NETWORK TRAFFIC ANALYSIS

Table of Contents :

Introduction :

Network traffic analysis is a crucial task for understanding network behavior, identifying potential security threats, and optimizing network performance. This project aims to develop a tool that captures network packets, analyzes their characteristics, and visualizes the traffic patterns. The tool allows users to filter packets based on various criteria and provides insights into the network traffic.

Methodology :

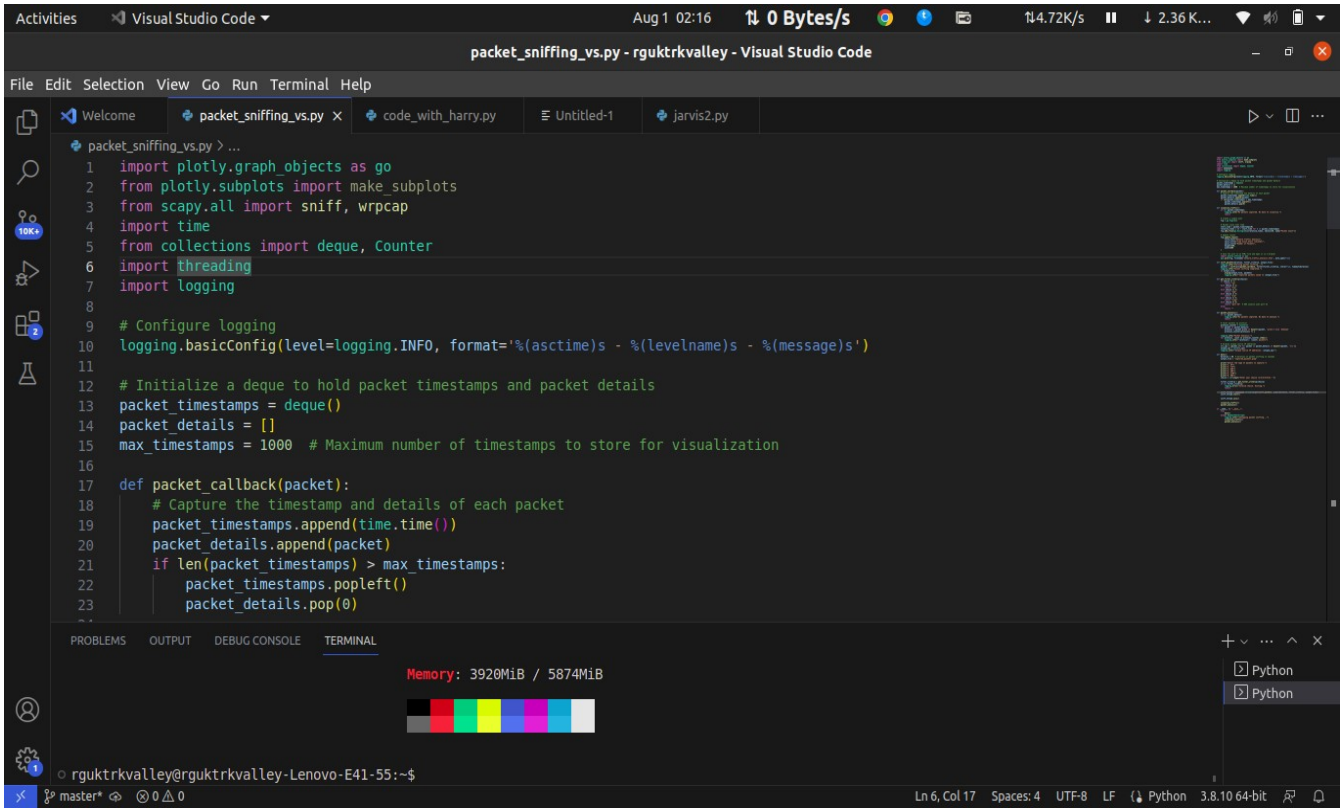
The project utilizes the following libraries and techniques:

1. **Scapy**: A powerful Python library for network packet manipulation and analysis. It is used for capturing and filtering network packets.
2. **Plotly**: A data visualization library for creating interactive and customizable plots. It is used to visualize the captured network traffic.
3. **Deque and Counter**: Data structures from the `collection` module used for efficient storage and analysis of packet timestamps and details.
4. **Multithreading**: Allows concurrent execution of packet sniffing and visualization to provide a responsive user experience.
5. **Logging**: Enables structured logging for informative output and error handling during the execution of the program.

* Sample Code:

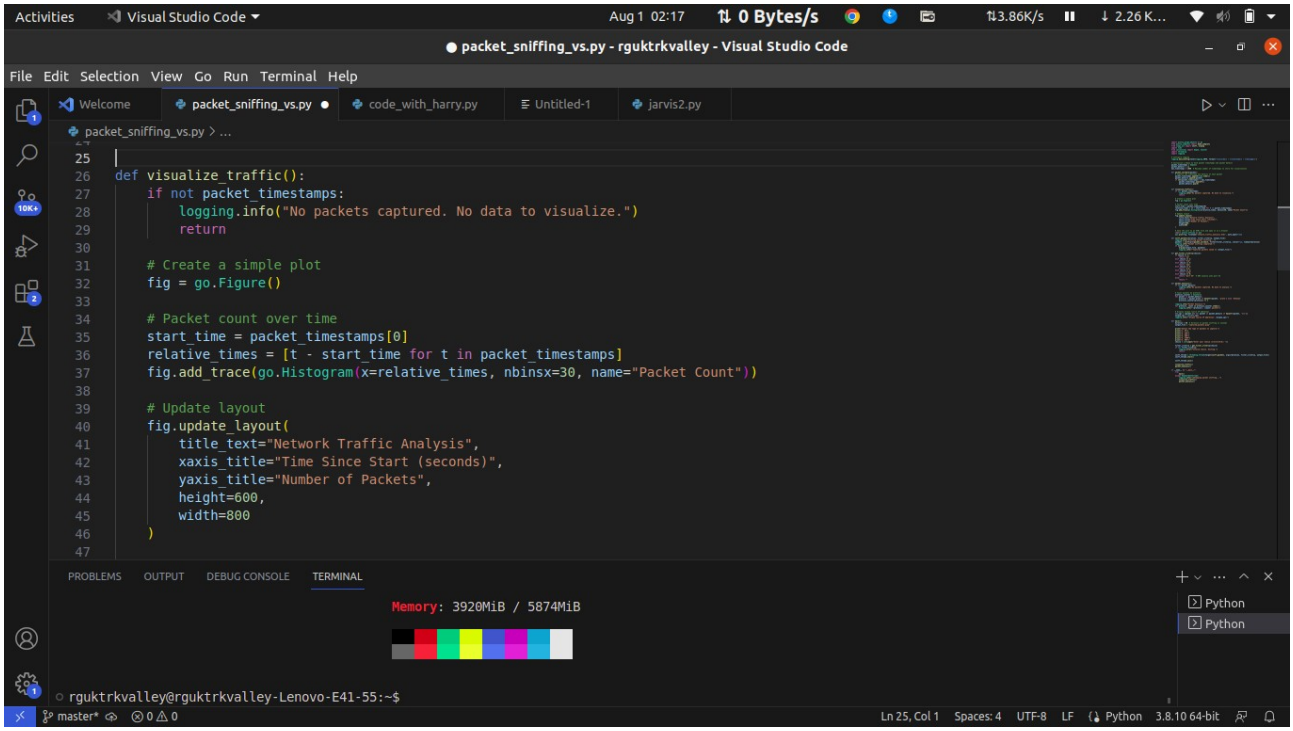
· Packet Sniffing :

The `sniff_packets` function captures packets based on user-defined filters and saves them to a file. The `packet_callback` function is used to process each packet, capturing its timestamp and details.



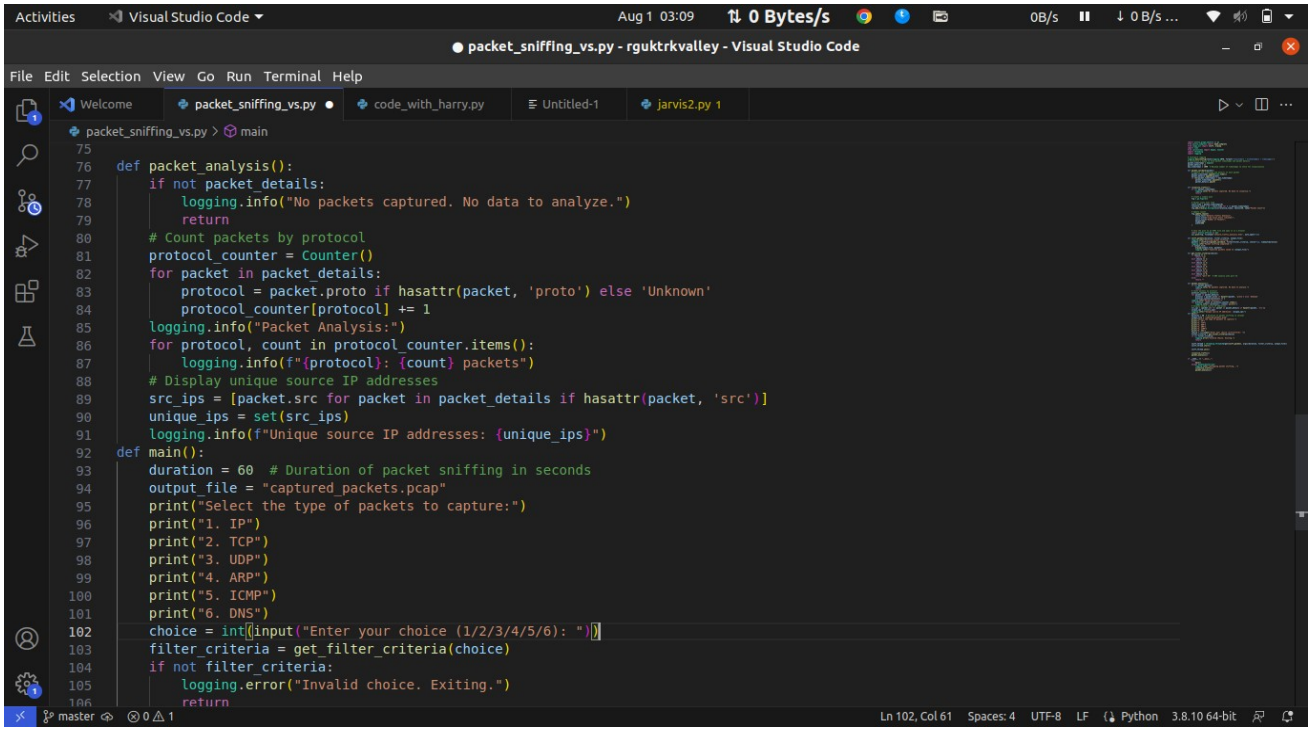
· Visualization :

The `visualize_traffic` function creates a histogram of packet counts over time using Plotly. This visual representation helps in understanding the traffic pattern.

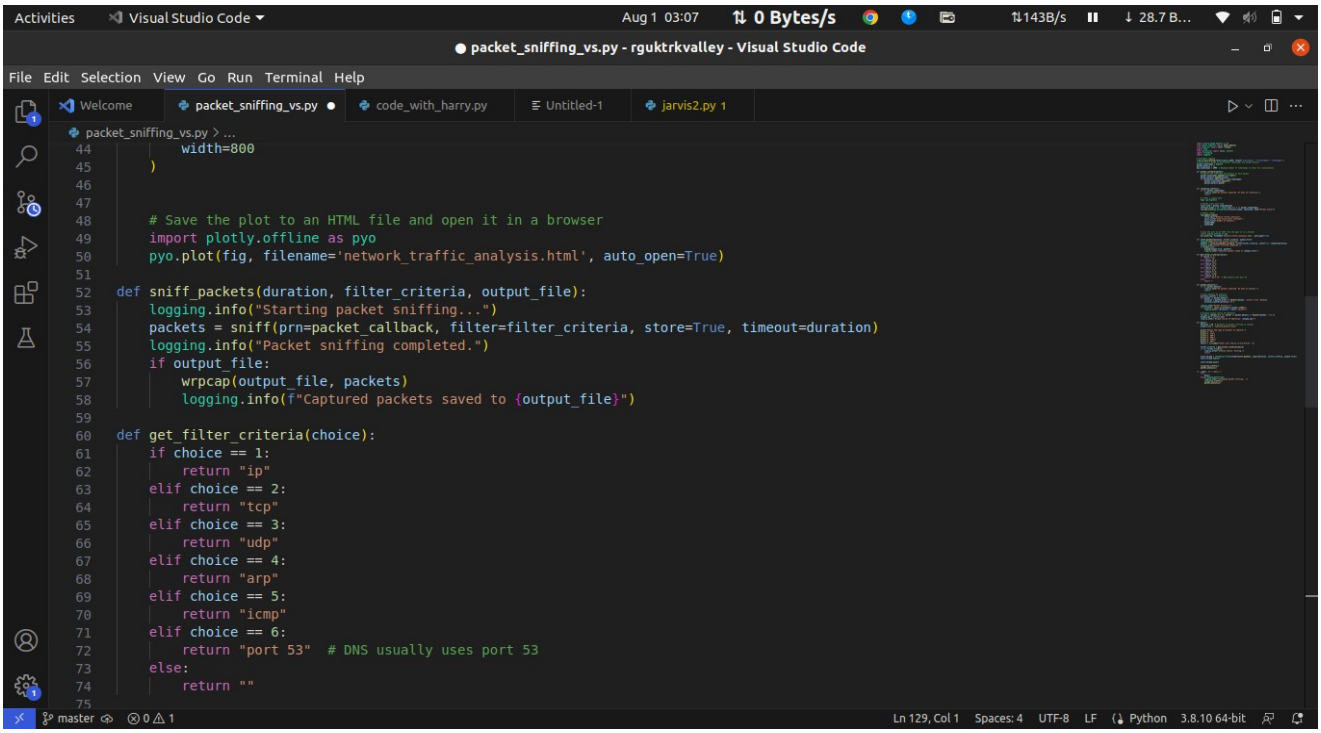


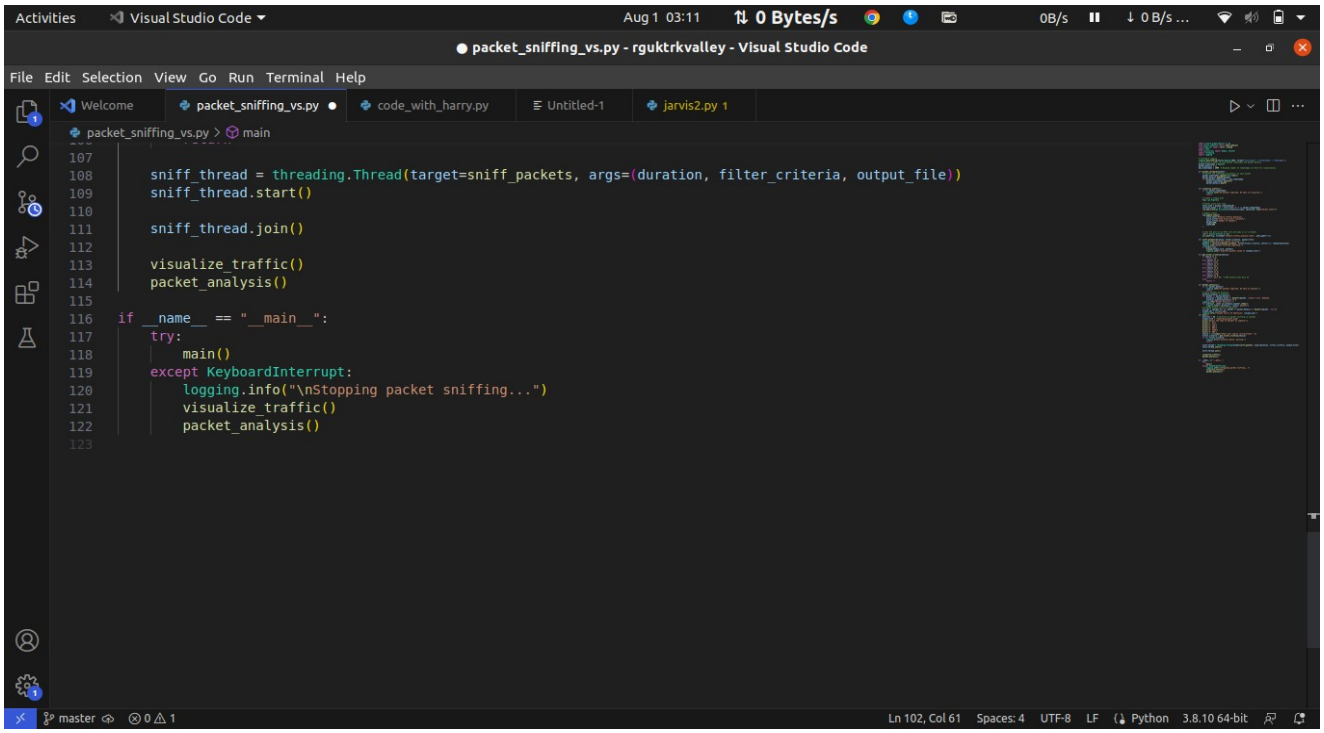
· **Packet Analysis :**

The `packet_analysis` function performs a basic analysis of the captured packets, counting packets by protocol and listing unique source IP addresses.



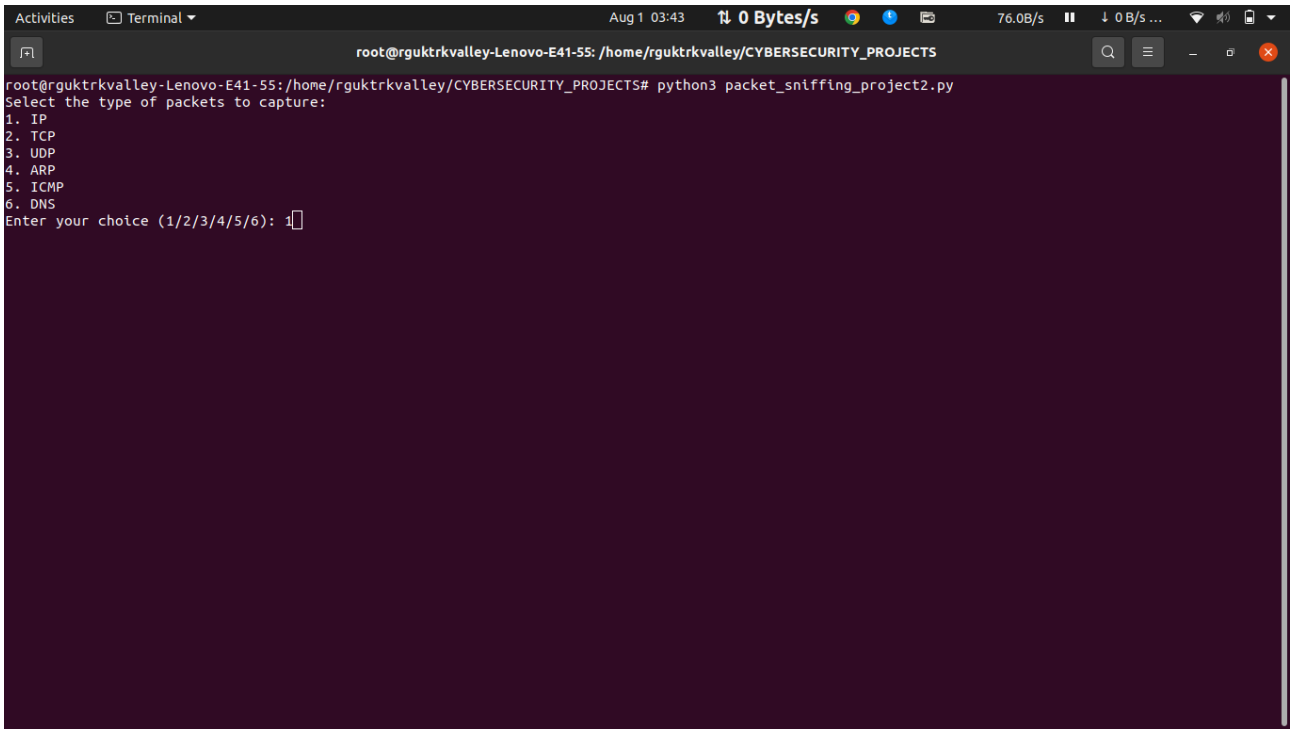
*** Select the Packet Type**: When prompted, choose the type of packets you want to capture (e.g., IP, TCP, UDP, ARP, ICMP, DNS).



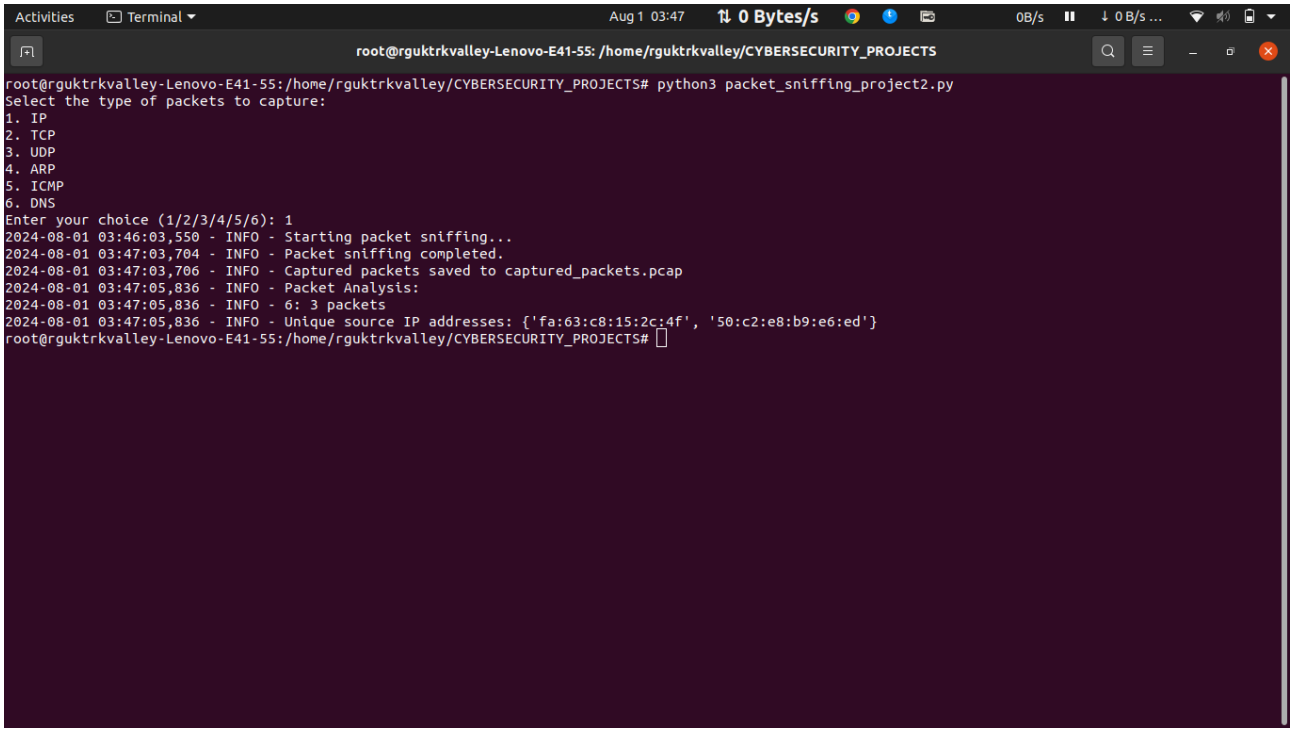


* Execution :

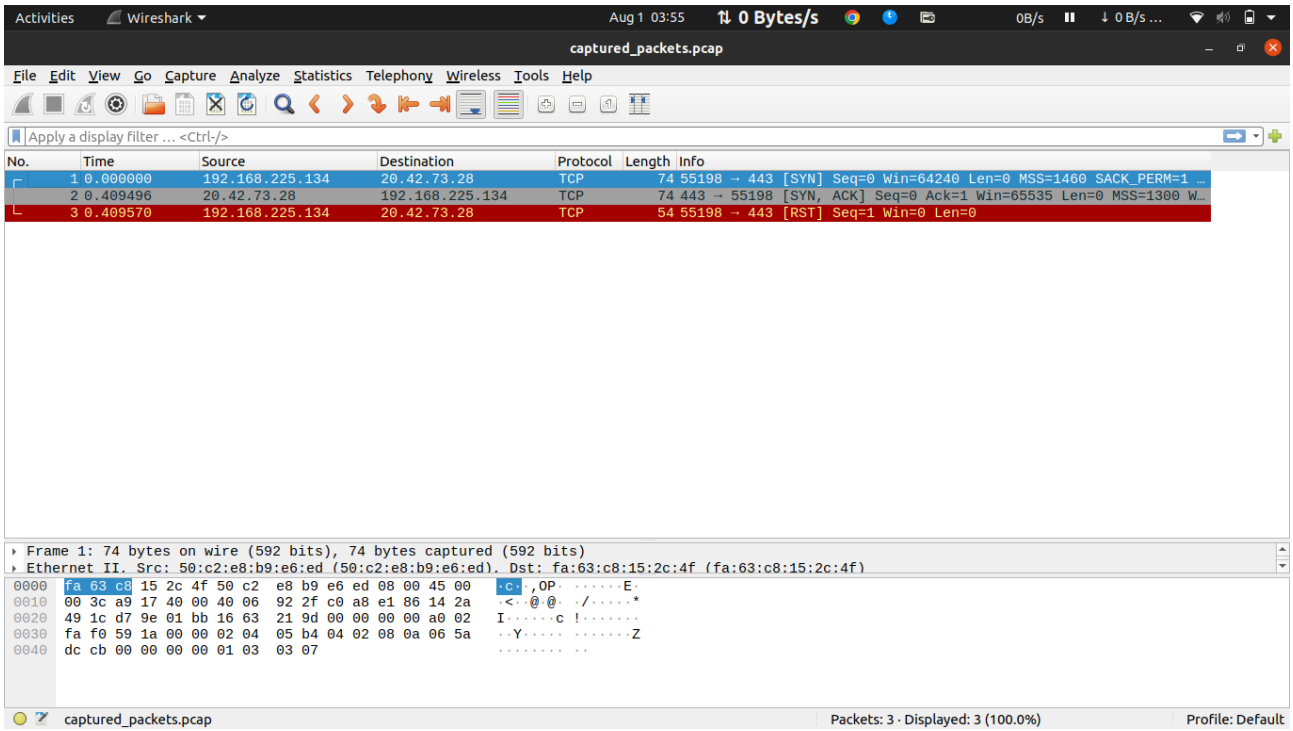
* When the code is executed we will need to choose the type of packets we need to capture .



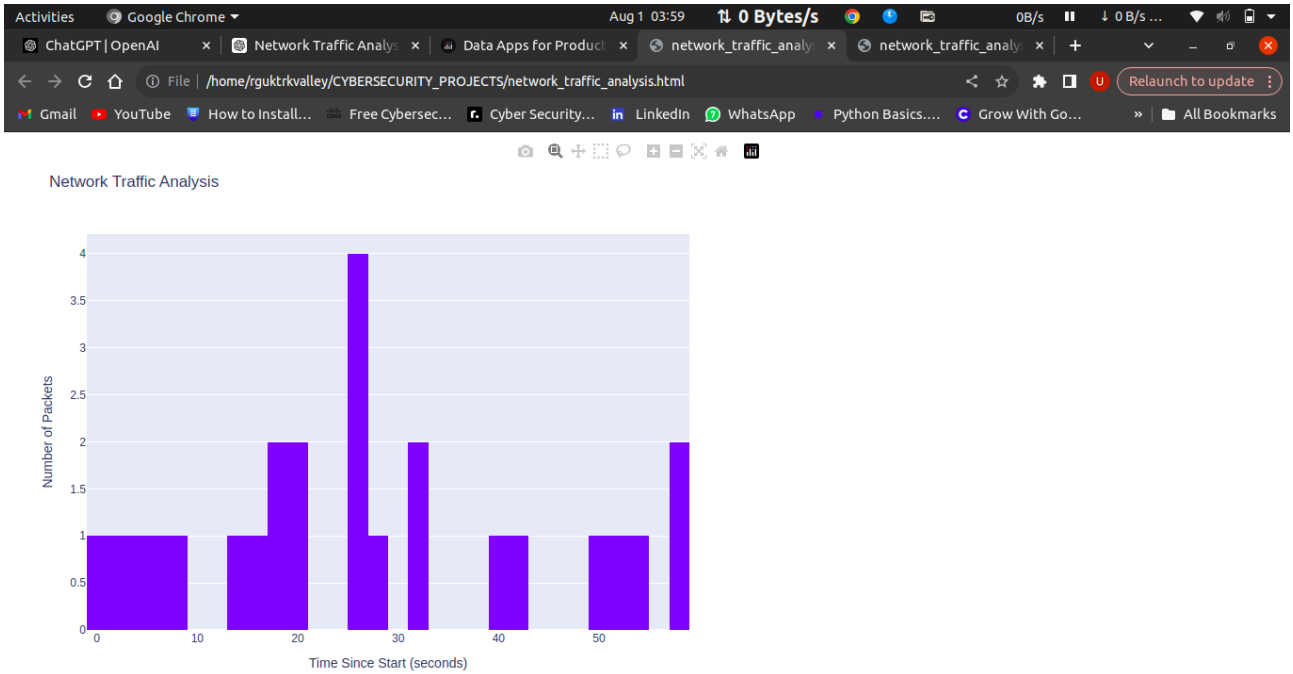
* In the above example I have taken the input as “1” where we will collect the packets of “IP”.



* As we can see the packets that are captured will be stored in “captured_packets.pcap”(The file can we viewed in Wireshark).



* The Visualization of the packets can be seen in the “network_traffic_analysis.html” file .



* Similarly we can capture the packets of all the other protocols giving their respective input.

* Conclusion :

This project provides a basic framework for capturing, visualizing, and analyzing network traffic. Future improvements could include real-time visualization updates, more advanced packet analysis features, and integration with other data sources.

* References :

- Plotly Documentation: https://plotly.com/python/
- Scapy Documentation: https://scapy.readthedocs.io/