Network Packet Sniffer with Anomaly Detection

Introduction

In the modern cybersecurity landscape, network traffic analysis is essential for detecting unauthorized access and identifying malicious activities early. This project presents a **Network Packet Sniffer with Anomaly Detection**, designed to monitor live network traffic, log essential details, and identify suspicious patterns such as DoS-like behaviors. The tool is tailored for security analysts, penetration testers, and interns who need to observe and report on network health in real-time.

The project emphasizes simplicity, reliability, and extensibility, offering a clean command-line interface and auto-generated visual summaries. It demonstrates the value of proactive monitoring in cybersecurity operations.

Abstract

This packet sniffer tool captures live network packets using Python's scapy library. It extracts and logs packet metadata such as timestamps, source/destination IPs, protocol types, and packet lengths. The data is stored in a local sqlite3 database for analysis and reporting.

The tool includes a basic anomaly detection system that flags and records high traffic rates, simulating potential DoS attacks by monitoring packets per second. Detected anomalies are logged and visualized using matplotlib charts. The solution can be extended to identify additional attack patterns (e.g., port scanning, SYN floods) and to generate more advanced reports.

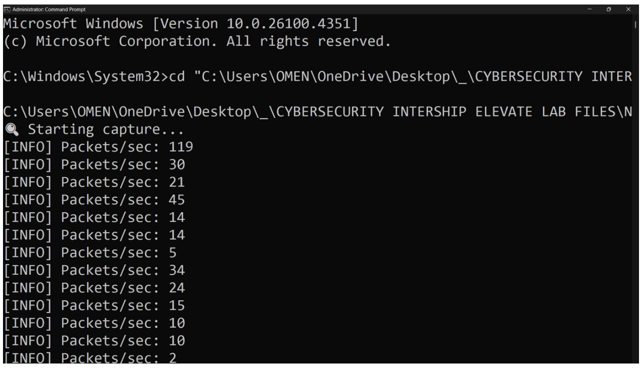
Tools Used

- **Python 3.12** – Core programming language for implementation.
- **scapy** – For real-time network packet sniffing and extraction of header data.
- **sqlite3** – For storing packet logs and anomaly records in a lightweight, persistent database.
- **matplotlib** – For generating visual reports on network traffic and anomalies.
- **threading** – For concurrent monitoring of packet rates without interrupting capture.
- **CSV module** – For exporting database contents to CSV files for external review.
- **Command Line Interface (CLI)** – For user interaction and report generation.

Steps Involved in Building the Project

Packet Capture:
Implemented a packet sniffer using scapy to collect live packets from the network interface.

The sniffer extracts important metadata: timestamp, source IP, destination IP, protocol, and length. These are processed in real-time and sent to the database handler.



```
Microsoft Windows [Version 10.0.26100.4351]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>cd "C:\Users\OMEN\OneDrive\Desktop\_\CYBERSECURITY INTER

C:\Users\OMEN\OneDrive\Desktop\_\CYBERSECURITY INTERSHIP ELEVATE LAB FILES\N
🔍 Starting capture...
[INFO] Packets/sec: 119
[INFO] Packets/sec: 30
[INFO] Packets/sec: 21
[INFO] Packets/sec: 45
[INFO] Packets/sec: 14
[INFO] Packets/sec: 14
[INFO] Packets/sec: 5
[INFO] Packets/sec: 34
[INFO] Packets/sec: 24
[INFO] Packets/sec: 15
[INFO] Packets/sec: 10
[INFO] Packets/sec: 10
[INFO] Packets/sec: 2
```

Database Design & Logging:

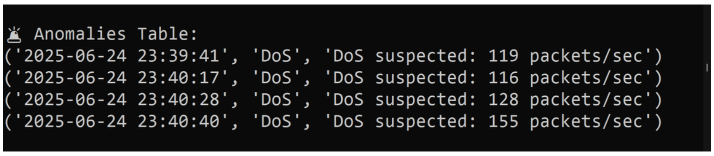
Designed and initialized an SQLite database with two tables:

- packets: stores packet metadata.
- anomalies: stores records of detected anomalies.

A background thread handles insertion of packet data to ensure efficient logging without data loss at higher capture rates.

Anomaly Detection:

Implemented a concurrent packet rate monitor. The tool calculates packets per second and compares it to a configurable threshold (default: 100 packets/sec). When the threshold is breached, an anomaly is logged in the database for review.



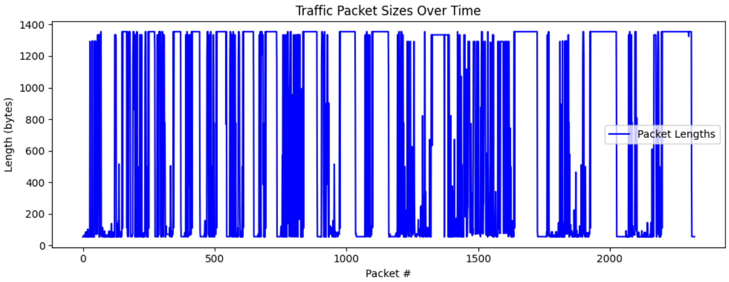
```
🚨 Anomalies Table:
('2025-06-24 23:39:41', 'DoS', 'DoS suspected: 119 packets/sec')
('2025-06-24 23:40:17', 'DoS', 'DoS suspected: 116 packets/sec')
('2025-06-24 23:40:28', 'DoS', 'DoS suspected: 128 packets/sec')
('2025-06-24 23:40:40', 'DoS', 'DoS suspected: 155 packets/sec')
```

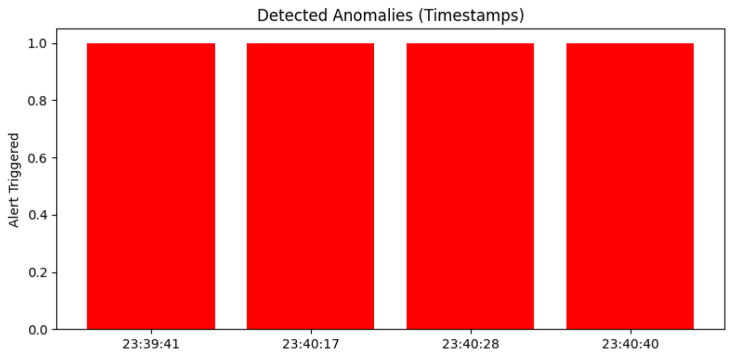
Graph Generation:

Developed functions to generate:

- A line graph of packet sizes over capture duration.
- A bar chart showing timestamps of detected anomalies.

These graphs help visualize network behavior and highlight periods of unusual traffic activity.





Data Review Utility:

Created a CLI tool (view_db.py) enabling:

- Viewing of all packet and anomaly records.
- Filtering packets by specific IP addresses.
- Exporting data to CSV files for external analysis.

This makes it easier to interpret the captured data and integrate it into broader security assessments.

Conclusion

The Network Packet Sniffer with Anomaly Detection achieves its objectives as a lightweight, functional network monitoring tool. It successfully demonstrates how network traffic can be captured, stored, analyzed, and visualized using accessible technologies. The system's modular design allows for easy extensions, including support for additional protocols, custom anomaly detection rules, or integration with larger Security Information and Event Management (SIEM) platforms.

The project underscores the importance of proactive monitoring in cybersecurity, offering an educational tool for interns and a foundation for more advanced security systems.