

CyberForge
ACADEMY

Wireshark 101

Packet Capture & Analysis

\$ whoami

- Software Engineer and Researcher at CyberForge Academy
- Research, Creating course content/setups, developing SaaS software and open source tools
- For past 6 months, interned with Singapore based Web3 education startup
- Final year, B. Tech. CSE @ LPU Jalandhar
- Interests:
 - Tech 🤖
 - Cricket 🏏
 - Food 🍕





Table of contents

01

Introduction

02

Benefits of
Packet Capture

03

TCP/IP
Analysis

04

Wireshark
Filters

05

Awesome
Hacks

06

CLI Options

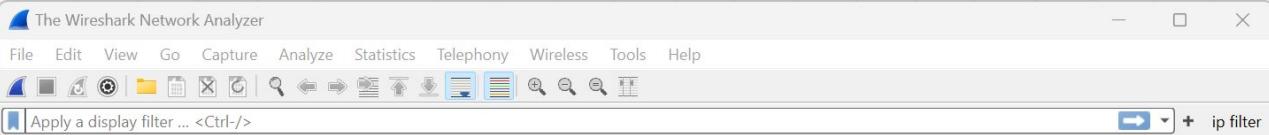


CyberForge
ACADEMY

What is Wireshark ?

- Most popular Network Protocol Analyzer
- Micro and macro analysis of captured packet data
- Released in 1998 by **Gerald Combs** as **Ethereal**
- Free and Open Source
<https://github.com/wireshark/wireshark>
- Cross-platform (Windows/Linux/MacOS)
- Can be extended in C or Lua
- Support plugins, easy way to extend functionality





Capture

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

All interfaces shown

- Wi-Fi
- Bluetooth Network Connection
- VMware Network Adapter VMnet8
- VMware Network Adapter VMnet1
- Local Area Connection* 2
- Local Area Connection* 1
- Ethernet 4
- Adapter for loopback traffic capture

Learn

[User's Guide](#) · [Wiki](#) · [Questions and Answers](#) · [Mailing Lists](#) · [SharkFest](#) · [Wireshark Discord](#) · [Donate](#)

You are running Wireshark 4.2.2 (v4.2.2-0-g404592842786). You receive automatic updates.

 Ready to load or capture

|| No Packets

|| Profile: Default

berForge
ACADEMY

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.103	239.255.255.250	SSDP	215	M-SEARCH * HTTP/1.1
2	12.223216	0a:00:27:00:00:11	Broadcast	ARP	42	Who has 192.168.56.103
3	12.223909	PCSSystemtec_18:44:d8	0a:00:27:00:00:11	ARP	60	192.168.56.103 is at 0a:00:27:00:00:11
4	12.223931	192.168.56.1	192.168.56.103	TCP	66	49988 → 8000 [SYN] Seq: 1
5	12.225437	192.168.56.103	192.168.56.1	TCP	66	8000 → 49988 [SYN, ACK]
6	12.225554	192.168.56.1	192.168.56.103	TCP	54	49988 → 8000 [ACK] Seq: 2
7	12.226024	192.168.56.1	192.168.56.103	HTTP	488	GET / HTTP/1.1
8	12.227594	192.168.56.103	192.168.56.1	TCP	60	8000 → 49988 [ACK] Seq: 3
9	12.245775	192.168.56.103	192.168.56.1	TCP	242	8000 → 49988 [PSH, ACK]

> Frame 7: 488 bytes on wire (3904 bits), 488 bytes captured (3904 bits) on interface \Device\NPF_{7BB2521C-
 > Ethernet II, Src: 0a:00:27:00:00:11 (0a:00:27:00:00:11), Dst: PCSSystemtec_18:44:d8 (08:00:27:18:44:d8)
 > Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.103
 > Transmission Control Protocol, Src Port: 49988, Dst Port: 8000, Seq: 1, Ack: 1, Len: 434
 > Hypertext Transfer Protocol

0030	04 02 ed 06 00 00	47 45 54 20 2f 20 48 54 54 50
0040	2f 31 2e 31 0d 0a 48 6f	73 74 3a 20 31 39 32 2e
0050	31 36 38 2e 35 36 2e 31	30 33 3a 38 30 30 30 0d
0060	0a 43 6f 6e 65 63 74	69 6f 6e 3a 20 6b 65 65
0070	70 2d 61 6c 69 76 65 0d	0a 55 70 67 72 61 64 65
0080	2d 49 6e 73 65 63 75 72	65 2d 52 65 71 75 65 73
0090	74 73 3a 20 31 0d 0a 55	73 65 72 2d 41 67 65 6e
00a0	74 3a 20 4d 6f 7a 69 6c	6c 61 2f 35 2e 30 20 28
00b0	57 69 6e 64 6f 77 73 20	4e 54 20 31 30 2e 30 3b
00c0	20 57 69 6e 36 34 3b 20	78 36 34 29 20 41 70 70
00d0	6c 65 57 65 62 4b 69 74	2f 35 33 37 2e 33 36 20
00e0	28 4b 48 54 4d 4c 2c 20	6c 69 6b 65 20 47 65 63
00f0	6b 6f 29 20 43 68 72 6f	6d 65 2f 31 32 30 2e 30
0100	2e 30 2e 30 20 53 61 66	61 72 69 2f 35 33 37 2e
0110	33 36 0d 0a 41 63 63 65	70 74 3a 20 74 65 78 74
0120	2f 68 74 6d 6c 2c 61 70	70 6c 69 63 61 74 69 6f
0130	6e 2f 78 68 74 6d 6c 2b	78 6d 6c 2c 61 70 70 70

.....GE T / HTTP
 /1.1 · Ho st: 192.
 168.56.1 03:8000.
 ·Connect ion: kee
 p-alive · ·Upgra
 -Insecu r e-Reques
 ts: 1 · U ser-Agen
 t: Mozil la/5.0 (Windows NT 10.0;
 Win64; x64) App
 leWebKit /537.36
 (KHTML, like Gec
 ko) Chro me/120.0
 .0.0 Saf ari/537.
 36 · Acce pt: text
 /html,ap plicatio
 n/xhtml+xml,appl

Why Packet Analysis?

- Diagnosing Network Issues

Example: Observe number of retransmissions in TCP packets.



- Performance Monitoring

Example: To monitor the response time of a web server.

- Security Analysis

Example: Analyzing network traffic to uncover abnormal communication, connections to malicious IPs, or unexpected data transfers.

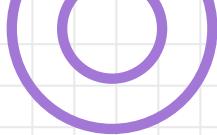
- Protocol Analysis

Example: Troubleshooting VoIP (Voice over IP) call quality issues.

- Forensic Analysis

Example: Analyzing packets over time to reconstruct events and pinpoint the compromise point.

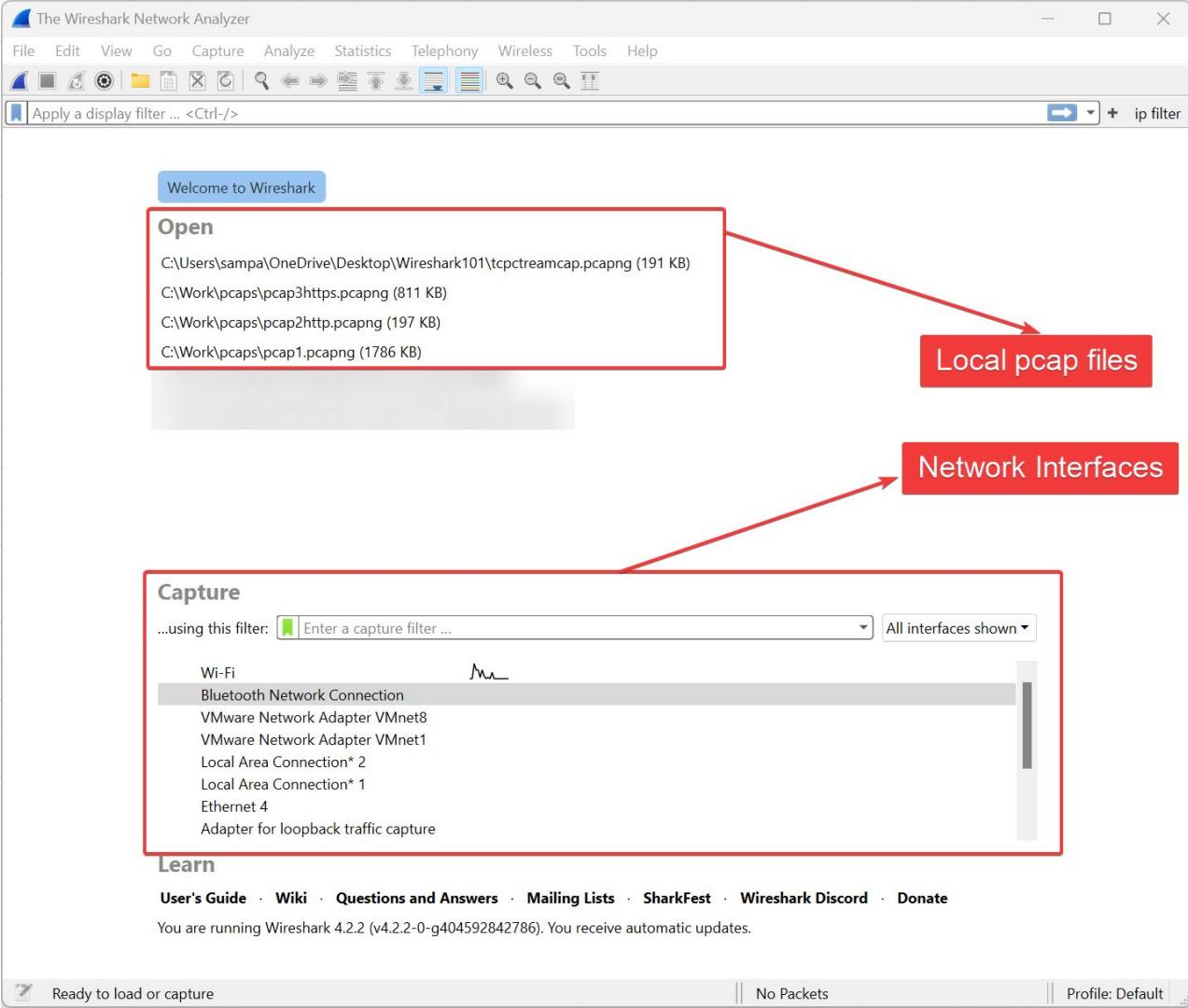


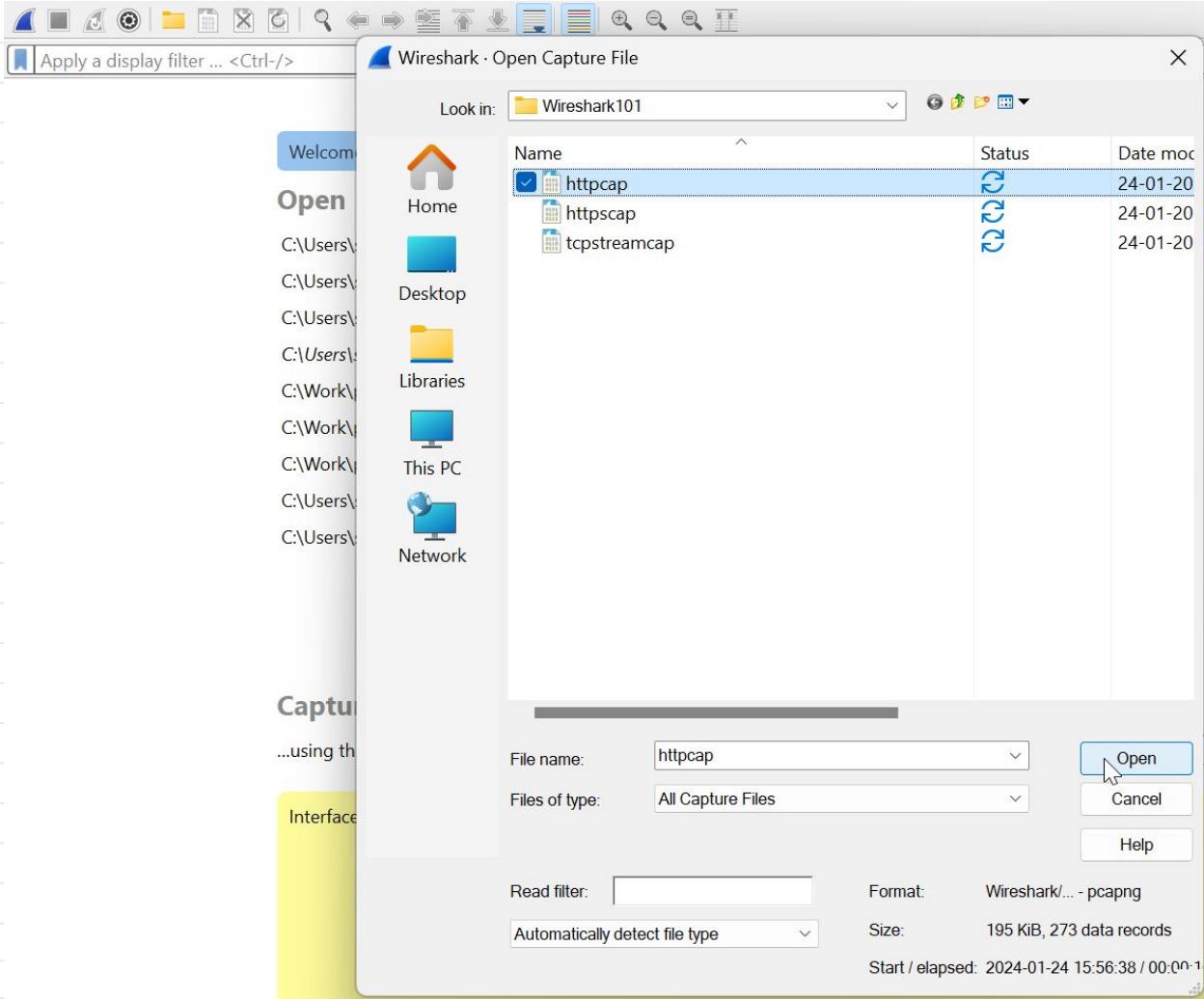


Capturing HTTP Packet

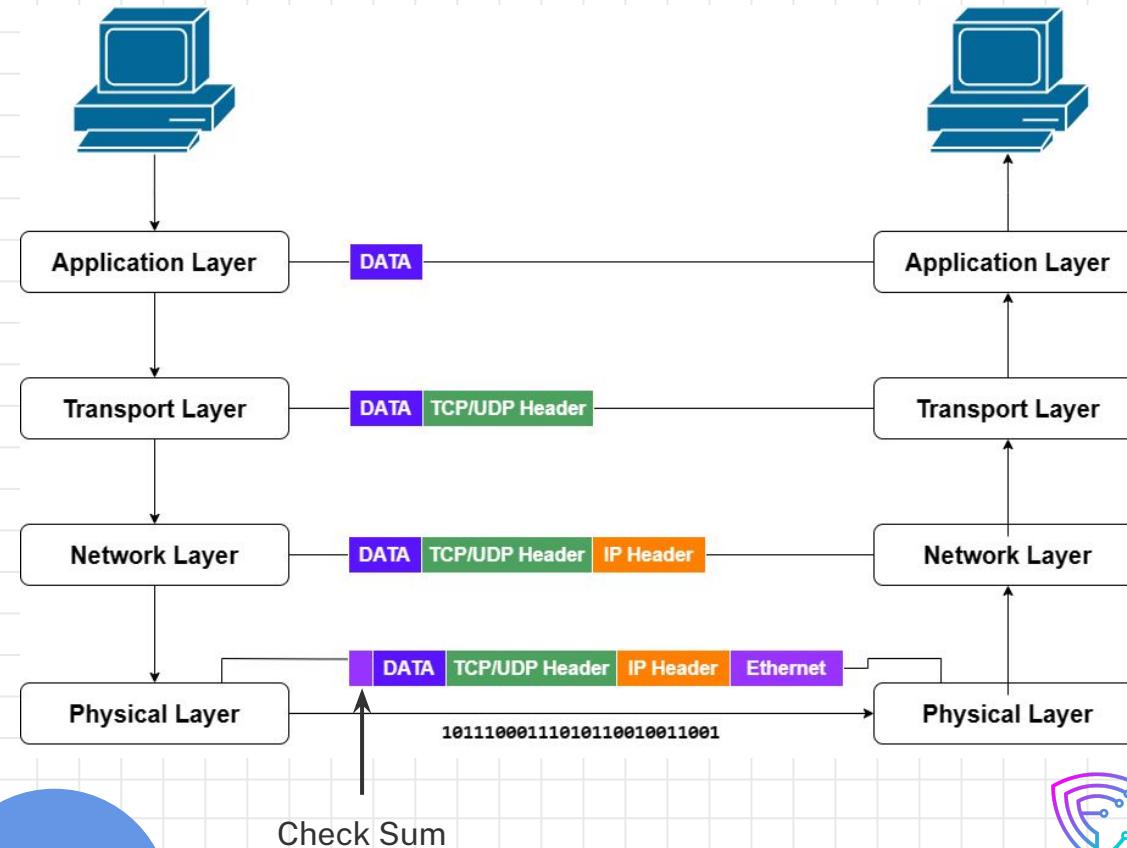


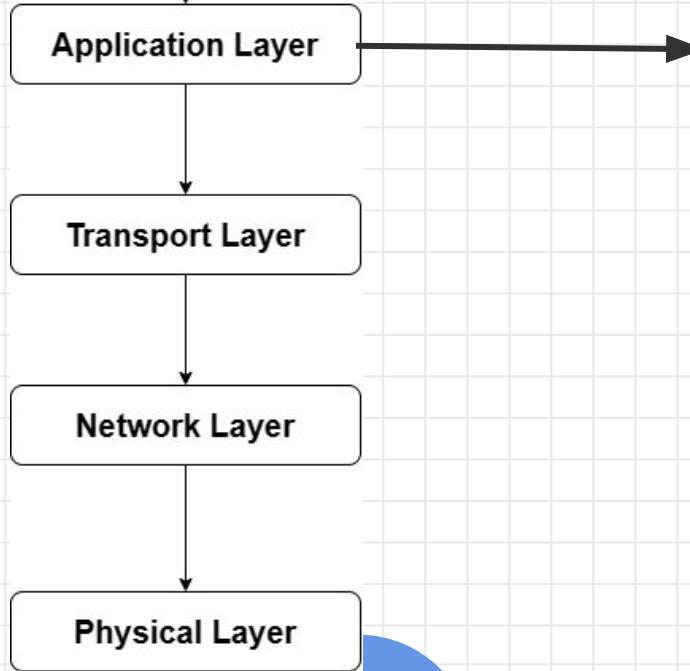
- Pcap: [Workshops/Wireshark101-OWASP/pcaps/httpcap - CyberForgeAcademy](#)
- Video: [Capturing HTTP Traffic using Wireshark \(youtube.com\)](#)





TCP/IP Model



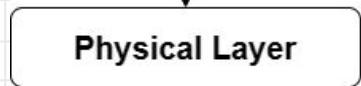
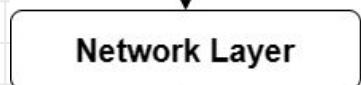
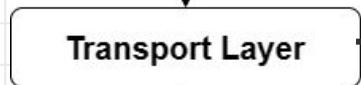
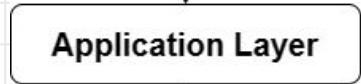


HTTP Response Packet with 200 code and page content

```
<ul style="list-style-type: none;">- ✓ Hypertext Transfer Protocol
- > HTTP/1.0 200 OK\r\nServer: SimpleHTTP/0.6 Python/3.10.12\r\nDate: Wed, 24 Jan 2024 10:11:07 GMT\r\nContent-type: text/html\r\n
- > Content-Length: 1654\r\nLast-Modified: Tue, 23 Jan 2024 19:46:49 GMT\r\n\r\n[HTTP response 1/1]  
[Time since request: 0.009803000 seconds]  
[Request in frame: 8]  
[Request URI: http://192.168.56.103:8000/]File Data: 1654 bytes
- ✓ Line-based text data: text/html (40 lines) 

```

The text block displays the contents of an HTTP response packet. It includes the protocol version (HTTP/1.0), status code (200 OK), server information (SimpleHTTP/0.6 Python/3.10.12), date (Wed, 24 Jan 2024 10:11:07 GMT), content type (text/html), content length (1654 bytes), last modified (Tue, 23 Jan 2024 19:46:49 GMT), and file data (1654 bytes). The second item in the list shows the line-based text data of the HTML file, starting with the DOCTYPE declaration and the beginning of the HTML document structure.

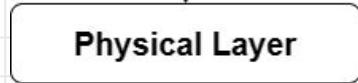
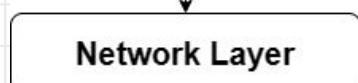
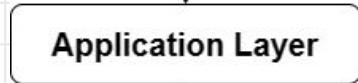


TCP Source port 8000 and Destination port 57066

✓ Transmission Control Protocol, Src Port: 8000, Dst Port: 57066, Seq: 1649, A
Source Port: 8000
Destination Port: 57066
[Stream index: 0]
➢ [Conversation completeness: Complete, WITH_DATA (31)]
[TCP Segment Len: 194]
Sequence Number: 1649 (relative sequence number)
Sequence Number (raw): 947992284
[Next Sequence Number: 1844 (relative sequence number)]
Acknowledgment Number: 435 (relative ack number)
Acknowledgment number (raw): 4092790033
0101 = Header Length: 20 bytes (5)
➢ Flags: 0x019 (FIN, PSH, ACK)
Window: 501
[Calculated window size: 64128]
[Window size scaling factor: 128]
Checksum: 0x33df [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
➢ [Timestamps]
➢ [SEQ/ACK analysis]
TCP payload (194 bytes)
TCP segment data (194 bytes)

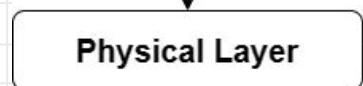
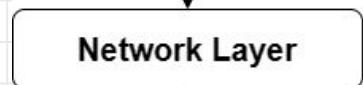
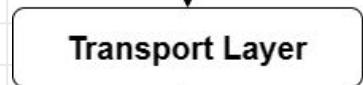
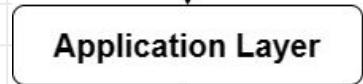


IPv4 Header, Source IP 192.168.56.103, Destination IP 192.168.56.1



- ✗ Internet Protocol Version 4, Src: 192.168.56.103, Dst: 192.168.56.1
 - 0100 = Version: 4
 - 0101 = Header Length: 20 bytes (5)
 - > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
 - Total Length: 234
 - Identification: 0x7546 (30022)
 - > 010. = Flags: 0x2, Don't fragment
 - ...0 0000 0000 0000 = Fragment Offset: 0
 - Time to Live: 64
 - Protocol: TCP (6)
 - Header Checksum: 0xd30e [validation disabled]
[Header checksum status: Unverified]
 - Source Address: 192.168.56.103
 - Destination Address: 192.168.56.1





MAC Header

Source MAC 08:00:27:18:44:d8, Destination MAC 0a:00:27:00:00:11

```
> Frame 13: 248 bytes on wire (1984 bits), 248 bytes captured (1984 bits) on interface \Device\NPF_{...}
  ✓ Ethernet II, Src: PCSSystemtec_18:44:d8 (08:00:27:18:44:d8), Dst: 0a:00:27:00:00:11 (0a:00:27:00:00:11)
    ✓ Destination: 0a:00:27:00:00:11 (0a:00:27:00:00:11)
      Address: 0a:00:27:00:00:11 (0a:00:27:00:00:11)
      .... ..1. .... .... .... = LG bit: Locally administered address (this is NOT
      .... ..0. .... .... .... = IG bit: Individual address (unicast)
    ✓ Source: PCSSystemtec_18:44:d8 (08:00:27:18:44:d8)
      Address: PCSSystemtec_18:44:d8 (08:00:27:18:44:d8)
      .... ..0. .... .... .... = LG bit: Globally unique address (factory default)
      .... ..0. .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
  > Internet Protocol Version 4, Src: 192.168.56.103, Dst: 192.168.56.1
  > Transmission Control Protocol, Src Port: 8000, Dst Port: 57066, Seq: 1649, Ack: 435, Len: 1442
  > [3 Reassembled TCP Segments (1842 bytes): #10(188), #11(1460), #13(194)]
  > Hypertext Transfer Protocol
  > Line-based text data: text/html (40 lines)
```

PCAP: [Workshops/Wireshark101-OWASP/pcaps/httpcap-CyberForgeAcademy](#)

Quiz

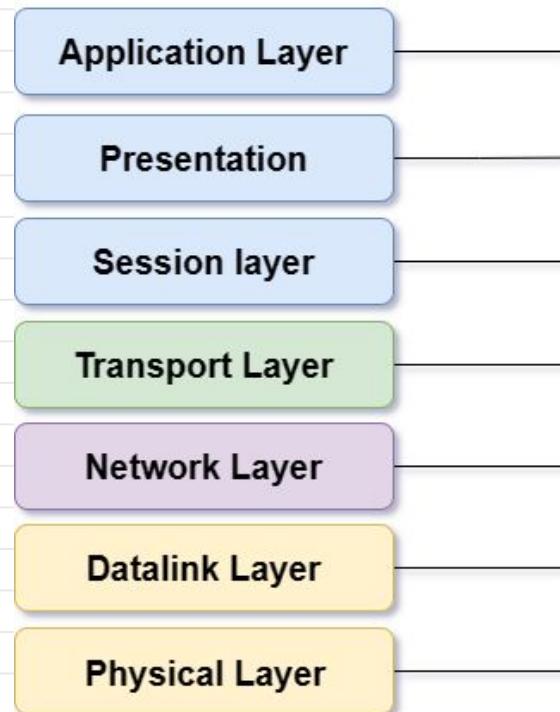
**Can we show the
OSI model in
Wireshark ?**

a) YES, we can !!

b) NO!!!!

c) Don't Know :(

OSI Model



TCP/IP Model



Wireshark Filters

- Criteria applied to captured packet data for targeting specific network traffic.
- Purpose:
 - Efficient analysis
 - Noise reduction
 - Troubleshooting
- Types of Filters:
 - Display Filters
 - Capture Filters



1. Protocol Based Filter

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

No.	Time	Source	Destination	Protocol	Length	Info
8	7.352496	192.168.56.1	192.168.56.103	HTTP	488	GET / HTT
13	7.362299	192.168.56.103	192.168.56.1	HTTP	248	HTTP/1.0
21	7.551271	192.168.56.1	192.168.56.103	HTTP	433	GET /pic1
24	7.553144	192.168.56.1	192.168.56.103	HTTP	434	GET /pic2
27	7.553679	192.168.56.1	192.168.56.103	HTTP	433	GET /pic3
38	7.563193	192.168.56.103	192.168.56.1	HTTP	70	HTTP/1.0
63	7.577103	192.168.56.103	192.168.56.1	HTTP	828	HTTP/1.0
214	7.633032	192.168.56.103	192.168.56.1	HTTP	156	HTTP/1.0
222	7.644380	192.168.56.1	192.168.56.103	HTTP	436	GET /favi
225	7.662237	192.168.56.103	192.168.56.1	HTTP	523	HTTP/1.0

> Frame 8: 488 bytes on wire (3904 bits), 488 bytes captured (3904 bits) on interface \D
> Ethernet II, Src: 0a:00:27:00:00:11 (0a:00:27:00:00:11), Dst: PCSSystemtec_18:44:d8 (0
> Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.103
> Transmission Control Protocol, Src Port: 57066, Dst Port: 8000, Seq: 1, Ack: 1, Len: 4
> Hypertext Transfer Protocol

Filtered for http traffic

2. Filter by IP Address

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.src = 192.168.56.1 ip filter

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.1	239.255.255.250	SSDP	217	M-SE
2	1.002871	192.168.56.1	239.255.255.250	SSDP	217	M-SE
3	2.004072	192.168.56.1	239.255.255.250	SSDP	217	M-SE
4	3.004880	192.168.56.1	239.255.255.250	SSDP	217	M-SE
5	7.350792	192.168.56.1	192.168.56.103	TCP	66	5706
7	7.352050	192.168.56.1	192.168.56.103	TCP	54	5706
8	7.352496	192.168.56.1	192.168.56.103	HTTP	488	GET
12	7.359008	192.168.56.1	192.168.56.103	TCP	54	5706
14	7.362488	192.168.56.1	192.168.56.103	TCP	54	5706
15	7.362822	192.168.56.1	192.168.56.103	TCP	54	5706
16	7.547398	192.168.56.1	192.168.56.103	TCP	66	5706

Filtered for source IP: 192.168.56.1

ip.dst == 192.168.56.1

No.	Time	Source	Destination	Protocol	Length	Info
6	7.351922	192.168.56.103	192.168.56.1	TCP	66	8000
9	7.353318	192.168.56.103	192.168.56.1	TCP	60	8000
10	7.358574	192.168.56.103	192.168.56.1	TCP	242	8000
11	7.358937	192.168.56.103	192.168.56.1	TCP	1514	8000
13	7.362299	192.168.56.103	192.168.56.1	HTTP	248	HTTP
19	7.550824	192.168.56.103	192.168.56.1	TCP	66	8000
22	7.552578	192.168.56.103	192.168.56.1	TCP	66	8000
25	7.553244	192.168.56.103	192.168.56.1	TCP	66	8000
28	7.555157	192.168.56.103	192.168.56.1	TCP	60	8000
29	7.556599	192.168.56.103	192.168.56.1	TCP	60	8000
30	7.556981	192.168.56.103	192.168.56.1	TCP	60	8000

Filtered for Destination IP: 192.168.56.1

ip.addr == 192.168.56.1

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.1	239.255.255.250	SSDP	217	M-SE
2	1.002871	192.168.56.1	239.255.255.250	SSDP	217	M-SE
3	2.004072	192.168.56.1	239.255.255.250	SSDP	217	M-SE
4	3.004880	192.168.56.1	239.255.255.250	SSDP	217	M-SE
5	7.350792	192.168.56.1	192.168.56.103	TCP	66	5706
6	7.351922	192.168.56.103	192.168.56.1	TCP	66	8000
7	7.352050	192.168.56.1	192.168.56.103	TCP	54	5706
8	7.352496	192.168.56.1	192.168.56.103	HTTP	488	GET
9	7.353318	192.168.56.103	192.168.56.1	TCP	60	8000
10	7.358574	192.168.56.103	192.168.56.1	TCP	242	8000
11	7.358937	192.168.56.103	192.168.56.1	TCP	1514	8000

Filtered for both source and destination IP: 192.168.56.1

3. Filter by MAC Address

eth.src == 0a:00:27:00:00:11						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.1	239.255.255.250	SSDP	217	M-SE
2	1.002871	192.168.56.1	239.255.255.250	SSDP	217	M-SE
3	2.004072	192.168.56.1	239.255.255.250	SSDP	217	M-SE
4	3.004880	192.168.56.1	239.255.255.250	SSDP	217	M-SE
5	7.350792	192.168.56.1	192.168.56.103	TCP	66	5706
7	7.352050	192.168.56.1	192.168.56.103	TCP	54	5706
8	7.352496	192.168.56.1	192.168.56.103	HTTP	488	GET
12	7.359008	192.168.56.1	192.168.56.103	TCP	54	5706
14	7.362488	192.168.56.1	192.168.56.103	TCP	54	5706
15	7.362822	192.168.56.1	192.168.56.103	TCP	54	5706
16	7.547398	192.168.56.1	192.168.56.103	TCP	66	5706

Filtered for source MAC: 0a:00:27:00:00:11

4. Filter by Frame content

frame contains "png"							X	→	ip filte
No.	Time	Source	Destination	Protocol	Length	Info			
8	7.352496	192.168.56.1	192.168.56.103	HTTP	488	GET / HT			
13	7.362299	192.168.56.103	192.168.56.1	HTTP	248	HTTP/1.0			
21	7.551271	192.168.56.1	192.168.56.103	HTTP	433	GET /pic			
24	7.553144	192.168.56.1	192.168.56.103	HTTP	434	GET /pic			
→ 27	7.553679	192.168.56.1	192.168.56.103	HTTP	433	GET /pic			
31	7.562094	192.168.56.103	192.168.56.1	TCP	242	8000 → 5			
66	7.588624	192.168.56.103	192.168.56.1	TCP	244	8000 → 5			
222	7.644380	192.168.56.1	192.168.56.103	HTTP	436	GET /fav			

```
> GET /pic3.png HTTP/1.1\r\n
Host: 192.168.56.103:8000\r\n
Connection: keep-alive\r\n
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, 1
```

Filtered by frame content: “png”

5. Saving Filters by custom names

The screenshot shows the Wireshark interface. On the left, a context menu is open over a selected filter. The menu items are:

- Current filter http
- Save this filter (highlighted with a red box and arrow 1)
- Remove this filter
- Manage Display Filters (highlighted with a red box and arrow 3)
- Filter Button Preferences...

The selected filter is highlighted with a blue box and arrow 2. The filter expression is: Ethernet address 00:00:5e:00:53:00: eth.addr == 00:00:5e:00:53:00.

On the right, the packet list pane shows several HTTP packets. A specific packet is highlighted with a blue box and arrow 2. The details pane shows the following information for the highlighted packet:

3904 bits) on interface : PCSSystemtec_18.56.103.00, Seq: 1, Ack: 1

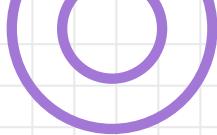
Protocol	Length
HTTP	488
HTTP	248
HTTP	433
HTTP	434
HTTP	433
HTTP	70
HTTP	828
HTTP	156
HTTP	436
HTTP	523

1. Open saved filters
2. Select from saved
3. Add custom name

Filter Name	Filter Expression
Ethernet address 00:00:5e:00:53:00	eth.addr == 00:00:5e:00:53:00
Ethernet type 0x0806 (ARP)	eth.type == 0x0806
Ethernet broadcast	eth.addr == ff:ff:ff:ff:ff:ff
No ARP	not arp
IPv4 only	ip
IPv4 address 192.0.2.1	ip.addr == 192.0.2.1
IPv4 address isn't 192.0.2.1	ip.addr != 192.0.2.1
IPv6 only	ipv6
IPv6 address 2001:db8::1	ipv6.addr == 2001:db8::1
TCP only	tcp
UDP only	udp
Non-DNS port	!(udp.port == 53 tcp.port == 53)
TCP or UDP port is 80 (HTTP)	tcp.port == 80 udp.port == 80
HTTP	http
No ARP and no DNS	not arp and not dns
Non-HTTP and non-SMTP to/from 192.0.2.1	ip.addr == 192.0.2.1 and tcp.port not in {80, 25}

<C:\Users\sample\AppData\Roaming\Wireshark\dfilters>

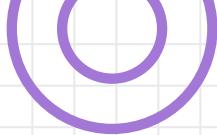
Menu to add custom names



TCP Stream Reconstruction



- Pcap: [Workshops/Wireshark101-OWASP/pcaps/tcpstreampcap -CyberForge Academy](#)
- Video: [Reconstructing TCP Stream \(youtube.com\)](#)



Extracting Resources from Traffic



- Pcap: [Workshops/Wireshark101 - OWASP/pcaps/objectcapture - CyberForgeAcademy](#)
- Video: [Exporting Objects from HTTP Traffic \(youtube.com\)](#)

Hack 1: Using Protocol Hierarchy

The screenshot shows the Wireshark interface with the Statistics menu open. The 'Protocol Hierarchy' option is highlighted with a red box and a cursor is hovering over it. The menu also includes options like Capture File Properties, Resolved Addresses, Conversations, Endpoints, Packet Lengths, I/O Graphs, Service Response Time, DHCP (BOOTP) Statistics, NetPerfMeter Statistics, ONC-RPC Programs, 29West, and ANCP.

No.	Time	Source
1	0.000000	192.16
2	1.002871	192.16
3	2.004072	192.16
4	3.004880	192.16
5	7.350792	192.16
6	7.351922	192.16
7	7.352050	192.16
8	7.352496	192.16
9	7.353318	192.16
10	7.358574	192.16
11	7.358937	192.16

The right side of the interface shows a list of captured network protocols. The table has columns for Protocol, Length, and Info. Several entries are highlighted with different colors (blue, green, yellow, grey) and some have arrows pointing to them, indicating they are being analyzed or selected.

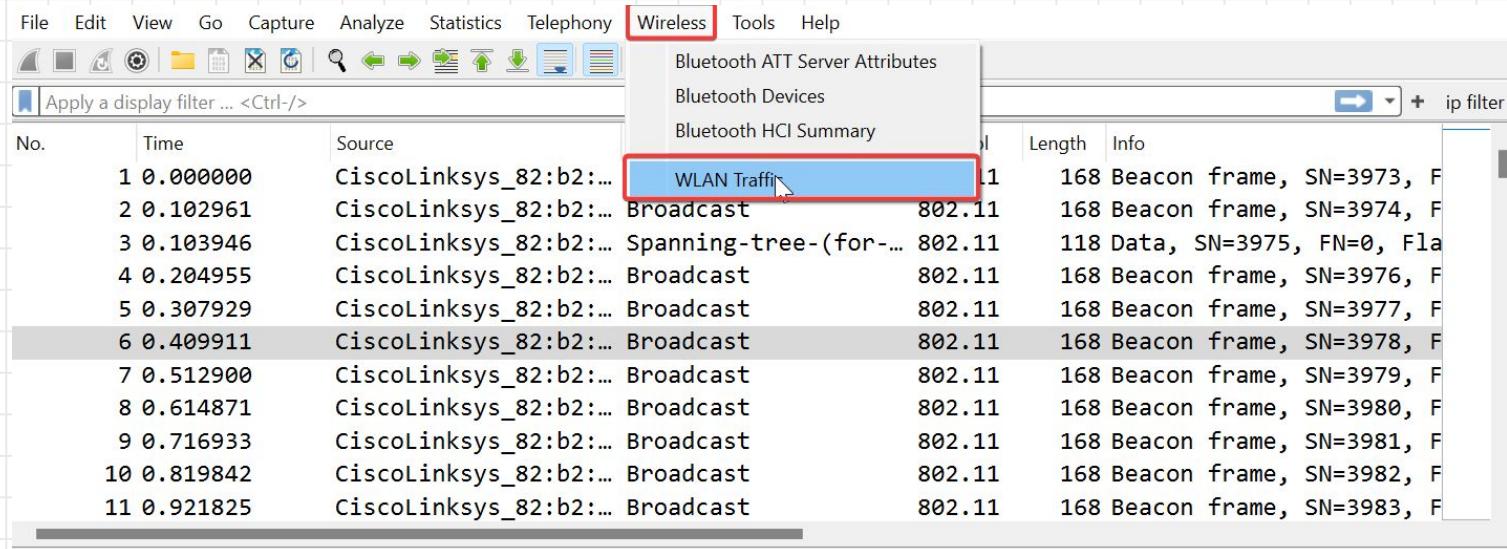
Protocol	Length	Info
DP	217	M-SE
DP	66	570E
DP	66	800E
DP	54	570E
TP	488	GET
DP	60	800E
DP	242	800E
DP	1514	800E



Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
Frame	100.0	244	100.0	190245	21 k	0	0	0	244
Ethernet	100.0	244	1.8	3494	388	0	0	0	244
Internet Protocol Version 4	99.2	242	2.5	4840	538	0	0	0	242
User Datagram Protocol	4.9	12	0.1	96	10	0	0	0	12
Simple Service Discovery Protocol	4.9	12	1.1	2096	233	12	2096	233	12
Transmission Control Protocol	94.3	230	94.4	179663	19 k	220	175954	19 k	230
Hypertext Transfer Protocol	4.1	10	92.0	174943	19 k	5	1954	217	10
Portable Network Graphics	0.8	2	78.2	148710	16 k	2	148710	16 k	2
Line-based text data	0.8	2	1.1	2123	236	2	2123	236	2
JPEG File Interchange Format	0.4	1	11.2	21214	2359	1	21214	2359	1
Address Resolution Protocol	0.8	2	0.0	74	8	2	74	8	2



Hack 2: Checking nearby devices over Wifi network



The screenshot shows the Wireshark interface with the 'Wireless' tab selected. A red box highlights the 'WLAN Traffic' option in the dropdown menu. The main pane displays a list of 11 captured frames, all from a source of 'CiscoLinksys_82:b2:...' and destination 'Broadcast'. The frames are IEEE 802.11 Beacon frames, ranging in time from 0.000000 to 0.921825. The details column shows frame lengths of 168 bytes and SN values from 3973 to 3983.

No.	Time	Source	Destination	Type	Length	Info
1	0.000000	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3973, F
2	0.102961	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3974, F
3	0.103946	CiscoLinksys_82:b2:...	Spanning-tree-(for...)	802.11	118	Data, SN=3975, FN=0, Fla
4	0.204955	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3976, F
5	0.307929	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3977, F
6	0.409911	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3978, F
7	0.512900	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3979, F
8	0.614871	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3980, F
9	0.716933	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3981, F
10	0.819842	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3982, F
11	0.921825	CiscoLinksys_82:b2:...	Broadcast	802.11	168	Beacon frame, SN=3983, F

PCAP: [Workshops/Wireshark101-OWASP/pcaps/wifiCapture.pcap](#) CyberForgeAcademy

Wireshark · Wireless LAN Statistics · wifiCapture.pcap

BSSID	Channel	SSID	Percent Packets	Percent Retry	Retry	Beacons	Data Pkts	Probe Reqs	Probe Resp	Auths	Deauths	Other	Protection
00:0c:41:82:b2:55	1	Coherer	98.2	4.9	35	398	284	0	26	2	0	3	Unknown
00:0c:41:82:b2:53			46.3	8.2	12	81	65	0	0	0	0	0	
00:0c:41:82:b2:55			17.8	32.1	18	23	2	0	26	2	0	3	Base station
00:0d:1d:06:e0:f2			0.3	0.0	0	1	0	0	0	0	0	0	
00:0d:93:82:36:3a			92.4	12.0	35	179	81	0	26	2	0	3	
01:00:5e:00:00:01			0.3	0.0	0	0	1	0	0	0	0	0	
01:00:5e:00:00:02			0.3	0.0	0	0	1	0	0	0	0	0	
01:00:5e:00:00:fb			4.4	0.0	0	0	14	0	0	0	0	0	
01:00:5e:7f:fff:fa			1.9	0.0	0	0	6	0	0	0	0	0	
01:80:c2:00:00:00			6.7	0.0	0	0	21	0	0	0	0	0	
06:0c:41:82:b2:53			0.3	0.0	0	0	1	0	0	0	0	0	
09:00:07:ff:ff:ff			15.6	0.0	0	0	49	0	0	0	0	0	
33:33:00:00:00:02			4.1	7.7	1	0	13	0	0	0	0	0	
33:33:ff:82:36:3a			1.9	16.7	1	0	6	0	0	0	0	0	
ff:ff:ff:ff:ff:ff			7.6	12.5	3	0	24	0	0	0	0	0	
98:d3:04:64:fa:55		<Broadcast>	0.1	0.0	0	0	1	0	0	0	0	0	
00:0d:93:82:36:3a			100.0	0.0	0	1	0	0	0	0	0	0	
33:33:ff:82:36:3a			100.0	0.0	0	0	1	0	0	0	0	0	
ff:ff:ff:ff:ff:ff		Coherer	1.0	0.0	0	0	0	7	0	0	0	0	
ff:ff:ff:ff:ff:ff		<Broadcast>	0.7	0.0	0	0	0	5	0	0	0	0	

List of all the nearby devices that are connected to Wifi pcap:



Hack 3: Check bluetooth connected devices

File Edit View Go Capture Analyze Statistics Telephony **Wireless** Tools Help

Bluetooth ATT Server Attributes
Bluetooth Devices
Bluetooth HCI Summary

No.	Time	Source			Length	Info
1	0.000000	host		HCI_CMD	9	Sent Inquiry
2	0.005959	controller	host	HCI_EVT	7	Rcvd Command Status (Inq)
3	2.792499	controller	host	HCI_EVT	18	Rcvd Inquiry Result
4	20.504540	controller	host	HCI_EVT	4	Rcvd Inquiry Complete
5	20.504967	host	controller	HCI_CMD	14	Sent Remote Name Request
6	20.510538	controller	host	HCI_EVT	7	Rcvd Command Status (Rem)
7	41.024117	controller	host	HCI_EVT	258	Rcvd Remote Name Request
8	82.957128	controller	host	HCI_EVT	13	Rcvd Connect Request
9	82.957154	host	controller	HCI_CMD	11	Sent Accept Connection R
10	82.972125	controller	host	HCI_EVT	7	Rcvd Command Status (Acc)
11	83.019118	controller	host	HCI_EVT	9	Rcvd PIN Code Request

> Frame 1: 9 bytes on wire (72 bits), 9 bytes captured (72 bits)
> Bluetooth
> Bluetooth HCI H4
> Bluetooth HCI Command - Inquiry

0000 01 01 04 05 33 8b 9e 10 00

PCAP: [Workshops/Wireshark101-OWASP/pcaps/BluetoothCapture.cap](#) CyberForgeAcademy

BD_ADDR	OUI	Name	LMP Version	LMP Subversion	Manufacturer	HCI Version	HCI Revision	Is Local Adapter
00:0e:6d:07:2e:fa	MurataManufa							

All Interfaces Show information steps

1 items; Right click for more option; Double click for device details

Bluetooth address of the connected user

Hack 4: Type Less & Drag More

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

frame.len == 217

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH
2	1.002871	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH
3	2.004072	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH
4	3.004880	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH

> Frame 4: 217 bytes on wire (1736 bits), 217 bytes captured (1736 bits) on interface \D
> Ethernet II, Src: 0a:00:27:00:00:11 (0a:00:27:00:00:11), Dst: IPv4mcast_7f:ff:fa (01:0
> Internet Protocol Version 4, Src: 192.168.56.1, Dst: 239.255.255.250
> User Datagram Protocol, Src Port: 59659, Dst Port: 1900
> Simple Service Discovery Protocol

0000	01 00 5e 7f ff fa 0a 00	27 00 00 11 08 00 45 00	..^..... '.....E.
0010	00 cb fa 2e 00 00 01 11	d6 4f c0 a8 38 01 ef ff 0..8...
0020	ff fa e9 0b 07 6c 00 b7	b6 ee 4d 2d 53 45 41 521... .M-SEAR
0030	43 48 20 2a 20 48 54 54	50 2f 31 2e 31 0d 0a 48	CH * HTT P/1.1..H
0040	4f 53 54 3a 20 32 33 39	2e 32 35 35 2e 32 35 35	OST: 239 .255.255

Source Hardware Address (eth.src), 6 bytes || Packets: 244 · Displayed: 4 (1.6%) || Profile: Default

Hack 5: Combine Filters

frame.len == 217

No.	Time	Protocol	Length
1	0.00000	255.250 SSDP	217
2	1.00287	255.250 SSDP	217
3	2.00407	255.250 SSDP	217
4	3.00488	255.250 SSDP	217

Context menu for the selected row (Frame 1):

- Expand Subtrees
- Collapse Subtrees
- Expand All
- Collapse All
- Apply as Column Ctrl+Shift+I
- Apply as Filter** (highlighted with red box)
- Prepare as Filter
- Conversation Filter
- Colorize with Filter
- Follow
- Copy
- Show Packet Bytes... Ctrl+Shift+O
- Export Packet Bytes... Ctrl+Shift+X
- Wiki Protocol Page
- Filter Field Reference
- Protocol Preferences
- Decode As... Ctrl+Shift+U
- Go to Linked Packet
- Show Linked Packet in New Window

Sub-menu for "Apply as Filter":

- Apply as Filter: ip.proto == 17
- Selected
- Not Selected
- ...and Selected** (highlighted with red box)
- ...or Selected
- ...and not Selected
- ...or not Selected

Protocol: UDP (17) (highlighted with red box)

Header checksum: 0xd64f [validation disabled]
[Header checksum status: Unverified]

Source Address: 192.168.56.1



((frame.len == 217) && (ip.proto == 17))						
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH
2	1.002871	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH
3	2.004072	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH
4	3.004880	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH

```
> Ethernet II, Src: 0a:00:27:00:00:11 (0a:00:27:00:00:11), Dst: IPv4mcast_7f:ff:fa (01:00:5e:7f:ff:fa)
`- Internet Protocol Version 4, Src: 192.168.56.1, Dst: 239.255.255.250
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
`- Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 203
    Identification: 0xfa2c (64044)
`- 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 1
    Protocol: UDP (17)
```



Combined frame length and protocol filters by AND operation

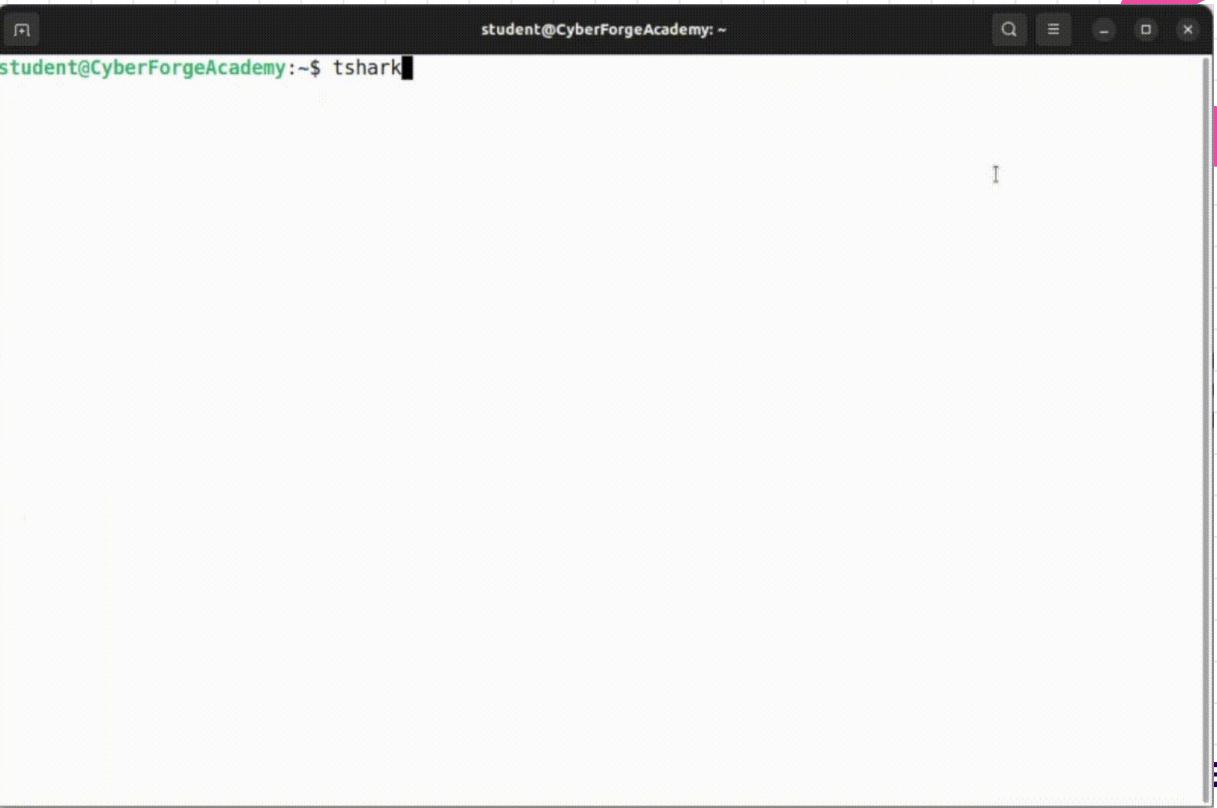


But, What on CLI?

Tshark (<https://tshark.dev>)

Documentation:

https://www.wireshark.org/docs/wsug_html_chunked/AppToolsTshark.html



A screenshot of a terminal window titled "student@CyberForgeAcademy: ~". The window shows the command "student@CyberForgeAcademy:~\$ tshark" entered at the prompt. The terminal has a dark background with light-colored text. The window is set against a background featuring large, stylized blue and pink letters "C" and "A" on the right side.

```
student@CyberForgeAcademy:~$ tshark
```

But, Can we do better?

Tcpdump

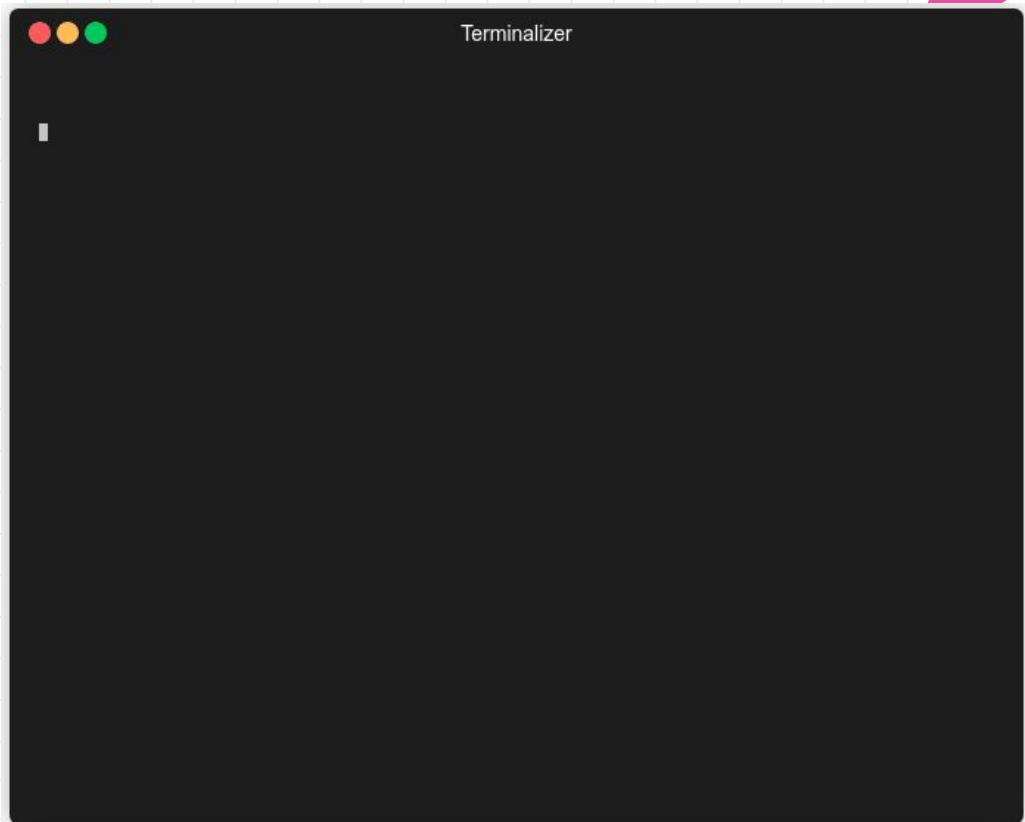
(<https://www.tcpdump.org/>)



```
student@CyberForgeAcademy:~$ sudo tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on enp0s3, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

Ok, but I want it like GUI :)

Termshark (<https://github.com/gcla/termshark>)



Install Wireshark on Linux

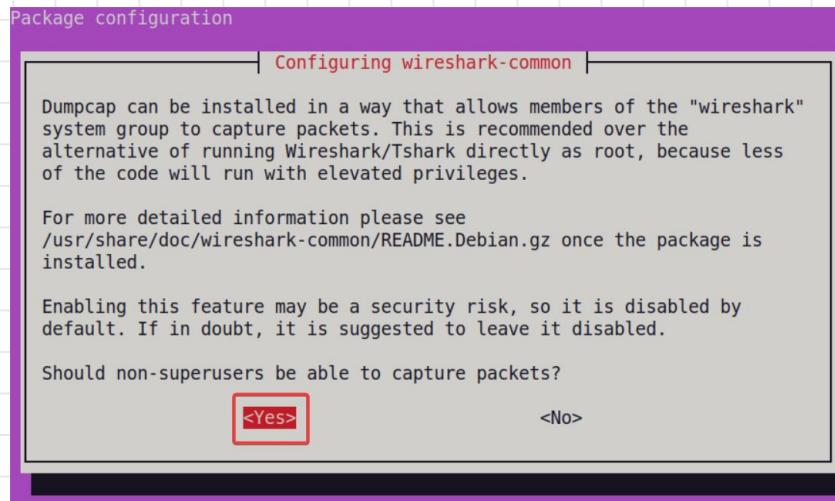
Step 1: Update device packages

Command: sudo apt-get update



Step 2: Download and install wireshark

Command: sudo apt install wireshark



Install Wireshark on Linux

Step 4: Add your current user to the wireshark group

Command: sudo adduser \$USER wireshark

Step 5: Restart system to start using Wireshark.

Thanks!

Do you have any questions?

contact@cyberforge.academy

+91 837537763

<https://cyberforge.academy>

<https://github.com/CyberForgeAcademy/Workshops>