**Open-Source Technologies**

**ON**

**PROJECT REPORT**

**INT 301**

**School of Computer Science & Engineering**

**LOVELY PROFESSIONAL UNIVERSITY**

**APRIL-2023**

****

**SUBMIT TO – DIPEN SAINI**

**NAME – HIMANSHU MISHRA**

**REGISTRATION NUMBER – 11907174**

**ROLL NUMBER – 48**

**Description -** Use any open-source software to generate report to capture packets from a network connection, trace connections, view the contents of suspect network transactions and identify bursts of network traffic.

**STEPS**

1. Open Wireshark and select the network interface you want to capture packets from.

2- Click on the "Capture" button to start capturing packets.

3- Let Wireshark capture packets for a desired period.

4- Once you have captured enough packets, stop the capture by clicking on the "Stop" button.

5- In the Wireshark interface, click on "Statistics" and select the type of report you want to generate.

6- For example, you can select "Protocol Hierarchy" to see a breakdown of the protocols used in the captured packets.

7- You can also use the "IO Graph" feature in Wireshark to identify bursts of network traffic.

8- To do this, select "Statistics" and then "IO Graph." In the IO Graph window, you can select the time range you want to analyse, and Wireshark will generate a graph of the traffic during that time.

9- To view the contents of a suspect network transaction, select the packet in Wireshark and click on "Packet Details.

10- " This will show you the contents of the packet, including any data that was transmitted.

11- You can also use the "Follow TCP Stream" feature in Wireshark to view the contents of a 12- TCP stream.

13- To do this, select a TCP packet in Wireshark and click on "Follow TCP Stream." This will show you the entire conversation between the two endpoints, including any data that was transmitted.

14- Once you have generated the desired reports and analysed the captured packets, you can save the capture file for future reference by selecting "File" and then "Save."

**Introduction**

**1.1 Objective of the project**

The objective of this project is to analyse network traffic using Wireshark to identify suspect transactions and bursts of traffic.

This analysis can help network administrators and security professionals detect and prevent malicious activity on the network.

**1.2 Description of the project**

The project involves capturing network traffic using Wireshark and analysing the packets to identify suspicious behaviour.

The analysis will include identifying unusual protocols, payloads, and patterns of network activity. The results of the analysis will be used to develop strategies to prevent and mitigate potential security threats.

**1.3 Scope of the project**

The scope of the project is limited to analysing network traffic captured in a controlled environment.

The project does not include any real-time monitoring or detection of network threats.

**System Description**

**2.1 Target system description**

The target system for this project is a network environment with multiple devices, including routers, switches, and servers.

The network traffic will be captured using Wireshark on a dedicated machine.

**2.2 Assumptions and Dependencies (If applicable)**

The project assumes that the network traffic captured in the controlled environment is representative of the traffic patterns in a real-world network environment.

The project also assumes that the Wireshark software is functioning correctly and that the captured packets are not corrupted or incomplete.

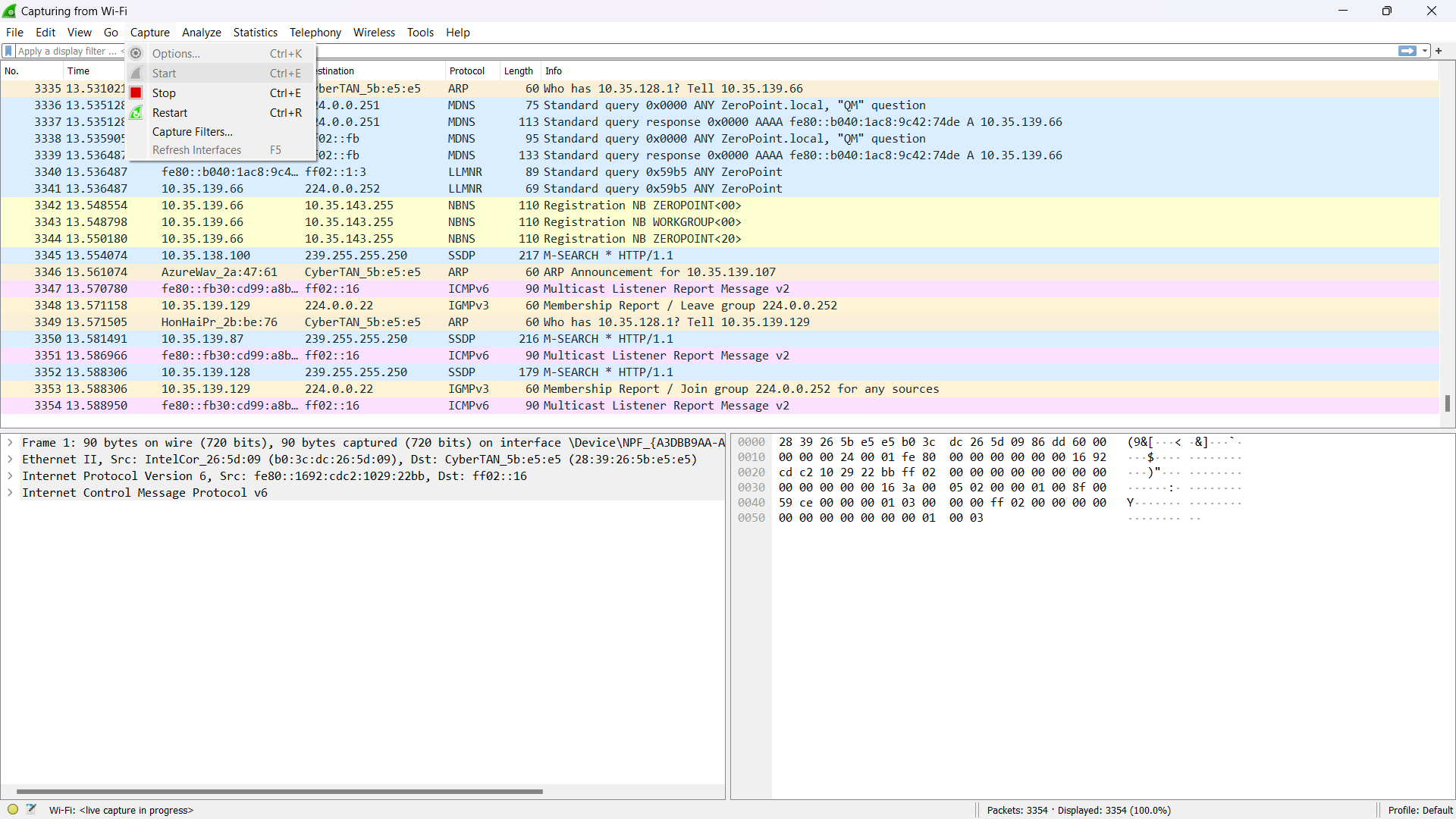
**2.3 Functional/Non-Functional Dependencies (if any)**

The project requires a machine capable of running Wireshark and capturing network traffic.

The machine must also have enough storage capacity to store the captured packets for analysis.

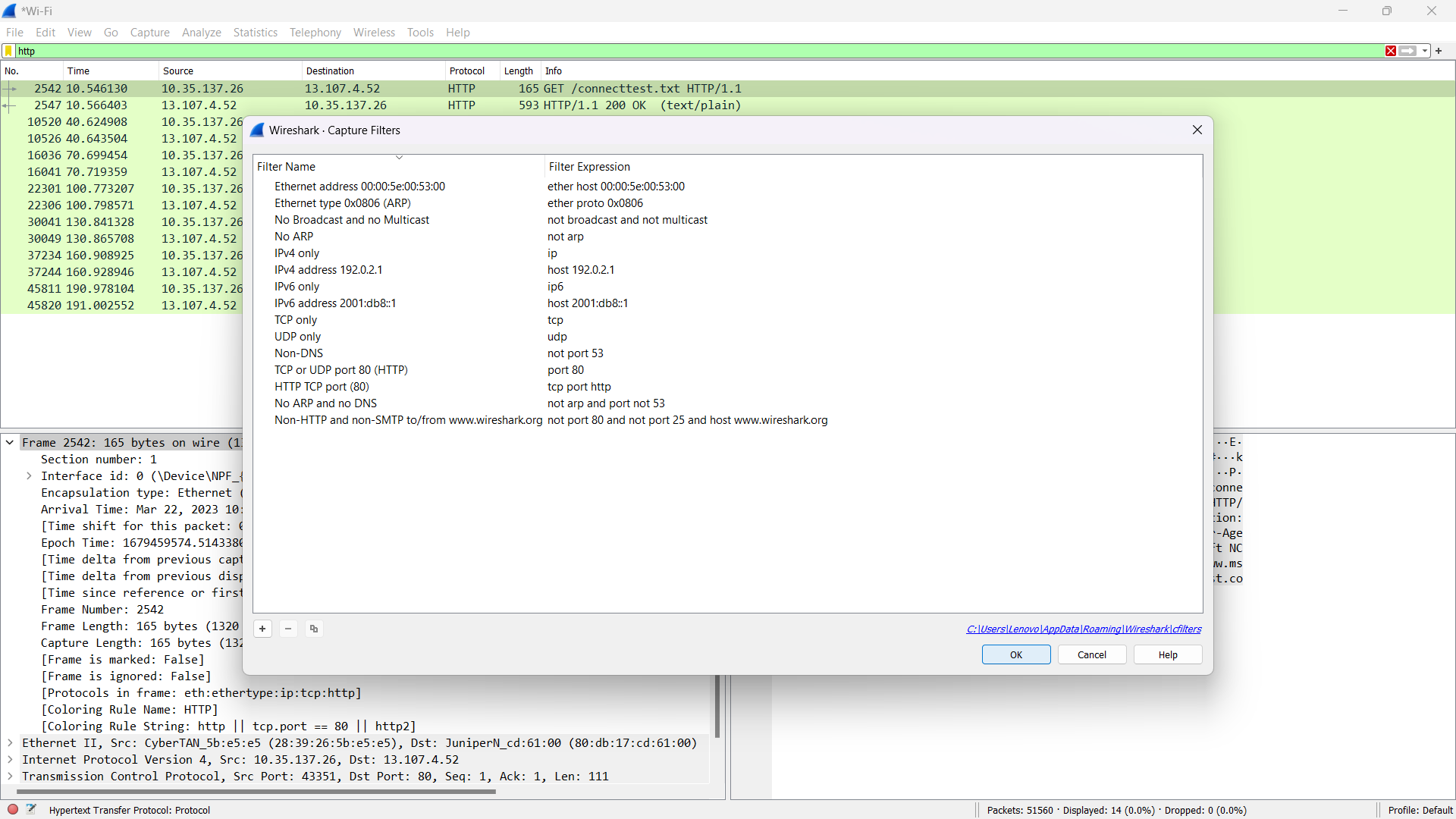
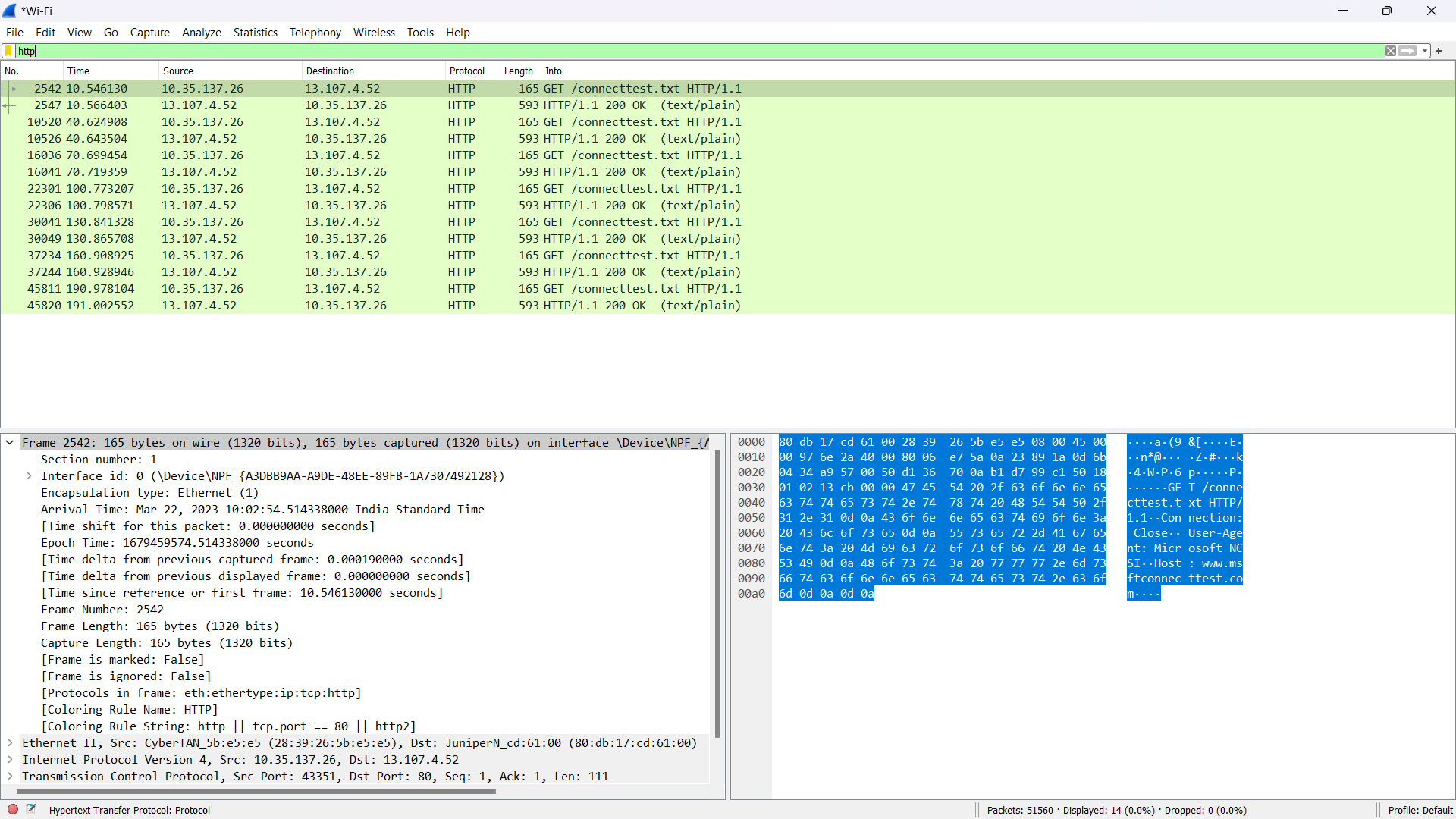
**Analysis Report**

**3.1 System snapshots and full analysis report**



Once the network interface is selected, you simply click the Start button to begin your capture. As the capture begins, it’s possible to view the packets that appear on the screen.

Graphical user interface

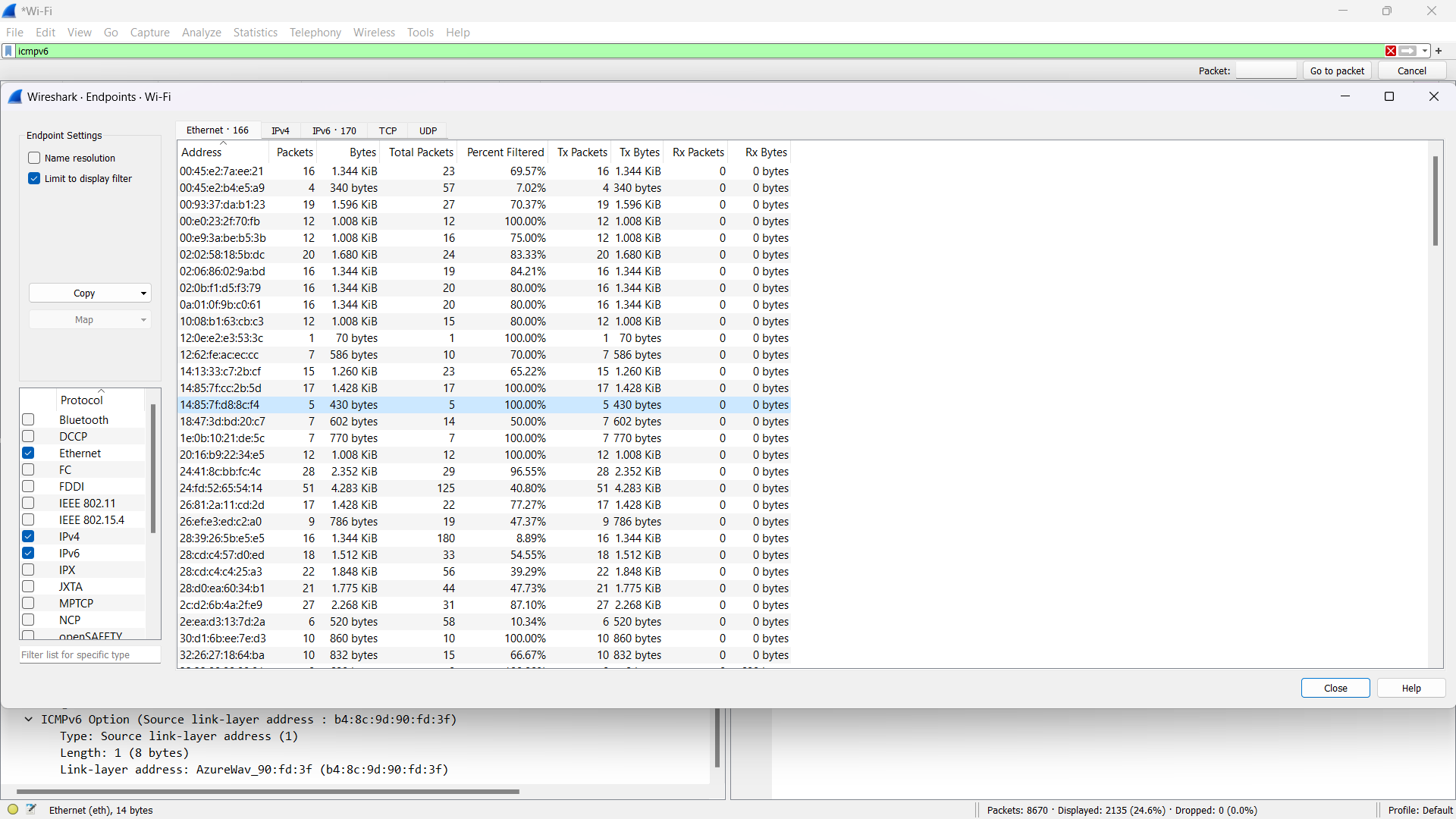
Description automatically generated

Wireshark capturing packets.

Graphical user interface, application

Description automatically generated

In this case, three major traffic bursts were generated. Many times, cybersecurity pros use Wireshark as a quick and dirty way to identify traffic bursts during attacks.



Viewing endpoint conversation in Wireshark

**Reference/ Bibliography**

Beale, J., & Jones, J. (2018). Practical Packet Analysis: Using Wireshark to Solve Real-World Network Problems. No Starch Press.

Wireshark User Guide. (n.d.). Retrieved from https://www.wireshark.org/docs/wsug\_html/