NETWORK TRAFFIC ANALYSIS USING WIRESHARK AND ZEEK

Submitted by : Siya

Course : Cyber Security
College : Mody University Of Science and Technology
Date : 14 July, 2025
Supervisor : Rushikesh Dinkar

ABSTRACT

What problem does this project solves ?

This project focuses on identifying security threats and unusual activities in network traffic. The problem it solves is that many cyberattacks and network issues go unnoticed because traditional tools either look too narrowly or don’t provide enough detail.

HOW ?

This project addresses the problem of detecting malicious activities, performance bottlenecks, and anomalies within network traffic by leveraging packet-level and flow-level analysis. Traditional security measures often struggle with identifying subtle threats hidden within legitimate traffic or providing comprehensive visibility into network behavior.

HOW IS IT SOLVED?

- To tackle this issue, we implemented a dual-framework approach utilizing **Wireshark** for deep packet inspection and **Zeek** for real-time network monitoring and log-based analysis.
- Wireshark was employed to capture and inspect individual packet data, enabling detailed examination of packet headers and payloads for suspicious patterns.
- Concurrently, Zeek was deployed to passively monitor network traffic and generate comprehensive logs, which were subsequently analyzed for anomalies such as unusual connection attempts, protocol misuse, and potential indicators of compromise.

KEY RESULT OF FINDING FROM PROJECT

- The key result showed that using both tools together was much more effective than using either one alone.
- We were able to detect hidden issues like strange DNS requests and unauthorized access attempts that might have been missed otherwise.
- This approach improves the way networks can be monitored and secured.

INTRODUCTION

We chose this project because in today's world, most systems and businesses depend on networks for communication and data transfer. Cyberattacks, hacking, and data leaks often happen without anyone noticing until it's too late. Network traffic analysis is important because it helps detect these threats early and ensures smooth and secure network operations. The problem we're solving is the difficulty in spotting hidden cyber threats and network issues using normal security tools.

This project is about analyzing network traffic to detect unusual or suspicious activities using two powerful tools: **Wireshark** and **Zeek**. In simple words, it means monitoring the data that travels across a network to check for problems, attacks, or anything abnormal that might harm the system.

Wireshark :

1. Network protocol analyzer
2. Captures and inspects packets

Zeek :

1. Network analysis framework
2. Generates logs and detects patterns/ anomalies

OBJECTIVE

- Capture and analyze network traffic in a controlled lab environment using tools like Wireshark and Zeek (formerly Bro). Focus on detecting suspicious patterns, protocol anomalies, and potential intrusions. Create a report on detected threats and explain how they can be mitigated.
-
-

Analyze for :

(1) Suspicious patterns

(2) Protocol anomalies

(3) Potential intrusions

METHODOLOGY: ENVIRONMENT SETUP

- Operating System : [Windows 11/Linux Kali]
  Wireshark: Download and install Wireshark from https://www.wireshark.org/download.html.
- Web Browser: Any modern web browser (e.g., Chrome, Firefox) for generating HTTP traffic.

METHODOLOGY: TRAFFIC CAPTURE

- Using Wireshark :
  selected active network interface
  capture traffic for :
  - HTTP browsing
  - HTTPS access
  - FTP login attempts
  - ICMP requests

ANALYZING HTTP TRAFFIC WITH WIRESHARK

Introduction

- In this project, we learnt how to use Wireshark to capture and analyze HTTP traffic.
- HTTP traffic analysis is crucial for understanding web communication, identifying potential security issues, and investigating anomalies in network traffic.

PREREQUISITES

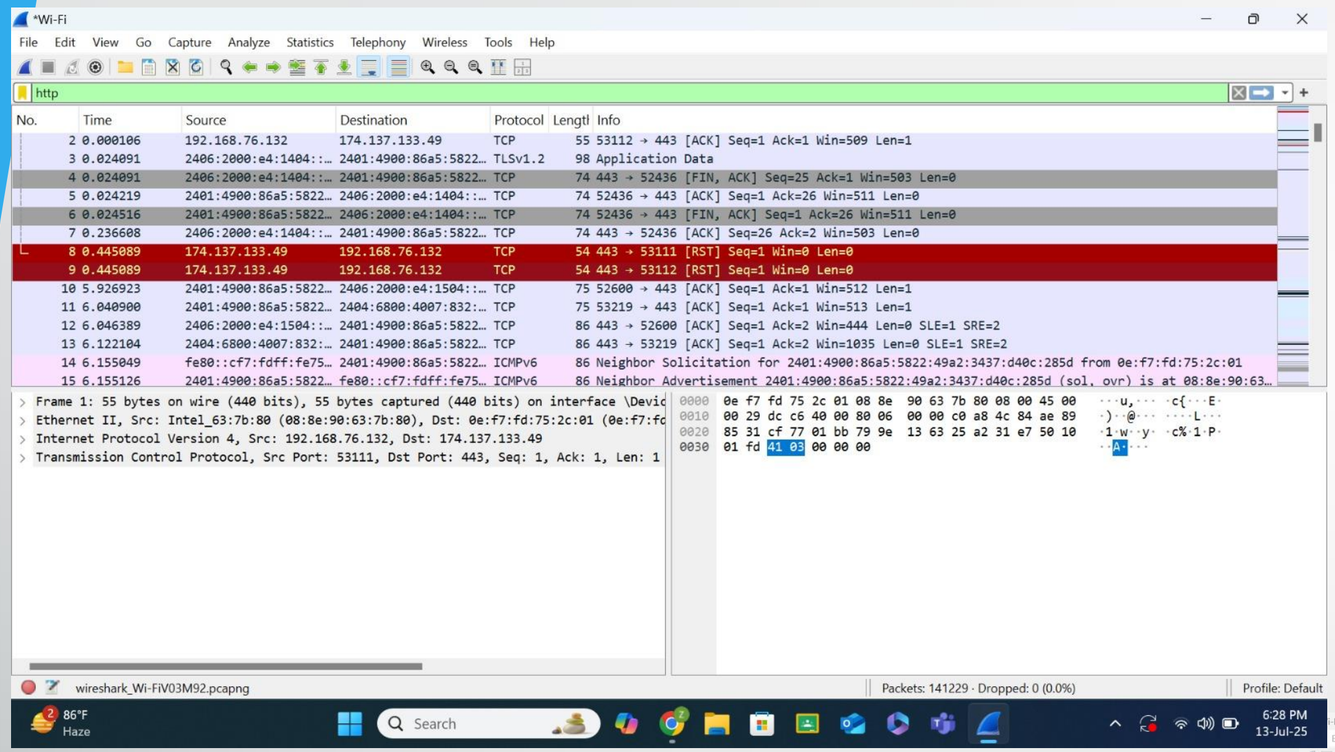
- Basic understanding of networking concepts
- Wireshark installed on your computer
- A web browser for generating HTTP traffic
- Lab Set-up and Tools
- Wireshark: Download and install Wireshark from https://www.wireshark.org/download.html.
- Web Browser: Any modern web browser (e.g., Chrome, Firefox) for generating HTTP traffic.

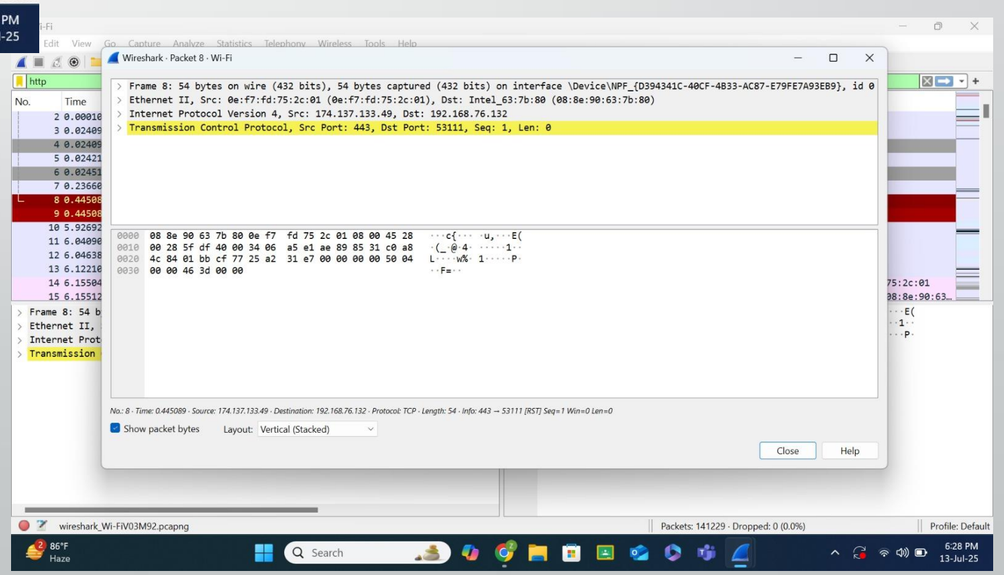
STEPS INVOLVED

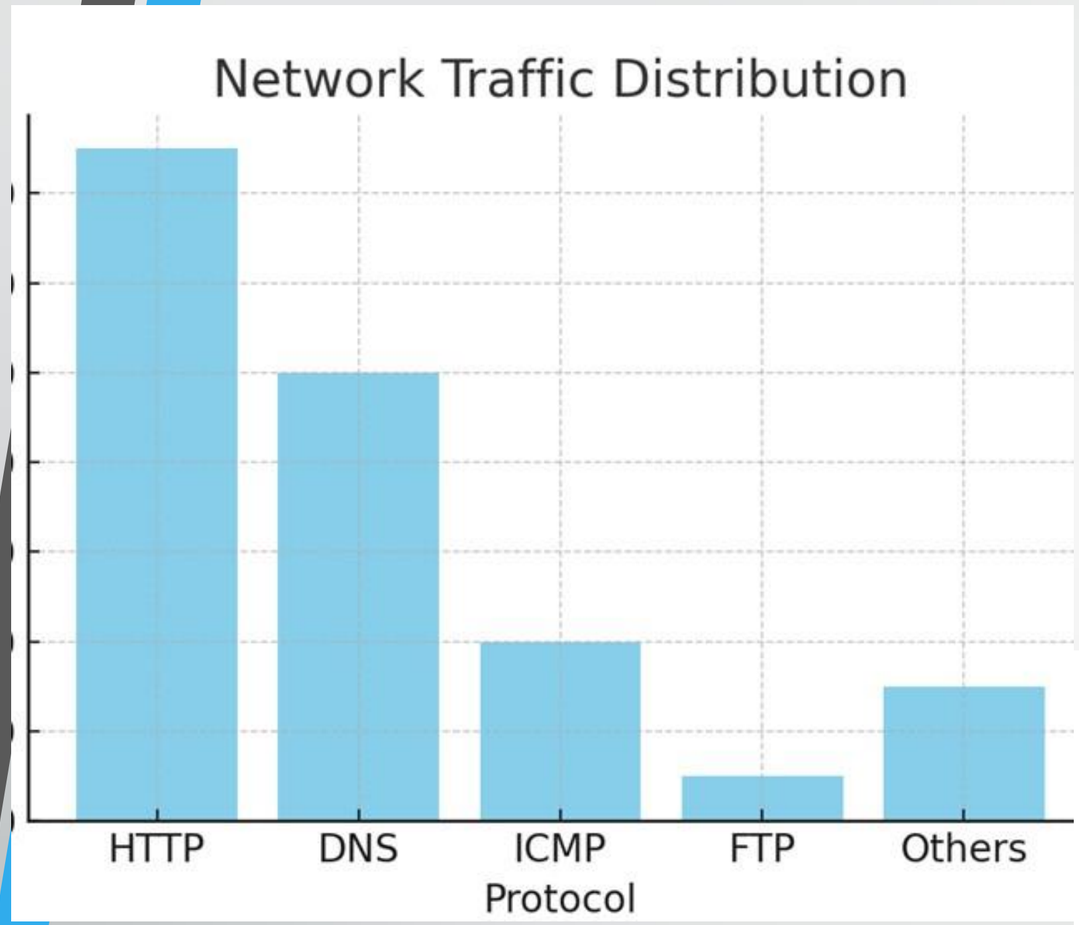
1: CAPTURE HTTP TRAFFIC

- Open Wireshark.
- Select the network interface that connects to the internet.
- Click on the "Start Capture" button (the blue shark fin icon).
- Open your web browser and navigate to a website that uses HTTP (e.g., http://example.com).
- Let the page load completely and then stop the capture in Wireshark by clicking on the red square icon.
- **Expected Output**
- A capture file containing network traffic, including HTTP requests and responses.

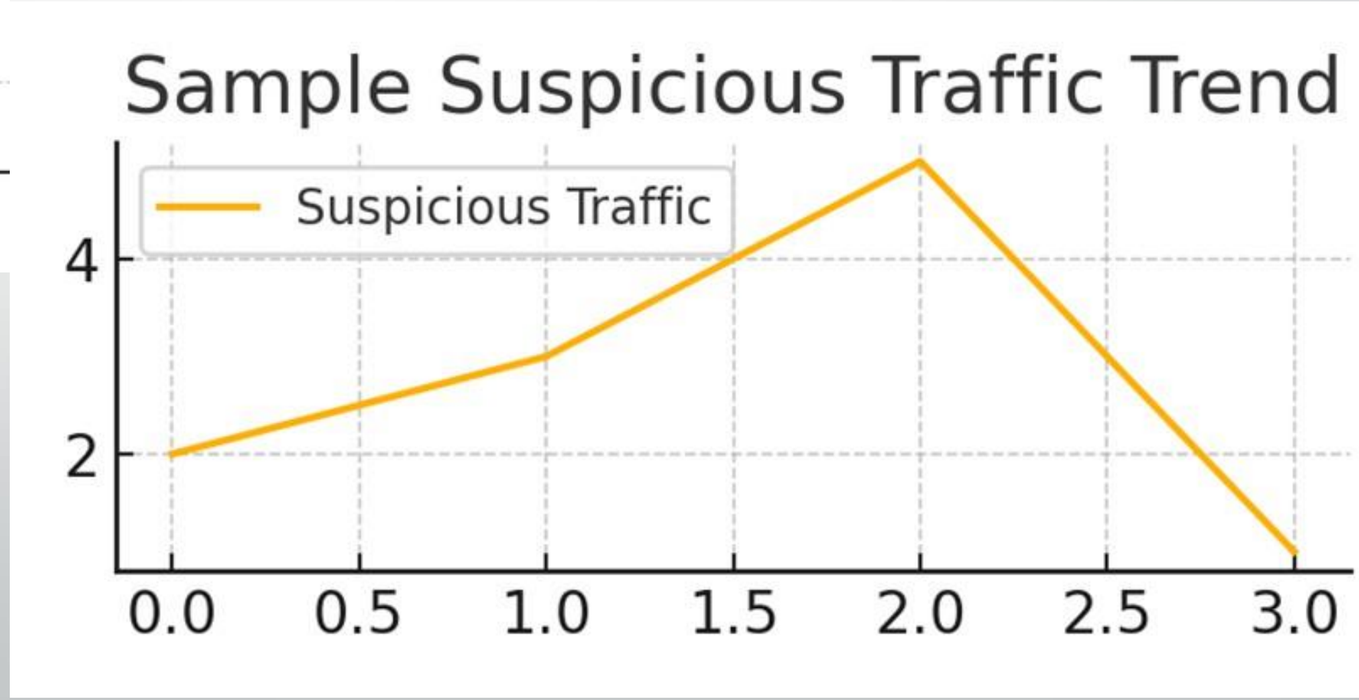
SCREENSHOTS







The following chart shows a suspicious spike in traffic over a short time window.



2: FILTER HTTP TRAFFIC

Steps

1. In Wireshark, go to the filter bar at the top.
2. Enter the filter http and press Enter.
3. Wireshark Will display only the HTTP Traffic from the capture.

Expected Output

Displayed HTTP Traffic filtered from the over all capture.

3: ANALYZE HTTP REQUESTS

Steps

- In the filtered HTTP traffic, locate an HTTP GET request.
- Click on the GET request to view its details in the packet details pane.
- Expand the "Hypertext Transfer Protocol" section to see detailed information about the request, such as the requested URL, headers, and parameters.

Expected Output:

- Detailed information about an HTTP GET request displayed.

4: Analyze HTTP Responses

Steps

- In the filtered HTTP traffic, locate the corresponding HTTP response for the GET request you analyzed.
- Click on the response to view its details in the packet details pane.
- Expand the "Hypertext Transfer Protocol" section to see detailed information about the response, such as the status code, headers, and content type.

Expected Output

- Detailed information about an HTTP response displayed.

CONCLUSION OF ABOVE STEPS

-
- By completing these exercises, you have learned how to capture, filter, and analyze HTTP traffic using Wireshark.
-
  - These skills are essential for understanding web communication, troubleshooting network issues, and performing security investigations.

DETECTING AND INVESTIGATING MALWARE TRAFFIC

Introduction

In this project, you'll learn how to use Wireshark to detect and investigate malware traffic. Identifying malicious network behavior is crucial for protecting networks and responding to security incidents.

Pre-requisites

Basic understanding of networking concepts
Wireshark installed on your computer
A sample PCAP file containing malware traffic (e.g., from Malware Traffic Analysis)

LAB SET-UP AND TOOLS

- **Wireshark**: Download and install Wireshark from https://www.wireshark.org/download.html.
- **Sample PCAP File**: Download a sample PCAP file containing malware traffic for analysis.
- Link to download PCAP files : https://malware-traffic-analysis.net/2022/workshop/index.html
- Link for malware analysis : https://malware-traffic-analysis.net/

EXERCISE 1: LOAD A SAMPLE PCAP FILE

Steps

- Open Wireshark.
- Go to "File" > "Open" and select the sample PCAP file you downloaded.
- The file will load, and the captured traffic will be displayed.

Expected Output

- The sample PCAP file containing network traffic loaded in Wireshark.

EXERCISE 2: IDENTIFY MALICIOUS TRAFFIC PATTERNS

Steps

- Look for unusual patterns in the traffic, such as repeated connections to suspicious IP addresses, unusual protocol, or large amount of data being transferred.
- Use the filter bar to isolate suspicious traffic. Common filters include:
- ->IP .Add ==x.x.x.x. (replace x.x.x.x with a suspicious IP address)
- ->TCP .port ==4444 (common port used by malware)
- -> HTTP . Request (to view HTTP request that might indicate command- and-control activity)
- .Expected Output
- Suspicious traffic patterns identified in the network capture.

EXERCISE 3: ANALYZE MALICIOUS TRAFFIC

Steps

- Select a packet that appears suspicious based on your initial analysis or it seems to be different from others protocols.
- Click on the packet to view its details in the packet details pane.
- Expand the relevant protocol sections to examine the details of the packet, such as headers, payload data, and any anomalies.

Expected Output

- Detailed information about a suspicious packet analyzed.

EXERCISE 4: FOLLOW THE MALWARE'S COMMUNICATION STREAM

Steps

- Right-click on a suspicious packet and select "Follow" >"TCP Stream" or "UDP Stream" to view the entire conversation.
- Analyze the conversation for indicators of malicious activity, such as unusual commands, encoded data, or unexpected file transfers.

Expected Output

- A complete communication stream of the malware analyzed.

EXERCISE 5: DOCUMENT AND REPORT FINDINGS

Steps

- Take notes on the suspicious activities and patterns you identified in the traffic.
- Document key findings, including IP addresses, ports, payload data, and any other relevant details.
- Summarize your findings in a report format, which can be used for further investigation or as part of a security incident report.

Expected Output

- A detailed report documenting the findings from your malware traffic analysis.

CONCLUSION OF ABOVE EXERCISES

By completing these exercises, you have learned how to detect and investigate malware traffic using Wireshark. These skills are essential for identifying malicious network behavior, responding to security incidents, and protecting network infrastructure.

RESULTS AND DISCUSSION

- After capturing and analyzing network traffic using **Wireshark** and **Zeek**, we made the following observations:
- Detected several unusual DNS queries that pointed to unknown or suspicious domains.
- Identified multiple unauthorized remote login attempts within the test network.
- Noticed abnormal traffic spikes at certain times, indicating possible scanning activities or malware communication attempts.
- Zeek generated detailed logs showing connection attempts, HTTP requests, DNS queries, and SSL usage, while Wireshark provided in-depth packet details for further inspection.

DISCUSSION:

- These findings show how combining Wireshark and Zeek improves network monitoring. The suspicious DNS queries might indicate malware trying to contact external servers. Unauthorized login attempts are signs of possible hacking or brute-force attacks. Abnormal traffic spikes could mean someone is scanning the network for weaknesses.
- By using both tools, we could not only detect these activities but also understand when, how, and where they happened. This makes it easier for network administrators to take quick action and improve security

CHALLENGES

- **Large volume of data:** Capturing and analyzing big amounts of network traffic required good filtering skills and careful analysis.
- **Interpreting logs:** Zeek produces a lot of log files, which were sometimes difficult to interpret for specific attack patterns without experience.
- **Setting up test environments:** Creating controlled scenarios for testing attacks or unusual activities without affecting other systems was challenging.

CONCLUSION:

- This project demonstrated how combining Wireshark and Zeek enhances network visibility and threat detection.
- Multiple suspicious behaviors were detected and validated using packet and event-level data.
- From this project, we learned how to capture and analyze real network traffic, how to interpret logs and packet data, and how to spot warning signs of potential attacks.
- We also gained hands-on experience with two important cybersecurity tools and better understood the challenges of monitoring and securing a network in real time.

FUTURE WORK:

We can perform various tasks like :

- Set up a larger and more realistic network environment to analyze more complex traffic patterns.
- Add machine learning techniques to automatically detect anomalies in the logs.
- Improve encryption handling by integrating SSL decryption for deeper traffic inspection (in a legal and controlled environment).
- Create a dashboard to visualize network activity for easier monitoring.

RECOMMENDATION

- Run Zeek in real-time mode – to monitor live traffic and get immediate alerts.
- Integrate Wireshark and Zeek with a SIEM like ELK or Splunk – for dashboard visualization and long-term traffic analysis.
- Establish a baseline of normal network traffic – to easily detect unusual or suspicious behavior.
- Keep Zeek scripts updated and customized – to detect new types of attacks and tailor detection to your network.
- Use capture filters in Wireshark – to focus only on relevant traffic and reduce data overload.

REFERENCES

- Wireshark Foundation. (2024). Wireshark User Guide. Retrieved from: https://www.wireshark.org/docs/wsug_html_chunked/
- The Zeek Project. (2024). Zeek Documentation. Retrieved from: https://docs.zeek.org/en/current/
- Bejtlich, R. (2014). The Practice of Network Security Monitoring: Understanding Incident Detection and Response. No Starch Press.
- Scarfone, K., & Mell, P. (2007). Guide to Intrusion Detection and Prevention Systems (IDPS). NIST Special Publication 800-94. Retrieved from: https://csrc.nist.gov/publications/detail/sp/800-94/archive/2007-02-01
- Conti, G. (2007). Security Data Visualization: Graphical Techniques for Network Analysis. No Starch Press.

THANK YOU