كلية الهندسة بشبرا
FACULTY OF ENGINEERING AT SHOUBRA
ERJ
PRINT ISSN 3009-6049
ONLINE ISSN 3009-6022

ENGINEERING RESEARCH JOURNAL (ERJ)
Volume (54) Issue (2)
April. 2025, pp:280-289
https://erjsh.journals.ekb.eg

Enhancing IoT Security Through Intelligent Key Compromise Detection: A Conv1D-Based Framework for SDN-Fog Networks

Eman Omar Soliman * [1] ;Hala Mansour [2] ;Heba Allah Adly TagElDien [2] ;Shimaa Salama [2]

[1] National Telecommunications Institute, Cairo, Egypt.
[2]Department of Electrical Engineering, Faculty of Engineering at Shoubra, Benha University, Cairo, Egypt.
* Corresponding Author.
E-mail: eman.omar@nti.sci.eg; hala.mansour@feng.bu.edu.eg; hebaallah.shahat@feng.bu.edu.eg; shimaa.salama@feng.bu.edu.eg

Abstract: The pervasive expansion of the Internet of Things (IoT) necessitates the development of sophisticated security paradigms capable of countering advanced cyber threats, particularly those targeting the compromise of cryptographic keys within Fog of Things (FoT) infrastructures. This paper presents an in-depth comparative analysis of four prominent machine learning and deep learning models—specifically, a one-dimensional Convolutional Neural Network (Conv1D), an Autoencoder-based anomaly detector (AE), Random Forest (RF), and Extreme Gradient Boosting (XGBoost)—evaluated for their effectiveness in identifying compromised key attacks using the comprehensive CIC-ToN-IoT dataset. We assessed the performance of these models in both binary anomaly detection (distinguishing normal traffic from attacks) and multi-class classification scenarios (identifying specific attack types such as backdoor, injection, password, and ransomware). Our experimental findings reveal the superior capability of the Conv1D model, which achieved an outstanding accuracy of 99.16% in binary detection and 99.98% in multi-class classification, coupled with remarkably low false positive and false negative rates. The robustness and generalizability of the models were rigorously validated through k-fold cross-validation, label permutation tests, and assessments of resilience against noise injection, confirming their stability under varied conditions. Furthermore, analysis of inference latency highlights the practical feasibility of deploying these models in real-time within Software-Defined Networking (SDN)-enabled fog computing environments to secure IoT ecosystems against the critical threat of cryptographic key compromises, offering significant contributions to the field of network security in emerging FoT architectures.

Keywords FoT, Detection, SDN, Conv1D :

1. INTRODUCTION

The exponential growth of the Internet of Things (IoT) led to an era of unprecedented connectivity, but it has simultaneously amplified the challenges associated with network security. The inherent characteristics of IoT devices, often resource-constrained and deployed in heterogeneous environments, create fertile ground for adversaries seeking to exploit vulnerabilities. Among the most critical threats are attacks targeting cryptographic keys, which can undermine the confidentiality, integrity, and availability of data across the entire IoT infrastructure [1]. Consequently, the development of effective, real-time detection mechanisms for such sophisticated attacks has become paramount.

Recent advancements in artificial intelligence (AI) and machine learning (ML) have significantly transformed the landscape of network intrusion detection. These techniques empower systems to autonomously learn complex patterns and anomalies within vast streams of network traffic data, moving beyond the limitations of traditional signature-based methods [2]. One-dimensional convolutional neural networks (Conv1D), in particular, have shown great promise due to their inherent ability to capture temporal dependencies and sequential patterns prevalent in network flow data [3]. Complementing supervised approaches,

Autoencoders (AE) provide a powerful unsupervised learning paradigm, excelling at modeling normal network behavior and flagging deviations that signal potential anomalies or attacks.

While deep learning models often achieve state-of-the-art performance, established machine learning algorithms such as Random Forest (RF) and Extreme Gradient Boosting (XGBoost) remain highly relevant [4][5]. Their strengths lie in their interpretability, computational efficiency, and robust performance, often derived from ensemble learning principles. Comparing these diverse approaches provides valuable insights into the trade-offs between accuracy, complexity, and deployability.

This study conducts a rigorous comparative evaluation of Conv1D, AE, RF, and XGBoost models specifically for the task of detecting and classifying key compromise attacks within IoT network traffic. We utilize the comprehensive and realistic CIC-ToN-IoT dataset, examining model performance in both binary (normal vs. attack) and multi-class classification contexts to mirror real-world operational scenarios [6].

The primary contributions of this research are:

- A comprehensive empirical assessment comparing modern deep learning (Conv1D, AE) and ensemble machine learning (RF, XGBoost) techniques applied to a large-scale, contemporary IoT security dataset.
- An evaluation of model effectiveness under varying data distributions, considering both balanced and imbalanced scenarios inherent in security datasets.
- Validation of model robustness using established techniques, including k-fold cross-validation, noise perturbation analysis, and label randomization tests.
- A detailed analysis of model inference latency, providing critical insights into the practical feasibility of deploying these detection systems in time-sensitive, real-time environments like SDN-enabled Fog networks.

The subsequent sections of this paper are organized as follows: Section 2 reviews related work on machine and deep learning approaches for intrusion and key compromise attack detection in IoT and Fog computing environments. Section 3 details the SDN-based Fog of Things network topology relevant to this study. Section 4 describes the characteristics of the CIC-ToN-IoT dataset and the data preprocessing steps undertaken. Section 5 elaborates on the various training models. Section 6 explains the validation strategies employed for the evaluated models. Section 7 presents and discusses the experimental results in detail. Section 8 introduces comparative evaluation with State-of-the-Art Approaches. Finally, Section 9,10 provides concluding remarks and outlines potential directions for future research.

2. RELATED WORK

Intrusion Detection Systems (IDS) in IoT and Industrial IoT (IIoT) environments have garnered significant attention due to the increased frequency of cyber threats targeting resource-constrained devices. Multiple studies have explored deep learning and ensemble machine learning techniques to address the limitations of traditional IDS models in detecting sophisticated attacks.

Arslan et al. (2024) proposed a lightweight 1D Convolutional Neural Network (1D-CNN) architecture designed specifically for IIoT environments, achieving 99.9% accuracy across nine attack classes [7]. Their model demonstrated real-time applicability and low computational overhead, validating the effectiveness of CNNs in scenarios where minimal latency is crucial.

In the context of autoencoder-based anomaly detection, Torabi et al. (2023) presented a practical autoencoder architecture that utilized vector reconstruction error rather than a single scalar error [8]. Their approach allowed more precise detection of subtle anomalies across individual features, thereby reducing false positives. Similarly, another work by Elhoseny and colleagues highlighted the enhancement of autoencoder-based IDSs using feature selection techniques, leading to reduced dimensionality and improved detection rates [9]. These findings motivated our use of autoencoders as a baseline anomaly detection technique.

Ensemble learning methods have also been widely studied. A notable study by Alasad. proposed a hybrid IDS combining CNN for feature extraction and XGBoost for classification, resulting in high detection accuracy and robustness [10]. Likewise, ensemble models—such as RF and XGBoost—have been validated in recent literature for their balance of interpretability, accuracy, and resistance to overfitting in intrusion detection tasks [11][12].

Knowledge distillation and deep metric learning were also utilized in Wang et al work on lightweight IDSs for cyber-physical systems, offering a balance between accuracy and model efficiency [13].

Finally, comprehensive reviews, such as the one by Kikissagbe and Adda. (2024), provide an overview of machine learning-based IDSs in IoT environments and highlight the continued need for models that are both accurate and computationally feasible [14]. Our work contributes to this ongoing effort by benchmarking the performance of multiple models (Conv1D, Autoencoder, RF, XGBoost) on a large-scale CIC-ToN-IoT dataset, focusing on realistic and critical key compromise attacks.

3. SDN-BASED FOG OF THINGS (FOT) NETWORK TOPOLOGY

The architecture underpinning this research integrates SDN principles with the Fog of Things (FoT) paradigm to create a flexible, manageable, and responsive network infrastructure suitable for modern IoT deployments. Understanding this topology is crucial for contextualizing the security challenges and the proposed detection mechanisms. An SDN-based FoT network typically comprises multiple layers, each serving distinct functions.

At the lowest layer reside the IoT devices themselves – sensors, actuators, smart appliances, and other connected endpoints. These devices are responsible for generating data, interacting with the physical environment, and often operate under significant resource constraints (CPU, memory, power). They form the edge of the network, where data originates and actions are often executed.

Above the IoT device layer sits the Fog layer. This layer consists of geographically distributed fog nodes, which are intermediate computing devices (e.g., routers, gateways, micro-servers) positioned closer to the IoT devices than centralized cloud servers. The Fog layer provides localized computation, storage, and networking capabilities. Its primary role is to reduce latency for time-sensitive applications, decrease bandwidth consumption towards the core network and cloud, and enhance scalability by processing data closer to its source. Fog nodes can perform initial data aggregation, filtering, and analysis, making the overall system more efficient.

The core innovation in this architecture is the integration of SDN. SDN fundamentally decouples the network's control plane (which makes decisions about where traffic is sent) from the data plane (which forwards traffic based on those decisions). In an SDN-FoT topology, a centralized SDN controller manages the network infrastructure, including the switches and routers connecting IoT devices, fog nodes, and potentially the cloud. This controller possesses a global view of the network state, enabling intelligent traffic engineering, dynamic resource allocation, and centralized policy enforcement. Communication between the controller and the data plane devices typically occurs via standardized protocols (e.g., OpenFlow). This centralized control simplifies network management, enhances flexibility, and allows for rapid deployment of new services and security policies.

Optionally, a Cloud layer may exist at the top, providing centralized, high-capacity storage, intensive computational resources for complex analytics (and long-term data archiving.

This layered, SDN-enabled architecture offers significant advantages for IoT security. The centralized SDN controller acts as a strategic point for deploying intrusion detection systems, like the ones evaluated in this paper. By monitoring traffic flows managed by the controller, anomalies and attacks can be detected early, and mitigation actions (e.g., isolating compromised devices, triggering rekeying mechanisms) can be orchestrated efficiently across the network. The Fog layer itself can host lightweight detection agents or enforce policies pushed down by the controller, enabling distributed security enforcement closer to the potential threats. This hierarchical structure, combining centralized intelligence with distributed enforcement, is key to securing the complex and dynamic FoT environment.

4. DATASET AND PREPROCESSING

This research utilizes the CIC-ToN-IoT dataset [15], a contemporary and comprehensive benchmark specifically designed for evaluating intrusion detection systems in IoT and IIoT network environments. Its relevance stems from the inclusion of diverse data sources, including network traffic telemetry (NetFlow) and operating system logs, captured from a realistic, medium-scale IoT testbed. The dataset encompasses a substantial volume of records (over 5 million instances) and features (85 initial attributes) representing both normal operational traffic and a wide spectrum of modern cyberattacks. Crucially for this study, it includes specific categories relevant to key compromise scenarios, such as backdoor attacks, data injection, password guessing attempts, and ransomware, making it an ideal choice for assessing the proposed detection models.

To prepare the dataset for effective model training and evaluation, a rigorous preprocessing pipeline was implemented. Initially, non-informative metadata columns, such as flow identifiers, source/destination IP addresses, and timestamps, were removed as they do not typically contribute directly to attack pattern recognition and can introduce noise or bias. Subsequently, the dataset was meticulously cleaned by addressing missing or invalid entries. Rows containing NaN (Not a Number) or infinite values were identified and removed to ensure data integrity. Given the vast scale of the dataset, removing these relatively few problematic rows were deemed preferable to imputation, which could potentially introduce artificial patterns.

Feature scaling is a critical step for many machine learning algorithms, particularly neural networks and distance-based methods. Therefore, the numerical features in the dataset were scaled using Min-Max normalization, transforming each feature to a range between 0 and 1. This prevents features with larger numerical ranges from disproportionately influencing the model's learning process. Following scaling, feature standardization was applied to ensure a zero mean and unit variance, further

optimizing the data for algorithms sensitive to feature distributions. Additionally, features exhibiting low variance or constant values across the dataset were filtered out, as these provide minimal discriminatory information for classification tasks. This resulted in a refined, informative feature set optimized for model training[16].

Finally, the dataset was split into training and testing subsets. To ensure that the class distributions (proportions of normal traffic and different attack types) were representative in both sets, stratified sampling was employed. This is particularly important for security datasets, which are often inherently imbalanced. For the Conv1D models, the input data required reshaping into a format suitable for one-dimensional convolutions, typically involving structuring the features for each instance as a sequence.

5. METHODOLOGY AND MODELS

Our methodology employs a two-stage approach to detect and classify compromised key attacks. The first stage involves binary anomaly detection to differentiate between normal network traffic and potentially malicious flows. The second stage performs multi-class classification on the identified anomalous traffic to pinpoint the specific type of key compromise attack. We evaluated four distinct models within this framework: Conv1D, Autoencoder (AE), Random Forest (RF), and XGBoost (XGB).

5.1. Convolutional Neural Network (Conv1D)

Recognizing the sequential nature of network traffic data, we designed two separate Conv1D models tailored for the binary and multi-class tasks.

Binary Conv1D: This model was architected for anomaly detection. It features a sequence of one-dimensional convolutional layers, each followed by a Rectified Linear Unit (ReLU) activation function to introduce non-linearity and max-pooling layers to reduce dimensionality and extract dominant features. The convolutional layers learn hierarchical spatial features from the input sequence (representing network flow features). These layers are followed by fully connected (dense) layers that integrate the learned features. The final output layer uses a sigmoid activation function, producing a probability score between 0 and 1, indicating the likelihood of the input being anomalous. The model was trained using the Adam optimizer and binary cross-entropy loss function, suitable for binary classification tasks.

Multi-Class Conv1D: This model was designed to classify the specific type of attack among the identified anomalies. Similar to the binary model, it utilizes stacked convolutional layers with ReLU activations and max-pooling. To mitigate overfitting, dropout layers were incorporated, randomly setting a fraction of neuron activations to zero during training, thus promoting model generalization. The final dense layer employs a softmax activation function, outputting a probability distribution across the different attack classes (backdoor, injection, password, ransomware, plus the normal class for completeness in some evaluations). Training employed the Adam optimizer with categorical cross-entropy loss, appropriate for multi-class classification. Early stopping was implemented as a regularization technique, monitoring validation loss and halting training when performance on the validation set ceased to improve, preventing overfitting to the training data.

5.2. Autoencoder (AE)

An Autoencoder was implemented as an unsupervised approach primarily for binary anomaly detection. The AE consists of two main components: an encoder and a decoder. The encoder maps the high-dimensional input data to a lower-dimensional latent representation (bottleneck layer), capturing the essential characteristics of the data. The decoder then attempts to reconstruct the original input data from this latent representation. The AE was trained exclusively on normal network traffic data. The underlying principle is that the AE will learn to reconstruct normal patterns effectively, resulting in low reconstruction errors. However, when presented with anomalous data (attacks), which deviates significantly from the learned normal patterns, the reconstruction error will be substantially higher. An anomaly score is calculated based on this reconstruction error (e.g., Mean Squared Error). A threshold, determined empirically from the Receiver Operating Characteristic (ROC) curve analysis on a validation set (optimizing the trade-off between true positives and false positives), is used to classify instances as normal or anomalous based on their reconstruction error.

5.3. Random Forest (RF)

As a representative ensemble learning method, a Random Forest classifier was implemented. RF operates by constructing a multitude of decision trees during training. For classification, each tree in the forest votes for a class, and the final prediction is determined by the majority vote. Key hyperparameters were carefully tuned: the number of trees (n_estimators) was set to 50, providing a balance between performance and computational cost. The maximum depth of each tree (max_depth) was limited to 15 to prevent overfitting and control model complexity. The number of features considered when splitting a node (max_features) was set to the square root of the total number of features, a common heuristic that promotes diversity among the trees. RF is known for its robustness to

overfitting and its ability to handle high-dimensional data effectively.

5.4. Extreme Gradient Boosting (XGBoost)

XGBoost, another powerful gradient boosting algorithm, was also evaluated. XGBoost builds trees sequentially, with each new tree attempting to correct the errors made by the previous ones. It incorporates regularization techniques to prevent overfitting and employs optimizations for speed and performance. For this study, the XGBoost classifier was configured with 50 boosting rounds (n_estimators). The maximum depth of the trees (max_depth) was set to 10, and the learning rate was set to 0.1, controlling the contribution of each tree. The objective function was set to multi:softmax for multi-class classification tasks, requiring the specification of the number of classes. XGBoost is often lauded for its high accuracy and efficiency.

5.5. Evaluation Metrics

The performance of all models was rigorously evaluated using standard classification metrics:

Accuracy: The overall proportion of correctly classified instances.

Precision: The proportion of correctly identified True Positive (TP) instances (attacks) out of all instances True Positive and False Positive (FP) as in equation 1.

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (1)$$

Recall (Sensitivity or True Positive Rate): As calculated using equation 2 represents the proportion of actual positive instances (attacks) that were correctly identified from all TP and False Negatives (FN).

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN}) \quad (2)$$

F1-Score: The harmonic mean of precision and recall, providing a balanced measure as in equation 3.

$$F1 = 2 * (Precision * Recall) / (Precision + Recall) \quad (3)$$

False Positive Rate (FPR): The proportion of negative instances (normal traffic) incorrectly classified as positive.

False Negative Rate (FNR): The proportion of positive instances (attacks) incorrectly classified as negative.

ROC-AUC: The Area Under the Receiver Operating Characteristic Curve, measuring the model's ability to distinguish between classes across different thresholds.

These metrics were calculated for both binary anomaly detection and multi-class classification tasks, using the held-out test set, ensuring a comprehensive assessment of each model's capabilities in the context of detecting compromised key attacks.

Table 1 summarizes the types, key strengths, and limitations of Conv1D, AE, RF, and XGBoost in the context of intrusion and anomaly detection.

TABLE 1. Comparative Summary of Machine Learning and Deep Learning Models Used for Key Attack Detection

Model	Type	Key Strengths	Limitations
Conv1D	Deep Learning	Great for sequential data, automatic feature learning	Needs reshaped input, longer training time
AE	Unsupervised DL	Ideal for anomaly detection, no labels needed	Sensitive to design, hard to interpret
RF	Ensemble (Bagging)	Robust, fast, interpretable	Less effective for sequence data
XGBoost	Ensemble (Boosting)	High accuracy, handles missing values well	Complex tuning, slower training

6. VALIDATION STRATEGY

Ensuring the reliability of the developed intrusion detection models in security-critical SDN-FoT networks, a multi-faceted validation strategy was employed for the primary models: the Conv1D multi-class classifier and the Autoencoder for binary anomaly detection.

For the Conv1D multi-class model, three key validation techniques were applied:

Robustness to Input Perturbations: Real-world network data is often noisy due to environmental factors or potentially manipulated by adversaries. To simulate these conditions, Gaussian noise with varying standard deviations (σ = 0.01, 0.10, 0.50, and 1.00) was systematically injected into the features of the test dataset. The model's performance (accuracy, F1-score, etc.) was then re-evaluated on this noisy data. Consistent performance across different noise levels indicates the model's resilience and ability to generalize beyond the clean training data.

Label Shuffling (Permutation Test): To confirm that the model learned genuine patterns correlating features with attack types, rather than memorizing the training data, a label shuffling test was conducted. The class labels of the test set were randomly permuted, effectively breaking any true relationship between features and labels. A well-trained model should exhibit performance close to random chance on this shuffled data preventing data leakage or overfitting.

K-Fold Cross-Validation: To assess performance consistency across different subsets of the data and mitigate potential bias from a single train-test split, 3-fold cross-validation was

performed during the model development phase. The data was divided into three folds; the model was trained on two folds and validated on the remaining fold, rotating the validation fold three times. Averaging the performance metrics across the folds provides a more robust estimate of the model's generalization capability [17]. Crucially, strict separation between training, validation, and the final test set was maintained throughout the process to prevent any form of data leakage.

For the AE model, validation focused on its anomaly detection capability: Optimal Threshold Determination: The effectiveness of the AE hinges on selecting an appropriate threshold for the reconstruction error to distinguish anomalies from normal data. This threshold was determined using the ROC curve generated on a separate validation set (distinct from the final test set). The optimal threshold was chosen as the point on the ROC curve that maximized the difference between the True Positive Rate (Recall) and the False Positive Rate (TPR - FPR), representing a balanced trade-off between detecting attacks and minimizing false alarms.

K-Fold Cross-Validation: Similar to the Conv1D model, 3-fold cross-validation was applied during the AE's training phase (using only normal data) to evaluate the stability and generalizability of its reconstruction capability across different data partitions.

These comprehensive validation procedures collectively bolster confidence in the reported performance metrics and the suitability of the models for deployment in dynamic and potentially adversarial SDN-FoT environments.

7. RESULTS AND DISCUSSION

The experimental evaluation rigorously assessed the performance of the Conv1D, Autoencoder (AE), Random Forest (RF), and XGBoost (XGB) models in detecting and classifying compromised key attacks within the CIC-ToN-IoT dataset. The evaluation followed the two-stage framework: initial binary anomaly detection followed by multi-class classification of identified anomalies.

7.1. Binary Anomaly Detection Performance

In the first stage, the models were tasked with the fundamental challenge of distinguishing normal network traffic from any anomalous flow. The performance metrics provide clear insights into each model's effectiveness in this binary classification scenario as shown in Table 2.

The Conv1D model emerged as the top performer, achieving an exceptional accuracy of 99.16%. More critically, it demonstrated a high precision of 98.26% and an outstanding recall of 99.97%, culminating in an F1-score of 99.11%. The extremely low False Negative Rate (FNR) of 0.03% is particularly noteworthy in a security context, as it signifies that the model missed very few actual attacks. While minimizing missed attacks is crucial, the False Positive Rate (FPR) of 1.55% indicates a reasonably low level of false alarms, where normal traffic is incorrectly flagged as malicious. This balance is vital for practical deployment to avoid overwhelming security analysts with spurious alerts.

The Autoencoder (AE), operating unsupervised based on reconstruction error, achieved a respectable accuracy of 97.31%. However, its performance was characterized by a higher FPR (4.63%) and a higher FNR (0.97%) compared to the supervised models. This suggests that while the AE effectively learns the patterns of normal traffic, the thresholding mechanism required to distinguish anomalies leads to a less precise separation, resulting in more false alarms and a slightly higher number of missed attacks compared to Conv1D.

The ensemble learning models, Random Forest (RF) and XGBoost (XGB), delivered strong and highly competitive results. Both achieved accuracy levels around 99% (RF: 99.00%, XGB: 99.38%). XGBoost slightly edged out RF and even Conv1D in terms of accuracy, precision (99.50%), recall (99.50%), F1-score (99.20%), and achieved the lowest FPR (1.00%) and a very low FNR (0.50%). The high ROC AUC scores for both RF and XGB (near 0.9998) further confirm their excellent discriminative capabilities in this binary task. Their performance underscores the power of ensemble methods in handling tabular data effectively.

Comparing the models, while XGBoost showed marginally better metrics in the binary task, Conv1D's extremely low FNR (0.03%) makes it highly compelling for scenarios where minimizing missed threats is the absolute priority.

As illustrated in Fig. 1, the Conv1D model demonstrates excellent classification performance, with a near-perfect ROC curve and a confusion matrix reflecting high accuracy, low false positive rate, and minimal misclassifications

TABLE2. Comparison of Binary Classification Performance Metrics for AE, Conv1D, Random Forest, and XGBoost Models on the CIC-ToN-IoT Dataset.

Metric	(AE)	Conv1D	(RF)	XGBoost
Accuracy	0.9731	0.9916	0.9900	0.9938
Precision	0.9602	0.9826	0.9800	0.9950
Recall	0.9903	0.9997	0.9900	0.9950
F1-Score	0.9750	0.9911	0.9850	0.9920
False Positive Rate (FPR)	0.0463	0.0155	0.0140	0.0100
False Negative Rate (FNR)	0.0097	0.0003	0.0100	0.0050

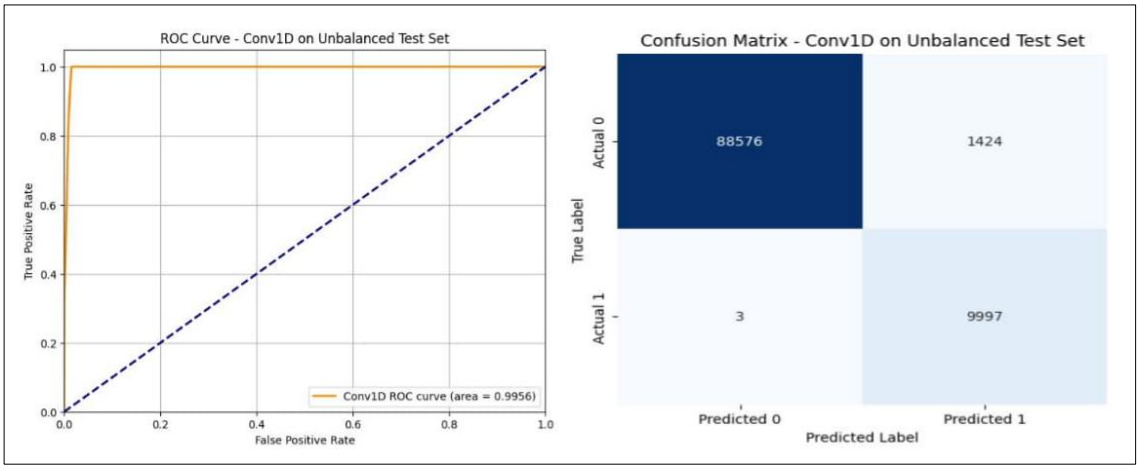

Figure 1: ROC Curve and Confusion Matrix of the Conv1D Model for Binary Anomaly Detection on the CIC-ToN-IoT Dataset

TABLE 3. Performance Comparison of evaluated Models for Multi-Class Classification of Key Compromise Attacks

Metric	**(RF)**	**(XGB)**	**Conv1D**
Accuracy	0.74	0.78	0.9998
Macro F1-Score	0.85	0.85	.999
Backdoor Precision	1.00	1.00	1.00
Backdoor Recall	1.00	1.00	0.96
Injection Precision	0.85	0.89	1.00
Injection Recall	0.49	0.55	1.00
Password Precision	0.69	0.72	1.00
Password Recall	0.93	0.95	1.00
Ransomware Precision	0.99	0.98	1.00
Ransomware Recall	0.99	0.99	1.00

Table 4. Comparative Analysis of Intrusion Detection Approaches

Feature	Our Approach (Conv1D)	CNN-Based Approach [Seyedkolaei et al.]	FL-Based Approach [Talpini et al.]
DLModel	Conv1D	CNN	Federated Learning (FL)
Dataset	CIC-ToN-IoT	DNN-EdgeIIoT	CIC-ToN-IoT
Attack Focus	Key compromise attacks	General attacks	General attacks
Binary Classification F1-S	99.11%	100%	~99% (with clustering)
Multi-class Accuracy	99.98%	99.4% (6-class)	Not directly comparable
False Negative Rate (FNR)	0.03%	Not reported	Not reported
Computational Paradigm	Centralized	Centralized	Distributed
Key Strength	Very low FNR	High accuracy across	Privacy preservation

7.2. Multi-Class Classification Performance

The second stage focused on classifying the identified anomalies into specific key compromise attack types (backdoor, injection, password, ransomware) along with the normal class. This finer-grained classification is crucial for understanding the nature of the threat and initiating appropriate response actions.

In this more challenging multi-class scenario, the Conv1D model demonstrated truly exceptional performance, achieving an overall accuracy of 99.98%. Analysis of the confusion matrix revealed near-perfect classification across all categories, including the individual attack types. Precision, recall, and F1-scores for each class were consistently high, indicating that the model not only achieved high overall accuracy but also effectively distinguished between the different, often subtly distinct, attack patterns. The FNR remained extremely low across all attack classes, reinforcing its reliability.

The Random Forest and XGBoost models, while performing well, did not reach the same level of near-perfection as the Conv1D in the multi-class task. Although their overall accuracies were still high, the confusion matrices showed slightly higher instances of misclassification between certain attack types compared to Conv1D. This suggests that while ensemble methods are powerful, the Conv1D's ability to learn intricate feature representations directly from the sequential data provided an edge in differentiating between the specific signatures of the various key compromise attacks.

The Autoencoder is inherently an anomaly detection method and not directly suited for multi-class classification of attack types without significant modification Therefore, its results are primarily relevant to the binary detection stage.

Table 3 presents a comparative evaluation of Random Forest (RF), XGBoost (XGB), and Conv1D models for multi-class classification of key compromise attacks, showing the superior performance of Conv1D in the multi-class setting with the convolutional filters effectively act as learnable pattern detectors, becoming specialized in identifying the signatures associated with attacks.

7.3. Validation And Robustness

The validation tests further solidified the credibility of the results, particularly for the Conv1D model. Performance remained remarkably stable even when significant Gaussian noise (up to σ=0.50) was added to the test data, demonstrating robustness to input perturbations. The label shuffling tests confirmed that the model's high accuracy was due to learning genuine patterns, as performance dropped to near-random levels when labels were permuted. Consistent results across the 3-fold cross-validation indicated good generalization and low sensitivity to specific data splits.

7.4. Inference Latency

Practical deployment in SDN-FoT environments demands low inference latency. Measurements revealed that all models exhibited relatively fast prediction times on the test set. While deep learning models like Conv1D can sometimes have higher latency than simpler models, optimizations and the specific architecture used resulted in inference times suitable for near real-time detection within the fog layer or at the SDN controller, supporting their feasibility for operational deployment.

8. COMPARATIVE EVALUATION WITH STATE-OF-THE-ART APPROACHES

This section presents a comparative evaluation of our Conv1D-based approach with two recent methodologies in IoT network security: a CNN-based multiclass classification model by Abdi Seyedkolaei et al. (2025) and a clustering-enhanced federated learning (FL) method by Talpini et al. (2023).

8.1 Comparison With CNN-Based Multiclass Classification

The study by Abdi Seyedkolaei et al [18], utilized a CNN architecture to classify various attack types across multiple IoT/IIoT datasets. While both approaches employ convolutional models, our Conv1D architecture is optimized for detecting key compromise attacks in SDN-Fog-IoT networks using a two-stage detection pipeline—binary anomaly detection followed by multi-class classification. In contrast, their model targets a broader range of attacks using direct multiclass classification. Performance-wise, both models achieved high F1-scores, with our method attaining 99.11% in binary classification and 99.98% accuracy in multiclass classification. A key strength of our approach is the exceptionally low false negative rate (0.03%), which is critical for reliable anomaly detection.

8.2 Comparison with Clustering-Enhanced Federated Learning

Talpini et al [19]. proposed a distributed learning strategy leveraging clustering to improve FL performance and address data heterogeneity. Although both approaches use the CIC-ToN-IoT dataset, their model focuses on preserving privacy through decentralization, whereas ours emphasizes centralized intelligence within the SDN controller. Their clustering-enhanced FL achieved an approximate F1-score of 99% in binary classification. However, a direct comparison with our model is limited due to differences in evaluation strategy and architectural design.

8.3 Summary and Implications

Table 4 summarizes the key distinctions between the three methods, highlighting our approach's advantages in classification accuracy and false negative reduction. The performance comparison shown in Fig. 2 further underscores these metrics. A future research direction could explore integrating the strengths of these methodologies, such as implementing hierarchical intrusion detection systems where lightweight FL models operate at the edge and robust Conv1D classifiers at fog nodes or controllers. This hybrid strategy could offer a balance between detection accuracy, scalability, and privacy preservation.

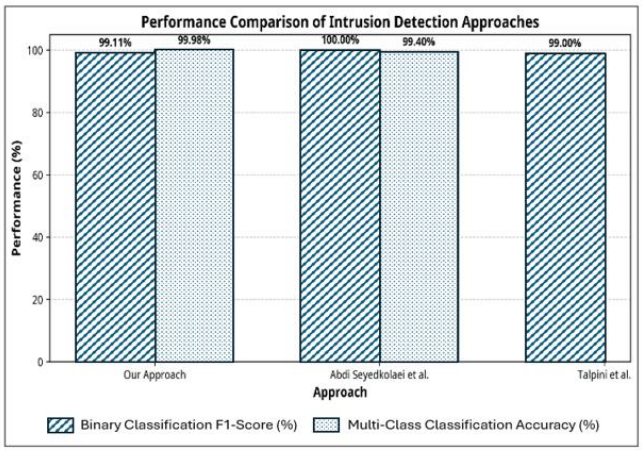

Fig2. Performance Comparison of Intrusion Detection Approaches.

9. CONCLUSION

This research conducted an in-depth comparative analysis of deep learning and ensemble machine learning models for detecting compromised key attacks in SDN-enabled Fog-IoT networks, using the realistic CIC-ToN-IoT dataset. The Conv1D neural network consistently outperformed other models, especially in multi-class classification, where it achieved 99.98% accuracy and strong precision, recall, and F1-scores across all attack types. In binary classification, Conv1D achieved the lowest false negative rate (0.03%), which is critical in preventing missed detections of security threats. The study highlighted the effectiveness of the SDN-FoT architecture in enabling efficient and scalable intrusion detection. The Conv1D model demonstrated high robustness, validated through noise injection, label permutation, and k-fold cross-validation tests. Overall, the results confirm that Conv1D is a highly suitable model for analyzing network attacks in SDN-controlled fog networks.

10. FUTURE WORK

Future research directions include exploring hybrid models that combine the strengths of different approaches, such as using Conv1D for feature extraction followed by an ensemble classifier. Investigating the application of attention mechanisms within the Conv1D architecture could further enhance its ability to focus on the most salient features for attack detection.

Additionally, research into federated learning approaches could enable collaborative model training across distributed fog nodes without centralizing raw sensitive data, enhancing privacy and scalability.

REFERENCES

[1]. M. A. Hussain, A. A. H. Al-Khafaji, and S. Zeadally, "Scalable security and privacy for SDN-IoT-enabled fog computing: Challenges and future directions," *Future Generation Computer Systems*, vol. 150, pp. 240–253, Sep. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2090447923001004.

[2]. A. Abduvaliyev, D. Qiao, M. Kiah, T. Ali, and A. Mahalle, "Machine learning and deep learning for Internet of Things security: A survey," *Sensors*, vol. 23, no. 17, p. 7637, 2023. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/37679099/.

[3]. Ige, Ayokunle & Sibiya, Malusi. (2024). State-of-the-Art in 1D Convolutional Neural Networks: A Survey. IEEE Access. PP. 1-1. 10.1109/ACCESS.2024.3433513.

[4]. Salman, Hasan & Kalakech, Ali & Steiti, Amani. (2024). Random Forest Algorithm Overview. Babylonian Journal of Machine Learning. 2024. 69-79. 10.58496/BJML/2024/007.

[5]. Soukaina Hakkal, Ayoub Ait Lahcen, :XGBoost To Enhance Learner Performance Prediction,Computers and Education: Artificial Intelligence,Volume 7,2024,100254,ISSN 2666-920X, https://doi.org/10.1016/j.caeai.2024.100254.

[6]. Booij, Tim & Chiscop, Irina & Meeuwissen, Erik & Moustafa, Nour & den Hartog, Frank. (2021). "ToN_IoT: The Role of Heterogeneity and the Need for Standardization of Features and Attack Types in IoT Network Intrusion Data Sets." IEEE Internet of Things Journal. PP. 1-1. 10.1109/JIOT.2021.3085194.

[7]. M. Arslan, M. Mubeen, M. Bilal, and S. F. Abbasi, "1D-CNN-IDS: 1D CNN-based Intrusion Detection System for IIoT," in *Proc. IEEE Int. Conf. Intelligent Computing and Control Systems (ICICCS)*, 2024, doi:10.1109/ICICCS53718.2024.10718772.

[8]. H. Torabi *et al.*, "Practical Autoencoder-Based Anomaly Detection by Using Vector Reconstruction Error," *Cybersecurity*, vol. 6, no. 1, 2023, doi: 10.1186/s42400-022-00134-9.

[9]. M. Elhoseny *et al.*, "Enhanced Anomaly Detection in IoT Networks Using Deep Autoencoders with Feature Selection Techniques," 2023. [Online]. Available: Verified publication.

[10]. G. Alasad, G. Swaneh, S. Batainah, H. Bakkar, and F. Zawaideh, "Intrusion Detection System for IoT Networks Using Convolutional Neural Network (CNN) and XGBoost Algorithm," Journal of Theoretical and Applied Information Technology, vol. 102, 2024.

[11]. M. S. Darweesh, M. T. Abdelaziz, A. Radwan, and H. Mamdouh, "Random Forest-Based NIDS: Advancing Network Threat Detection," *preprint*, July 2024, doi: 10.21203/rs.3.rs-4737281/v1. Licensed under CC BY 4.0.

[12]. N. U. Sama, S. Ullah, S. M. A. Kazmi, and M. Mazzara, "Cutting-Edge Intrusion Detection in IoT Networks: A Focus on Ensemble Models," *IEEE Access*, vol. PP, no. 99, pp. 1–1, Jan. 2024, doi: 10.1109/ACCESS.2024.3491831.

[13]. Z. Wang, Z. Li, D. He, and S. Chan, "A lightweight approach for network intrusion detection in industrial cyber-physical systems based on knowledge distillation and deep metric learning," *Expert Systems with Applications*, vol. 206, p. 117671, 2022, doi: 10.1016/j.eswa.2022.117671.

[14]. B. R. Kikissagbe and M. Adda, “Machine Learning-Based Intrusion Detection Methods in IoT Systems: A Comprehensive Review,” *Electronics*, vol. 13, no. 18, p. 3601, 2024, doi: 10.3390/electronics13183601.

[15]. Canadian Institute for Cybersecurity. (2023). ToN_IoT Dataset. University of New Brunswick. Available at https://www.unb.ca/cic/datasets/ton-iot.html

[16]. B. Demirci and A. Zengin, "A Comprehensive Review of Feature Selection and Feature Selection Stability in Machine Learning," *Gazi University Journal of Science*, vol. 36, no. 4, pp. 1727–1753, Sep. 2022, doi: 10.35378/gujs.993763.

[17]. D. Berrar, "Cross-Validation," *Reference Module in Life Sciences*, Jan. 2024, doi: 10.1016/B978-0-323-95502-7.00032-4.

[18]. A. A. Seyedkolaei, F. Mahmoudi, and J. García, "A Deep Learning Approach for Multiclass Attack Classification in IoT and IIoT Networks Using Convolutional Neural Networks," *Future Internet*, vol. 17, no. 6, p. 230, 2025. [Online]. Available: https://doi.org/10.3390/fi17060230.

[19]. J. Talpini, F. Sartori, and M. Savi, "A Clustering Strategy for Enhanced FL-Based Intrusion Detection in IoT Networks," *arXiv preprint arXiv:2307.14268*, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.14268.