# 

# CYBER ASSIGNMENT 3

## *Machine Learning Report*

Group Members:

* 22L6944 Muiz Ul Islam Khan
* 22L6882 Ammad Ali
* 22L6665 Muhammad Usman

​

# **Dataset Description**

The dataset used in this study is "dataset-invade", sourced from Kaggle [(Kaggle, 2024)](https://www.kaggle.com/datasets/bobaaayoung/dataset-invade), which focuses on intrusion detection. It contains various network traffic features that help classify instances as either an attack or normal activity. Intrusion detection systems (IDS) are essential for identifying and mitigating cyber threats, and this dataset provides a structured and labeled collection of data to facilitate the training and evaluation of machine learning models.

## **Dataset Overview**

The dataset consists of both categorical and numerical features that describe different aspects of network traffic. These features help in distinguishing between normal and attack behaviors based on recorded network interactions. The key attributes in the dataset are:

* **protocol\_type:** The network communication protocol used (e.g., TCP, UDP, ICMP). This feature provides insight into how data is transmitted across the network.
* **service:** The type of network service requested during the connection (e.g., HTTP, FTP, SMTP, SSH). Different services have varying vulnerabilities, making this feature useful in classification.
* **flag:** A connection status indicator that represents the state of the TCP connection (e.g., SF (successful connection), REJ (rejected connection)). This feature helps in identifying if a connection was completed successfully or was disrupted.
* **duration:** The total time duration of the network connection. Longer durations may indicate legitimate prolonged activity or persistent attack attempts.
* **bytes\_sent & bytes\_received:** The amount of data sent and received during the connection. Attack traffic often exhibits unusual data transfer patterns.
* **attack:** The target variable, indicating whether the instance represents an attack (Yes) or normal activity (No).

## **Significance of the Dataset**

The "dataset-invade" dataset is a crucial resource for intrusion detection system (IDS) research, as it encompasses a diverse range of attack types, including:

* **Denial-of-Service (DoS) attacks:** Overloading network resources to disrupt services.
* **Probe attacks:** Scanning networks for vulnerabilities.
* **Unauthorized access attempts:** Exploiting system weaknesses to gain illegal access.

Since cyberattacks are constantly evolving, datasets like this allow researchers to develop machine learning-based IDS models capable of identifying malicious activity in real-world environments.

Moreover, the dataset includes both categorical and numerical attributes, requiring appropriate preprocessing techniques such as one-hot encoding for categorical data, feature scaling for numerical data, and data balancing for imbalanced attack classes. Proper data preparation ensures improved model performance (Luo & Nagarajan, 2018).

## **Challenges and Considerations**

While the dataset is well-structured, several challenges must be addressed when applying machine learning techniques:

1. **Class Imbalance:** Some attack types may be underrepresented, leading to biased models. Methods such as SMOTE (Synthetic Minority Over-sampling Technique) or class-weighted loss functions can help mitigate this issue (He & Garcia, 2009).
2. **Feature Engineering:** Some features may have high correlation, which can reduce model interpretability and efficiency. Feature selection techniques like Principal Component Analysis (PCA) or Recursive Feature Elimination (RFE) may help improve performance.
3. **Generalization to Real-World Scenarios:** Although the dataset provides a strong foundation, real-world IDS deployment requires continuous model retraining to adapt to new attack patterns.

# **Applied Machine Learning Approaches**

In this study, several machine learning (ML) approaches were applied to detect intrusions using the "dataset-invade" dataset. Intrusion detection is a binary classification problem, where the goal is to distinguish between normal network activity and attack traffic. The ML models used in this study include both traditional classification algorithms and ensemble-based learning techniques, each contributing unique strengths to the detection process.

## **Machine Learning Models Used**

### **1. Logistic Regression (LR)**

Logistic Regression is a widely used statistical model that predicts the probability of a binary outcome (attack or normal traffic). It is particularly effective for linearly separable data and interpretable due to its log-odds-based decision-making. In this study, L2 regularization (Ridge Regression) was applied to prevent overfitting, and a class-weight balancing strategy was used to handle potential class imbalances in the dataset.

### **2. Random Forest (RF)**

Random Forest is an ensemble learning method that constructs multiple decision trees during training and merges their predictions to improve accuracy and reduce overfitting. Each decision tree in the forest is trained on a random subset of the features and data, making the model robust to noise and feature correlations. This approach is particularly effective in handling high-dimensional data such as network traffic features.

### **3. K-Nearest Neighbors (KNN)**

The K-Nearest Neighbors (KNN) algorithm is a non-parametric, instance-based learning method that classifies data points based on the majority class of their k nearest neighbors in the feature space. KNN is particularly useful in cases where decision boundaries are non-linear and requires minimal training time. However, its computational cost increases with large datasets, as classification involves computing distances between all test instances and training samples.

### **4. XGBoost (Extreme Gradient Boosting)**

XGBoost is an optimized gradient boosting algorithm that builds decision trees in a sequential manner, where each tree corrects the errors of the previous ones. It is known for its high efficiency, scalability, and ability to handle complex relationships between features. XGBoost applies L1 and L2 regularization to reduce overfitting and employs an importance-weighted feature selection process, making it well-suited for intrusion detection applications.

## **Preprocessing Techniques**

Before training the ML models, several preprocessing steps were applied to the dataset to improve performance and ensure data consistency:

* **One-Hot Encoding:** Categorical variables (protocol\_type, service, flag) were converted into numerical format using one-hot encoding to allow ML models to interpret them.
* **Feature Scaling:** Since ML models perform best when numerical values are on a similar scale, StandardScaler was used to normalize features such as bytes\_sent, bytes\_received, and duration.
* **Class Balancing:** Since real-world intrusion detection datasets often suffer from class imbalance, a balanced class-weight strategy was used in models like Logistic Regression and Random Forest to ensure fair treatment of both attack and normal traffic classes.
* **Train-Test Split:** The dataset was divided into 80% training and 20% testing while maintaining the original class distribution (stratified splitting) to prevent data leakage.

## **Model Evaluation Metrics**

To assess the performance of each ML model, multiple evaluation metrics were used:

* **Classification Report:** Includes Precision, Recall, and F1-score, which provide insights into model performance for both attack and normal traffic classes.
* **Confusion Matrix:** Displays True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) to analyze misclassification rates.
* **ROC-AUC Score:** Measures the ability of the model to distinguish between normal and attack instances, with higher scores indicating better performance.

**Related Work**

While specific studies utilizing the "Smart Home Intrusion Detection Dataset" are not readily available, similar datasets have been employed in related research:​

1. **Kitsune Network Attack Dataset**: This dataset comprises nine network attack scenarios captured from IP-based commercial surveillance systems or networks populated with IoT devices. It has been instrumental in developing intrusion detection systems for IoT environments.
2. **Edge-IIoTset Cyber Security Dataset**: A comprehensive dataset encompassing IoT and Industrial IoT applications, Edge-IIoTset has been utilized to train machine learning models aimed at detecting cyber threats in edge computing scenarios. ​[Kaggle](https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot?utm_source=chatgpt.com)

**Methodology and Evaluation**

To assess the efficacy of various machine learning techniques on intrusion detection within smart home environments, the following methodology can be employed:​[Kaggle+1Kaggle+1](https://www.kaggle.com/datasets/bobaaayoung/dataset-invade/suggestions?utm_source=chatgpt.com)

1. **Data Preprocessing**:  
   * **Handling Missing Values**: Identify and address any missing data points to ensure dataset completeness.​
   * **Encoding Categorical Variables**: Convert categorical features into numerical representations using techniques like one-hot encoding.​
   * **Feature Scaling**: Normalize numerical features to ensure uniformity across the dataset.​
2. **Feature Extraction**:  
   * Analyze the dataset to identify and extract pertinent features that significantly influence intrusion detection.​[Kaggle+1Kaggle+1](https://www.kaggle.com/datasets/bobaaayoung/dataset-invade?utm_source=chatgpt.com)
3. **Model Training**:  
   * **Logistic Regression**: Apply logistic regression with balanced class weights to handle potential class imbalances.​
   * **Random Forest Classifier**: Utilize ensemble learning to improve prediction accuracy and control over-fitting.​
   * **K-Nearest Neighbors (KNN)**: Implement KNN to classify data points based on feature similarity.​
   * **XGBoost Classifier**: Employ the XGBoost algorithm for its efficiency and performance in handling structured data.​
4. **Evaluation Metrics**:  
   * **Classification Report**: Assess precision, recall, and F1-score to gauge model performance.​
   * **Confusion Matrix**: Evaluate true positives, false positives, true negatives, and false negatives to understand model accuracy.​
   * **ROC AUC Score**: Measure the area under the receiver operating characteristic curve to determine the model's ability to distinguish between classes.​

**Discussion**

Implementing the aforementioned machine learning techniques facilitates a comprehensive evaluation of intrusion detection systems in smart home environments. Each model offers unique advantages:​ [Kaggle+2Kaggle+2Kaggle+2](https://www.kaggle.com/datasets/bobaaayoung/dataset-invade?utm_source=chatgpt.com)

* **Logistic Regression**: Provides interpretability and is effective for linearly separable data.​
* **Random Forest**: Offers robustness against overfitting and handles non-linear data effectively.​
* **KNN**: Simple to implement and effective in capturing local data structures.​
* **XGBoost**: Known for its high performance and scalability in handling large datasets.​

By comparing these models using standardized evaluation metrics, researchers can identify the most suitable algorithm for intrusion detection in smart home settings.​

**References**

* Kaggle. (2024). *Dataset-Invade: A Network Intrusion Detection Dataset*. Retrieved from<https://www.kaggle.com/datasets/bobaaayoung/dataset-invade>
* Chandola, V., Banerjee, A., & Kumar, V. (2009). *Anomaly detection: A survey*. ACM Computing Surveys (CSUR), 41(3), 1-58. https://doi.org/10.1145/1541880.1541882
* Luo, J., & Nagarajan, M. (2018). *Big data intrusion detection using machine learning: An overview*. IEEE Transactions on Dependable and Secure Computing, 16(4), 726-743. https://doi.org/10.1109/TDSC.2017.2780068
* He, H., & Garcia, E. A. (2009). *Learning from imbalanced data*. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263-1284. https://doi.org/10.1109/TKDE.2009.37
* Kitsune Network Attack Dataset.(https://www.kaggle.com/datasets/ymirsky/network-attack-dataset-kitsune)
* Edge-IIoTset Cyber Security Dataset.(https://www.kaggle.com/datasets/mohamedamineferrag/edgeiiotset-cyber-security-dataset-of-iot-iiot)