

TASK 5: CAPTURE AND ANALYZE NETWORK TRAFFIC USING WIRESHARK.

Objective:

The objective of this task is to use Wireshark to capture live network traffic, identify various communication protocols in use, and analyze the packet details.

Tools Used:

- Wireshark.
- Web Browser (Google Chrome).

Procedure:

Step 1. Install Wireshark

- Download and install Wireshark from the official website: <https://www.wireshark.org>.

Step 2. Start Packet Capture

- Launch Wireshark.
- Select your active network interface (e.g., Wi-Fi or Ethernet).
- Click Start Capturing Packets (the blue shark fin icon).

Step 3. Generate Network Traffic

- While capturing, open a web browser and visit a website (e.g., <https://example.com>) or use the terminal/command prompt to run `ping google.com`.

Step 4. Stop the Capture

- After about 1 minute of activity, click the Stop button (red square icon).

Step 5. Filter by Protocol

- Use the filter bar to type protocols such as `http`, `dns`, or `tcp` and hit Enter to view specific packets.

Step 6. Identify Protocols

- Examine the captured packets and identify at least three different protocols (e.g., HTTP, DNS, TCP, UDP, ICMP).

Step 7. Export the Capture

- Go to File > Export Specified Packets.
- Save the capture as a .pcap file

Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter... <Ctrl+F>

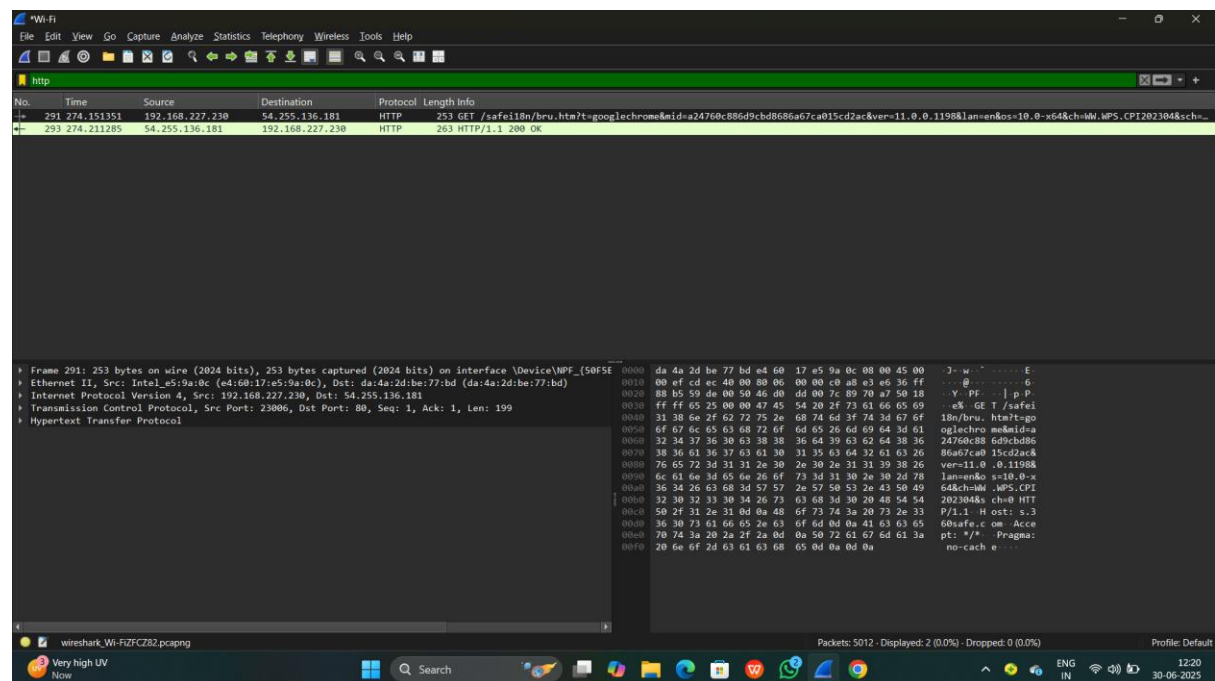
No.	Time	Source	Destination	Protocol	Length	Info
4594	322.702585	2409:40f4:1007:7837...	2404:6800:4002:806c...	QUIC	1296	Protected Payload (KP0), DCID=F89cef10f9a4bde
4595	322.702659	2409:40f4:1007:7837...	2404:6800:4002:806c...	QUIC	92	Protected Payload (KP0), DCID=F89cef10f9a4bde
4596	322.900122	2409:40f4:1007:7837...	2404:6800:4002:806c...	QUIC	1296	Protected Payload (KP0), DCID=F89cef10f9a4bde
4597	322.929468	2404:6800:4002:806c...	2409:40f4:1007:7837...	QUIC	95	Protected Payload (KP0)
4598	322.959213	2404:6800:4002:806c...	2409:40f4:1007:7837...	QUIC	311	Protected Payload (KP0)
4599	322.959213	2404:6800:4002:806c...	2409:40f4:1007:7837...	QUIC	85	Protected Payload (KP0)
5000	322.560384	2409:40f4:1007:7837...	2404:6800:4002:806c...	QUIC	101	Protected Payload (KP0), DCID=F89cef10f9a4bde
5001	322.996063	2409:40f4:1007:7837...	2404:6800:4002:806c...	QUIC	95	Protected Payload (KP0), DCID=F89cef10f9a4bde
5002	323.011237	2409:40f4:1007:7837...	2404:6800:4003:c00c...	TCP	75	[TCP Keep-Alive] 23013 > 5228 [ACK] Seq=2037 Ack=8357 Win=64768 Len=1
5003	323.050178	2404:6800:4002:806c...	2409:40f4:1007:7837...	QUIC	89	Protected Payload (KP0)
5004	323.063837	2404:6800:4002:806c...	2409:40f4:1007:7837...	QUIC	89	Protected Payload (KP0)
5005	323.076422	2404:6800:4003:c00c...	2409:40f4:1007:7837...	TCP	90	[TCP Keep-Alive ACK] 5228 > 23013 [ACK] Seq=8357 Ack=2038 Win=267776 Len=0 SLE=2037 SRE=2038
5006	324.778986	2001:4860:4802:34:1::	2409:40f4:1007:7837...	TLSv1.3	113	Application Data
5007	324.779981	2409:40f4:1007:7837...	2001:4860:4802:34:1::	TCP	74	23028 > 443 [FIN, ACK] Seq=4817 Ack=20271 Win=4256 Len=0
5008	324.818417	2001:4860:4802:34:1::	2409:40f4:1007:7837...	TCP	74	443 > 23028 [ACK] Seq=20271 Ack=4818 Win=265477 Len=0
5009	324.847595	2001:4860:4802:34:1::	2409:40f4:1007:7837...	TCP	74	443 > 23028 [FIN, ACK] Seq=20271 Ack=4818 Win=265472 Len=0
5010	324.847937	2409:40f4:1007:7837...	2001:4860:4802:34:1::	TCP	74	23028 > 443 [ACK] Seq=4818 Ack=20272 Win=64256 Len=0
5011	324.849770	2409:40f4:1007:7837...	2404:6800:4007:826c...	TCP	75	[TCP Keep-Alive] 23015 > 443 [ACK] Seq=1798 Ack=7527 Win=64512 Len=1
5012	324.883794	2404:6800:4007:826c...	2409:40f4:1007:7837...	TCP	90	[TCP Keep-Alive ACK] 443 > 23015 [ACK] Seq=7527 Ack=1799 Win=267776 Len=0 SLE=1798 SRE=1799

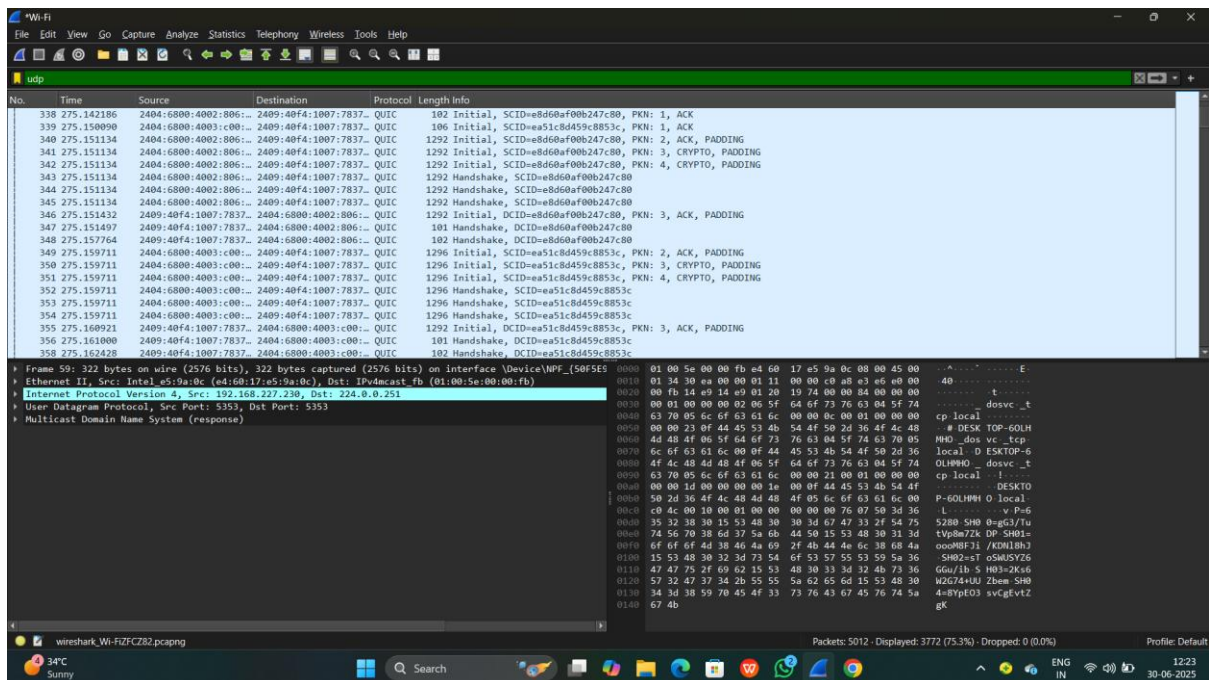
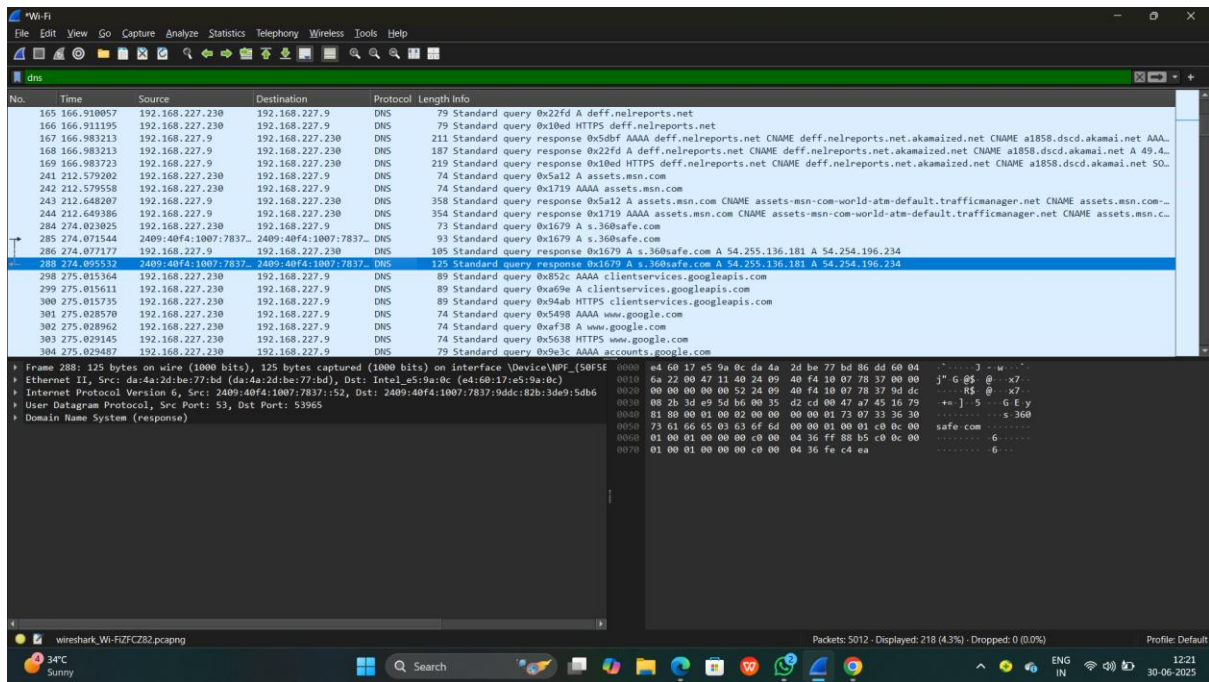
Frame 1: 133 bytes on wire (1064 bits), 133 bytes captured (1064 bits) on interface \Device\NPF_{50F5E9C0010} Ethernet II, Src: da:4a:2d:be:77:bd (da:4a:2d:be:77:bd), Dst: Intel_e5:9a:0c (e4:60:17:e5:9a:0c) Internet Protocol Version 6, Src: 64:f9b9:134b:4f6d, Dst: 2409:40f4:1007:7837:9ddc:82d:35e9:5d86 Transmission Control Protocol, Src Port: 443, Dst Port: 22089, Seq: 1, Ack: 1, Len: 59 Transport Layer Security

0000 e4 60 17 e5 9a 0c da 4a 2d be 77 bd 86 dd 60 00
0010 00 00 00 4f 06 2b 00 64 ff 9b 00 00 00 00 00 00
0020 00 00 34 bb 4f 6d 24 09 fa 10 07 78 37 9d dc
0030 00 2b 3d 49 5d b6 01 bb 59 37 c3 cd 10 b4 31 e
0040 f3 7a 50 18 00 0b 1c db 00 00 17 03 03 00 3d 20
0050 85 00 e2 2a fe 8a 56 5b 79 e7 b5 69 36 fa 8c f1
0060 7a d5 c4 e7 ac 2f 1a fe 5f c7 01 11 9f 05 5d 6e
0070 83 1e 1e df 55 74 e7 98 50 8e aa aa 98 b2 b4 11
0080 3a fe a0 c5 d1

wireshark-Wi-FiZC82.pcapng
Packets: 5012 · Dropped: 0 (0.0%)
Profile: Default
12:19 30-06-2023

Sports headline
Lando Norris wi...





Conclusion:

The experiment successfully demonstrated the ability of Wireshark to capture and filter live network traffic. Key protocols such as DNS, HTTP, and ICMP were identified, and packet details were analyzed. This task provided a clear understanding of how different protocols work in real-time communication.