Journal of Information and Cyber Security (JICS)
Volume 2, Issue 1, Date: 30- June -2026
ISSN: 2978-6797, DOI: 10.64820/AEPJICS.21.1.8.62026

A Data-Centric and Explainable Framework for Trustworthy Network Intrusion Detection

Mayes Nasser Ahmad [1] and Qasem Abu Al-Haija [2, *]
[1, 2] Faculty of Computer and Information Technology, Jordan University of Science and Technology, Jordan
Email: [1] mnahmad23@just.edu.jo ,[2] qsabuhaija@just.edu.jo,
*Corresponding Author

***Abstract*—Intrusion Detection Systems (IDSs) play a critical role in protecting modern networks against increasingly sophisticated cyber threats. This paper presents a data-centric and explainable machine learning framework for network intrusion detection using the CIC-IDS2017 benchmark dataset. The proposed framework integrates data preprocessing, SMOTE-based class imbalance mitigation, supervised machine learning, and explainable artificial intelligence techniques to improve both detection performance and transparency. Three widely used classifiers—Logistic Regression, Random Forest, and XGBoost—are evaluated using security-oriented metrics including accuracy, precision, recall, F1-score, confusion matrices, and ROC-AUC. Experimental results demonstrate the superiority of XGBoost, achieving 99.79% accuracy, 99.85% precision, 99.79% recall, and 99.81% F1-score, while achieving an ROC-AUC of 1.0 for binary intrusion detection. Furthermore, SHAP-based explainability analysis identifies the most influential network-flow features contributing to attack detection decisions. The results confirm that combining data-centric preprocessing, imbalance-aware learning, and explainable AI can significantly enhance the robustness, interpretability, and practical applicability of machine learning-based intrusion detection systems.**

***Keywords*—Intrusion Detection, Machine Learning, Network Security, XGBoost, Explainable AI, Random Forest.**

I. INTRODUCTION

Computer networks are becoming increasingly complex, leading to a rapid surge in cyberattacks, including Distributed Denial-of-Service (DDoS), penetration attempts, and application-layer threats. Traditional intrusion detection systems (IDSs), which rely primarily on signature-based approaches, are often ineffective in detecting emerging and zero-day attacks [1]. This limitation has driven the need for more reliable, data-driven security solutions. To address these challenges, machine learning (ML)-based intrusion detection systems have been widely adopted [2]. These systems learn patterns from network traffic data and automatically distinguish between benign and malicious activities. Realistic and diverse network traffic can be obtained from benchmark datasets such as CIC-IDS2017 [3], which provide labeled flow-based features representing various attack scenarios. However, despite the high accuracy reported in many studies, their effectiveness in real-world security applications remains limited due to issues such as class imbalance and lack of interpretability.

Recent studies have further emphasized the importance of machine learning, deep learning, feature selection, adversarial robustness, and large-scale datasets for improving intrusion detection systems [16]–[20]. In this work, we propose a data-centric intrusion detection framework based on the CIC-IDS2017 dataset. The proposed approach evaluates multiple supervised learning models, including Logistic Regression, Random Forest, and Extreme Gradient Boosting (XGBoost). Unlike prior studies, this work emphasizes a data-centric pipeline that integrates data preprocessing, imbalance handling, and explainability using SHAP to enhance security-driven evaluation. Special attention is given to security-relevant metrics, including recall, F1-score, confusion matrices, and ROC-AUC, with a focus on minimizing the number of missed attacks. As illustrated in Figure 1, the proposed approach follows a structured data-centric pipeline that prioritizes data quality, balanced learning, and interpretability for effective intrusion detection.

Figure 1 illustrates the motivation behind this study. While traditional signature-based intrusion detection systems struggle to identify emerging and zero-day attacks, machine learning-based approaches offer improved detection capabilities through automated pattern learning. Nevertheless, several challenges remain, including class imbalance, high-dimensional network features, and limited model explainability. To address these issues, the proposed framework integrates data-centric preprocessing, class-balancing with SMOTE, explainable AI via SHAP, and security-driven evaluation metrics to improve the robustness, generalizability, and trustworthiness of intrusion detection systems.

A. Background

Intrusion Detection Systems (IDSs) play a critical role in monitoring network traffic and identifying malicious activities that may compromise system security. Traditional signature-based IDSs rely on predefined rules to detect known threats but struggle to identify zero-day attacks. In contrast, anomaly-based IDSs detect deviations from normal behavior and are better suited to identifying unknown attacks; however, they often suffer from high false-positive rates [4]. Recent advances in machine learning have significantly enhanced anomaly-based IDSs by enabling them to learn complex patterns from labeled network traffic data. Models such as Logistic Regression, Random Forests, and gradient-

Received: 29-3-2026
Revised: 28-5-2026
Published: 30-6-2026

boosting techniques (e.g., XGBoost) are widely used due to their balance of performance, robustness, and interpretability [5]. Nevertheless, several challenges remain.

One of the primary challenges is class imbalance, where benign traffic dominates while certain attack types are underrepresented. Without appropriate handling techniques such as resampling or cost-sensitive learning, models tend to perform poorly in detecting rare but critical attacks. Additionally, network datasets often contain high-dimensional statistical features that require careful preprocessing and normalization to ensure reliable model performance [6]. Furthermore, traditional evaluation based solely on accuracy is insufficient for security applications. Metrics such as recall, F1-score, confusion matrices, and ROC-AUC provide a more comprehensive assessment of detection performance, particularly in identifying high-risk attacks. In this context, Explainable Artificial Intelligence (XAI) techniques, such as SHAP, play an essential role in improving transparency and enabling security analysts to interpret model decisions and take informed actions [7].

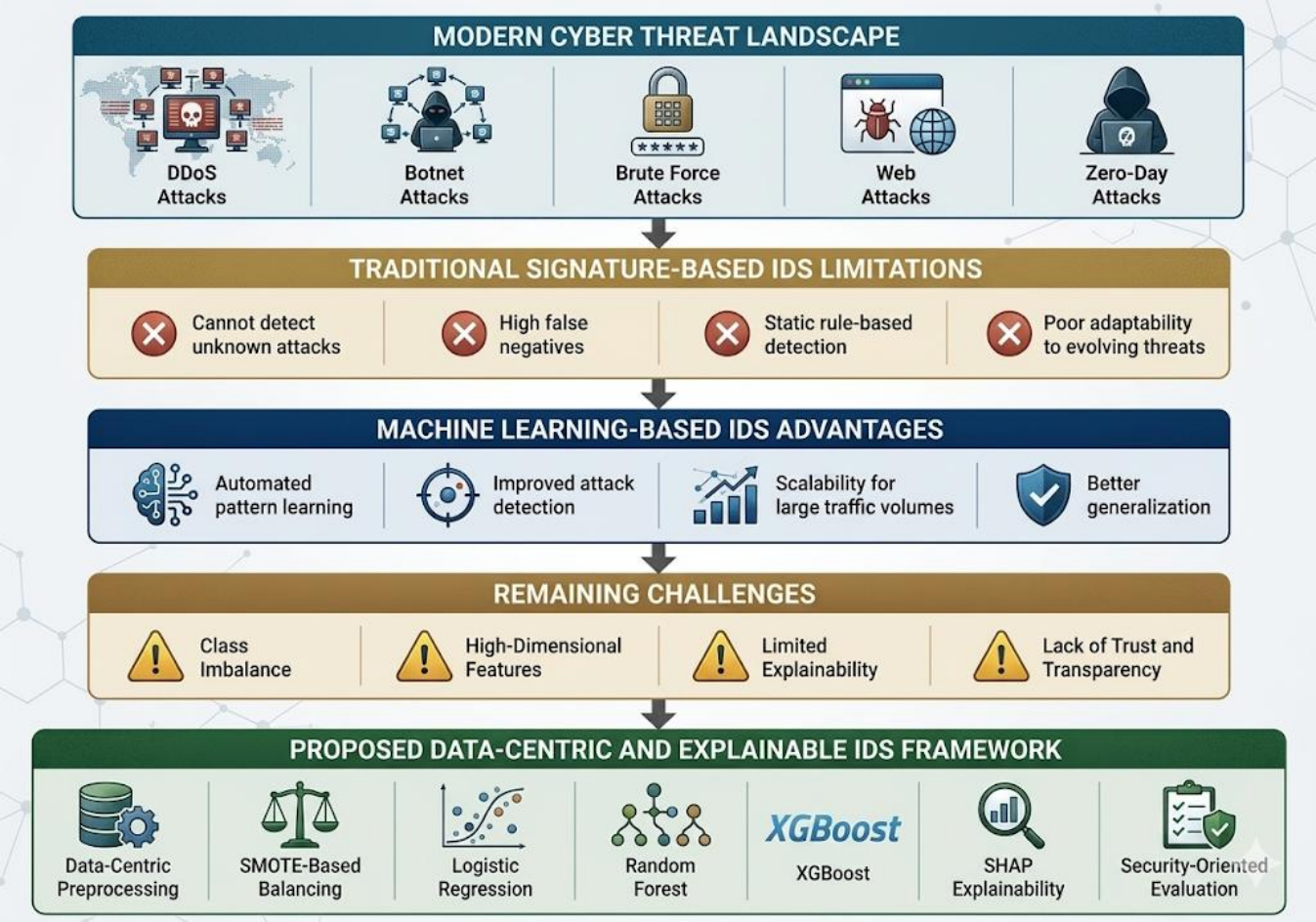

Fig. 1. Evolution of intrusion detection systems and the motivation for the proposed framework

B. Our Contributions

The main contributions of this work are summarized as follows:

- We propose a data-centric machine learning framework for network intrusion detection using the CIC-IDS2017 benchmark dataset, emphasizing data quality, preprocessing, and balanced learning.
- We integrate class imbalance mitigation through a hybrid sampling strategy based on SMOTE and undersampling to improve the detection of minority attack classes.
- We conduct a comparative evaluation of three widely used supervised learning models, namely Logistic Regression, Random Forest, and XGBoost, using security-oriented performance metrics.
- We incorporate Explainable Artificial Intelligence (XAI) using SHAP to improve the transparency and interpretability of intrusion detection decisions.
- We demonstrate that XGBoost achieves superior intrusion detection performance, attaining 99.79% accuracy, 99.85% precision, 99.79% recall, and 99.81% F1-score on the CIC-IDS2017 dataset.

C. Paper Structure

The remainder of this paper is organized as follows. Section II reviews related studies on machine learning-based intrusion detection, class imbalance handling, and explainable AI. Section III presents the proposed research methodology, including dataset description, preprocessing, feature engineering, imbalance mitigation, model training, and explainability analysis. Section IV discusses the experimental results and evaluates the performance of the

proposed framework using multiple security-oriented metrics. Finally, Section V concludes the paper and outlines future research directions.

II. Related Work

In recent years, the application of machine learning (ML) to network intrusion detection has gained significant attention, particularly with the availability of realistic benchmark datasets such as CIC-IDS2017 and CIC-IDS2018. These datasets provide high-volume, diverse, and labeled network traffic that closely resembles real-world enterprise environments, enabling effective evaluation of data-driven intrusion detection systems (IDS) [3].

A. Machine Learning IDS on CIC-IDS2017

Early studies utilizing CIC-IDS2017 demonstrated the effectiveness of classical supervised learning models. Sharafaldin et al. [3] introduced the dataset and highlighted its suitability for flow-based intrusion detection using statistical features extracted via CICFlowMeter [8]. Building on this, several studies employed baseline classifiers such as Logistic Regression (LR), Decision Trees, and k-Nearest Neighbors, achieving moderate performance while emphasizing the efficiency and interpretability of LR [9]. As research progressed, ensemble learning models, particularly Random Forest (RF), emerged as highly effective for intrusion detection. Abdelaziz et al. [10] applied RF with permutation feature importance and achieved a weighted F1-score of 99.8%, demonstrating its ability to capture nonlinear attack patterns and handle noisy data. Similarly, Ferrag et al. [4] reported that RF consistently outperforms individual classifiers because it can combine multiple decision boundaries. More recently, gradient boosting methods, especially XGBoost, have shown superior performance on the CIC-IDS2017 dataset. Several studies reported accuracy exceeding 99.9% and ROC-AUC values above 0.99 in both binary and multiclass classification settings [11], highlighting the effectiveness of boosting techniques in modeling complex network traffic behaviors.

B. Class Imbalance Handling

A major challenge in CIC-IDS2017 is severe class imbalance, where benign traffic dominates while certain attack types, such as SQL Injection, are underrepresented. Without appropriate handling, machine learning models tend to be biased toward majority classes, leading to poor detection of rare but critical attacks. To address this issue, the Synthetic Minority Over-sampling Technique (SMOTE) has been widely adopted. Originally proposed by Chawla et al. [12], SMOTE generates synthetic samples for minority classes to improve class distribution. Subsequent studies, including Saranya et al. [13], demonstrated that SMOTE significantly enhances detection performance for low-frequency attack categories.

C. Feature Selection and Explainability

Another important challenge in CIC-IDS2017 is high feature dimensionality, as the dataset contains more than 80 flow-based features. Feature selection techniques are commonly used to reduce complexity while maintaining high detection performance. Recent studies have increasingly focused on explainability to improve trust in intrusion detection systems. For example, Chen et al. [14] applied SHAP-based feature selection, reducing the feature space to fewer than 15 attributes while achieving over 98% accuracy. Explainable AI (XAI) methods, such as SHAP and permutation feature importance, provide valuable insights into model behavior, enabling security analysts to understand decision-making processes and identify critical attack indicators [15].

D. Research Gap

Despite these advancements, most existing studies primarily focus on improving model performance without adequately addressing data-centric aspects of intrusion detection, such as preprocessing, handling imbalanced data, and integrated feature engineering. Additionally, explainability is often treated as a separate component rather than being tightly coupled with the detection pipeline. Therefore, there is a need for a unified, data-centric, and explainable framework that integrates preprocessing, class imbalance handling, and interpretable machine learning to enhance both detection performance and practical usability in real-world security environments. To provide a clearer comparison of existing approaches, Table 1 summarizes key studies by model, contributions, strengths, and limitations.

TABLE I. Comparison of Related Work on CIC-IDS2017

Ref.	Study	Model(s) Used	Dataset	Key Contribution	Strengths	Limitations
[3]	Sharafaldin et al.	Statistical features (CICFlowMeter)	CIC-IDS2017	Dataset creation and feature extraction	Realistic dataset; diverse attacks	No advanced ML modeling
[9]	Early ML Studies	LR, DT, k-NN	CIC-IDS2017	Baseline ML models for IDS	Fast, interpretable models	Limited detection performance
[10]	Abdelaziz et al.	Random Forest	CIC-IDS2017	RF with feature importance	High F1-score (99.8%); robust to noise	Limited interpretability
[4]	Ferrag et al.	RF, ML models	Multiple datasets	Comparative ML evaluation	Strong ensemble performance	Limited focus on data preprocessing
[11]	Recent Studies	XGBoost	CIC-IDS2017	Boosting-based IDS models	Very high accuracy (>99.9%)	Risk of overfitting; weak explainability
[13]	Saranya et al.	ML + SMOTE	CIC datasets	Class imbalance handling	Improved minority class detection	May introduce synthetic bias
[14]	Chen et al.	ML + SHAP	CIC datasets	Feature selection + explainability	Reduced features; interpretable results	Limited pipeline integration
This Work	Proposed Framework	LR, RF, XGBoost + SHAP	CIC-IDS2017	Data-centric + explainable IDS pipeline	Balanced learning + interpretability	Requires further real-world validation

III. RESEARCH METHODOLOGY

A. Overall System Architecture

The pipeline-based system includes data acquisition, preprocessing, feature engineering, training, and security-centric evaluation. This architecture is also designed to guarantee scalability, robustness against class imbalance, and high accuracy in detecting network attacks. The architecture of the proposed IDS is shown in Figure 2. The methodology adopts a flow-based intrusion detection approach, where network traffic is represented using statistical features extracted at the flow level, enabling efficient, real-time analysis compared to packet-level inspection.

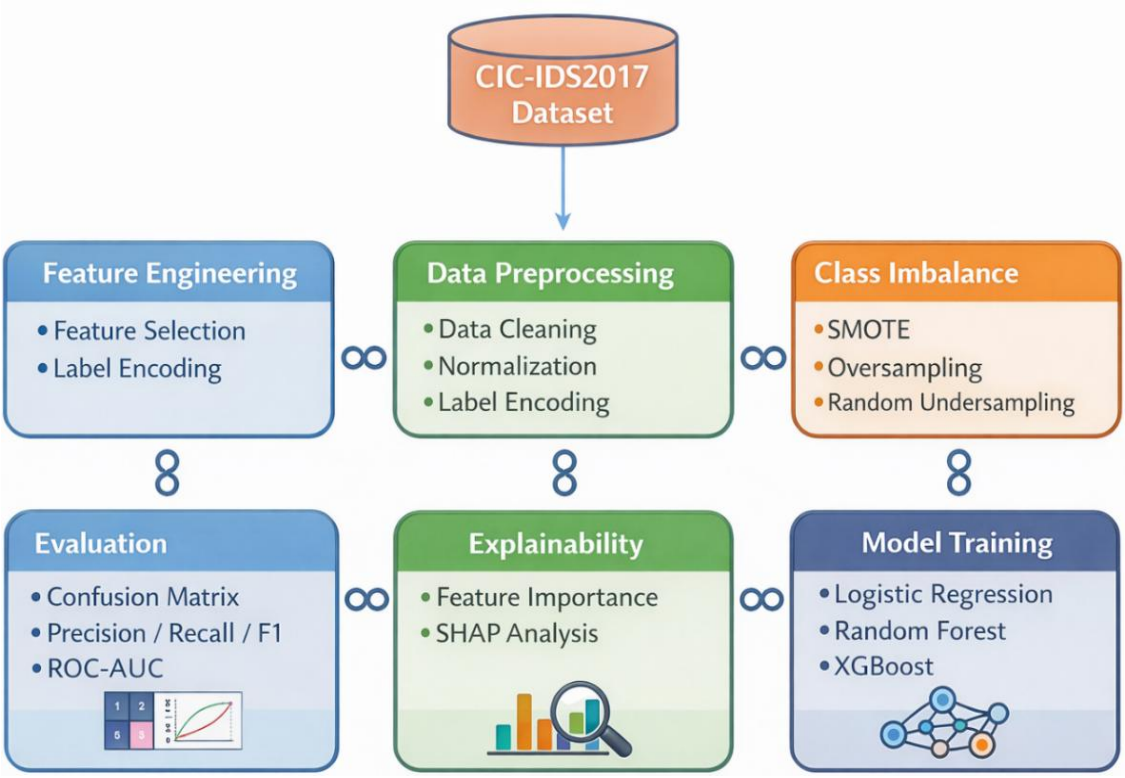

Fig. 2. Architecture of the proposed machine learning–based intrusion detection framework.

B. Dataset

The experiments here are performed using the CIC-IDS2017 dataset, a publicly available benchmark created by the Canadian Institute for Cybersecurity. The dataset includes labeled network flow records for benign traffic, as well as a wide range of contemporary attack scenarios: brute force, denial-of-service, web-based, infiltration, botnet, and DDoS. Network flows were generated from real traffic captures and analyzed by CICFlowMeter, yielding more than 80 statistical features per flow. This dataset is widely used in intrusion detection research and is considered a valid reflection of real-world network behavior [3].

C. Data Preprocessing and Cleaning

Due to the large scale and heterogeneity of CIC-IDS2017, extensive preprocessing is applied before model training:

1. **Column normalization:** Column names are cleaned to remove trailing and leading spaces. 2.
2. **Missing and infinite values:** Infinite values are replaced with NaN, and missing values are imputed using the median of each numeric feature.
3. **Zero-variance feature removal:** Features with no variance across samples are removed, as they do not contribute to classification.
4. **Duplicate removal:** Duplicated flow logs are identified and eliminated.
5. **Data type optimization:** Numeric variables are downcast to make them occupy less space in memory without losing the information.

These operations improve data quality, numerical stability, and efficient training.

D. Feature Engineering and Label Encoding

The dataset consists of over 80 numerical flow features, organized by packet length statistics, inter-arrival times, a few statistical flow-duration parameters, and protocol-level behavior.

- **Feature selection:** We keep only the numeric features for modeling.
- **Label encoding:** The attack type labels are encoded into numerical form for multiclass classification.
- **Binary target generation:** A secondary binary label is obtained to be used as a support in security examination, e.g., an analysis of ROC-AUC.

This dual-label approach allows achieving high-resolution attack localization and high-level threat recognition.

E. Handling Class Imbalance

To address this challenge, in our setting, we use SMOTE to generate synthetic minority attack class samples:

— Utilization of random under-sampling is used to minimize the dominant majority benign class.

— A mixed sampling pipeline that enables a more balanced class distribution and still maintains enough training data.
— This is an important step toward improving recall for high-risk attack types despite their rarity.

F. Model Training

The supervised machine learning models employed:

1. **Logistic Regression (LR):** Because of its simplicity, fast training time, and interpretability, it also happens to be the baseline model.
2. **Random Forest (RF):** A tree-based ensemble model that uses multiple decision trees to detect nonlinear attack patterns and improve robustness.
3. **XGBoost:** An optimized gradient boosting approach for large-scale data, with predictive power and high computational efficiency.

The dataset is split into training and testing sets using a 70:30 stratified split to maintain class distributions. To maintain numerical stability, feature scaling is performed using StandardScaler.

G. Evaluation Strategy

Finally, the models were evaluated against real-world cybersecurity standards using security-oriented metrics, e.g., Accuracy, precision, recall (F1-score for multiclass), confusion matrices (or attack-level error analysis), ROC-AUC for binary benign vs. malicious detection, and per-class recall identifying high-risk missed attacks.

H. Explainability and Security Insights

To enhance transparency and trust in the IDS, Feature importance analysis was performed for tree-based models to identify key attack indicators, and SHAP was applied to explain both global feature influence and individual predictions.

I. Experimental Setup

All experiments were conducted using Google Colab Pro running Python 3.11. The computational environment consisted of an NVIDIA Tesla T4 GPU (16 GB of VRAM), 25 GB of RAM, and a Linux-based runtime. Data preprocessing and analysis were performed using Pandas (v2.2.2), NumPy (v1.26.4), and Scikit-learn (v1.5.1). Machine learning models were implemented using Scikit-learn and XGBoost (v2.1.1), while data visualization was performed using Matplotlib (v3.9.2) and Seaborn (v0.13.2). The dataset was split into a 70:30 stratified training and test set. Logistic Regression was configured with max_iter=1000 and the LBFGS optimizer. Random Forest employed 100 decision trees with a maximum depth of 20 and the Gini impurity criterion. XGBoost was configured with 100 estimators, learning rate=0.1, maximum depth=6, subsample=0.8, colsample_bytree=0.8, and objective='multi'. Feature standardization was performed using StandardScaler, and class imbalance was addressed using SMOTE combined with random undersampling. Random seeds were fixed at 42 to ensure reproducibility of experimental results.

IV. Results and Discussion

This section presents the experimental results obtained from applying machine learning models to the CIC-IDS2017 dataset and provides a security-oriented discussion of their effectiveness.

A. Performance Analysis

The results shown in Table 1 show that Logistic Regression provides a baseline level of performance but struggles with complex and minority attack patterns. Random Forest significantly improves detection accuracy through its ensemble learning. XGBoost achieves near-perfect performance across all metrics, demonstrating superior generalization and robustness. The high performance is influenced by dataset characteristics (class separability and preprocessing), and further validation on unseen datasets is required.

TABLE II. Overall Performance Comparison on CIC-IDS2017

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	0.8552	0.9644	0.8552	0.8998
Random Forest	0.9370	0.9949	0.9370	0.9643
XGBoost	0.9979	0.9985	0.9979	0.9981

B. Confusion Matrix Analysis

Figure 3 presents the normalized confusion matrices and illustrates that XGBoost minimizes both false positives and false negatives across most attack categories. Logistic Regression exhibits higher confusion between benign traffic and rare attacks such as Web-based intrusions. Random Forest shows improved classification but still misclassifies some low-frequency attack types. These results highlight the advantage of the boosted tree models in learning complex decision boundaries. Figure 3 shows that XGBoost significantly reduces false negatives, which is critical for security applications.

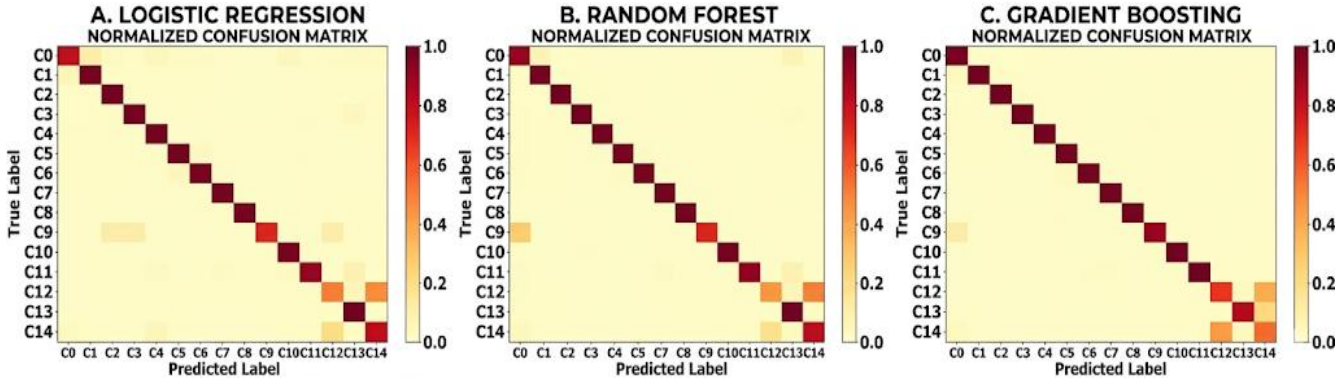

Fig. 3. Normalized Confusion Matrices of Evaluated Intrusion Detection Models

C. *Per-Class Detection Performance*

The per-class recall analysis reveals that high-impact attacks, such as DDoS, DoS Hulk, PortScan , and SSH, Achieve Recall Values Close to 1.0 in XGBoost. Nevertheless, rare types of attack, including Web Attacks, remain difficult to predict, though the performance comparison is clear in Figure 4. This indicates that it is essential to handle class imbalance and assess feature relevance for reliable intrusion detection. Flow duration and packet length were dominant indicators of attack behavior.

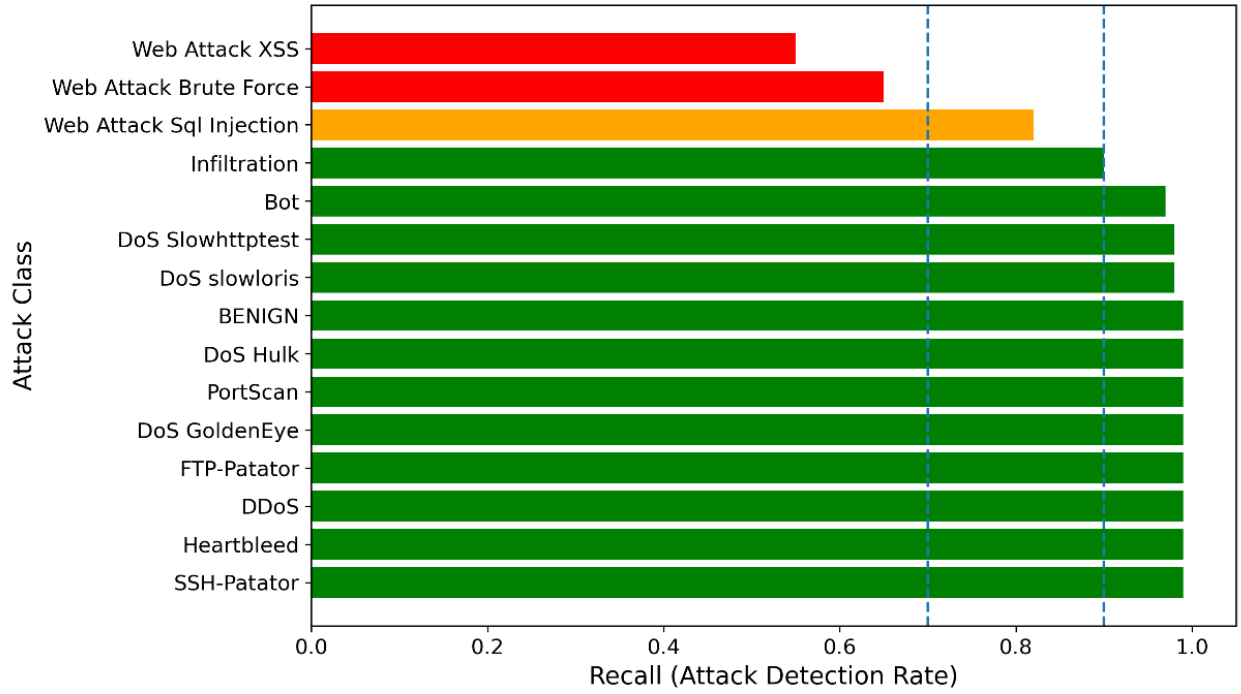

Fig. 4. Per-Class Recall of Attack Categories Using XGBoost

D. *ROC Curve Analysis*

We further validate the model's performance by generating Binary classification plots of benign versus malicious traffic. A Random Forest AUC of 0.9983 and an XGBoost AUC of 1.0000 indicate high discrimination performance even at low false-positive rates (Figure 5). This performance is crucial for deployment in the real world, where excessive false alerts can overwhelm security analysts.

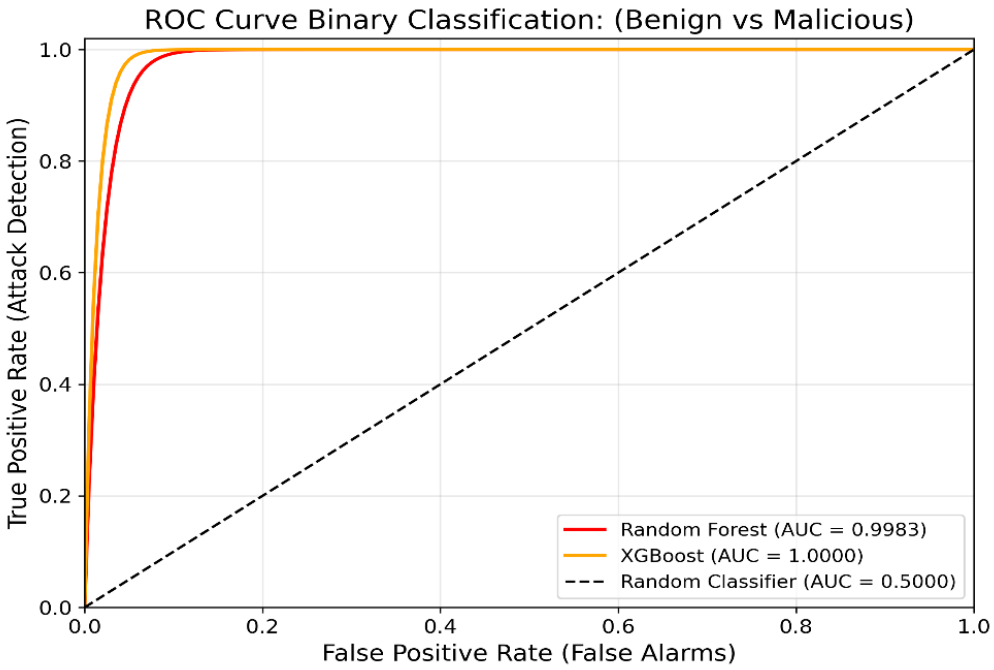

Fig. 5. ROC Curve for Binary Classification of Benign vs. Malicious Traffic

E. Security Discussion

From a cybersecurity perspective, these findings suggest that:

— Ensemble and boosting approaches strongly increase attack detection accuracy.

— Since XGBoost has the highest recall, it minimizes the chances of overlooking intrusions.

— Combining feature-driven learning with imbalance handling enables reliable detection of both common and rare attacks.

In general, experimental data indicate that tree-based ensemble learning is highly effective for modern intrusion detection on the CIC-IDS2017 dataset.

V. Conclusion & Future Work

This study proposes a machine-learning-based, data-driven intrusion detection system, evaluated on the CIC-IDS2017 dataset, to recognize malicious network traffic. The findings demonstrate that ensemble models perform much better than linear methods, and that XGBoost achieves the best accuracy, recall, F1-score, and ROC-AUC. The findings confirm the significance and role of using security metrics, such as recall and per-class analysis, to detect both common and rare attacks effectively. Future work could focus on using deep learning models to capture more complex traffic patterns, using real-time or streaming-based detection in operational environments, and deeper integration of interpretable AI solutions to improve analysts' decision-making. Further robustness and generalizability could be supported by testing the system on additional datasets or real network traffic.

References

[1] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in 2010 IEEE Symposium on Security and Privacy, 2010, pp. 305-316: IEEE.

[2] A. Pinto, L.-C. Herrera, Y. Donoso, and J. A. J. S. Gutierrez, "Survey on intrusion detection systems based on machine learning techniques for the protection of critical infrastructure," vol. 23, no. 5, p. 2415, 2023.

[3] I. Sharafaldin, A. H. Lashkari, and A. A. J. I. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," vol. 1, no. 2018, pp. 108-116, 2018.

[4] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, H. J. J. O. I. S. Janicke, and Applications, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," vol. 50, p. 102419, 2020.

[5] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, and S. J. I. Venkatraman, "Robust intelligent malware detection using deep learning," vol. 7, pp. 46717-46738, 2019.

[6] Y. Yang, K. Zheng, C. Wu, and Y. J. S. Yang, "Improving the classification effectiveness of intrusion detection by using an improved conditional variational autoencoder and a deep neural network," vol. 19, no. 11, p. 2528, 2019.

[7] S. M. Lundberg et al., "From local explanations to global understanding with explainable AI for trees," vol. 2, no. 1, pp. 56-67, 2020.

[8] A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and A. A. A. Ghorbani, "Characterization of tor traffic using time-based features," in International Conference on Information Systems Security and Privacy, 2017, vol. 2, pp. 253-262: SciTePress.

[9] C. J. I. A. Wong, "Designs for safer signal-controlled intersections by statistical analysis of accident data at accident blackspots," vol. 7, pp. 111302-111314, 2019.

[10] M. T. Abdelaziz et al., "Enhancing network threat detection with random forest-based NIDS and permutation feature importance," vol. 33, no. 1, p. 2, 2025.

[11] A. Al Farsi, A. Khan, M. M. Bait-Suwailam, and M. R. Mughal, "Comparative Performance Evaluation of Machine Learning Algorithms for Cyber Intrusion Detection," 2024.

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. J. J. o. a. i. r. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," vol. 16, pp. 321-357, 2002.

[13] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. A. J. P. C. S. Khan, "Performance analysis of machine learning algorithms in intrusion detection system: A review," vol. 171, pp. 1251-1260, 2020.

[14] Z. Saharuna, T. Ahmad, and R. M. J. V. C. Ijtihadie, "Shape-based feature selection and masv-weighted smote for enhanced attack detection in VANETs," p. 100970, 2025.

[15] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. J. A. c. s. Pedreschi, "A survey of methods for explaining black box models," vol. 51, no. 5, pp. 1-42, 2018.

[16] Q. A. Al-Haija and A. Droos, “A comprehensive survey on deep learning-based intrusion detection systems in Internet of Things (IoT),” Expert Systems, 2024/2025.

[17] S. Altamimi and Q. A. Al-Haija, “Maximizing intrusion detection efficiency for IoT networks using extreme learning machine,” Discover Internet of Things, 2024.

[18] M. Al-Omari and Q. A. Al-Haija, “Performance Analysis of ML-Based Intrusion Detection with Hybrid Feature Selection,” Computer Systems Science and Engineering, 2024.

[19] Q. A. Al-Haija and S. A. Tamimi, “A State-of-the-Art Survey of Adversarial Reinforcement Learning for IoT Intrusion Detection,” Computers, Materials & Continua, 2026.

[20] Q. A. Al-Haija, Z. Masoud, A. Yasin, K. Alesawi, and Y. Alkarnawi, “Revolutionizing Threat Hunting in Communication Networks: Introducing a Cutting-Edge Large-Scale Multiclass Dataset,” ICICS, IEEE, 2024.