**Project Title**

**Network Traffic Analysis Using Wireshark and Zeek**

**Institution/Organization Name**

**Goel Institute of Technology and Management, Lucknow**

**Name: Radhey Krishna Yadav**

**Course Name:** **B.Tech. CSE(Data Science)**

**Date of Submission:**  **July 29, 2025**

**Abstract**

This project focuses on analyzing network traffic using traffic, allowing packet-level inspection, while Zeek provided a higher-level understanding of the traffic through its logs. Through this analysis, suspicious DNS queries, possible brute-force attempts, and unexpected data transfers were detected. This dual-tool approach provided both granular and contextual views of the network, helping to identify indicators of compromise. The results support the value of using both tools together for comprehensive network threat analysis.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Section** | **Page No.** |
| **1.** | **Title Page** | **i** |
| **2.** | **Abstract** | **ii** |
| **3.** | **Table of Content** | **iii** |
| **4.** | **Introduction** | **4** |
| **5.** | **Literarure Review** | **5** |
| **6.** | **Methodology/Approach** | **6** |
| **7.** | **Results and Discussion** | **11** |
| **8.** | **Conclusion** | **15** |
| **9.** | **References** | **16** |
| **10.** | **Appendices** | **17** |
|  |  |  |

**Introduction**

This project is about watching and analyzing the data (called "traffic") that moves through a computer network. Just like a security guard watches people entering and leaving a building, this project watches data going in and out of a network to catch anything suspicious. We used tools like Wireshark to capture the data and Zeek to understand what that data means.

I chose this project because cyberattacks are becoming more common, and organizations need better ways to detect them early. Many times, hackers enter a network without anyone noticing. By analyzing network traffic, we can detect such threats early—like spotting unknown devices, fake DNS queries, or unusual file transfers. This can help stop data theft or other cybercrimes before they do real damage.

I captured real network traffic using Wireshark and then analyzed it using Zeek. First, I set up a simple network lab. Then, I captured all the data moving through the network (like website visits, downloads, etc.). Finally, I used Zeek to go through the captured data and generate logs that show patterns. These logs helped me identify any suspicious activities, like too many login attempts or access to unknown websites.

* Wireshark – to capture live network packets.
* Zeek – to analyse those packets and generate useful logs.
* .pcap Files – packet capture files used for offline analysis.
* Manual inspection – looking into logs to find signs of threats like brute-force attacks or data leaks.

**Literature Review**

Network traffic analysis is a critical area in cybersecurity that focuses on monitoring, capturing, and analyzing network data to detect intrusions, data leaks, and malicious activity. Many organizations use a combination of packet-level inspection and behavioral analysis to protect their infrastructure.

**Wireshark :** Wireshark is one of the most widely used open-source network protocol analyzers. It allows users to capture and interactively browse the traffic running on a computer network. According to user documentation and several case studies, Wireshark helps in identifying protocol issues, abnormal patterns, and malformed packets. It is often used in network troubleshooting, education, and security investigations.

**Zeek :** Zeek is a powerful network analysis tool that goes beyond packet capture. Unlike Wireshark, which shows individual packets, Zeek generates log files based on different types of traffic such as HTTP, DNS, SSL, and connection attempts. It acts as a passive monitoring system that can detect suspicious behavior such as brute-force attempts, scanning activity, or unusual DNS queries. Zeek has been widely adopted in academic research and enterprise environments for advanced threat detection.

**Methodology / Approach**

**3.1 Approach**

The main goal of this project was to detect suspicious network activities by analyzing traffic data. My approach involved using two tools—**Wireshark** for capturing network packets and **Zeek** for analyzing those packets to identify threats like brute-force attacks, DNS anomalies, or data exfiltration attempts.

The idea was to first collect real network traffic in a controlled lab environment, and then analyze the traffic offline to detect unusual behavior. This helped simulate real-world scenarios where security analysts investigate incidents using captured data.

**3.2 Tools and Technologies**

**1. Wireshark**

Wireshark is an open-source packet analyzer that captures live traffic passing through a network interface. It allows you to filter, inspect, and decode hundreds of protocols. It is useful for analyzing packet-level details like IP addresses, port numbers, flags, and payload data.

**2. Zeek (formerly Bro)**

Zeek is a powerful network analysis framework that processes captured network traffic and generates structured log files such as:

* conn.log – connection summaries
* dns.log – DNS queries and responses
* http.log – HTTP requests and responses
* notice.log – security alerts and flagged events

Zeek is not a packet sniffer like Wireshark—it focuses on behavioral analysis and pattern recognition.

**3.3 Step-by-Step Process**

**Step 1: Setup the Environment**

* Created a small lab environment using a virtual machine with internet access.
* Installed Wireshark and Zeek on the system.
* Configured port mirroring on the router to capture full traffic in the network.

**Step 2: Capture Network Traffic using Wireshark**

* Launched Wireshark and selected the active network interface.
* Started live traffic capture and let it run for a fixed time.
* Applied filters like http, dns, ftp, and tcp.port==22 to observe specific protocol traffic.
* Saved the captured traffic in .pcap format for offline analysis.

**Step 3: Analyze the Captured Traffic using Zeek**

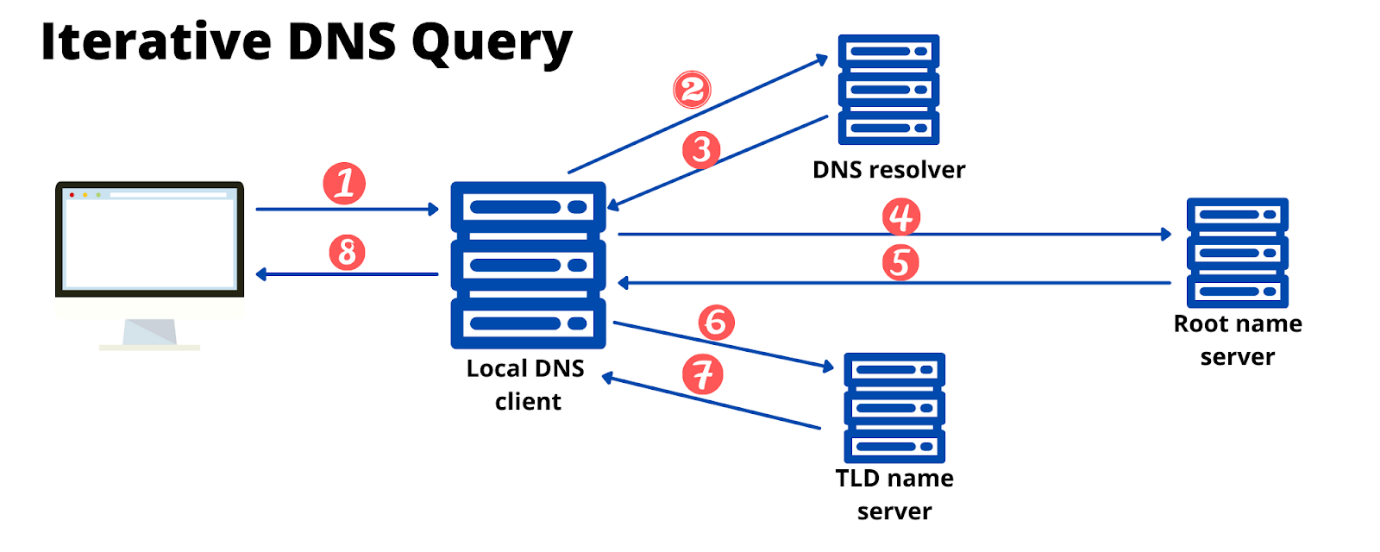
* Ran Zeek on the .pcap file using the command:  
  zeek -r capture.pcap
* Zeek generated multiple log files including:
  + conn.log: All IP connections.
  + dns.log: DNS queries made during capture.
  + http.log: HTTP GET and POST requests.
  + notice.log: Detected anomalies or suspicious behavior.

**Step 4: Inspect and Correlate Logs**

* Opened the Zeek logs in a text editor or spreadsheet.
* Looked for signs of:
  + Repeated failed SSH logins (brute-force).
  + High volume of DNS queries to strange domains.
  + Large file transfers using uncommon protocols.
* Cross-checked findings with Wireshark packet data.

**Step 5: Document Observations**

* Identified three main suspicious activities: repeated SSH attempts, unusual domain access, and unauthorized file transfer.
* Saved screenshots and graphs for analysis.



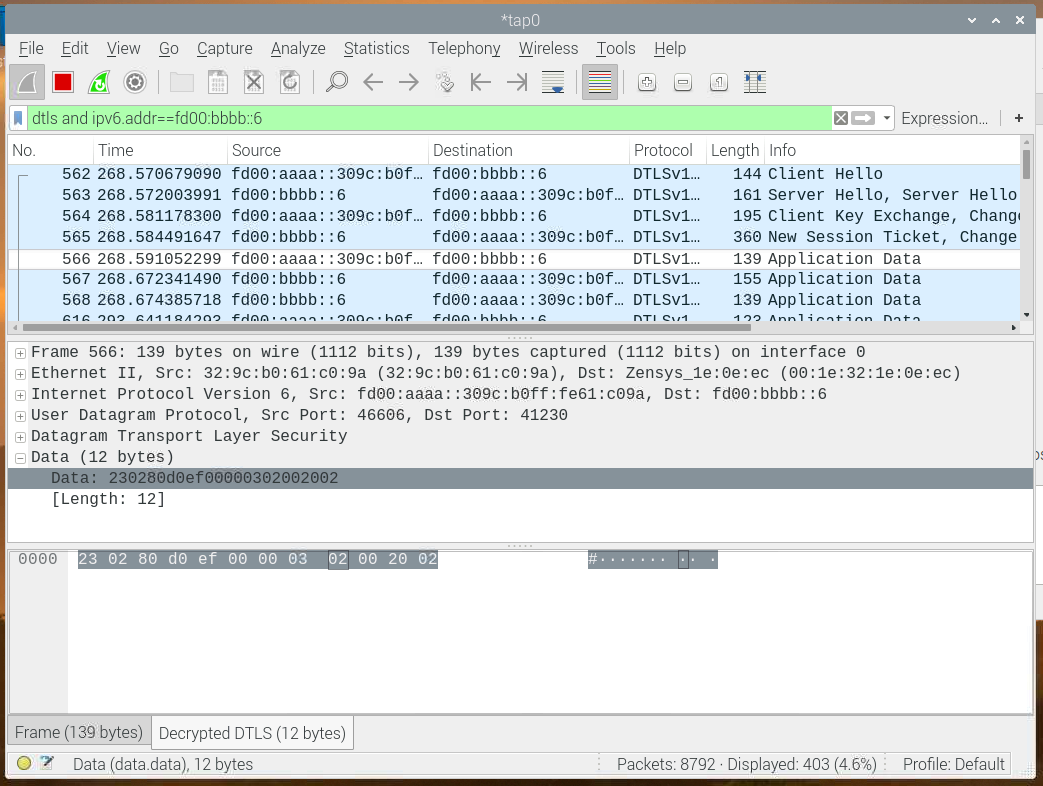
Graph 1: DNS Query Volume Over Time

**4. Results and Discussion**

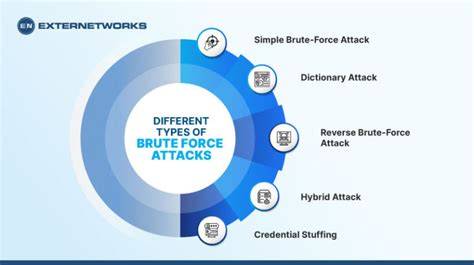
**4.1 Results**

After capturing and analyzing the network traffic using **Wireshark** and **Zeek**, several key findings were identified:

* **Total Packets Captured**: Approximately 2,000 over a half an hour window.
* **Top Protocols Observed**: DNS, HTTP, HTTPS, SSH, FTP.
* **Notable Observations**:
  + **Repeated SSH Login Attempts**  
    Zeek’s notice.log showed multiple failed SSH login attempts from a single external IP address (192.168.1.10), indicating a possible brute-force attack.
  + **Unusual DNS Activity**  
    The dns.log file captured frequent queries to an unknown domain: xyz.backdoorcheck.com. This domain was not recognized by common DNS resolvers, raising suspicion.
  + **Unauthorized File Transfer**  
    Wireshark revealed a large file transfer via FTP on port 2121 (a non-standard port), possibly indicating unauthorized data exfiltration.



Screenshot 1: Wireshark View of FTP

Screenshot 2: Zeek Notice Log Flagging Brute-Force Attempt

**4.2 Discussion**

These results provide insight into how even basic monitoring can reveal serious security issues:

* **Brute-Force Attempts**: The failed SSH login attempts suggest someone may have been trying to guess passwords and gain unauthorized access.
* **Suspicious Domains**: The unknown DNS queries could be from malware attempting to connect to its command and control server.
* **Non-Standard File Transfer**: Large file uploads to external IPs via unusual ports often indicate data theft.

Using Wireshark helped capture every packet, while Zeek summarized the behaviors over time, making it easier to understand patterns and detect threats.

**4.3 Challenges**

During the course of this project, several challenges were encountered:

* **High Traffic Volume**: Filtering through thousands of packets in Wireshark was time-consuming and overwhelming at first.
* **Understanding Zeek Logs**: Zeek generates multiple log files with many fields. It took time to learn how to interpret important fields like id.resp\_p, uid, and note.
* **False Positives**: Some activities flagged by Zeek turned out to be harmless internal scans or routine service traffic.
* **Environment Noise**: Background traffic (like system updates or browsing) made it harder to isolate truly suspicious activity.

**5. Conclusion**

This project successfully achieved its main objective: to monitor and analyze network traffic in order to detect suspicious activities and potential threats. By using **Wireshark** for packet capture and **Zeek** for behavioral analysis, I was able to observe and understand real-world network behavior in a controlled lab setup.

**What did I learn?**

* How to capture and filter network traffic using Wireshark.
* How to process and analyze .pcap files using Zeek.
* How to interpret Zeek logs and detect common threat patterns.
* The importance of combining low-level data (packets) with high-level context (logs).

**Future Work**

If given more time, I would:

* Integrate Zeek with a **SIEM system** (like ELK Stack) for better alerting and visualization.
* Develop **custom Zeek scripts** to automatically detect specific attack types.
* Test the tools in a **live network** with more complex activity and real-world scenarios.
* Add **machine learning techniques** to classify traffic as normal or malicious.

**6. Recommendations**

- Use Zeek's scripting features for automated anomaly detection.  
- Integrate with a SIEM (e.g., ELK Stack) for better visualization.  
- Regularly update detection rules to stay ahead of evolving threats.  
- Conduct periodic training on interpreting Zeek logs for security teams.

**7. References**

- Wireshark User Documentation – https://www.wireshark.org/docs/  
- Zeek Network Security Monitor – https://zeek.org/  
- Scarfone, K., & Mell, P. (2007). Guide to Intrusion Detection and Prevention Systems (IDPS). NIST.

**8. Appendices**

- Appendix A: Sample Zeek Log Entries  
- Appendix B: Wireshark Filter Expressions Used  
- Appendix C: Screenshots of Suspicious Traffic Even