Student Name: Carlos Nakagomi

Instructor: Quynh Nguyen

Langara College

Cyber Security Case Study

1. Scope of the proposed ML project
   1. Purpose

The objective of this project is to develop an AI-powered Intrusion Detection System (AI-IDS) that can accurately classify network traffic into normal or malicious activity. The AI-IDS leverages machine learning techniques, particularly Random Forest (RF), to detect security threats such as Denial-of-Service (DoS) attacks, phishing, and malware intrusions in real-time. The system aims to enhance cybersecurity resilience at Queensland University, ensuring data protection and regulatory compliance.

* + 1. Dataset Boundaries

The model will be trained on the CICIDS2017 dataset, using all available features to maximize intrusion detection accuracy. These include network traffic attributes, packet-based metrics, statistical measures, flow characteristics, TCP flag indicators, and rate-based metrics. The target variable classifies network traffic as benign or malicious, covering attack types such as DoS, Botnet, and Infiltration.

* + 1. Generalization and Applicability

The model is designed to detect intrusions in real-time within Queensland University’s network, ensuring continuous monitoring and protection. Its adaptability is maintained through continuous learning and periodic retraining with updated attack data.

* 1. Goals

The project aims to develop a machine learning-powered Intrusion Detection System (IDS) capable of detecting and classifying network intrusions in real time. The focus is on optimizing the Random Forest model to achieve high detection accuracy while minimizing false positive rates. Ensuring adaptability and scalability is essential, allowing the IDS to be deployed across various network environments. The system will enhance cybersecurity resilience by enabling automated threat detection and response while aligning with cybersecurity standards and best practices to meet regulatory compliance requirements.

* 1. Deliverables

The project deliverables include comprehensive data preprocessing and feature engineering to ensure data integrity and model performance. This involves handling missing values, duplicates, and outliers while normalizing and scaling network traffic features. Techniques such as SMOTE and class weighting will be applied to balance the dataset. The model development and optimization phase will focus on training a Random Forest-based IDS using the CICIDS2017 dataset, conducting feature importance analysis to enhance efficiency. Performance evaluation will involve measuring accuracy, precision, recall, F1-score, and AUC-ROC. Cross-validation techniques will be used to ensure model robustness and reliability. The final deployment and integration phase will implement the IDS in real-time network monitoring systems while ensuring scalability for university-wide security applications.

1. Feature Selection
   1. Handling missing values

The dataset has no missing values, which means that all observations are complete. This ensures that no additional preprocessing is needed for handling null values, allowing to proceed with feature selection, dimensionality reduction (such as PCA), and model training without concerns about data imputation.

* 1. Handling Duplicated Rows

The initial dataset contained 2,438,052 rows and 122 columns. A total of 2,360 duplicate rows were identified and removed, meaning that the new dataset has 2,435,692 rows and 122 columns.

* 1. Handling outliers

It was plotted boxplots, and it was revealed a significant outliers in many numeric features, which were handled using an interquartile range (IQR)-based method. For each column, values outside 1.5 times the IQR from the first and third quartiles were identified as outliers and replaced with the column’s median to preserve the dataset's structure and integrity. This process ensures a clean and robust dataset, ready for further analysis and modeling, while minimizing the risk of outlier-induced bias.

* 1. Correlation

A Pearson correlation analysis was conducted to explore feature relationships within the dataset. Features with a correlation above 0.85 were identified as highly related, helping to understand redundancy and potential dependencies. The goal was to gain insights into data structure rather than feature selection for modeling. By analyzing correlation patterns, we observed significant associations among multiple features, highlighting areas where redundant information may exist.

* 1. Splitting data

The dataset was split into training and testing sets to facilitate model evaluation. The features were separated from the target variable before applying an 80-20 split using train\_test\_split from Scikit-Learn. Eighty percent of the data was allocated for training (X\_train, y\_train), ensuring the model learns patterns effectively, while the remaining 20% (X\_test, y\_test) was reserved for testing to assess generalization performance.

1. Features selection
   1. Standardize and PCA

Principal Component Analysis (PCA) was applied to check the number of variables that should be kept, the result was 34. Numeric columns were standardized using StandardScaler to ensure uniform feature scaling. The number of components needed to retain 95% variance was identified, ensuring essential attack patterns were preserved. This is crucial in cybersecurity for effective anomaly detection and classification, improving model efficiency while maintaining accuracy.

* 1. Chi-Square test

The Chi-Square test was conducted to evaluate the relationship between categorical features and the target variable in a cybersecurity dataset, aiding in selecting significant features for modeling. Categorical features in X\_train were identified and encoded numerically before applying the Chi-Square test. Features with p-values below 0.05 were selected as statistically significant, ensuring that only relevant features are retained to improve detection accuracy and reduce noise. The test results showed strong associations with the target variable: flow\_id (Chi2 Score: 2.45 × 10¹¹, P-Value: 0.0), timestamp (1.61 × 10¹¹, 0.0), src\_ip (1.16 × 10⁸, 0.0), dst\_ip (1.15 × 10⁸, 0.0), and protocol (2.38 × 10⁵, 0.0).

* 1. ANOVA

The ANOVA F-test was performed to assess the relationship between numerical features and the target variable in the cybersecurity dataset. This method helps identify the most relevant numerical features for classification tasks. Numerical features in X\_train were identified and used for the ANOVA test. Features with p-values below 0.05 were considered statistically significant, ensuring that only relevant attributes were retained for model efficiency and improved detection accuracy. The test results showed strong associations between several features and the target variable. Notable features with high significance include src\_port (ANOVA Score: 1675.47, P-Value: 0.0), dst\_port (8689.18, 0.0), duration (3167.00, 0.0), packets\_count (28997.94, 0.0), and fwd\_packets\_count (32175.74, 0.0).

* 1. Mutual Information

The analysis revealed several key features with strong associations to the target variable. The highest-ranked features were timestamp (0.8803), flow\_id (0.7585), mean\_header\_bytes (0.6383), total\_header\_bytes (0.6145), dst\_ip (0.5973), src\_ip (0.5800), fwd\_packets\_IAT\_max (0.5684), fwd\_packets\_IAT\_total (0.5636), and packet\_IAT\_total (0.5565). These results indicate that packet timing (IAT), IP addresses, and header-related metrics are highly informative in distinguishing network behavior. The high importance of timestamp and flow\_id suggests that traffic patterns over time play a crucial role in classification.

* 1. Relief

The Relief algorithm was applied to evaluate the relevance of features in the cybersecurity dataset by assigning importance scores based on their ability to distinguish between different classes. The top-ranked features with the highest Relief scores include src\_ip (0.5659), max\_header\_bytes (0.5509), fwd\_min\_header\_bytes (0.5498), mean\_header\_bytes (0.4780), bwd\_max\_header\_bytes (0.4675), and down\_up\_rate (0.4565). These results highlight the significance of IP addresses, packet headers, and transmission rates in identifying network behavior patterns. The dominance of src\_ip and flow\_id suggests that source identification and flow tracking play a crucial role in differentiating between normal and malicious traffic.

* 1. Random forest

To identify the most important features for intrusion detection, a Random Forest model was trained on the dataset. Since the dataset contained non-numeric columns (flow\_id, timestamp, src\_ip, dst\_ip, and protocol), they were first converted into numeric format using label encoding. The encoded dataset was then used to train a Random Forest classifier with 100 trees to compute feature importance scores. The top-ranked features identified by the model were flow\_id (0.118924), src\_ip (0.114593), timestamp (0.052368), rst\_flag\_counts (0.043380), and dst\_ip (0.039295). The dominance of flow\_id and src\_ip suggests that network flow identifiers and source IP addresses are critical in distinguishing attack patterns. The presence of rst\_flag\_counts further highlights the significance of TCP reset flags in detecting malicious activity.

* 1. Selected features

The final feature set consists of categorical and numerical features that exhibited high statistical significance and relevance for classification. The essential categorical features include flow\_id, timestamp, src\_ip, dst\_ip, and protocol. Flow\_id was identified as highly significant in the Chi-Square test and ranked as the most important feature in Random Forest, confirming its strong relationship with the target variable. Timestamp had the highest Mutual Information score, reinforcing the idea that network event timing plays a crucial role in attack detection. Source and destination IP addresses were found to be important in Chi-Square, Relief, and Random Forest, highlighting their relevance in classifying network behavior. The protocol feature was selected in Chi-Square, showing its importance in distinguishing different types of traffic and attack patterns. The key numerical features retained include src\_port, dst\_port, duration, packets\_count, and fwd\_packets\_count, which were identified as significant in the ANOVA test and Random Forest, demonstrating their role in measuring network activity and detecting anomalies. Mean\_header\_bytes and total\_header\_bytes ranked high in Mutual Information and Relief, reinforcing the importance of packet structure in attack detection. Fwd\_packets\_IAT\_max and fwd\_packets\_IAT\_total were important in Mutual Information, suggesting that inter-arrival times between packets can indicate anomalies. Rst\_flag\_counts, highlighted in Random Forest, is a crucial feature for identifying malicious TCP reset flags. Max\_header\_bytes and min\_header\_bytes showed strong presence in Relief, confirming that packet size variations are highly relevant for classification. Additional numerical features retained for classification include bwd\_packets\_count, fwd\_avg\_bytes, bwd\_avg\_bytes, fwd\_header\_bytes, bwd\_header\_bytes, fwd\_IAT\_mean, bwd\_IAT\_mean, active\_mean, idle\_mean, fwd\_psh\_flags, bwd\_psh\_flags, fwd\_urg\_flags, bwd\_urg\_flags, and packet\_loss\_ratio. Several features were excluded due to their low statistical significance, minimal variance, or lack of contribution to classification. The removed features include bwd\_ece\_flag\_counts, bwd\_rst\_flag\_counts, active\_min, idle\_max, fwd\_bulk\_total\_size, and subflow\_bwd\_packets, which had a Relief score of zero, indicating no contribution to classification. Payload\_bytes\_variance had minimal impact compared to other selected features. Features with near-zero variance across samples were eliminated as they do not provide meaningful variation for classification.

1. Balancing classes

Before SMOTE, the dataset was overwhelmingly dominated by benign traffic, accounting for 73.31% of the total observations, while several attack categories had extremely low representation. To reverse this issue, SMOTE was applied to the dataset, generating synthetic samples for underrepresented attack classes. After applying SMOTE, every class was upsampled to 1,428,992 instances, effectively balancing the dataset. With this transformation, no class had disproportionate representation, with each category now accounting for 7.14% of the total dataset.

1. Comparison of the models

The baseline model demonstrated exceptional results, achieving an accuracy of 0.9982 along with precision, recall, and F1-score values that also reached 0.9982. As for the proposed model, it showed an accuracy of 0.7300, precision of 0.9661, recall of 0.7300, and an F1 score of 0.8244. The baseline model demonstrated a classification performance with an AUC score close to 1.0, this result means that the baseline model was highly effective in distinguishing between benign and malicious network traffic. The proposed model exhibited a lower AUC score of 0.9294.

1. SHAP

The SHAP analysis was conducted to identify the most influential features in the model’s predictions. The results highlighted src\_port, duration, and total\_payload\_bytes as key contributors. Src\_port exhibited distinct clusters, indicating certain port ranges are strongly linked to malicious activity. A graph of different colored lines

AI-generated content may be incorrect.A graph of different colored lines

AI-generated content may be incorrect.

In the first SHAP plot, features such as src\_port, duration, and total\_payload\_bytes emerged as the most influential variables in the model’s decision-making process. The distribution of SHAP values for src\_port shows clear clusters, indicating that certain source port ranges are closely associated with specific types of network behavior.

The second SHAP plot reinforced the significance of packets\_count, protocol, and fwd\_packets\_count, further emphasizing that packet volume and protocol types are crucial indicators of network intrusions. Consistent prominence of src\_port and duration across both plots validated their role in improving detection accuracy.

1. Conclusion

By strategically addressing threshold values, dataset reduction, and balancing techniques, this study successfully developed a Random Forest-based Intrusion Detection System capable of effectively identifying malicious network traffic. The combination of feature selection methods, class balancing, and model tuning ensured robust performance across diverse attack scenarios. Moving forward, the AI-IDS can be further enhanced by incorporating ensemble methods, adaptive learning techniques, and additional real-world testing to strengthen its resilience against evolving cyber threats.

1. Further considerations

In the development of this AI-powered Intrusion Detection System (AI-IDS), several key considerations were addressed to ensure optimal performance, data integrity, and model generalization. These aspects were critical in refining the dataset, tuning model parameters, and achieving robust results. A significant aspect of this study was fine-tuning the threshold values used for classification. While Random Forest can handle classification well, adjusting the probability threshold for class prediction was necessary to strike a balance between precision and recall. By trying different threshold values, it was possible to identify an optimal threshold of **0.45.** This threshold effectively reduced false positives without compromising detection accuracy, which is critical for cybersecurity applications where false alarms can overwhelm security teams.