Network Traffic Analysis Using Wireshark and Zeek

Name : Mohammad Sameer

Institution: United College Of Engineering and Research

Course Name: B Tech

Date : 01-08-2025

Supervisor: Mr. Ayush Kumar

# Abstract

This project explores the detection of suspicious network activities using Wireshark and Zeek (formerly Bro), two powerful network traffic analysis tools. In a controlled lab environment, network traffic was captured and analyzed to detect protocol anomalies, signs of intrusion, and unusual behavior. Wireshark was used for deep packet inspection, while Zeek provided high-level analysis through scripts and logs. The results highlighted DNS tunneling, malformed HTTP requests, and port scanning attempts. This report discusses the methodology, findings, and proposes mitigation strategies like firewall hardening and real-time alert systems. The project showcases how open-source tools can support intrusion detection and enhance cybersecurity awareness.

# Table of Contents

1. Title Page

2. Abstract

3. Table of Contents

4. List of Figures and Tables

5. Introduction

6. Literature Review

7. Methodology

8. Results and Discussion

9. Conclusion

10. Recommendations

11. References

12. Appendices

# List of Figures and Tables

Figure 1: Wireshark Interface – Packet Capture View

Figure 2: Zeek Log Output – Notice.log Sample

Table 1: Comparison of Normal vs Suspicious Traffic Patterns

Figure 3: Detected DNS Tunneling Timeline

# Introduction

Network traffic analysis is crucial for detecting cyber threats and understanding communication patterns. This project focuses on using Wireshark and Zeek to analyze network data in a lab setting. With increasing threats like malware, data exfiltration, and reconnaissance attacks, real-time and post-event traffic inspection has become essential. The goal of this project is to identify indicators of compromise (IoCs), understand common anomaly patterns, and demonstrate how open-source tools help in threat detection and prevention.

# Literature Review

Wireshark is a well-established packet analyzer used to inspect packet-level data in real time or from saved captures. Zeek, on the other hand, is a network security monitor that creates structured logs based on traffic behaviors. Research suggests that combining both tools gives security analysts a deep and broad view of network activity. Studies have shown that Zeek is particularly effective at detecting anomalies like port scans and suspicious DNS queries, while Wireshark is ideal for manual packet inspection and protocol analysis.

# Methodology

## Approach

We simulated normal and suspicious traffic in a controlled lab environment using virtual machines. The objective was to capture traffic, analyze anomalies, and create a threat report.

## Tools and Technologies

- Wireshark: For packet-level traffic capture and protocol dissection.  
- Zeek (Bro): For behavior-based logging and detection.  
- Kali Linux and Ubuntu VMs: For generating controlled traffic.  
- TCPDump: For command-line packet capturing.

## Step-by-Step Process

1. Set up a virtual network using VirtualBox with one attacker VM and one victim VM.  
2. Installed Zeek and Wireshark on a monitoring node.  
3. Used tools like `nmap`, `curl`, and `dns2tcp` to generate scanning and tunneling traffic.  
4. Captured traffic with Wireshark and TCPDump.  
5. Ran Zeek on pcap files to generate logs: conn.log, dns.log, http.log, notice.log.  
6. Analyzed Zeek logs and correlated them with Wireshark findings.

# Results and Discussion

## Results

- Port Scan Detected: Zeek's notice.log flagged a horizontal scan across ports.  
- DNS Tunneling: Detected via abnormal DNS queries to suspicious domains.  
- HTTP Anomalies: Wireshark revealed malformed HTTP headers and suspicious User-Agent strings.  
- Bandwidth Spike: Spotted unusual data upload patterns consistent with data exfiltration.

## Discussion

The combination of Zeek and Wireshark proved effective in identifying different stages of an attack. Zeek's structured logs reduced the manual burden of filtering through packets, while Wireshark validated findings visually. One challenge was dealing with the massive amount of captured data, requiring careful filtering and scripting.

# Conclusion

This project successfully demonstrated how network traffic analysis with Wireshark and Zeek can uncover security threats. The ability to detect port scans, tunneling attempts, and malformed packets highlights the effectiveness of this method in early threat detection. This hands-on experience enhances understanding of real-world attack signatures and network behavior.

## Future Work

- Automate threat detection with Zeek scripting and SIEM integration.  
- Expand traffic scenarios to include malware communication.  
- Integrate real-time alerting with email/SMS triggers from Zeek logs.

# Recommendations

- Use Zeek in conjunction with firewall logs for full visibility.  
- Regularly update Zeek detection scripts to catch evolving threats.  
- Train security analysts in traffic pattern recognition using tools like Wireshark.

# References

Paxson, V. (1999). "Bro: A System for Detecting Network Intruders in Real-Time".

Wireshark Documentation – https://www.wireshark.org/docs/

Zeek Documentation – https://docs.zeek.org

Scarfone, K., & Mell, P. (2007). "Guide to Intrusion Detection and Prevention Systems (IDPS)". NIST.

# Appendices

Appendix A: Zeek conn.log Sample  
Appendix B: Wireshark .pcapng Snapshots  
Appendix C: Script used for DNS tunneling test