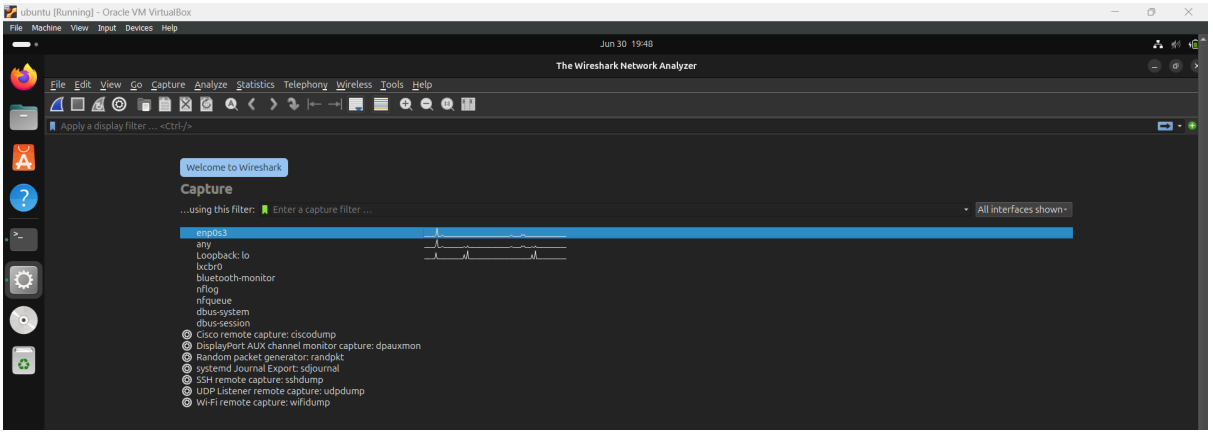
Task 5: Capture and Analyze Network Trac Using Wireshark

1)Install Wireshark
Here, i installed the wireshark.

```
vamsi@vamsi-VirtualBox:~$ sudo apt update
[sudo] password for vamsi:
Hit:1 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu noble InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu noble-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
9 packages can be upgraded. Run 'apt list --upgradable' to see them.
vamsi@vamsi-VirtualBox:~$ sudo apt install wireshark -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
wireshark is already the newest version (4.2.2-1.1build3).
The following package was automatically installed and is no longer required:
  python3-netifaces
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 9 not upgraded.
vamsi@vamsi-VirtualBox:~$
```

2)Start capturing on your active network interface.
This is the starting interface of the wireshark.



3)Browse a website or ping a server to generate traffic.
After starting the capture, a new terminal window was opened and the following command was run to ping Google, which generates ICMP packets.

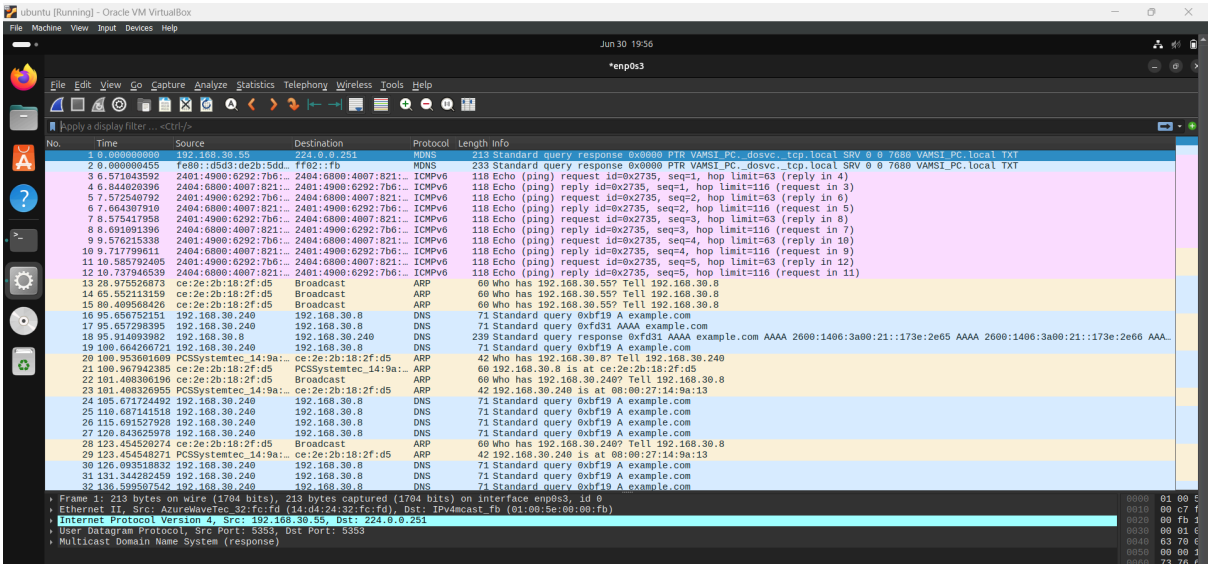
ping google.com -c 5
Then, to generate HTTP, HTTPS, and DNS traffic, a website was accessed using

curl https://example.com
This simulates real-world browsing activity and helps capture various protocol packets like DNS (for name resolution), TCP (for reliable transport), and HTTP/HTTPS (for web content).
Here, we can see that packets are sent .

```
vamsi@vamsi-VirtualBox:~$ ping google.com -c 5
PING google.com (2404:6800:4007:821::200e) 56 data bytes
64 bytes from maa05s26-in-x0e.1e100.net (2404:6800:4007:821::200e): icmp_seq=1 ttl=116 time=273 ms
64 bytes from maa05s26-in-x0e.1e100.net (2404:6800:4007:821::200e): icmp_seq=2 ttl=116 time=91.8 ms
64 bytes from maa05s26-in-x0e.1e100.net (2404:6800:4007:821::200e): icmp_seq=3 ttl=116 time=116 ms
64 bytes from maa05s26-in-x0e.1e100.net (2404:6800:4007:821::200e): icmp_seq=4 ttl=116 time=142 ms
64 bytes from maa05s26-in-x0e.1e100.net (2404:6800:4007:821::200e): icmp_seq=5 ttl=116 time=152 ms

--- google.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4013ms
rtt min/avg/max/mdev = 91.818/154.951/273.018/62.652 ms
vamsi@vamsi-VirtualBox:~$ curl https://example.com
```

After pressing the command in the terminal , these are the packets generated.

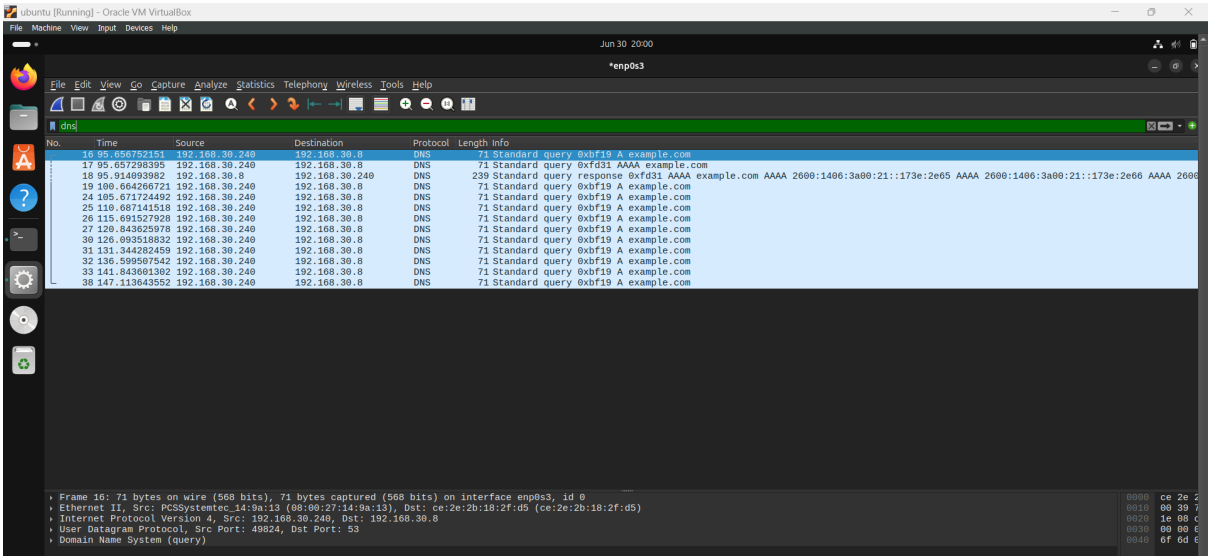


4).Stop capture after a minute.

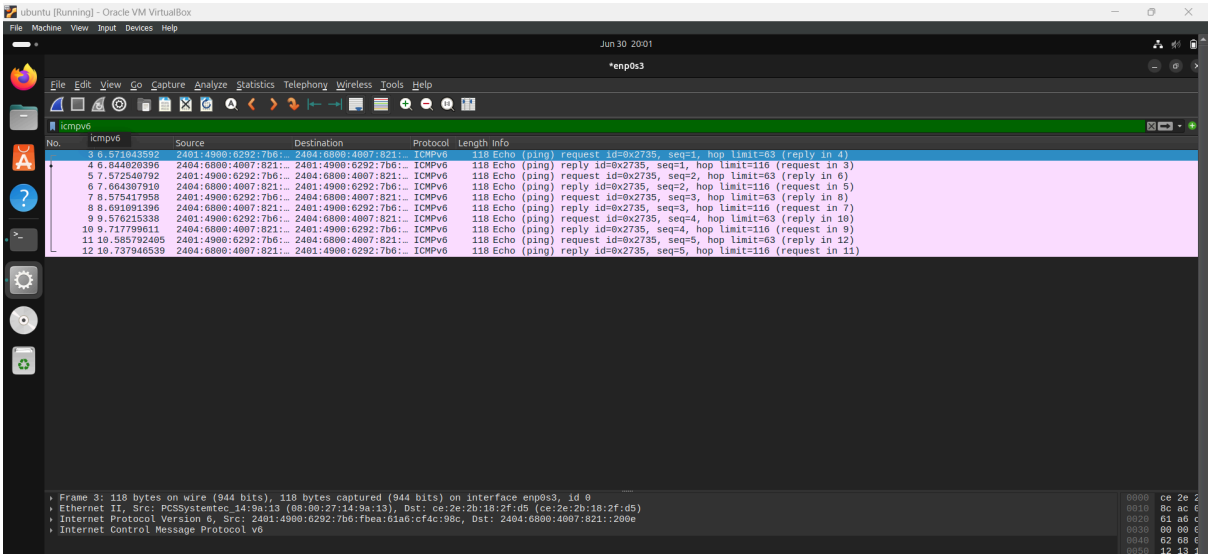
I stopped capturing the packets after a minute by clicking on the button.

5).Filter captured packets by protocol (e.g., HTTP, DNS, TCP).

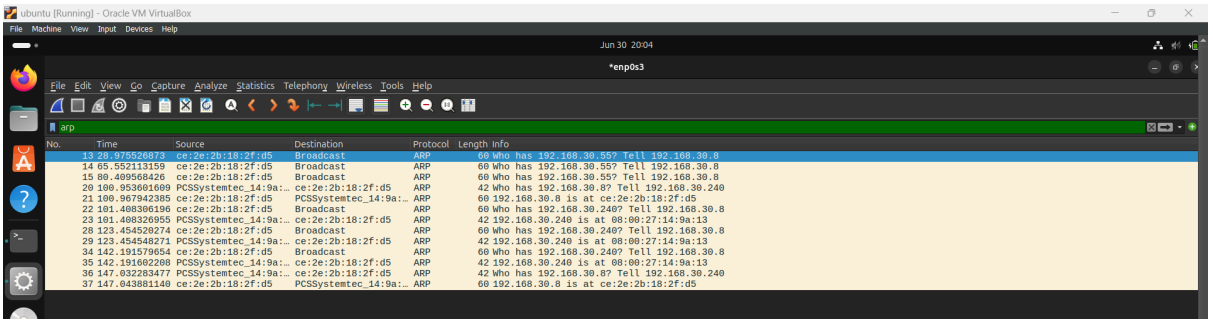
These are dns packets.



These are icmpv6 packets.



These are arp packets.



6).Identify at least 3 different protocols in the capture.

1. ICMP – Internet Control Message Protocol

- Purpose:
  Used primarily for network diagnostics and error reporting.

- Example:
  Executing `ping google.com` sends ICMP Echo Request packets and receives Echo Reply packets in return.

- Use Case:

    - Verifies if a host or device is reachable on the network.

    - Measures round-trip time (RTT) to diagnose latency or packet loss issues.

2. TCP – Transmission Control Protocol

- Purpose:
  Provides reliable, ordered, and error-checked delivery of data between applications.

- Example:
  Protocols such as HTTP, HTTPS, FTP, and SSH use TCP to ensure that data is transmitted correctly.

- Use Case:

    - Initiates a 3-way handshake (SYN → SYN-ACK → ACK) to establish a connection.

    - Guarantees that all data is delivered without loss or duplication, and in the correct order.

3. DNS – Domain Name System

  Purpose:
  Translates human-readable domain names into machine-readable IP addresses .

Example:
When accessing , your device sends a DNS query to resolve the domain name to its corresponding IP address.

Use Case:
Acts as the first step in any web browsing session.
Without DNS resolution, the browser cannot locate the server hosting the website.

7)Export the capture as a .pcap file.
I have completely exported the .pcap files.

8)Summarize your findings and packet details.

1. ICMP – Internet Control Message Protocol

- Purpose:
  Used for network diagnostics and error reporting.

- Typical Usage:
  Commonly utilized in tools like `ping` to check the reachability of a host and to measure latency (round-trip time).

- Key Functions:

  - Sends Echo Requests and receives Echo Replies.

  - Helps troubleshoot network issues like unreachable hosts or packet loss.

2. TCP – Transmission Control Protocol

- Purpose:
  A connection-oriented protocol designed to ensure reliable, ordered, and error-checked delivery of data.

- Typical Usage:
  Forms the backbone of major internet applications such as:

  - Web browsing: HTTP/HTTPS

    - File transfer: FTP
    - Email: SMTP/IMAP/POP3
- Key Features:
    - Establishes a reliable connection via the 3-way handshake process (SYN → SYN-ACK → ACK).
    - Guarantees that data arrives complete and in the correct sequence.

---

3. DNS – Domain Name System

- Purpose:
  Resolves human-readable domain names into machine-usable IP addresses.
- Typical Usage:
  DNS queries are essential before any website, application, or cloud service can be accessed.
- Key Details:
    - Operates over UDP (default) or TCP (for larger queries), using port 53.
    - Enables seamless web browsing and service accessibility by mapping domain names to IPs.