Real-Time Network Intrusion Detection System (IDS)

Project Overview

This project implements a fully functional, real-time Intrusion Detection System (IDS) using Python and advanced machine learning techniques. The system captures live network traffic, extracts key packet features, and applies an unsupervised anomaly detection model to identify and flag potential threats or suspicious activities that deviate from normal network behavior. The approach combines network programming, data preprocessing, and intelligent pattern recognition for robust cybersecurity analysis.

Methodology and Technical Stack

The proposed IDS architecture integrates concepts from both networking and machine learning domains, ensuring a structured, data-driven approach to network anomaly detection. The following components form the technical backbone of the system:

Component-wise Implementation:

1. Packet Capture: Implemented using the Scapy library to intercept and parse network traffic packets on a specified network interface (default: 'en0'). Scapy provides low-level access to live packet data, allowing real-time monitoring of network behavior.

2. Anomaly Detection: The anomaly detection logic utilizes Isolation Forest (from Scikit-learn), a tree-based ensemble algorithm capable of identifying outliers in high-dimensional data. This model requires no prior labeled attack data, making it suitable for identifying unknown (zero-day) network threats.

3. Data Preprocessing: Key packet features such as Time, Length, and Protocol are structured using Pandas. Protocol types are factorized and encoded numerically to enable compatibility with ML algorithms. Standard scaling via StandardScaler ensures that features contribute equally to the model’s decision-making process.

4. Visualization: The detection results are plotted using Matplotlib. Normal traffic is represented in green, while anomalies are highlighted in red on a scatter plot, providing a visual understanding of traffic irregularities.

Key Features

• Real-Time Monitoring: Captures and analyzes network packets live from the host interface, ensuring up-to-date threat visibility. • Unsupervised Learning: Uses an anomaly-based approach, eliminating the need for predefined attack signatures and enhancing adaptability. • DSA Integration: Efficiently structures and transforms dynamic network packet data using data structuring and transformation algorithms for optimal ML performance. • Visual Reporting: Provides clear anomaly visualizations, aiding in easier interpretation of network events.

Getting Started

1. Install Dependencies: Ensure Python and required libraries are installed. Run the following command: pip install scapy pandas numpy scikit-learn matplotlib 2. Run the Script: Execute the Python script and modify the interface variable (‘en0’) to match your system’s Wi-Fi adapter. 3. Analyze Results: The system runs for a defined duration, displaying total packets processed, number of anomalies detected, and a visual scatter plot differentiating between normal and abnormal traffic behavior.

Conclusion

This Real-Time Intrusion Detection System effectively bridges networking concepts with machine learning-based anomaly detection, providing a foundation for proactive cybersecurity solutions. Its unsupervised nature and adaptability make it highly suitable for real-world network environments where attack patterns continuously evolve.