Threat Detection and Anomaly Identification Framework – AI-driven security mechanism to detect suspicious or malicious activity.

Akhila Sunesh,
Miza Haris P,
Shone Sajan

Mrs. Gayathri J L
Department of Computer Science and Engineering
Saintgits College of Engineering, Kottayam, Kerala
June 28, 2025

Abstract - In an era of rapidly evolving cyber threats, traditional network security models often struggle to keep up due to their heavy computational requirements, time-consuming setup for preliminary analysis, and overly technical interfaces that limit accessibility. To address these challenges, a lightweight, AI-powered anomaly detection framework has been designed for fast, efficient, and user-friendly network traffic analysis. By combining accessibility with analytical power, Mini Joy serves as an effective tool for early-stage network monitoring and education.

***Keywords*:** Network Security, Machine Learning, Anomaly Detection, PCAP, Intrusion Detection, Isolation Forest, Streamlit, AI Feedback.

1 INTRODUCTION

Our proposed model introduces Mini Joy, a compact and modular anomaly detection tool inspired by the Joy model used in network flow analysis. As cyber threats grow in complexity, traditional rule-based systems struggle to detect novel attacks. Mini Joy addresses this by offering a lightweight, ML-based framework that prioritizes accessibility and rapid analysis. It supports both PCAP and CSV formats, extracting flow-level features such as packet count, byte rate, and inter-arrival time. Using unsupervised learning algorithms like Isolation Forest and KMeans, it detects abnormal traffic behavior effectively. A Streamlit GUI and AI-driven feedback module make it user-friendly for both researchers and students.

2 MATERIALS AND METHODS

The dataset chosen is CICIDS2017 dataset, a comprehensive and widely used benchmark for intrusion detection research. Developed by the Canadian Institute for Cybersecurity (CIC), this dataset simulates realistic network traffic by combining benign activities with various modern attack scenarios such as DoS, DDoS, Brute Force, Botnet, Infiltration, and Web Attacks. It includes detailed flow-level features such as packet lengths, durations, protocol information, and traffic direction, making it suitable for both supervised and unsupervised anomaly detection tasks. The dataset is structured in CSV format and is also compatible with raw PCAP files, allowing flexibility in preprocessing and model training.

To identify anomalies in the network flows, this project implements two unsupervised machine learning algorithms: Isolation Forest and KMeans. Isolation Forest is an ensemble-based anomaly detection method that isolates observations by randomly selecting features and split values. The core idea is that anomalies are more susceptible to isolation and thus require fewer splits to separate from the rest of the data. This makes Isolation Forest highly efficient, especially with high-dimensional datasets, and robust against outliers.

KMeans, on the other hand, is a centroid-based clustering algorithm that partitions data into k clusters by minimizing the variance within each cluster. After the clustering process, flows that are located far from any cluster centroid are considered potential anomalies. Although KMeans is sensitive to initialization and requires selecting an appropriate value for k, it provides an intuitive way to group similar traffic patterns and flag outliers. Both models are well-suited for network anomaly detection, offering complementary strengths in terms of speed, interpretability, and effectiveness.

3 FILE ARCHITECTURE

The Mini Joy project follows a modular architecture. At its core, app.py powers the Streamlit-based graphical user interface, allowing users to upload network data, visualize flows, and interact with anomaly detection results. Configuration variables shared across the system are defined in config.py, while main.py serves as a command-line interface. The feedback_generator.py module provides AI-generated, non-technical explanations for detected anomalies, enhancing interpretability. Within the models/ directory, anomaly_model.py handles machine learning training and scoring using unsupervised algorithms like Isolation Forest and KMeans. The utils/ directory contains key preprocessing scripts: csv_loader.py manages CSV input, feature_extractor.py computes statistical and temporal flow-level features, and pcap_processor.py parses raw PCAP files into analyzable flow data. Output results—including model scores, flagged anomalies, and plots—are stored in the outputs/ folder, while the sample_data/ directory provides example PCAP and CSV files for testing. Finally, the README.md file documents setup instructions, usage guidelines, and project details.

4 SYSTEM ARCHITECTURE

The Mini Joy system is designed to be flexible in terms of input formats, supporting both raw and preprocessed network data. It accepts PCAP (packet capture) files, which are parsed using Scapy, a powerful Python library for packet manipulation. The system constructs 5-tuple flows (source IP, destination IP, source port, destination port, and protocol) from the captured packets. In addition, it supports CSV files, typically derived from benchmark datasets such as CICIDS2017, which contain precomputed flow-level statistics.

Once the input data is prepared, Mini Joy proceeds with feature extraction, an essential step in transforming raw network traffic into a structured format suitable for machine learning. Each flow is characterized using a set of statistical and temporal features. These include the number of packets, total bytes transmitted, and statistical summaries (mean, minimum, maximum) of packet lengths. It also computes inter-arrival time metrics to capture temporal behavior between packets, which is often indicative of anomalies such as DoS attacks or traffic spikes.

The system offers two unsupervised machine learning models for anomaly detection. The Isolation Forest algorithm isolates anomalies based on random partitioning of feature space, which is efficient and effective for high-dimensional data. Alternatively, users can opt for the KMeans clustering algorithm, where normal flows are grouped into clusters, and potential anomalies are identified based on their distance from cluster centroids. This flexibility allows users to test and compare different detection approaches within the same framework.

To enhance interpretability, Mini Joy includes a visualization and feedback module. After scoring flows, the system generates a histogram of anomaly scores, helping users understand the distribution of detected threats. The AI Feedback Module further interprets flagged flows by providing human-readable explanations. For example, it may indicate that a flow was marked suspicious due to unusually large packet sizes or bursts of traffic within a short time window.

The entire system is integrated into a user-friendly interface built with Streamlit, allowing smooth interaction without requiring technical expertise. Through the GUI, users can upload PCAP or CSV files, select their preferred machine learning model, and view the results in real time. Additionally, the system offers the option to download flagged anomalies in JSON format, enabling further analysis or integration with other tools. This architecture ensures Mini Joy remains lightweight yet powerful, making it ideal for rapid prototyping and educational use.

5 ANOMALY SCORE DISTRIBUTION ANALYSIS

To better understand the decision-making process of the Isolation Forest model, a histogram of anomaly scores was generated using the CICIDS2017 dataset. In this visualization, the x-axis represents the anomaly score assigned to each network flow, while the y-axis indicates the number of flows within each score range. The scoring system is such that lower (more negative) values correspond to flows deemed normal by the model, whereas scores closer to zero or positive suggest higher levels of anomalous behavior.

A significant observation from the histogram is the high concentration of flows in the score range between -0.25 and -0.15, peaking around -0.22. This clustering indicates that a large proportion of network traffic is considered benign. This is expected in real-world traffic, where malicious flows are typically a minority. The distribution of scores is distinctly right-skewed, showing a long tail toward the higher score ranges. This tail represents the relatively small number of flows that the model considers to be potential anomalies.

To flag anomalies, a threshold is applied based on score percentiles. Specifically, the top 5% of flows with the highest anomaly scores are identified as suspicious and extracted for review. These flows are often indicative of security incidents and are saved in a separate output for further investigation. This approach ensures that even without labeled data, the model can effectively isolate unusual behavior based on statistical deviation from the norm.

Interpreting the results, the flagged outliers could correspond to various types of network attacks. For instance, unusually large packet counts within short time intervals may signal DoS or DDoS attacks, while a high number of short-duration flows may indicate port scanning activity. Similarly, exceptionally large byte transfers within a single flow might point to data exfiltration or tunneling. The histogram thus not only serves as a visual summary of model behavior but also reinforces the effectiveness of unsupervised learning in identifying meaningful security threats in network traffic.

6 CONCLUSION

Mini Joy represents a practical and accessible entry point into the field of AI/ML-based network anomaly detection. Developed with modularity and simplicity at its core, it provides a flexible framework for analyzing network traffic using both raw PCAP and preprocessed CSV data formats. This adaptability makes it suitable for a wide range of users—from students and educators aiming to understand the basics of flow-based traffic analysis to developers and researchers interested in prototyping new detection methods.

One of the key advantages of Mini Joy is its focus on usability and interpretability. The system integrates a user-friendly Streamlit GUI, allowing users to upload files, select models, and view results in real time without needing deep technical knowledge. Additionally, the AI-driven feedback module translates raw anomaly scores into clear, human-readable explanations, making the output more meaningful to non-specialists.

The use of well-established unsupervised learning algorithms such as Isolation Forest and KMeans allows Mini Joy to effectively flag suspicious network behavior without requiring labeled training data. Its architecture is also deep learning–ready.

ACKNOWLEDGMENTS

We acknowledge the mentors, institutional heads and industrial mentors for their guidance and support. We thank Intel Corporation for such an opportunity.

REFERENCES

1. Canadian Institute for Cybersecurity. "CICIDS2017 Dataset." [Online]. Available: `https://www.unb.ca/cic/datasets/ids-2017.html`

2. UNSW Canberra. "UNSW-NB15 Dataset." [Online]. Available: `https://research.unsw.edu.au/projects/unsw-nb15-dataset`

3. Cisco. "Joy: A Package for Capturing and Analyzing Network Flow Data." [Online]. Available: `https://github.com/cisco/joy`

4. Scikit-learn: Machine Learning in Python. [Online]. Available: `https://scikit-learn.org`

5. Streamlit Documentation. [Online]. Available: `https://docs.streamlit.io`