc:4b:be:12)
    Sender IP address: 192.168.220.144
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.220.146
```

arp

No.	Time	Source	Destination	Protocol	Length	Info
1903	7.866539233	7a:ad:fc:4b:be:12	IntelCor_31:2e:be	ARP	42	Who has 192.168.220.146? Tell 192.168.220.144
1904	7.866557724	IntelCor_31:2e:be	7a:ad:fc:4b:be:12	ARP	42	192.168.220.146 is at 34:f6:4b:31:2e:be

```
▸ Frame 1904: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface wlp1s0, id 0
▸ Ethernet II, Src: IntelCor_31:2e:be (34:f6:4b:31:2e:be), Dst: 7a:ad:fc:4b:be:12 (7a:ad:fc:4b:be:12)
▾ Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: IntelCor_31:2e:be (34:f6:4b:31:2e:be)
    Sender IP address: 192.168.220.146
    Target MAC address: 7a:ad:fc:4b:be:12 (7a:ad:fc:4b:be:12)
    Target IP address: 192.168.220.144
```

Gratuitous ARP

- A gratuitous ARP request is an Address Resolution Protocol request packet where the source and destination IP are both set to the IP of the machine issuing the packet and the destination MAC is the broadcast address `ff:ff:ff:ff:ff:ff`.
-

Example Traffic

```
Ethernet II, Src: 02:02:02:02:02:02, Dst: ff:ff:ff:ff:ff:ff
    Destination: ff:ff:ff:ff:ff:ff (Broadcast)
    Source: 02:02:02:02:02:02 (02:02:02:02:02:02)
    Type: ARP (0x0806)
    Trailer: 000000000000000000000000000000000000
Address Resolution Protocol (request/gratuitous ARP)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    Sender MAC address: 02:02:02:02:02:02 (02:02:02:02:02:02)
    Sender IP address: 192.168.1.1 (192.168.1.1)
    Target MAC address: ff:ff:ff:ff:ff:ff (Broadcast)
    Target IP address: 192.168.1.1 (192.168.1.1)
0000  ff ff ff ff ff ff 02 02 02 02 02 02 08 06 00 01   ................
0010  08 00 06 04 00 01 02 02 02 02 02 02 c0 a8 01 01   ................
0020  ff ff ff ff ff ff c0 a8 01 01 00 00 00 00 00 00   ................
0030  00 00 00 00 00 00 00 00 00 00 00 00               ............
```

ICMP

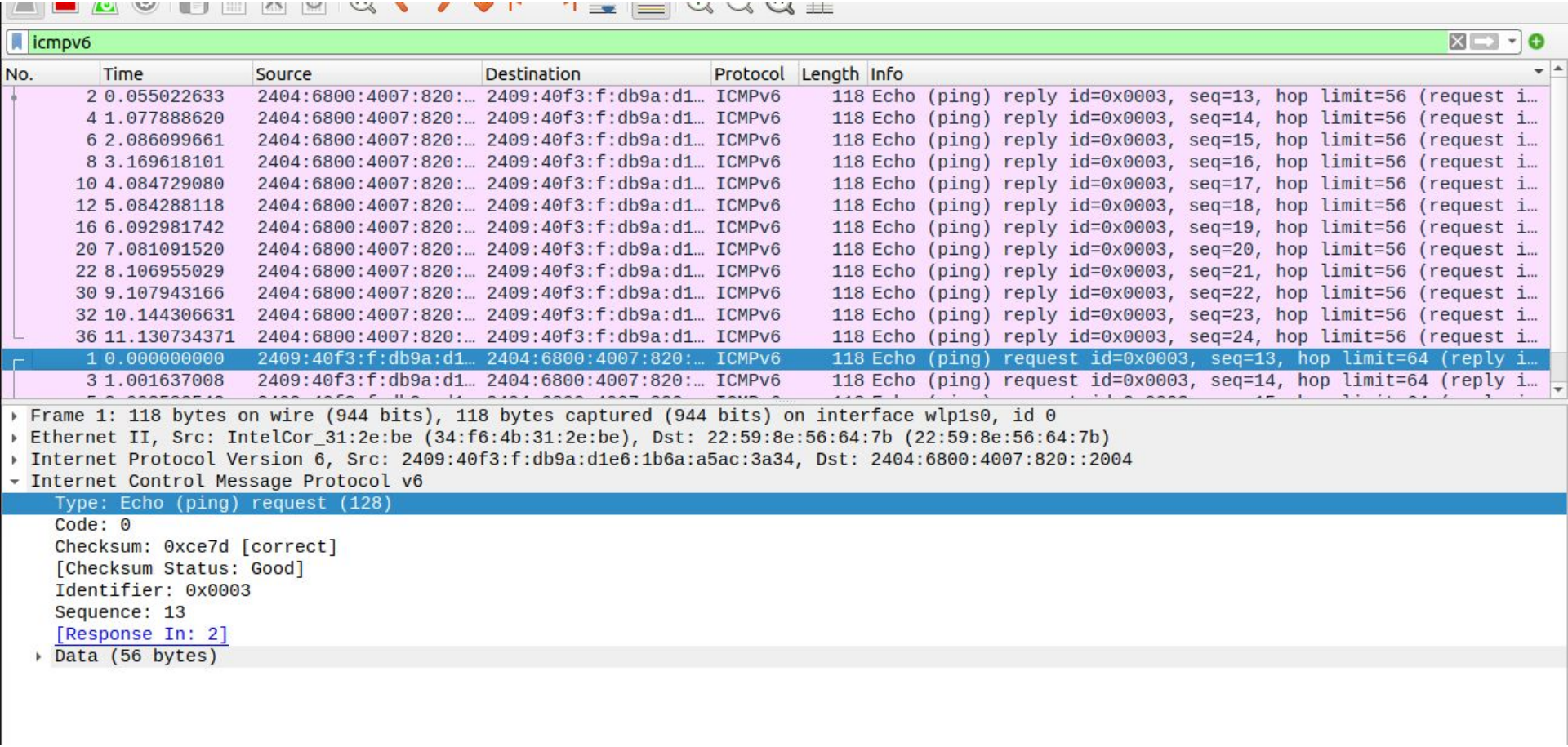
Internet Control Message Protocol

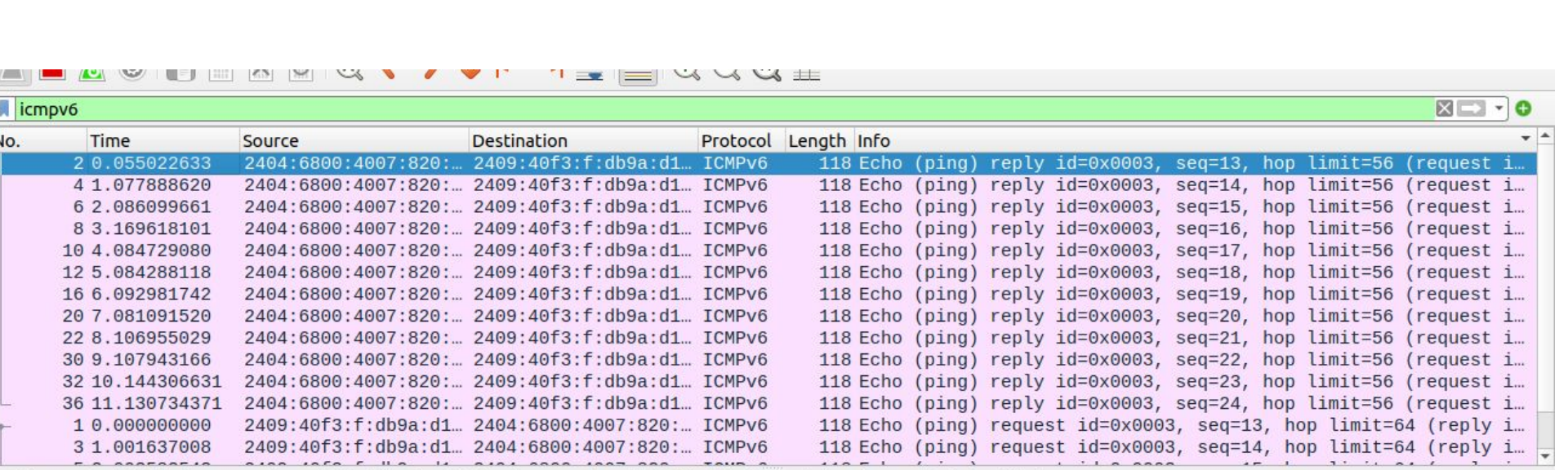
Introduction

- Unlike the Transport Control Protocol (TCP) and User Datagram Protocol (UDP), the Internet Control Message Protocol (ICMP) is not designed for carrying data.
- The most typical use of ICMP is for error reporting.
- Error data in ICMP is carried in two values: the type and the code.
- The well-known ping program sends an ICMP type 8 code 0 message to the specified host.
- The destination host, seeing the echo request, sends back a type 0 code 0 ICMP echo reply
- Another interesting ICMP message is the source quench message.
- Its original purpose was to perform congestion control— to allow a congested router to send an ICMP source quench message to a host to force that host to reduce its transmission rate.

ICMP message types

ICMP Type	Code	Description
0	0	echo reply (to ping)
3	0	destination network unreachable
3	1	destination host unreachable
3	2	destination protocol unreachable
3	3	destination port unreachable
3	6	destination network unknown
3	7	destination host unknown
4	0	source quench (congestion control)
8	0	echo request
9	0	router advertisement
10	0	router discovery
11	0	TTL expired
12	0	IP header bad


icmpv6
No. Time Source Destination Protocol Length Info
2 0.055022633 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=13, hop limit=56 (request i…
4 1.077888620 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=14, hop limit=56 (request i…
6 2.086099661 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=15, hop limit=56 (request i…
8 3.169618101 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=16, hop limit=56 (request i…
10 4.084729080 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=17, hop limit=56 (request i…
12 5.084288118 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=18, hop limit=56 (request i…
16 6.092981742 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=19, hop limit=56 (request i…
20 7.081091520 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=20, hop limit=56 (request i…
22 8.106955029 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=21, hop limit=56 (request i…
30 9.107943166 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=22, hop limit=56 (request i…
32 10.144306631 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=23, hop limit=56 (request i…
36 11.130734371 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=24, hop limit=56 (request i…
1 0.000000000 2409:40f3:f:db9a:d1… 2404:6800:4007:820:… ICMPv6 118 Echo (ping) request id=0x0003, seq=13, hop limit=64 (reply i…
3 1.001637008 2409:40f3:f:db9a:d1… 2404:6800:4007:820:… ICMPv6 118 Echo (ping) request id=0x0003, seq=14, hop limit=64 (reply i…
Frame 1: 118 bytes on wire (944 bits), 118 bytes captured (944 bits) on interface wlp1s0, id 0
Ethernet II, Src: IntelCor_31:2e:be (34:f6:4b:31:2e:be), Dst: 22:59:8e:56:64:7b (22:59:8e:56:64:7b)
Internet Protocol Version 6, Src: 2409:40f3:f:db9a:d1e6:1b6a:a5ac:3a34, Dst: 2404:6800:4007:820::2004
Internet Control Message Protocol v6
Type: Echo (ping) request (128)
Code: 0
Checksum: 0xce7d [correct]
[Checksum Status: Good]
Identifier: 0x0003
Sequence: 13
[Response In: 2]
Data (56 bytes)


icmpv6
No. Time Source Destination Protocol Length Info
2 0.055022633 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=13, hop limit=56 (request i…
4 1.077888620 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=14, hop limit=56 (request i…
6 2.086099661 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=15, hop limit=56 (request i…
8 3.169618101 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=16, hop limit=56 (request i…
10 4.084729080 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=17, hop limit=56 (request i…
12 5.084288118 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=18, hop limit=56 (request i…
16 6.092981742 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=19, hop limit=56 (request i…
20 7.081091520 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=20, hop limit=56 (request i…
22 8.106955029 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=21, hop limit=56 (request i…
30 9.107943166 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=22, hop limit=56 (request i…
32 10.144306631 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=23, hop limit=56 (request i…
36 11.130734371 2404:6800:4007:820:… 2409:40f3:f:db9a:d1… ICMPv6 118 Echo (ping) reply id=0x0003, seq=24, hop limit=56 (request i…
1 0.000000000 2409:40f3:f:db9a:d1… 2404:6800:4007:820:… ICMPv6 118 Echo (ping) request id=0x0003, seq=13, hop limit=64 (reply i…
3 1.001637008 2409:40f3:f:db9a:d1… 2404:6800:4007:820:… ICMPv6 118 Echo (ping) request id=0x0003, seq=14, hop limit=64 (reply i…
Frame 2: 118 bytes on wire (944 bits), 118 bytes captured (944 bits) on interface wlp1s0, id 0
Ethernet II, Src: 22:59:8e:56:64:7b (22:59:8e:56:64:7b), Dst: IntelCor_31:2e:be (34:f6:4b:31:2e:be)
Internet Protocol Version 6, Src: 2404:6800:4007:820::2004, Dst: 2409:40f3:f:db9a:d1e6:1b6a:a5ac:3a34
Internet Control Message Protocol v6
Type: Echo (ping) reply (129)
Code: 0
Checksum: 0xcd7d [correct]
[Checksum Status: Good]
Identifier: 0x0003
Sequence: 13
[Response To: 1]
[Response Time: 55.023 ms]
Data (56 bytes)

THANK YOU