Wireshark Traffic Analysis Report

Internship: Cyber Security Internship
Task Number: 5
Name: Dhruvik Variya
Date: 30 June 2025
Tool Used: Wireshark

Objective

To perform live packet capture on an active network interface using Wireshark and analyze network traffic by identifying key protocols such as **ICMP**, **DNS**, and **TCP**. This task aims to improve hands-on skills in traffic analysis and protocol behavior observation.

Tools & Environment

- **Operating System:** Windows
- **Software:** Wireshark GUI (Dark Theme enabled)
- **Capture File Format:** `.pcapng`
- **Network Activity Simulated:**
  - `ping` to generate ICMP traffic
  - Browsing websites like YouTube and Google for DNS and TCP traffic

---

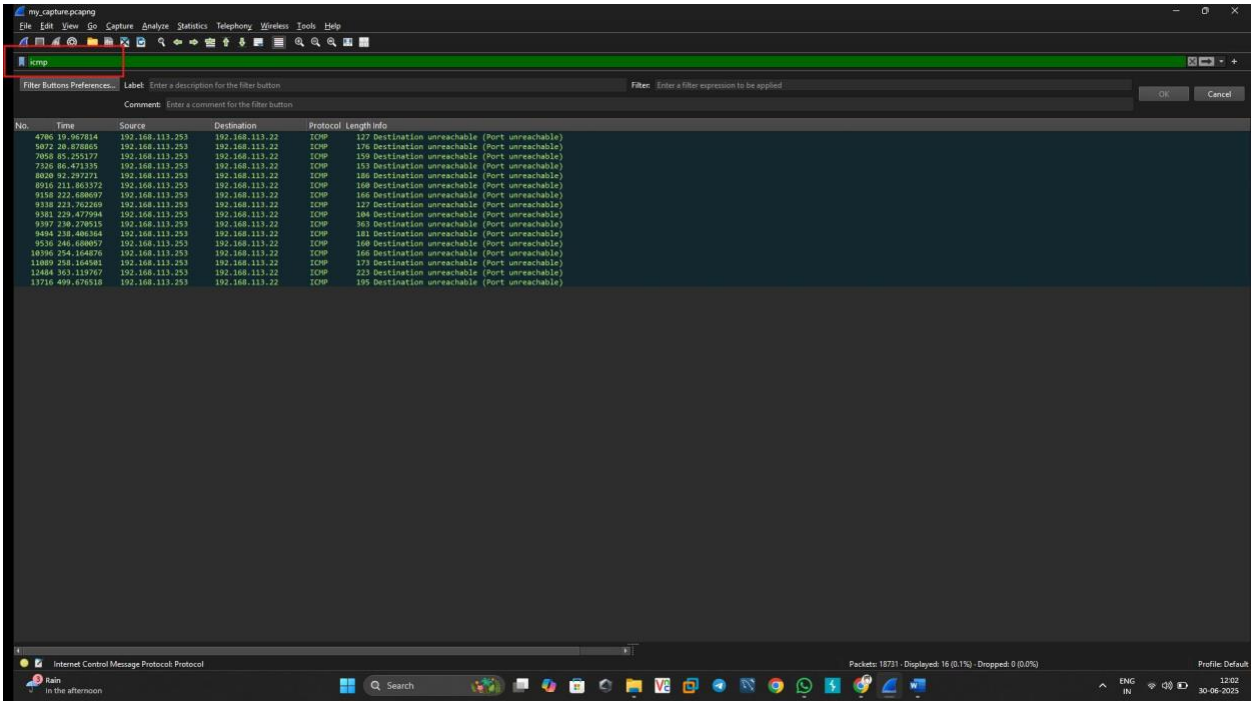
Protocol Analysis

---

1. ICMP – Internet Control Message Protocol

- **Wireshark Filter Used:** `icmp`
- **Captured Behavior:**
  - All ICMP packets show: `Destination Unreachable (Port unreachable)`
  - **Source IP:** `192.168.113.253`
  - **Destination IP:** `192.168.113.22`
- **Explanation:**
  - ICMP is used for diagnostics and error messages.

    - These ICMP packets likely resulted from trying to reach services on closed or blocked ports.
- **Use Case:**
    - Helps detect unreachable hosts or services
    - Common in `ping` and `traceroute` utilities
-

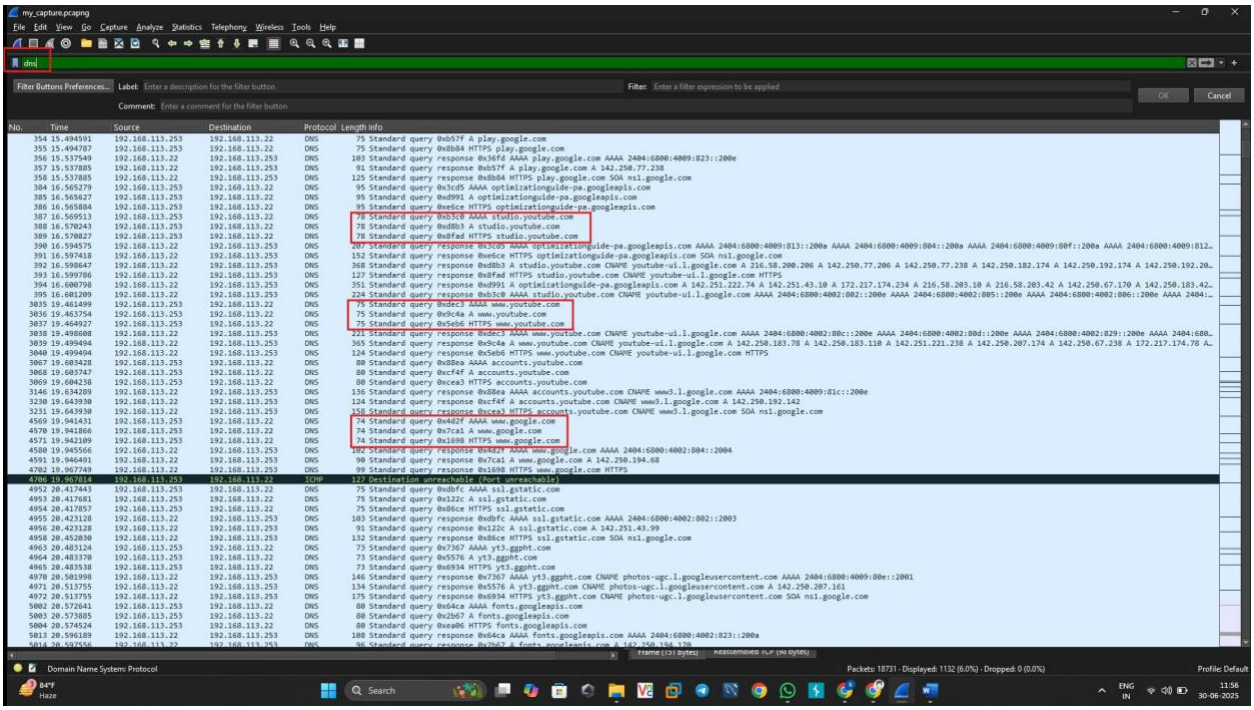


---

2. DNS – Domain Name System

- **Wireshark Filter Used:** `dns`
- **Captured Behavior:**
    - Standard query and response packets observed.
    - Query targets included:
        - `www.google.com`
        - `www.youtube.com`
        - `studio.youtube.com`
        - `optimizationguide-pa.googleapis.com`
    - Various **record types** captured:

    - `AAAA`: IPv6 Address
    - `CNAME`: Canonical Name (redirect)
    - `A`: IPv4 Address
- **Explanation:**
    - DNS is critical for resolving human-readable domains to IP addresses.
    - It's usually the first step in browsing activity.
- **Use Case:**
    - Essential for network forensics to track what websites or services are being accessed.
    - Attackers also exploit DNS for tunneling or exfiltration.



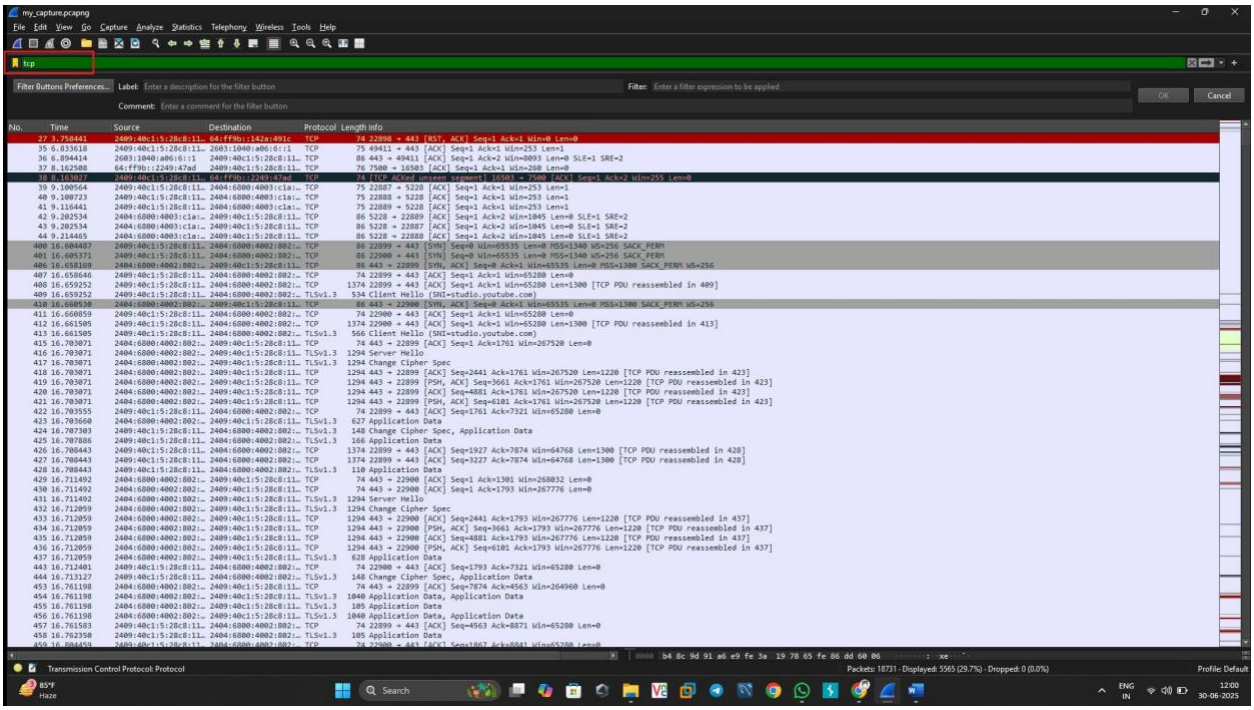
---

3. TCP – Transmission Control Protocol

- **Wireshark Filter Used:** `tcp`
- **Captured Behavior:**
    - TCP 3-way handshakes: `SYN`, `SYN-ACK`, `ACK`
    - Many connections observed on **port 443** (HTTPS)
    - Protocol stack includes:
        - TLSv1.3 handshakes

- Application Data (encrypted traffic)

- **Explanation:**
  - TCP ensures reliable delivery of data using acknowledgement and retransmission.
  - Encrypted communication is common on port 443.
- **Use Case:**
  - Useful for detecting secure traffic, session reassembly, and performance debugging.
  - In real-world forensics, TCP reassembly can reveal full HTTP content or downloads.



🗂 Files Submitted

- `my_capture.pcapng` — Captured packet file
- `README.md` — Protocol summary (for GitHub)
- Screenshots:
  - `icmp.png`
  - `dns.png`
  - `tcp.png`

☑ Conclusion & Learning Outcomes

Through this task, I gained:

- Practical experience with **live packet capturing**.
- Ability to **filter and isolate protocols**.
- Understanding of **how protocols behave in real time**.
- Insight into network diagnostics, layered protocol behavior, and encrypted vs. unencrypted traffic.
