Report : Network Traffic Analysis using Wireshark

-Joshua Livingston R

**Overview :**

This report presents the results of a network traffic capture and analysis exercise using Wireshark. The objective was to observe and analyze different types of network protocols in real time, identify at least three distinct protocols, and understand their roles in typical internet activity.

**Table of Content:**

1. Introduction

2. Methodology

• Tools and Setup

• Procedure

3. Protocols Identified and Analyzed

• Domain Name System (DNS)

• Hypertext Transfer Protocol (HTTP)

• Internet Control Message Protocol (ICMP)

• Transmission Control Protocol (TCP)

4. Summary of Findings

5. Conclusion

1**. Introduction:**

This report presents the results of a network traffic capture and analysis exercise using Wireshark. The objective was to observe and analyze different types of network protocols in real time, identify at least three distinct protocols, and understand their roles in typical internet activity.

2. **Methodology:**

2.1 Tools and Setup:

• Tool: Wireshark (version [Insert Version]).

• Operating System: [e.g., Windows 10 / Ubuntu 22.04].

• Network Interface: [e.g., Wi-Fi (wlan0) or Ethernet (eth0)].

2.2 Procedure:

1. Wireshark was installed and launched.

2. A live capture session was started on the active network interface.

3. To generate traffic: o A web browser was used to visit https://example.com. o The ping command was run in the terminal to reach google.com.

4. The capture ran for approximately one minute before being stopped.

5. Filters were applied to isolate and analyze traffic by protocol.

6. The capture was saved as

network\_capture.pcap.

3. **Protocols Identified and Analyzed**:

3.1 Domain Name System (DNS):

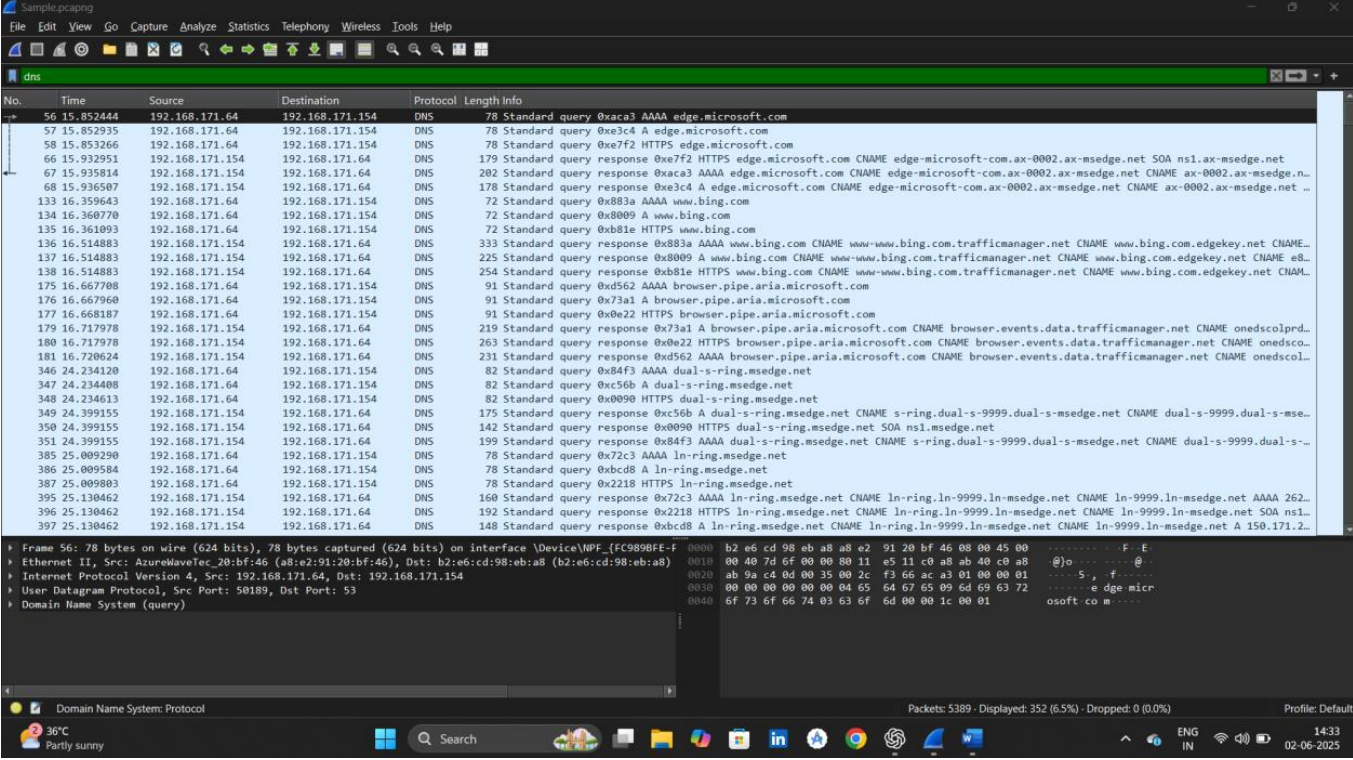
• Protocol Type: UDP

• Port: 53

• Purpose: Translates human-readable domain names (e.g., google.com) into IP addresses.

• Observation: Several DNS query packets were observed when browsing a website and pinging a domain. Each query had a corresponding response from a DNS server.

Example packet:



3.2 Hypertext Transfer Protocol (HTTP):

• Protocol Type: TCP

• Port: 80

• Purpose: Facilitates the transfer of web pages and content over the internet.

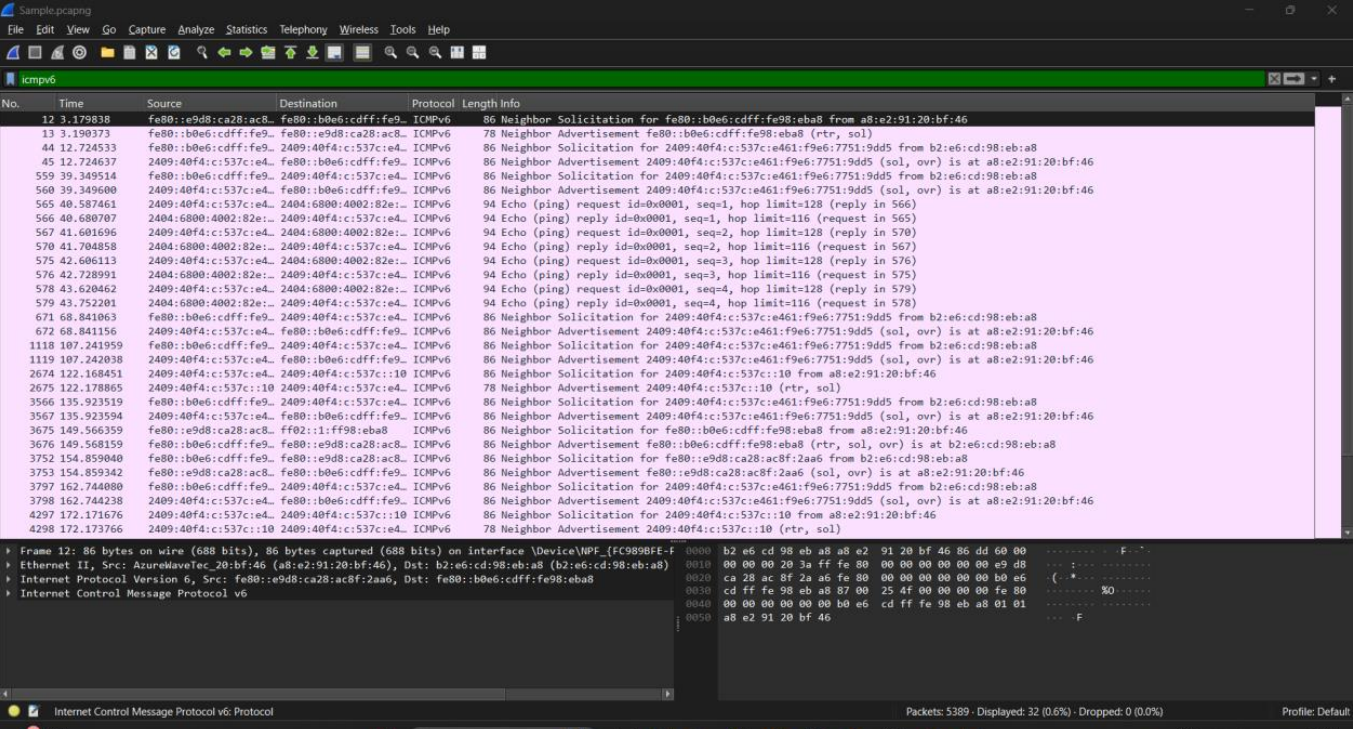
• Observation: HTTP GET requests were seen when visiting the example website. Response packets contained HTML content. Example Packet: Source IP: 192.168.1.5 Destination IP: 93.184.216.34 (example.com) Request: GET /index.html HTTP/1.1 Host: example.com.

3.3 Internet Control Message Protocol (ICMP):

• Protocol Type: ICMP

• Purpose: Used for diagnostics and testing, e.g., ping command.

• Observation: ICMP Echo Request and Echo Reply packets were generated during the ping google.com command.

Example Packet:

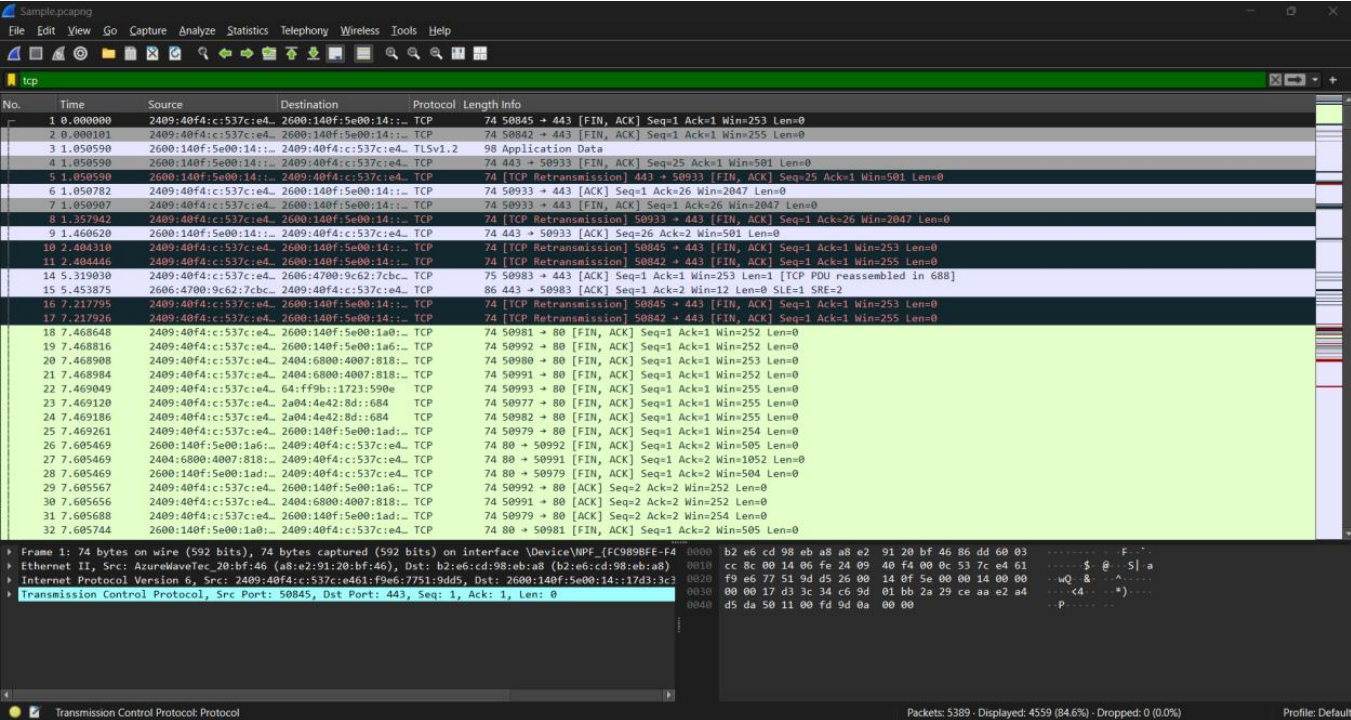
3.4 Transmission Control Protocol (TCP):

• Protocol Type: TCP

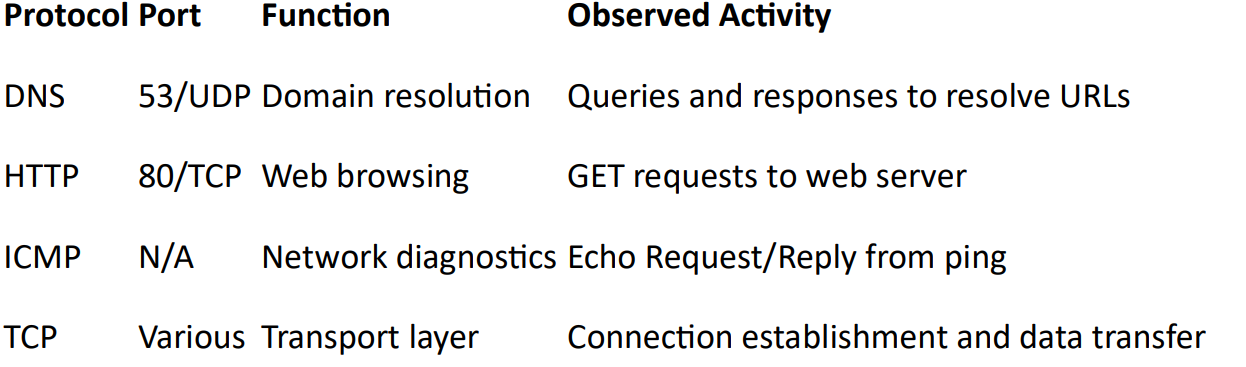
• Purpose: Reliable, ordered delivery of data.

• Observation: TCP packets showed connection establishment through a 3-way handshake and were used in HTTP communication.

Handshake Example:



**4. Summary of Findings**:



5. **Conclusion:**

This exercise successfully demonstrated the use of Wireshark to capture and analyze realtime network traffic. The capture showed typical protocol usage during everyday activities such as browsing a website and pinging a domain. DNS, HTTP, ICMP, and TCP protocols were identified and understood in context. The .pcap file has been saved and can be revisited for deeper analysis.