**FMDB** *Transactions on Sustainable Intelligence and Security*

# Explainable Cyber Attack Detection Using Machine Learning and SHAP-Based Feature Interpretation on Network Traffic Data

**Christoph Wech[1], K. Srinarayani[2,*], Chiara Brüggemann[3], Andrin Bertschi[4], Fabian Richter[5], Deepa Nair[6]**

[1,4]Department of Computer Science, Secure and Trustworthy Systems Group, ETH Zurich, Zürich, Switzerland.
[2]Department of Computer Science in Artificial Intelligence and Machine Learning, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[3,5]Department of Physics, Palffy-Buß Group, University of Wurzburg, Würzburg, Bavaria, Germany.
[6]Department of Logistics, Tampines Industries Pte. Ltd., East Region, Singapore.
chris.wech@inf.ethz.ch[1], srinarak@srmist.edu.in[2], chiara.brueggemann@uni-wuerzburg.de[3], andrin.bertschi@inf.ethz.ch[4], fabian.richter@uni-wuerzburg.de[5], deepa.nair@tampinesindustries.sg[6]

**Abstract:** Cybersecurity threats continue to evolve in complexity, making accurate and interpretable intrusion detection systems essential for modern network environments. This paper presents an explainable cyber attack detection framework utilizing machine learning techniques and SHapley Additive exPlanations (SHAP) for transparent decision-making. The proposed framework was evaluated on the CIC-IDS2017 dataset, which contains both benign and malicious network traffic representing various cyber attack scenarios. Three machine learning approaches were investigated, namely Isolation Forest, One-Class Support Vector Machine (One-Class SVM), and Random Forest. Isolation Forest and One-Class SVM were employed as unsupervised anomaly detection techniques, while Random Forest was used as a supervised attack classification model. Experimental results demonstrated that Random Forest significantly outperformed the anomaly detection approaches, achieving a Precision of 99.64%, Recall of 98.86%, F1-Score of 99.25%, and ROC-AUC of 99.98%. To improve transparency, SHAP was integrated to identify and explain the most influential network traffic features contributing to attack detection. The analysis revealed that destination port characteristics and packet length statistics were among the most significant indicators of malicious activity. The proposed framework combines high detection performance with explainability, making it suitable for practical cybersecurity applications and security monitoring environments.

**Keywords:** Intrusion Detection System; Cyber Attack Detection; Machine Learning (ML); Random Forest; Isolation Forest; Anomaly Detection; Shapley Additive Explanations (SHAP).

**Received on:** 30/08/2025, **Revised on:** 09/11/2025, **Accepted on:** 18/12/2025, **Published on:** 11/09/2026

**Journal Homepage:** https://www.fmdbpub.com/user/journals/details/FTSIS

**DOI:** https://doi.org/10.69888/FTSIS.2026.000804

**Cite as:** C. Wech, K. Srinarayani, C. Brüggemann, A. Bertschi, F. Richter, and D. Nair, "Explainable Cyber Attack Detection Using Machine Learning and SHAP-Based Feature Interpretation on Network Traffic Data," *FMDB Transactions on Sustainable Intelligence and Security,* vol. 1, no. 3, pp. 168–181, 2026.

**Copyright** © 2026 C. Wech *et al.,* licensed to Fernando Martins De Bulhão (FMDB) Publishing Company. This is an open access article distributed under CC BY-NC-SA 4.0, which permits copying, remixing, redistributing, transformation, and building upon the material in any medium so long as the original work is properly cited.

## 1. Introduction

*Corresponding author.

The rapid expansion of digital technologies, cloud computing platforms, Internet of Things (IoT) devices, and interconnected enterprise networks has significantly increased the exposure of organizations to cyber threats. Modern networks continuously generate massive volumes of traffic, creating opportunities for attackers to exploit vulnerabilities through denial-of-service attacks, brute-force attacks, botnet activities, web application attacks, and other sophisticated intrusion techniques. As cyberattacks continue to evolve in complexity and scale, traditional signature-based intrusion detection systems struggle to identify previously unseen attack patterns and often suffer from limited adaptability [8]; [10]. Machine learning has emerged as a promising solution for intelligent intrusion detection due to its capability to learn complex patterns from network traffic data and automatically distinguish malicious activities from legitimate communications [1]; [2]. Supervised machine learning techniques utilize labeled datasets to learn attack signatures and behavioral characteristics, while unsupervised approaches focus on identifying anomalies that deviate from normal network behavior. These approaches have demonstrated significant improvements in detection accuracy compared to conventional rule-based security mechanisms. The effectiveness of machine learning-based intrusion detection systems largely depends on the availability of realistic and comprehensive datasets. Earlier benchmark datasets such as KDD Cup 99 have been widely criticized for containing redundant records and failing to represent modern network environments [5]. To address these limitations, the CIC-IDS2017 dataset was developed to provide realistic network traffic containing both benign activities and contemporary attack scenarios, including Distributed Denial of Service (DDoS), brute-force attacks, web attacks, botnet activities, and infiltration attacks [3]; [4]. Consequently, CIC-IDS2017 has become one of the most widely adopted datasets for evaluating intrusion detection systems. Several studies have investigated the use of machine learning algorithms such as Random Forest, Support Vector Machines, Decision Trees, and ensemble learning techniques for cyber attack detection [7]; [11]; [12]; [13].

Among these approaches, Random Forest has consistently demonstrated strong performance due to its ability to handle high-dimensional data, reduce overfitting, and provide robust classification capabilities [6]; [18]. Similarly, anomaly detection techniques such as Isolation Forest and One-Class Support Vector Machines have gained attention for their capability to identify unknown attacks without relying on labeled attack data [10]; [15]; [16]; [17]. Despite the success of machine learning-based intrusion detection systems, a significant challenge remains in the interpretability of their decisions. Many models operate as black-box systems, making it difficult for cybersecurity analysts to understand why a particular network flow is classified as malicious. This lack of transparency can reduce trust and hinder practical deployment in real-world security operations environments [19]; [20]. Recent developments in Explainable Artificial Intelligence (XAI) have provided mechanisms to interpret machine learning predictions and identify the factors influencing classification decisions. SHapley Additive exPlanations (SHAP) has emerged as one of the most effective approaches for explaining model outputs by quantifying the contribution of individual features toward predictions. Motivated by these challenges, this paper proposes an explainable cyber attack detection framework that combines supervised and unsupervised machine learning techniques with SHAP-based feature interpretation. The proposed framework evaluates the performance of Isolation Forest, One-Class SVM, and Random Forest models on the CIC-IDS2017 dataset. Furthermore, SHAP is employed to provide transparent explanations for attack predictions and identify the most influential network traffic characteristics responsible for malicious behavior detection [1]. Experimental results demonstrate that the Random Forest classifier achieves superior detection performance while SHAP-based explainability enhances the transparency and trustworthiness of the intrusion detection process. The major contributions of this work are as follows:

- Implementation and comparative evaluation of Isolation Forest, One-Class SVM, and Random Forest models for cyber attack detection.
- Performance assessment using standard evaluation metrics including Precision, Recall, F1-Score, and ROC-AUC.
- Integration of SHAP-based explainability to interpret model predictions and identify significant network traffic features.
- Analysis of feature importance and attack characteristics using the CIC-IDS2017 intrusion detection dataset.
- Development of an explainable intrusion detection framework that combines high detection accuracy with model transparency.

## 2. Literature Review

Machine learning-based intrusion detection has become an active area of cybersecurity research due to its ability to automatically learn attack patterns from large-scale network traffic data. Over the past decade, researchers have explored various supervised, unsupervised, and hybrid learning approaches to improve detection accuracy while reducing false alarm rates. Farnaaz and Jabbar [6] investigated the application of Random Forest algorithms for network intrusion detection and demonstrated that ensemble-based approaches can effectively classify malicious traffic while maintaining high accuracy. Their findings highlighted the robustness of Random Forest in handling complex network traffic features. Similarly, Belavagi and Muniyal [7] conducted a comparative evaluation of supervised machine learning algorithms for intrusion detection. They reported that ensemble-based models generally outperform individual classifiers in terms of accuracy and stability. Buczak and Guven [8] presented a comprehensive survey of machine learning and data mining methods for cybersecurity applications.

Their work emphasized the growing importance of machine learning in modern intrusion detection systems and identified challenges related to scalability, feature engineering, and evolving attack patterns.

Ambusaidi et al. [9] proposed a feature selection-based intrusion detection framework and demonstrated that reducing irrelevant network attributes can significantly improve detection performance and computational efficiency. Anomaly-based intrusion detection has also received considerable attention due to its capability to identify previously unseen attacks. Garcia-Teodoro et al. [10] provided an extensive review of anomaly-based intrusion detection techniques and discussed their advantages in detecting novel threats. Chitrakar and Chuanhe [15] combined Support Vector Machines with clustering techniques to improve anomaly detection performance. Similarly, Thaseen and Kumar [16] introduced a hybrid intrusion detection model based on Principal Component Analysis and optimized SVM. At the same time, Mehmood and Rais [17] explored the use of Ant Colony Optimization for feature selection in SVM-based network anomaly detection systems. The development of realistic datasets has played a crucial role in advancing intrusion detection research. Tavallaee et al. [5] critically analyzed the KDD Cup 99 dataset and identified several limitations that affect the generalizability of machine learning models.

To overcome these issues, Sharafaldin et al. [4] introduced the CIC-IDS2017 dataset, which provides realistic network traffic and modern attack scenarios. Panigrahi and Borah [14] further analyzed the CIC-IDS2017 dataset and demonstrated its suitability for evaluating contemporary intrusion detection systems. Several recent studies have focused on improving intrusion detection through ensemble learning and advanced machine learning architectures. Ahmim et al. [11] proposed a hierarchical intrusion detection framework combining decision trees and rule-based approaches. Zhou et al. [12] developed an efficient intrusion detection system using feature selection and ensemble classification techniques, achieving significant improvements in detection performance. Chen et al. [13] introduced an ADASYN-Random Forest model that addressed class imbalance issues while maintaining high classification accuracy. More recently, Abdelaziz et al. [18] demonstrated the effectiveness of Random Forest-based intrusion detection systems and emphasized the importance of feature importance analysis in understanding attack behavior. Although significant progress has been achieved in attack detection accuracy, the interpretability of machine learning models remains a major challenge.

Molnar et al. [19] highlighted the importance of explainable machine learning and discussed various approaches for interpreting model decisions. Explainability is particularly important in cybersecurity environments where analysts require clear justification for attack alerts and automated decisions. Sharma et al. [20] applied explainable artificial intelligence techniques to intrusion detection systems and demonstrated that feature attribution methods can improve trust, transparency, and analyst understanding of attack predictions. Despite these advancements, existing studies often focus primarily on detection performance while providing limited insight into the reasoning behind model decisions. Furthermore, comparative evaluations involving both anomaly detection techniques and supervised classifiers combined with explainability mechanisms remain relatively limited. Therefore, this work addresses these gaps by evaluating Isolation Forest, One-Class SVM, and Random Forest models within a unified framework and integrating SHAP-based explainability to provide transparent and interpretable cyber attack detection.

## 3. Proposed Methodologies

The proposed Explainable Cyber Attack Detection Framework integrates anomaly detection techniques, supervised machine learning, and Explainable Artificial Intelligence (XAI) to identify malicious network activities and provide interpretable attack predictions. The methodology consists of six major stages: data preprocessing, binary attack labeling, anomaly detection using Isolation Forest, anomaly detection using One-Class Support Vector Machine (One-Class SVM), supervised attack classification using Random Forest, and explainability analysis using SHAP (SHapley Additive ExPlanations).

### 3.1. Data Preprocessing

The CIC-IDS2017 dataset consists of network traffic flow records collected over multiple days containing both benign and malicious activities. Initially, all CSV files were merged into a unified dataset. Missing and invalid values were identified and removed to improve data quality and model reliability. The original dataset contained 2,830,743 records and 79 attributes. After removing records containing missing values, the final dataset consisted of 2,827,876 records. Each network flow contains 78 numerical features describing packet statistics, flow characteristics, timing information, and protocol-specific attributes. To formulate a binary classification problem, all attack categories were grouped into a single attack class.

Benign traffic was assigned a label value of 0, while all malicious traffic categories were assigned a label value of 1. To simplify the classification task, the original multiclass labels were converted into a binary format by mapping Benign Traffic to 0 and Attack Traffic to 1. This binary label mapping enables the machine learning models to focus on distinguishing normal network behavior from malicious activities, making the detection process more efficient and suitable for binary intrusion detection.

After the label transformation, the dataset consisted of 2,271,320 benign traffic samples and 556,556 attack traffic samples, resulting in a total of 2,827,876 network traffic instances used for model training and evaluation.

### 3.2. Isolation Forest-Based Anomaly Detection

Isolation Forest is an unsupervised anomaly detection algorithm designed to identify unusual observations through random partitioning of the feature space. Unlike distance-based methods, Isolation Forest isolates anomalies by recursively splitting the dataset using randomly selected features and split values. Anomalous instances are generally easier to isolate because they differ significantly from normal observations. Consequently, anomalies tend to have shorter path lengths within the isolation trees. The anomaly score is calculated as:

$$s(x,n) = 2^{\left(-\left(E(h(x))/c(n)\right)\right)} \quad (1)$$

The anomaly score in the Isolation Forest algorithm is computed as $s(x,n)$, where $x$represents a data sample and $n$denotes the total number of samples in the dataset. The score is determined using the average path length $E(h(x))$, which represents the average number of splits required to isolate the sample across all isolation trees, and $c(n)$, which is the average path length of unsuccessful searches in a binary search tree of size $n$. The anomaly score ranges from 0 to 1, where values close to 1 indicate that the sample is highly likely to be anomalous, values around 0.5 suggest behavior similar to normal network traffic, and values below 0.5 indicate normal observations. In this study, the Isolation Forest algorithm was employed as an unsupervised anomaly detection technique to identify suspicious network traffic patterns without using attack labels during the training process, enabling the detection of previously unseen or unknown cyber threats.

### 3.3. One-Class Support Vector Machine

One-Class Support Vector Machine is another unsupervised anomaly detection technique that learns the boundary of normal network traffic. During training, the model attempts to create a decision boundary that encloses the majority of normal samples. Any traffic instance located outside this boundary is considered anomalous. The optimization objective of One-Class SVM is expressed as:

$$min(1/2)|\quad |w|\quad |^2 + (1/\nu n)\Sigma\xi_i - \rho \quad (2)$$

Subject to:

$$\left(w \cdot \varphi(x_i)\right) \geq \rho - \xi_i \quad (3)$$

The One-Class Support Vector Machine (One-Class SVM) algorithm constructs a decision boundary to distinguish normal data from anomalous observations. In this formulation, $w$represents the weight vector that defines the decision boundary, while $\rho$denotes the bias or offset parameter. The parameter $\nu$is a user-defined value that controls the expected proportion of anomalies and influences the trade-off between the decision boundary and classification errors. Here, $n$refers to the total number of training samples, $\xi_i$is the slack variable that permits a limited number of misclassifications, and $\phi(x_i)$represents the transformed feature space of the input sample $x_i$. The primary objective of the algorithm is to maximize the separation between normal traffic and anomalous observations while minimizing classification errors. In the proposed framework, One-Class SVM was trained exclusively on normal network traffic patterns and subsequently employed to detect suspicious network activities, making it effective for identifying unknown or previously unseen cyberattacks without requiring attack samples during training.

### 3.4. Random Forest-Based Attack Classification

Random Forest is a supervised ensemble learning algorithm consisting of multiple decision trees. Each tree is trained using a random subset of training samples and features. Final predictions are obtained through majority voting among all trees. The impurity of a node is measured using the Gini Index:

$$Gini(D) = 1 - \Sigma(p_i^2) \quad (4)$$

Where:

- Gini(D) = Impurity of dataset D
- $p_i$ = Probability of class i
- i = Class index

Lower Gini values indicate purer nodes and better class separation. The final Random Forest prediction is computed as:

$$\hat{y} = Mode\big(T_1(x), T_2(x), \ldots, T_n(x)\big) \quad (5)$$

The Random Forest classifier predicts the class label, denoted by $\hat{y}$, by combining the predictions of multiple decision trees through a majority voting mechanism. In this formulation, $T_1, T_2, \ldots, T_n$represent the individual decision trees within the ensemble, $n$denotes the total number of trees, Mode() refers to the majority voting operation used to determine the final prediction, and $x$represents the input network traffic sample. Each decision tree independently classifies the input sample, and the class receiving the highest number of votes is selected as the final prediction. For this research, the Random Forest classifier was configured with 200 decision trees, a maximum tree depth of 20, and a random state of 42 to ensure reproducibility of the experimental results. The model was trained using labeled network traffic data and subsequently evaluated on previously unseen test samples to assess its ability to accurately distinguish between benign and malicious network traffic.

### 3.5. Performance Evaluation Metrics

The effectiveness of the proposed framework was assessed using four standard classification metrics. Precision measures the proportion of predicted attacks that are actually attacks:

$$Precision = TP/(TP + FP) \quad (6)$$

In the evaluation of the proposed intrusion detection models, Precision measures the proportion of correctly identified attack instances among all samples predicted as attacks. Here, True Positives (TP) represent the number of malicious network traffic instances that are correctly classified as attacks. At the same time, False Positives (FP) denote the number of benign traffic instances that are incorrectly classified as attacks. A high Precision value indicates that the model generates a low false alarm rate, meaning that most traffic identified as malicious is indeed an actual attack. This metric is particularly important in intrusion detection systems, as excessive false positives can overwhelm security analysts with unnecessary alerts and reduce the overall effectiveness of cybersecurity monitoring. Recall measures the proportion of actual attacks that are successfully detected:

$$Recall = TP/(TP + FN) \quad (7)$$

In the evaluation of the proposed intrusion detection models, Precision measures the proportion of correctly identified attack instances among all samples predicted as attacks. Here, True Positives (TP) represent the number of malicious network traffic instances that are correctly classified as attacks. At the same time, False Positives (FP) denote the number of benign traffic instances that are incorrectly classified as attacks. A high Precision value indicates that the model generates a low false alarm rate, meaning that most traffic identified as malicious is indeed an actual attack. This metric is particularly important in intrusion detection systems, as excessive false positives can overwhelm security analysts with unnecessary alerts and reduce the overall effectiveness of cybersecurity monitoring:

$$F1 = 2 \times \big((Precision \times Recall)/(Precision + Recall)\big) \quad (8)$$

The F1-Score is a harmonic mean of Precision and Recall, providing a balanced measure of a model's classification performance, particularly when dealing with imbalanced datasets. A high F1-Score is achieved only when both Precision and Recall are simultaneously high, indicating that the model can accurately identify malicious traffic while maintaining a low false alarm rate. In addition to the F1-Score, the Receiver Operating Characteristic – Area Under the Curve (ROC-AUC) is used to evaluate the model's ability to distinguish between benign and malicious network traffic across various classification thresholds.

The ROC-AUC value ranges from 0 to 1, where a value of 1.0 represents perfect classification with complete separation between attack and benign traffic. In contrast, a value of 0.5 indicates performance equivalent to random guessing. Higher ROC-AUC values therefore reflect better discriminative capability and greater robustness of the intrusion detection model in identifying cyberattacks.

### 3.6. SHAP-Based Explainability Analysis

Although machine learning models can achieve high classification accuracy, understanding the reasoning behind their decisions is equally important in cybersecurity applications. To improve transparency, SHapley Additive exPlanations (SHAP) were incorporated into the proposed framework. SHAP is based on cooperative game theory and assigns a contribution value to each feature according to its influence on the model prediction. The SHAP value for a feature is calculated as:

$$\varphi_{\mathrm{i}} = \Sigma[(|S|!\,(|N| - |S| - 1)!)/|N|!] \times [f(S \cup i) - f(S)] \quad (9)$$

The SHAP (SHapley Additive exPlanations) value for a feature, denoted as $\phi_i$, measures the contribution of feature $i$to the model's prediction. In the SHAP formulation, $S$represents a subset of all available features, $N$denotes the complete set of features, $|\ S\ |$is the number of features in subset $S$, and $|\ N\ |$is the total number of features. The function $f(S)$corresponds to the model prediction using the subset of features, while $f(S \cup \{i\})$represents the prediction after adding feature $i$to the subset. The SHAP value quantifies how much a particular feature increases or decreases the prediction score by comparing these two predictions across all possible feature combinations. Positive SHAP values indicate that a feature increases the likelihood of the traffic being classified as an attack. In contrast, negative SHAP values suggest that the feature contributes toward a benign classification. In this study, SHAP was applied to the Random Forest model to determine the most influential network traffic features responsible for cyberattack detection. The resulting feature importance rankings provide transparent and interpretable insights into the model's decision-making process, thereby enhancing the explainability, reliability, and trustworthiness of the proposed intrusion detection framework.

### 3.7. Overall Framework Workflow

The complete workflow of the proposed system can be summarized as follows: The proposed methodology begins by collecting and merging the CIC-IDS2017 network traffic dataset into a unified dataset for analysis. The merged data is then preprocessed by removing missing, duplicate, and invalid records to ensure data quality and reliability. Next, the original attack categories are converted into binary labels, where normal traffic is represented as 0 and attack traffic as 1, simplifying the intrusion detection task into a binary classification problem. The cleaned dataset is subsequently divided into training and testing subsets for model development and performance evaluation. Two unsupervised anomaly detection algorithms, Isolation Forest and One-Class SVM, are first trained to identify suspicious network traffic patterns without relying on attack labels during training. A Random Forest classifier is then trained using the labeled data to perform supervised cyberattack classification. The performance of all models is assessed using standard evaluation metrics, including Precision, Recall, F1-Score, and ROC-AUC, to measure their effectiveness in detecting malicious traffic (Figure 1).

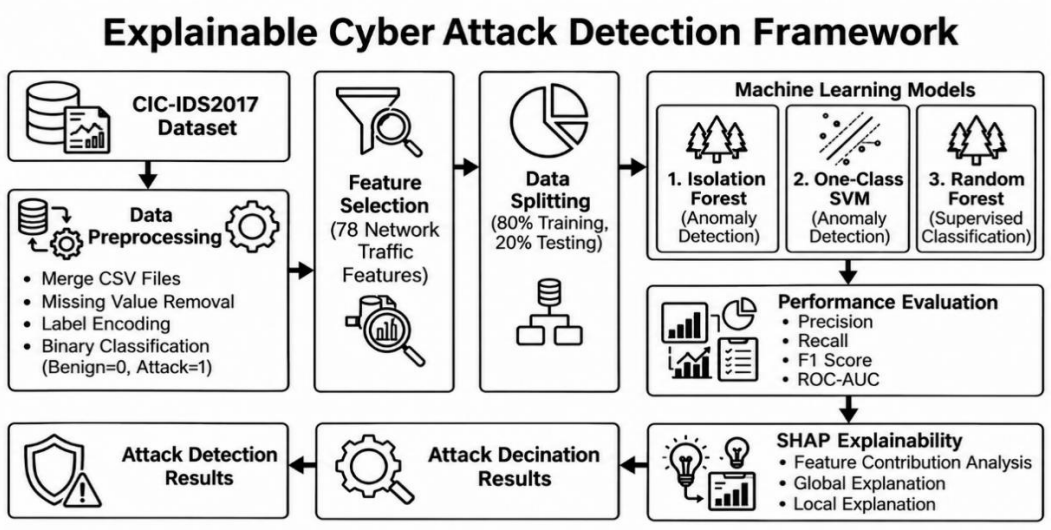



**Figure 1:** System architecture of the proposed explainable cyber-attack detection framework

To enhance the interpretability of the proposed framework, SHAP (SHapley Additive exPlanations) is applied to the Random Forest model to analyze the contribution of individual network traffic features toward prediction outcomes. Finally, the framework generates interpretable cyberattack predictions along with feature importance rankings, providing both accurate intrusion detection and transparent explanations of the model's decision-making process.

### 3.8. Dataset Description

#### 3.8.1. CIC-IDS2017 Dataset Overview

The experiments conducted in this study utilized the CIC-IDS2017 dataset developed by the Canadian Institute for Cybersecurity (CIC). The dataset was designed to overcome the limitations of earlier intrusion detection datasets by providing

realistic network traffic that closely resembles modern enterprise environments. It contains both normal user activities and multiple categories of cyber attacks executed in a controlled network infrastructure. The dataset captures complete network flow information generated from real-world scenarios involving web browsing, email communication, file transfers, and network services. In addition to benign activities, various attack scenarios were performed using attacker machines targeting victim systems. These attacks include Distributed Denial of Service (DDoS), Denial of Service (DoS), Port Scanning, Brute Force Attacks, Botnet Activities, Web Attacks, Infiltration Attacks, and Heartbleed Exploitation. The CIC-IDS2017 dataset has become one of the most widely used benchmark datasets in intrusion detection research due to its realistic traffic patterns, comprehensive feature representation, and diversity of attack categories.

### 3.8.2. Dataset Structure

The CIC-IDS2017 dataset is distributed across multiple CSV files, with each file representing network traffic captured during different days and time periods of organizational network activity. These network flow records were generated using CICFlowMeter, which extracts bidirectional flow-based features from raw packet capture (PCAP) files. The original dataset consists of eight CSV files corresponding to different traffic scenarios, including Monday-WorkingHours, Tuesday-WorkingHours, Wednesday-WorkingHours, Thursday-WorkingHours-Morning-WebAttacks, Thursday-WorkingHours-Afternoon-Infiltration, Friday-WorkingHours-Morning, Friday-WorkingHours-Afternoon-PortScan, and Friday-WorkingHours-Afternoon-DDos. Each record represents a single network flow and contains 78 numerical features along with a class label indicating whether the traffic is benign or belongs to a specific attack category. These features capture various aspects of network communication, including packet statistics, flow duration, protocol information, communication rates, packet inter-arrival times, and header-based characteristics. Important attributes include Destination Port, Flow Duration, Total Forward Packets, Total Backward Packets, Packet Length Mean, Packet Length Standard Deviation, Average Packet Size, Flow Bytes per Second, Flow Packets per Second, and Packet Length Variance. Collectively, these flow-based features provide comprehensive information about network behavior, enabling machine learning models to effectively distinguish between normal and malicious traffic patterns for accurate cyberattack detection.

### 3.8.3. Attack Categories

The CIC-IDS2017 dataset includes a diverse set of cyberattack categories that represent a wide range of real-world network security threats commonly encountered in enterprise environments. These attack classes encompass various types of denial-of-service attacks, brute-force authentication attempts, network scanning activities, botnet communications, web application attacks, infiltration attempts, and protocol vulnerabilities. Specifically, the dataset contains DDoS, DoS Hulk, DoS GoldenEye, DoS Slowloris, DoS SlowHTTPTest, Port Scan, FTP-Patator, SSH-Patator, Bot, Web Attack – Brute Force, Web Attack – XSS, Web Attack – SQL Injection, Infiltration, and Heartbleed attack categories. Each attack type exhibits distinct network traffic characteristics, such as variations in packet transmission rates, flow duration, protocol usage, connection patterns, and communication behavior. The inclusion of these diverse attack scenarios enables a comprehensive evaluation of intrusion detection models by assessing their ability to accurately detect and differentiate malicious activities from normal network traffic across multiple cyber threat types.

### 3.8.4. Dataset Preprocessing

Before model development, all individual CIC-IDS2017 CSV files were combined into a single unified dataset to facilitate comprehensive analysis and model training. During this integration process, feature names were standardized by removing unnecessary spaces, correcting formatting inconsistencies, and ensuring uniform attribute naming across all files. The merged dataset initially contained 2,830,743 network traffic records with 79 attributes, including 78 network flow features and 1 class label. As part of the preprocessing stage, the dataset was examined for missing and invalid values. Missing values were primarily identified in network traffic attributes such as Flow Bytes/s and Flow Packets/s, which could adversely affect the performance and reliability of machine learning models. To maintain data integrity, all records containing missing values were removed rather than imputed. After the cleaning process, the final dataset consisted of 2,827,876 records while retaining all 79 attributes, with a total of 2,867 invalid records removed. The resulting cleaned dataset provided a high-quality and consistent foundation for training, testing, and evaluating the proposed intrusion detection models.

### 3.8.5. Binary Class Construction

To formulate the intrusion detection task as a binary classification problem, all individual attack categories in the CIC-IDS2017 dataset were consolidated into a single malicious class, while normal network traffic was retained as the benign class. Accordingly, Benign Traffic was assigned the label 0, and Malicious Traffic was assigned the label 1. Following this binary label transformation, the dataset comprised 2,271,320 benign records and 556,556 attack records, resulting in a total of 2,827,876 network traffic instances. This binary representation simplifies the classification task by enabling machine learning

models to focus solely on distinguishing normal network behavior from malicious activity, thereby providing an effective framework for evaluating intrusion detection performance using standard binary classification metrics.

### 3.8.6. Class Distribution Analysis

Analysis of the processed CIC-IDS2017 dataset revealed a moderate class imbalance between benign and malicious network traffic. Benign traffic accounts for approximately 80.3% of the dataset, whereas attack traffic represents approximately 19.7% of the total network flow records. This distribution closely reflects real-world network environments, where normal communication significantly outweighs malicious activities (Figure 2).

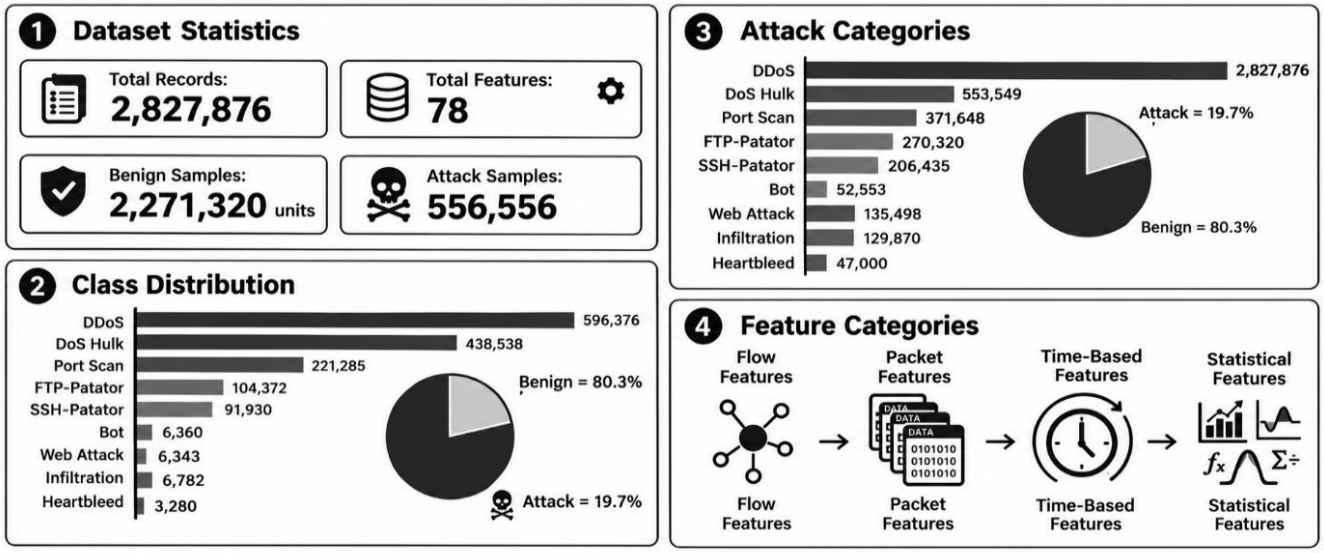



**Figure 2:** Overview of the CIC-IDS2017 dataset used in this study

Consequently, intrusion detection models must be capable of accurately identifying the relatively small proportion of attack traffic while maintaining a low false positive rate to avoid unnecessary security alerts. Evaluating models on such an imbalanced dataset provides a more realistic assessment of their effectiveness and robustness in practical cybersecurity applications.

### 3.8.7. Feature Categories

The 78 numerical features in the CIC-IDS2017 dataset can be broadly classified into several categories based on the type of network information they represent. Flow-based features describe the overall characteristics of network communication, including flow duration, packet counts, byte counts, and transmission rates. Packet-based features capture statistical properties of packet lengths, such as minimum, maximum, mean, standard deviation, and variance, providing insights into packet size distributions within each network flow. Time-based features characterize communication timing by measuring packet inter-arrival times, active and idle durations, and other temporal behaviors of network traffic. Header-based features contain protocol-specific information, including TCP flag counts such as SYN, ACK, PSH, FIN, and RST, which help identify abnormal connection patterns commonly associated with cyberattacks. Additionally, statistical features provide aggregated measures derived from network flows, summarizing communication behavior through various descriptive statistics. The combination of these diverse feature categories enables machine learning algorithms to effectively learn both normal network communication patterns and the distinctive behavioral characteristics of malicious traffic, thereby improving the accuracy and robustness of intrusion detection systems.

### 3.8.8. Dataset Significance

The CIC-IDS2017 dataset provides a comprehensive representation of modern network environments and attack scenarios. Its realistic traffic generation methodology, diverse attack categories, and extensive feature representation make it highly suitable for evaluating machine learning-based intrusion detection systems. Furthermore, the large number of records and inclusion of multiple attack types enable robust performance evaluation and generalization analysis. These characteristics make CIC-IDS2017 an appropriate benchmark dataset for developing explainable cyber attack detection frameworks.

## 4. Results and Discussions

This section presents the experimental results obtained from the implementation of Isolation Forest, One-Class Support Vector Machine (One-Class SVM), and Random Forest models on the CIC-IDS2017 dataset. The objective was to evaluate the effectiveness of anomaly detection techniques and supervised machine learning approaches for cyber attack detection. In addition, SHAP explainability analysis was performed to identify the most influential features contributing to attack classification.

### 4.1. Performance Comparison of Intrusion Detection Models

The performance of the three machine learning models was evaluated using Precision, Recall, and F1-Score. Table 1 summarizes the obtained results. The Random Forest classifier significantly outperformed both anomaly detection approaches across all evaluation metrics. The model achieved a Precision of 99.64%, Recall of 98.86%, and F1-Score of 99.25%, demonstrating its ability to distinguish malicious traffic from benign network activities accurately.

**Table 1:** Performance comparison of machine learning models

| Model | Precision | Recall | F1-Score |
|---|---|---|---|
| Isolation Forest | 0.448 | 0.450 | 0.449 |
| One-Class SVM | 0.371 | 0.490 | 0.422 |
| Random Forest | 0.996 | 0.989 | 0.993 |

In contrast, Isolation Forest and One-Class SVM exhibited lower detection performance due to their unsupervised learning nature and inability to leverage attack labels during training (Figure 3).

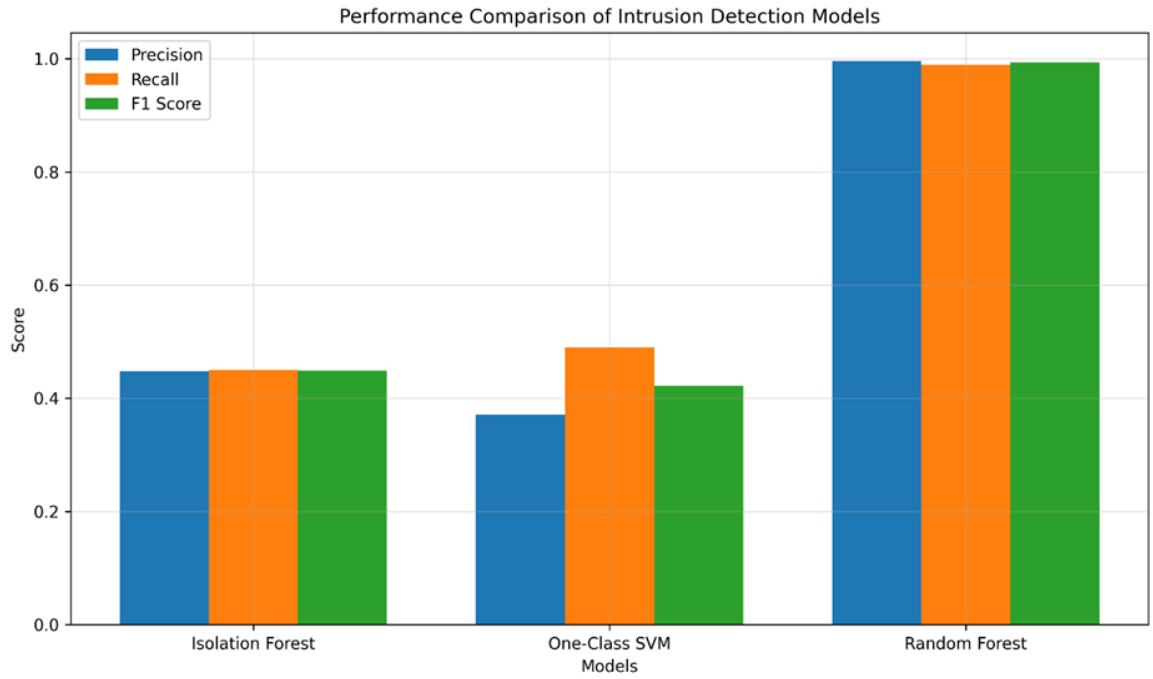



**Figure 3:** Shows the comparison of precision, recall, and F1-score for isolation forest, one-class SVM, and random forest

The superior performance of Random Forest can be attributed to its ensemble learning mechanism, which combines multiple decision trees to reduce variance and improve generalization. Furthermore, the algorithm effectively captures complex relationships among network traffic features.

### 4.2. Confusion Matrix Analysis

To further analyze the performance of the Random Forest classifier, a confusion matrix was generated using the testing dataset. The confusion matrix provides detailed information regarding correctly and incorrectly classified instances. The Random Forest

model correctly classified 16,036 benign samples and 3,905 attack samples. Only 14 benign samples were incorrectly identified as attacks, while 45 attack samples were misclassified as benign traffic. These results indicate an extremely low false positive rate and false negative rate. The confusion matrix demonstrates the robustness of the proposed framework in distinguishing between normal and malicious network traffic, making it suitable for real-world intrusion detection applications (Figure 4).

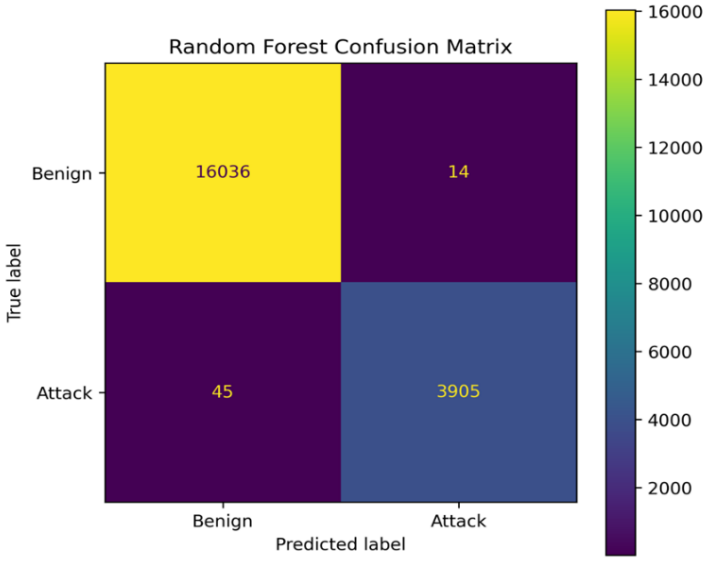



**Figure 4:** Expands the confusion matrix of the random forest classifier

### 4.3. ROC Curve Analysis

The Receiver Operating Characteristic (ROC) curve was used to evaluate the classification capability of the Random Forest model across multiple decision thresholds. The obtained ROC-AUC score was 0.9998, indicating near-perfect discrimination between benign and attack traffic (Figure 5).

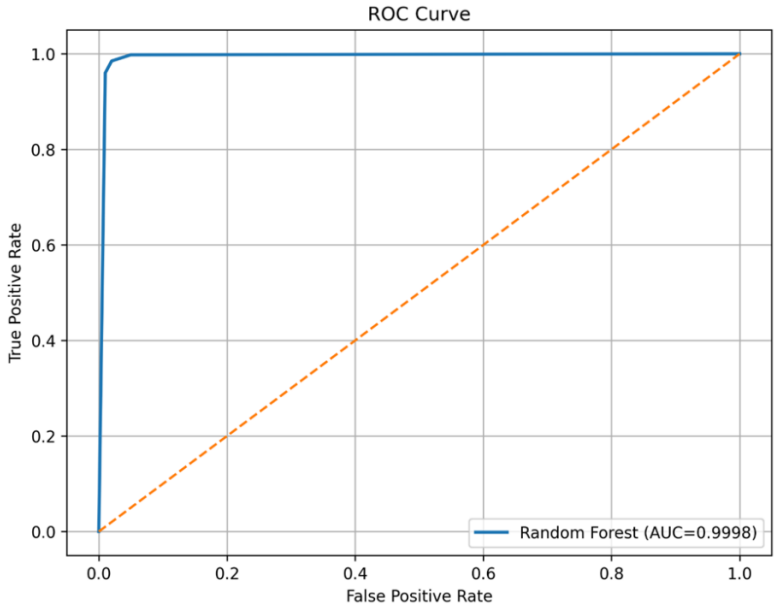



**Figure 5:** Shows the ROC curve of the random forest classifier

A ROC-AUC value close to 1 signifies that the classifier can effectively separate positive and negative classes with minimal overlap. This result confirms the excellent predictive capability of the Random Forest classifier on the CIC-IDS2017 dataset.

### 4.4. Dataset Class Distribution

The binary version of the CIC-IDS2017 dataset contains both benign and malicious network traffic samples. Analysis of the dataset revealed that benign traffic constituted approximately 80.3% of all records, while attack traffic represented approximately 19.7%. Although the dataset exhibits class imbalance, the Random Forest classifier successfully learned discriminative patterns and achieved high detection performance. The class distribution closely resembles real-world network environments, where malicious traffic represents only a small fraction of overall communications (Figure 6).

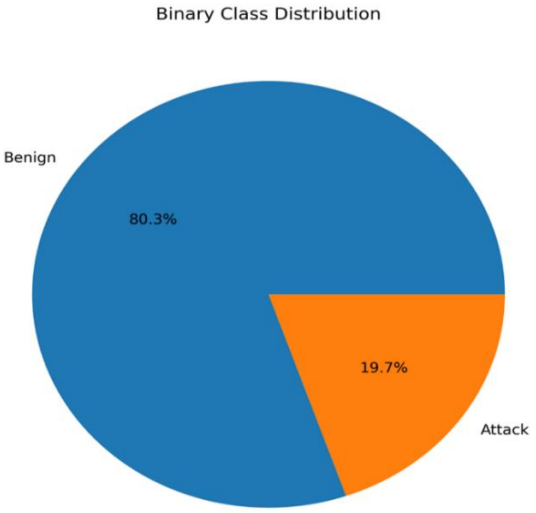



**Figure 6:** Shows the distribution of benign and attack samples in the CIC-IDS2017 dataset

### 4.5. Random Forest Feature Importance Analysis

Feature importance analysis was performed to identify the most significant network traffic attributes contributing to cyber attack detection (Figure 7).

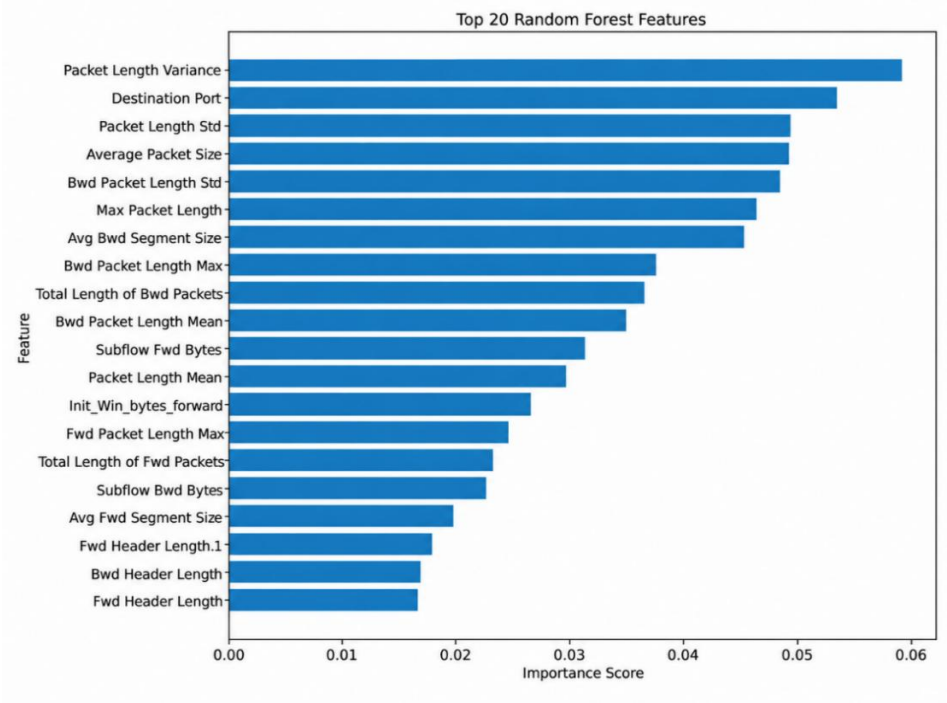



**Figure 7:** Illustrates the top twenty important features identified by the random forest classifier

The Random Forest classifier ranked Packet Length Variance, Destination Port, Packet Length Standard Deviation, Average Packet Size, and Backward Packet Length Standard Deviation among the most influential features. These results indicate that packet-size characteristics, communication patterns, and port utilization play critical roles in differentiating attack traffic from normal network behavior. Such features capture abnormalities in communication structure and flow statistics commonly exhibited by malicious activities.

### 4.6. SHAP Explainability Analysis

To improve transparency and interpretability, SHAP (SHapley Additive exPlanations) analysis was applied to the Random Forest classifier. SHAP computes the contribution of each feature toward the final prediction and provides a global understanding of model behavior. The SHAP analysis identified Destination Port as the most influential feature affecting attack classification, followed by Backward Packet Length Standard Deviation, Maximum Packet Length, Packet Length Variance, and Packet Length Standard Deviation (Figure 8).

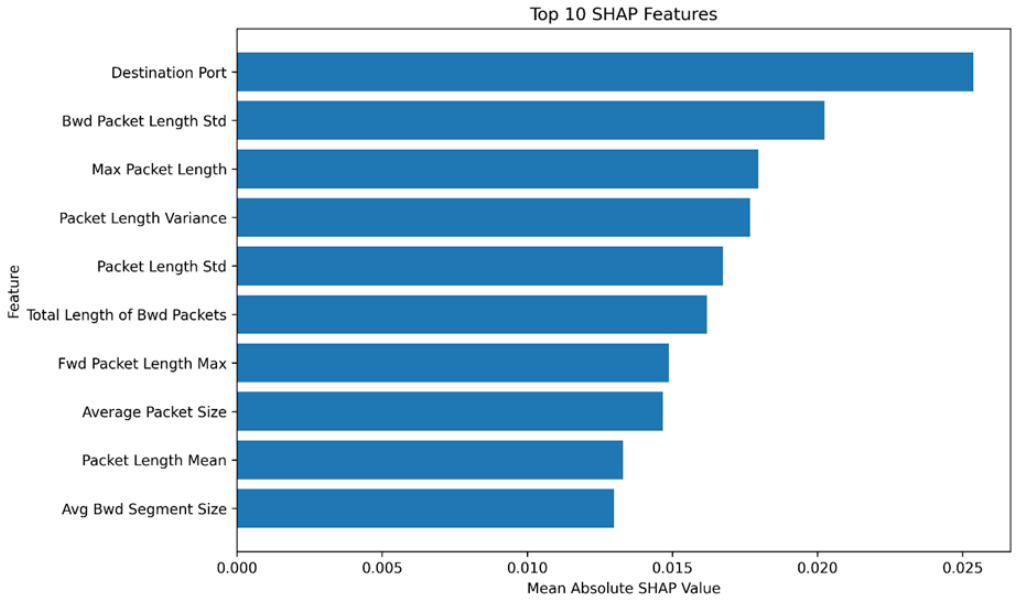



**Figure 8:** Shows the SHAP-based feature importance ranking highlighting the most influential network traffic attributes used for cyber-attack detection

These findings align closely with the Random Forest feature importance rankings, indicating consistency between model behavior and explainability results. The analysis demonstrates that attacks exhibit distinct packet-size distributions and communication characteristics compared to benign traffic. Therefore, these features provide valuable information for accurate cyber-attack detection. The experimental findings demonstrate that supervised learning approaches significantly outperform unsupervised anomaly detection methods when labeled attack data is available. While Isolation Forest and One-Class SVM successfully identified a portion of malicious traffic, their detection performance was considerably lower than that of Random Forest. The Random Forest classifier achieved an F1-Score of 99.25% and a ROC-AUC score of 99.98%, indicating exceptional classification performance. Furthermore, SHAP explainability analysis provided meaningful insights into the decision-making process by identifying the network traffic features most responsible for attack predictions. The combination of high detection accuracy and explainability makes the proposed framework suitable for deployment in practical cybersecurity environments where both performance and transparency are critical requirements.

## 5. Conclusion and Future Work

### 5.1. Conclusion

Cyber-attacks continue to pose significant threats to modern digital infrastructures, making the development of accurate and reliable intrusion detection systems an important research challenge. This study presented an Explainable Cyber Attack Detection Framework that combines anomaly detection techniques, supervised machine learning, and Explainable Artificial Intelligence (XAI) for effective network intrusion detection. The proposed framework was evaluated using the CIC-IDS2017 dataset containing both benign and malicious network traffic records. Three machine learning approaches were investigated,

namely Isolation Forest, One-Class Support Vector Machine (One-Class SVM), and Random Forest. The experimental results demonstrated that Random Forest significantly outperformed the anomaly detection models, achieving a Precision of 99.64%, Recall of 98.86%, F1-Score of 99.25%, and ROC-AUC of 99.98%. The performance comparison revealed that supervised learning approaches are highly effective when labeled attack data is available. Although Isolation Forest and One-Class SVM were capable of detecting abnormal network behavior, their detection accuracy was considerably lower than that of the Random Forest classifier. The confusion matrix analysis further confirmed the robustness of the proposed framework by demonstrating extremely low false positive and false negative rates. To improve transparency and interpretability, SHAP explainability analysis was integrated into the framework. The SHAP results identified Destination Port, Backward Packet Length Standard Deviation, Maximum Packet Length, Packet Length Variance, and Packet Length Standard Deviation as the most influential features contributing to attack detection. The explainability analysis provided valuable insights into the decision-making process of the classifier and enhanced the trustworthiness of the intrusion detection system. Overall, the proposed framework successfully combines high detection accuracy with model interpretability, making it suitable for practical cybersecurity applications, enterprise network monitoring, and intelligent threat detection environments.

### 5.2. Future Work

Although the proposed framework achieved excellent performance for binary intrusion detection, several opportunities exist for further enhancement and extension. First, future research can extend the framework from binary classification to multi-class attack classification, enabling the identification of specific attack categories such as DDoS, Port Scan, Botnet, Brute Force, Infiltration, and Web Attacks. Such an extension would provide more detailed threat intelligence and improve incident response capabilities. Second, deep learning architectures such as Long Short-Term Memory (LSTM), Gated Recurrent Units (GRU), Convolutional Neural Networks (CNN), and Transformer-based models can be explored to capture complex temporal and sequential relationships within network traffic data. Third, real-time intrusion detection can be implemented by integrating the trained model with live network monitoring tools and packet capture systems. This would enable continuous monitoring and immediate detection of malicious activities in operational environments. Fourth, Reinforcement Learning techniques can be incorporated to develop adaptive security systems capable of automatically responding to detected threats. Such systems could dynamically update defense strategies based on changing attack patterns and network conditions. Fifth, additional explainability methods such as LIME, Integrated Gradients, and Counterfactual Explanations can be investigated to provide complementary interpretations of attack predictions and improve analyst understanding. Finally, the proposed framework can be evaluated on additional intrusion detection datasets such as UNSW-NB15, CIC-DDoS2019, CICIoT2023, and CSE-CIC-IDS2018 to assess its generalization capability across different cybersecurity scenarios. By incorporating these enhancements, future intrusion detection systems can become more intelligent, adaptive, explainable, and capable of addressing emerging cybersecurity challenges in modern network environments.

**Acknowledgment:** The authors gratefully acknowledge the Secure and Trustworthy Systems Group, SRM Institute of Science and Technology at Ramapuram, University of Würzburg, and Tampines Industries Pte. Ltd. for their valuable support and contributions to this work.

**Data Availability Statement:** The data supporting this study are available from the corresponding author upon reasonable request.

**Funding Statement:** This research received no financial support or external funding.

**Conflicts of Interest Statement:** All authors declare no conflicts of interest regarding this publication.

**Ethics and Consent Statement:** The study followed ethical guidelines, and informed consent was obtained from all participants.

### References

1. J. Heaton, "Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning" *Genetic Programming and Evolvable Machines*, vol. 19, no. 10, pp. 305–307, 2018.
2. T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction," *Springer*, New York, United States of America, 2009.
3. A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of Tor traffic using time based features," *in Proc. 3rd Int. Conf. Inf. Syst. Security Privacy (ICISSP)*, Porto, Portugal, 2017.
4. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," *in Proc. 4th Int. Conf. Inf. Syst. Security Privacy (ICISSP)*, Funchal, Madeira, Portugal, 2018.

5. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," *in Proc. IEEE Symp. Comput. Intell. Security Defense Appl. (CISDA)*, Ottawa, Ontario, Canada, 2009.
6. N. Farnaaz and M. A. Jabbar, "Random forest modeling for network intrusion detection system," *Procedia Computer Science,* vol. 89, no. 8, pp. 213–217, 2016.
7. M. C. Belavagi and B. Muniyal, "Performance evaluation of supervised machine learning algorithms for intrusion detection," *Procedia Computer Science*, vol. 89, no. 8, pp. 117–123, 2016.
8. A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1153–1176, 2016.
9. M. A. Ambusaidi, X. He, P. Nanda, and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," *IEEE Trans. Computers*, vol. 65, no. 10, pp. 2986–2998, 2016.
10. P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *Computers & Security*, vol. 28, no. 1–2, pp. 18–28, 2009.
11. A. Ahmim, L. Maglaras, M. A. Ferrag, M. Derdour, and H. Janicke, "A novel hierarchical intrusion detection system based on decision tree and rules-based models," *in Proc. 15th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, Santorini, Greece, 2019.
12. Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, no. 6, pp. 1–17, 2020.
13. Z. Chen, L. Zhou, and W. Yu, "ADASYN–Random Forest based intrusion detection model," *in Proc. 4th Int. Conf. Signal Process. Mach. Learn. (SPML 2021)*, Beijing, China, 2021.
14. R. Panigrahi and S. Borah, "A detailed analysis of CICIDS2017 dataset for designing intrusion detection systems," *International Journal of Engineering & Technology*, vol. 7, no. 3.24, pp. 479–482, 2018.
15. R. Chitrakar and H. Chuanhe, "Anomaly detection using support vector machine classification with k-Medoids clustering," *in Proc. 3rd Asian Himalayas Int. Conf. Internet (AH-ICI)*, Kathmundu, Nepal, 2012.
16. I. S. Thaseen and C. A. Kumar, "Intrusion detection model using fusion of PCA and optimized SVM," *in Proc. 2014 Int. Conf. Contemporary Comput. Informatics (IC3I)*, Mysore, India, 2014.
17. T. Mehmood and H. B. M. Rais, "SVM for network anomaly detection using ACO feature subset," *in Proc. 2015 Int. Symp. Math. Sci. Comput. Res. (iSMSC)*, Ipoh, Malaysia, 2015.
18. M. T. Abdelaziz, A. Radwan, H. Mamdouh, A. S. Saad, A. S. Abuzaid, A. A. AbdElhakeem, S. Zakzouk, K. Moussa, and M. S. Darweesh, "Enhancing network threat detection with random forest-based NIDS and permutation feature importance," *Journal of Network and Systems Management*, vol. 33, no. 10, pp. 1–36, 2024.
19. C. Molnar, G. Casalicchio, and B. Bischl, "Interpretable machine learning—A brief history, state-of-the-art and challenges," *arXiv:2010.09337,* 2020. [Accessed by 19/06/2025].
20. B. Sharma, L. Sharma, C. Lal, and S. Roy, "Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning based approach," *Expert Systems with Applications*, vol. 238, no. 3, pp. 1–18, 2024.

***Publisher's Note:*** *The publisher remains impartial concerning jurisdictional claims in published maps and institutional affiliations. Responsibility for the content rests entirely with the authors and does not necessarily reflect the publisher's perspectives.*