TED UNIVERSITY

SENG 484 Ethical Hacking and Countermeasures

Project Progress Report

IoTGuard: AI-Based Intrusion Detection and Prevention for IoT Networks

Ateş Öztürk

Mehmet Alp Demiral

Ozan Alp Sarıdoğan

1. Introduction **3**
2. Project Progress Summary **3**
2.1 Dataset Analysis **3**
2.2 Preprocessing Pipeline **3**
2.3 Class Consolidation and Mapping **4**
2.4 Machine Learning Architecture **4**
2.5 Results and Discussions **5**
3. Conclusion **6**
4. References **6**

1. Introduction

The IoTGuard project aims to design an AI-powered Intrusion Detection System (IDS) capable of identifying malicious behaviors in IoT network traffic. The solution combines traditional packet-analysis tools such as Suricata with machine learning models to detect complex, multi-class cyberattacks in IoT environments.

In this reporting period, the team focused on exploratory data analysis, dataset preprocessing, and establishing a scalable streaming-based machine-learning pipeline using the CIC-IoT-2023 dataset. The goal was to understand the dataset structure, analyze class distributions, and prepare a training architecture that can handle continuous ingestion of IoT network traffic.

2. Project Progress Summary

2.1 Dataset Analysis

The CIC-IoT-2023 dataset contains multiple CSV files representing IoT network traffic with 33 labeled attack categories and 1 benign labeled category. The following work was completed:

- Inspected dataset structure using pandas (data types, null values, distributions)
- Encoded categorical attack labels using LabelEncoder
- Generated a 50×50 correlation heatmap to examine feature relevance
- Calculated unique value counts for all columns across all files
- Identified strong correlations in rate-based, temporal, and flag-based features

2.2 Preprocessing Pipeline

A modular preprocessing pipeline was implemented:

- Cleaning infinite values and replacing missing data
- Standardization via a streaming StandardScaler (partial_fit)
- Feature selection using 46 consistently available numerical features
- Extraction of behavioral, statistical, and protocol-based attributes

Key Features used

The following 46+ features were identified and consistently extracted across all files:

- Flow duration
- Protocol and flag indicators (e.g. TCP, UDP, DHCP, ICMP, flag counters)
- Statistical signal features (e.g. Min, Max, AVG, Std)
- Packet/byte-based metrics
- Behavioral metrics (IAT, magnitude, radius, covariance)

Cleaning Operations Implemented

- Removal and replacement of null values.
- Handling missing values.
- Consistent column-feature selection across all splits

Scaling

Standardization approach implemented was implemented via partial_fit:

```
scaler = StandardScaler()
scaler.partial_fit(df[X_columns])
```

This allowed scaling large files without memory overhead and ensured consistent normalization.

2.3 Class Consolidation and Mapping

Three class-level configurations were prepared:

- Full 34 class classification
- 7 class grouped classification (e.g. DDoS, DoS, Mirai, Recon, Spoofing, Web, BruteForce, Benign)
- Binary mode (e.g. Attack vs Benign)

These mappings allow evaluation at different abstraction levels.

2.4 Machine Learning Architecture

A streaming classifier pipeline using SGDClassifier (logistic regression loss) was implemented:

Main components:

- Streaming scaler fitting
- Dynamic class weight calculation to handle imbalance
- Streaming training loop (one CSV file per iteration)
- Evaluation pipeline producing precision, recall, F1-score, and confusion matrices

2.5 Results and Discussions

In this reporting stage, the IoTGuard project successfully completed extensive dataset analysis, feature extraction, cleaning, and encoding. A robust and scalable streaming-based machine learning pipeline was implemented, enabling training on extremely large IoT datasets without memory limitations. Correlation analyses and class-mapping strategies provide a strong foundation for the next phases, where LightGBM, anomaly detection, and real-time integration with Suricata will be developed.

Overall, the project is on track and has completed all required milestones for the initial progress stage.

High-Volume Attack Classes

The classifier performed extremely well on large DDoS and Mirai categories with highly distinctive traffic patterns:

- DDoS-ICMP_Flood, DDoS-RSTFINFlood, Mirai-udpplain all achieved F1-scores above 0.95
- Other major DDoS variants had F1-scores between 0.75 and 0.85

This shows the model is highly effective at recognizing large-scale, repetitive attack behaviors.

Moderate and Low-Frequency Classes

Performance dropped sharply for smaller classes:

- Many categories such as BrowserHijacking, XSS, SqlInjection, Uploading_Attack had F1-scores below 0.05
- Reconnaissance subcategories showed inconsistent results due to overlapping feature patterns

Reconnaissance subcategories showed inconsistent results due to overlapping feature patterns

Benign Traffic

- BenignTraffic: F1 = 0.52 (Precision 0.86, Recall 0.37

The lower recall indicates the model tends to misclassify benign flows as attacks, likely due to the overwhelming presence of attack samples in the dataset.

3. Conclusion

In this reporting stage, the IoTGuard project successfully completed extensive dataset analysis, feature extraction, cleaning, and encoding. A robust and scalable streaming-based machine learning pipeline was implemented, enabling training on extremely large IoT datasets without memory limitations. Correlation analyses and class-mapping strategies provide a strong foundation for the next phases, where LightGBM, anomaly detection, and real-time integration with Suricata will be developed.

Overall, the project is on track and has completed all required milestones for the initial progress stage.

4. References

[1] IoT-23 Dataset, Stratosphere Laboratory, 2020.
[2] BoT-IoT Dataset, University of New South Wales, 2019.
[3] S. M. H. Bamakan et al., “Cyber Threat Detection for IoT Using Machine Learning,” *IEEE Access*, 2022.