

**Name:** Bint e Fatima Niazi

**Date:** July 2025

**Contact: -**

-Email: fatimabinte4@gmail.com

-LinkedIn: linkedin.com/in/bint-e-fatima-niazi

-GitHub: github.com/fatima-niazi

Project: Network Packet Analysis using Wireshark

# Introduction

This document outlines the project for practical cybersecurity training. The focus was on capturing and analyzing real network traffic using Wireshark. Live packet inspection, protocol dissection, filtering, and analysis of traffic such as HTTP, DNS, TCP, and ICMP were performed.

# Tools & Environment

- Wireshark  
- Kali Linux   
- Terminal tools: ping, nslookup  
- Browser

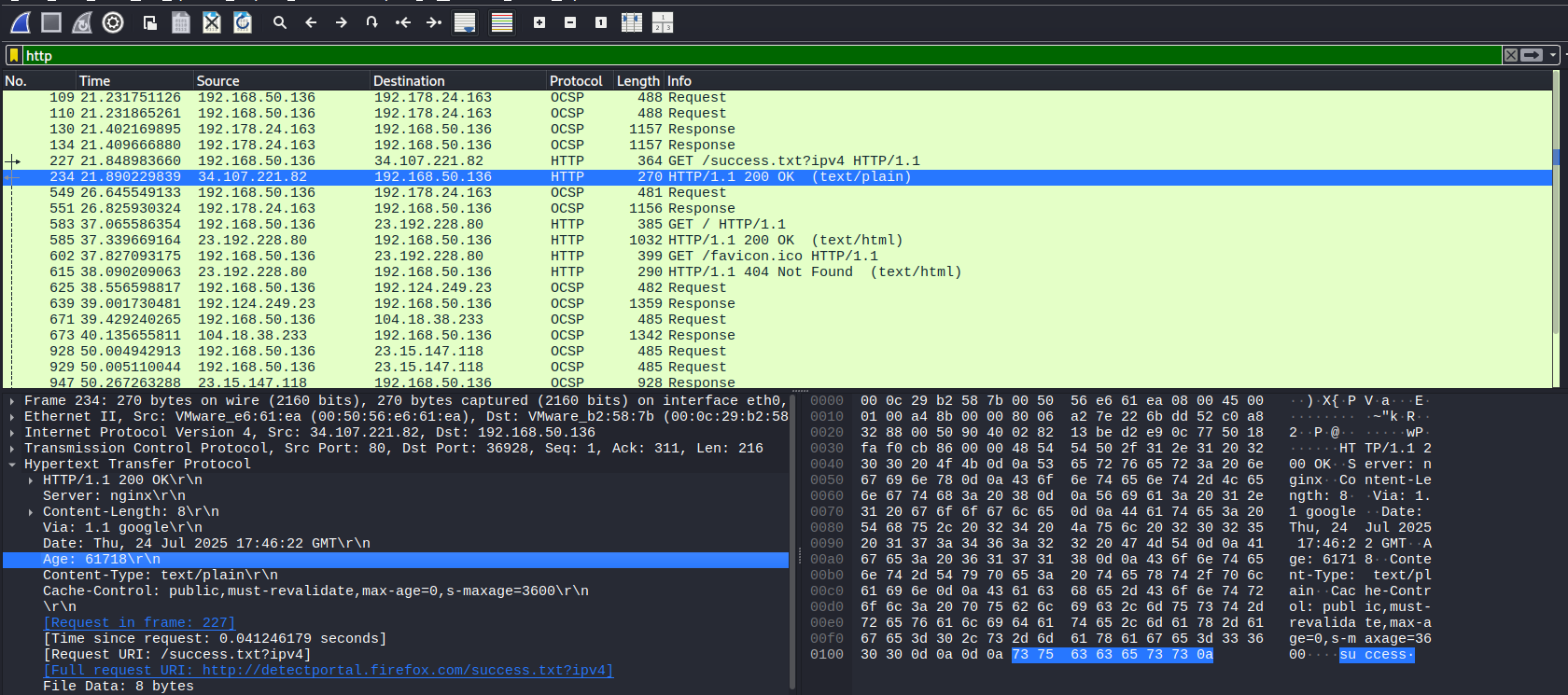
# Packet Capture Process

Wireshark was launched and the active network interface (wlan0 or eth0) was selected. Live capture was started while browsing websites, pinging, and performing DNS lookups to simulate real network activity. Capture was saved as `capture.pcapng` for analysis.

# Filters Applied & Observations

* Filter: http

Captured HTTP requests and headers such as Host, User-Agent, etc.



* Filter: dns

Monitored DNS queries and responses for domain name resolution.

A screenshot of a computer

AI-generated content may be incorrect.

* Filter: icmp

Captured ping traffic, including Echo Request and Echo Reply.

A screenshot of a computer

AI-generated content may be incorrect.

* Filter: ip.addr == YOUR\_IP

Isolated all traffic to and from the system's IP address.

A screenshot of a computer

AI-generated content may be incorrect.

* Filter: tcp.flags.syn == 1 && tcp.flags.ack == 0

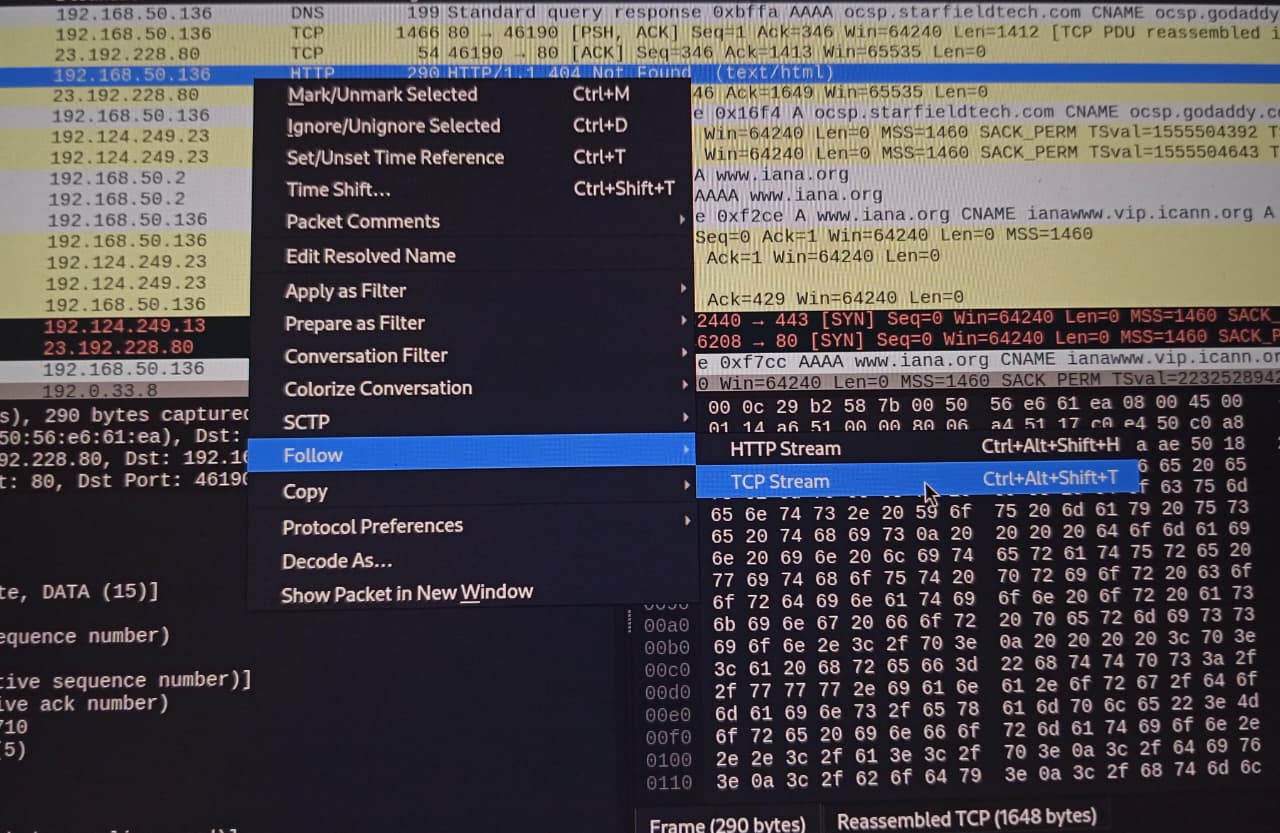
Observed SYN packets used in the TCP handshake initiation.

A screenshot of a computer

AI-generated content may be incorrect.

# Packet Analysis Summary

Multiple protocols were analyzed in-depth. DNS queries showed how domain names are resolved to IPs. HTTP GET requests revealed web traffic details like URLs and headers. TCP handshakes were dissected to understand how connections form. ICMP packets validated ping traffic and its round-trip time. TCP streams were followed to reconstruct full HTTP conversations.



A screenshot of a computer

AI-generated content may be incorrect.

# Conclusion

This project introduced practical network analysis using Wireshark. It developed hands-on skills in monitoring live traffic, identifying common protocols, and analyzing the structure of network communication. These are essential capabilities for SOC analysts, network defenders, and cybersecurity learners.