Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.038276
Article

Multi-Attack Intrusion Detection System for Software-Defined Internet of Things Network

Tarcízio Ferrão[1,*], Franklin Manene[2] and Adeyemi Abel Ajibesin[3]

[1]Pan African University Institute for Basic Sciences, Technology and Innovation, Nairobi, 6200-00200, Kenya
[2]Department of Electrical and Control Engineering, Egerton University, Nakuru, 536-20115, Kenya
[3]School of Engineering, American University of Nigeria, PMB 2250 Yola, Nigeria

*Corresponding Author: Tarcízio Ferrão. Email: ferraotarcizio@gmail.com
Received: 06 December 2022; Accepted: 23 February 2023

Abstract: Currently, the Internet of Things (IoT) is revolutionizing communication technology by facilitating the sharing of information between different physical devices connected to a network. To improve control, customization, flexibility, and reduce network maintenance costs, a new Software-Defined Network (SDN) technology must be used in this infrastructure. Despite the various advantages of combining SDN and IoT, this environment is more vulnerable to various attacks due to the centralization of control. Most methods to ensure IoT security are designed to detect Distributed Denial-of-Service (DDoS) attacks, but they often lack mechanisms to mitigate their severity. This paper proposes a Multi-Attack Intrusion Detection System (MAIDS) for Software-Defined IoT Networks (SDN-IoT). The proposed scheme uses two machine-learning algorithms to improve detection efficiency and provide a mechanism to prevent false alarms. First, a comparative analysis of the most commonly used machine-learning algorithms to secure the SDN was performed on two datasets: the Network Security Laboratory Knowledge Discovery in Databases (NSL-KDD) and the Canadian Institute for Cybersecurity Intrusion Detection Systems (CICIDS2017), to select the most suitable algorithms for the proposed scheme and for securing SDN-IoT systems. The algorithms evaluated include Extreme Gradient Boosting (XGBoost), K-Nearest Neighbor (KNN), Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LR). Second, an algorithm for selecting the best dataset for machine learning in Intrusion Detection Systems (IDS) was developed to enable effective comparison between the datasets used in the development of the security scheme. The results showed that XGBoost and RF are the best algorithms to ensure the security of SDN-IoT and to be applied in the proposed security system, with average accuracies of 99.88% and 99.89%, respectively. Furthermore, the proposed security scheme reduced the false alarm rate by 33.23%, which is a significant improvement over prevalent schemes. Finally, tests of the algorithm for dataset selection showed that the rates of false positives and false negatives were reduced when the XGBoost

and RF algorithms were trained on the CICIDS2017 dataset, making it the best for IDS compared to the NSL-KDD dataset.

Keywords: Dataset selection; false alarm; intrusion detection systems; IoT security; machine learning; SDN-IoT security; software-defined networks

1 Introduction

The Software-Defined Network (SDN) is a recently developed architecture that makes network programming more flexible and practical configurable. It divides traditional networks into control and data planes [1]. This technology allows developers to centrally program, control, and manage network resources using an SDN controller [2]. The SDN introduces features that facilitate virtualization in implementing Internet of Things (IoT) networks. It has been developed with solid centralization and virtualization of control to allow devices in the physical network to be transformed into software because such limitations of traditional architectures as a lack of agility and speed in service provision have lately become prominent [3]. The implementation of the SDN promises to reduce the cost of complex network customization. As shown in Fig. 1, the SDN architecture comprises three layers [4]: application, control, and infrastructure. The application layer consists of programs that communicate with the controller regarding the necessary tools through an Application Programming Interface (API). The control layer is a logical object that responds to requests or actions from the application layer, and transmits them to the components of the network on a specific device according to the type of action required. The infrastructure or the data layer contains devices that control the capacities of the network for processing and forwarding the data.

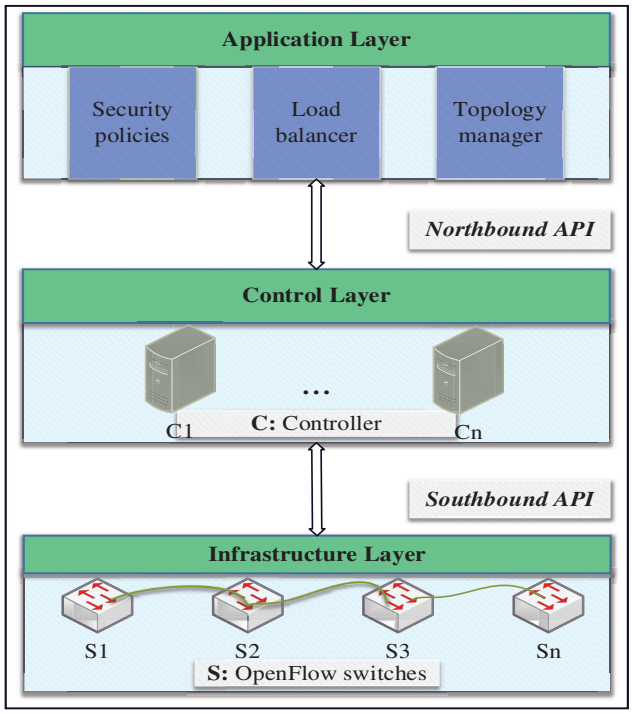


Figure 1: Architecture of the SDN

The Northbound API is a communication interface between the SDN and the application layer that offers an abstract perspective on the network and expresses how it behaves and what it needs. The Southbound API is an interface that connects data planes with network equipment, such as switches, to enable the direct expression of the behaviour and requirements of the network [5].

OpenFlow (OF) is a vendor-independent standard that allows heterogeneous devices to communicate [6,7]. Fig. 2 shows the architecture of a conventional network in which the applications must request access to resources of the terminal hardware because it uses a specific API, which makes it challenging to program the network [8]. Change management and network configuration according to the given organizational policies are complex tasks [9]. The SDN architecture solves this problem by separating control and the data into two planes, rather than exposing them to a higher application plane.

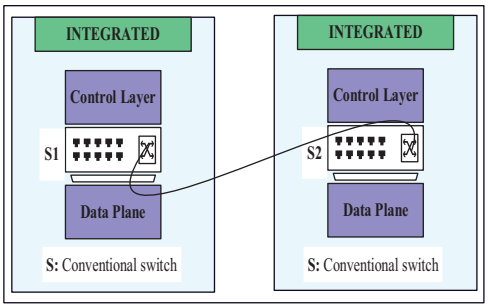


Figure 2: Architecture of a traditional network

The IoT technology creates data from which different forms of knowledge can be extracted to provide value-added services across various application areas. The SDN architecture can increase the bandwidth and flexibility of the IoT [8,10,11].

Due to the benefits of the SDN for the IoT, researchers have investigated methods to secure this connective environment from external attacks. The most common attack of the SDN architecture [4] is the Denial-of-Service/Distributed Denial-of-Service (DoS/DDoS), which overloads the processing resources of the victim until they are inaccessible to authorized users. Further attacks use an API to release unauthorized data and execute packet sniffing, which involves gathering data from the network where this is not authorized. Threats to the network layer include interference with traffic passing through OF switches to inhibit legitimate users from communicating with others. Attacks on the application layer involve unauthorized access to network programs. Many attackers prefer to harm the network controller in the control layer to cause errors in data transmission [12,13]. By contrast, threats to the computer network can be divided into active, passive, and physical attacks (that cause material damage) [14]. Cybernetic attacks include Man-in-the-Middle (MITM), spoofing, jamming, and malicious entry attacks.

Software-Defined IoT Networks (SDN-IoT) are subject to several threats. Many researchers have proposed methods of security against specific attacks, but have ignored a wider range of threats. Furthermore, the defensive schemes proposed for Intrusion Detection Systems (IDS) do not guarantee sufficient resources to control false alarms when categorizing the network flow in real-time, thus compromising the reliability of the system. As explained in [15,16], the IDS can identify numerous threats to the network. However, raising false alarms is a significant problem that reduces the effectiveness of the model. The IDS for SDN-IoT architectures can be improved through machine-learning algorithms applications. These algorithms can replace currently used techniques for image

reconstruction [17], distributed power control in tunnels [18] and can be used in intelligent vehicles as an efficient parking navigation system [19] to predict scenarios of transit and the availability of space. Due to the sheer variety of these algorithms, it is necessary to compare them to select the most suitable one for each problem. Furthermore, there are no effective methods for choosing an appropriate dataset in developing machine learning-based security mechanisms.

To solve the above problems, this paper is divided into three phases. In the first phase, five most commonly used machine-learning algorithms in IDS to secure SDN were analyzed on two datasets to identify the best one to use in security mechanisms for IoT environments. This analysis includes Extreme Gradient Boosting (XGBoost), K-Nearest Neighbor (KNN), Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression (LR) algorithms. As has been noted in [20,21], the Naive Bayes (NB), SVM, KNN, RF, and Decision Tree (DT) are popular algorithms for securing the SDN. In the second phase, the Multi-Attack Intrusion Detection System (MAIDS) based on dual machine learning was proposed to improve the security of SDN-IoT. The scheme includes a mechanism to control false alarms to prevent False Positives (FP) and False Negatives (FN) and to ensure that the alerts of the system are accurate. The last phase proposes a reliable method for selecting the appropriate dataset for training a machine-learning algorithm.

The remainder of this paper is structured as follows: Section 2 provides an overview of the methods used to secure SDN-based networks and Section 3 details the proposed model. Section 4 describes the simulations used to test the proposed method and an analysis of the results. Finally, the primary findings of this study are summarized in Section 5.

2 Related Work

Several researchers have studied security techniques for the SDN architecture. Their solutions have aimed to guarantee its operability, integrity, availability of information, and resistance to external threats. Many researchers consider the application of machine learning in IDS as an alternative that can improve SDN security. However, researchers differ in terms of the best algorithm that can be applied to design security systems for this new network infrastructure. Two detailed taxonomic studies [22] have been developed for the SDN security, comparing LR, RF and XGBoost algorithms to detect DDoS attacks. The authors concluded that XGBoost plays an important role in improving DDoS detection rate capability, including low training and testing time. This technique measures bandwidths to identify attack traffic. However, other researchers argue that SVM achieves greater accuracy in detecting DDoS attacks. Thus, an SVM-based IDS with a selective log for Internet Protocol (IP) tracking was proposed in [23–25]. During the "packet-in" event in the controller, this approach detects network intrusion and periodically obtains statistics on flow from OF switches. The authors considered attributes such as the number of packets and the time in seconds to classify attack traffic on the network.

As is known, SDN encompasses several advantages for IoT. One of the security solutions for this integrated technology has been the application of deep learning in IDS. Therefore, an SDN-enabled architecture was proposed in [26]. The system was designed for IoT devices with restricted resources. The authors used the deep neural network-based long short-term memory (CuDNNLSTM) network and the deep neural network-based gated recurrent unit (CuDNNGRU) algorithm to build the scheme. Their solution is scalable, efficient, and accurate, with a detection accuracy rate of 99.74%. In addition to a dual deep learning system, a time resource-based backpropagation neural network can also be used to detect DDoS attacks on SDN. This approach can lower system modelling costs while still achieving high accuracy. A scheme employing this technique was proposed in [27], which gathers statistics on

OF switches, such as the number and duration of inputs to each stream, and uses them to determine the hit rate. Additionally, it contains a method to dynamically retrieve the port of the victim device. While some studies have proposed applying the SDN architecture to protect the IoT environment, it may not be as effective as proposals that incorporate machine learning techniques. The authors of [11] examined the advantages of SDN technology in securing IoT networks. They proposed a security controller with a role-based architecture, called Rol-Sec, and SDN-based solutions to improve security. They also provided a comprehensive review of recently proposed SDN-based solutions to enhance the security of the IoT environment. They concluded that few studies in the literature had dealt with integrated SDN-IoT systems.

Deep learning methods have gained greater relevance in IDS applications as they present better results in attack detection. To model these algorithms, it is necessary to select a suitable dataset. As seen in the empirical study on the effectiveness of deep learning and ensemble methods for IDS [28], the authors considered the UNSW-NB15 [29] dataset as recent, containing empirical data on attacks. However, the deep learning models fit balanced datasets better than unbalanced datasets. On the other hand, the authors of [30] also investigated the Knowledge Discovery in Databases (KDD) dataset to highlight design challenges in wireless intrusion detection. They identified the challenges of methods used to track network traffic to train machine-learning algorithms for IDS.

Other IDS methods for SDN include the application of supervised and unsupervised learning. The application of pre-trained models for different classifications of traffic and flow was described in [31]. A flow-clustering approach was used to identify flows frequently observed in concurrence. The method collects information on traffic flow based on machine learning and integrates it into an SDN controller to predict anomalies in the network.

The applications of machine learning can be extended to networks with multiple controllers. Although this structure is very complex to implement, a scheme for detecting and preventing intrusion using machine learning was proposed in [32]. This model accepts nodes using a random forest model (CBNA-RF) that configures suitable security rules and automates defensive actions in a large-scale SDN framework. A machine learning-based detection method with the same objective was proposed in [33]. It implements learning algorithms using features with a few OF packets to detect abnormal traffic in SDN-related data and the control layers. This scheme can accurately identify low-rate DDoS attack traffic without considerably affecting system performance.

As can be seen, many researchers have proposed machine learning-based methods to detect DDoS attacks on SDN and have also applied deep learning methods to model IDS for SDN-IoT. However, gaps in robust and effective models for detecting multiple attacks remain. The authors of [34] described the difficulties of applying machine learning and SDN to handle the latest security risks in the IoT. They claimed that challenges to the next-generation IoT require a new vision of a secure design in which threats are proactively addressed and that machine learning and SDN are essential for providing reconfigurability and intelligence for IoT devices. Their research offered a new perspective on IoT network security based on design, polymorphism, and SDN. Other research suggests combining machine learning and deep learning to ensure SDN security. Although RF exhibits greater accuracy in detecting DDoS attacks targeting an SDN layer [35], XGBoost can be applied as a cloud-based method to detect attacks on SDN because it has higher accuracy, lower False Positive Rate (FPR), higher processing speed, and higher adaptability than RF, SVM, multilayer perceptron (MLP) and DT [36]. In addition, advanced features obtained by extracting data from network stream headers can be applied to detect SDN attacks [20]. Although it does not consider the False Negative Rate (FNR), this technique applies machine learning to increase attack detection accuracy and decrease FPR.

3 Proposed Intrusion Detection System and Dataset Selection Method

This section describes the proposed MAIDS scheme and the method to select an appropriate dataset to train a machine-learning algorithm for IDS.

In the first phase, were analyzed five machine-learning algorithms, XGBoost, KNN, RF, SVM, and LR, on the Network Security Laboratory Knowledge Discovery in Databases (NSL-KDD) dataset [37] and the Canadian Institute for Cybersecurity Intrusion Detection Systems (CICIDS2017) dataset [38]. A comparative analysis of these algorithms aims to select the best two algorithms that can be applied to the MAIDS scheme, which contain mechanisms to control FP and FN alarms. Fig. 3 illustrates the modelling and analysis of the proposed machine learning-based algorithms. The datasets were pre-processed to eliminate null values and choose columns containing the necessary features. The results of training on each dataset were visualized and compared by using the following parameters:

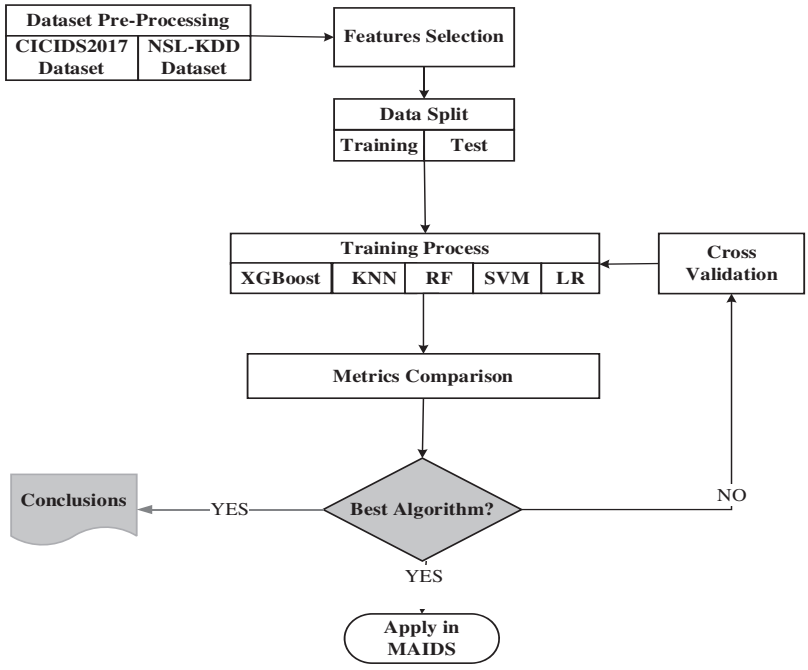


Figure 3: Flowchart of modelling the algorithms

Accuracy (A): is the percentage of the given set that is correctly classified. It includes the True Positives (TP), True Negatives (TN), FP and FN.

$$A = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

Precision (P): corresponds to the ratio of the dataset classified as positive concerning the complete set classified as TP (including false positives).

$$P = \frac{TP}{TP + FP} \tag{2}$$

Recall (R): corresponds to the ratio of TP in comparison with the overall number of positive features in the dataset.

$$R = \frac{TP}{TP + FN} \tag{3}$$

F1-score (F1): corresponds to the harmonic mean of precision P and recall R.

$$F1 = \frac{2PR}{P + R} \tag{4}$$

3.1 Datasets Description and Feature Selection

The NSL-KDD dataset was modified from the KDD dataset to train and evaluate machine learning-based classifiers to identify DDoS attacks [39]. It contains five classes, subdivided into data containing normal traffic and attack-related traffic. It has 41 features, including numeric, nominal, and binary resource values. Randomly 108,400 logs were used in the simulations [40]. The dataset consists of four subcategories, KDDTest+, KDDTest-21, KDDTrain+, and KDDTrain-21, encompassing the DoS, Remote-to-Local (R2L), User-to-Root (U2R), and probe attacks. KDDTest+ and KDDTrain+ include complete components, while the remainder comprises 20% of the entire set [41].

The CICIDS2017 dataset is recent, and has been used for updated attacks on the IDS that reflect the realistic application scenarios [42]. This dataset contains information on normal traffic, categorized as "benign", and different attack categories captured in five days, as shown in Table 1. It contains 83 features, built from 25 users of networks based on the Hypertext Transfer Protocol (HTTP), Hypertext Transfer Protocol Secure (HTTPS), File Transfer Protocol (FTP), and Secure Shell (SSH) [38]. The CICIDS2017 is suitable for binary and multi-class classification.

Table 1: Description of the NSL-KDD and CICIDS2017 datasets

Dataset	Types of attacks	Description
NSL-KDD	DoS	Interruption of services for legitimate users.
	U2R	The attacker acquires administrator permissions on the network.
	R2L	The attacker gains access to a device on the local network by sending packets without permission.
	Probe	It involves collecting information to exploit local network vulnerabilities.
CICIDS2017	Brute force (FTP-Patator and SSH-Patator)	It involves gaining unauthorized access to personal accounts.
	DoS/DDoS	It includes DoS Slowloris and Hulk.
	Web attack	It includes injection attacks, such as cross-site scripting (XSS) and structured query language (SQL) injection.
	BotNet ARES, PortScans, and DDoS LOIT	It is a set of devices on the same network for attacks.

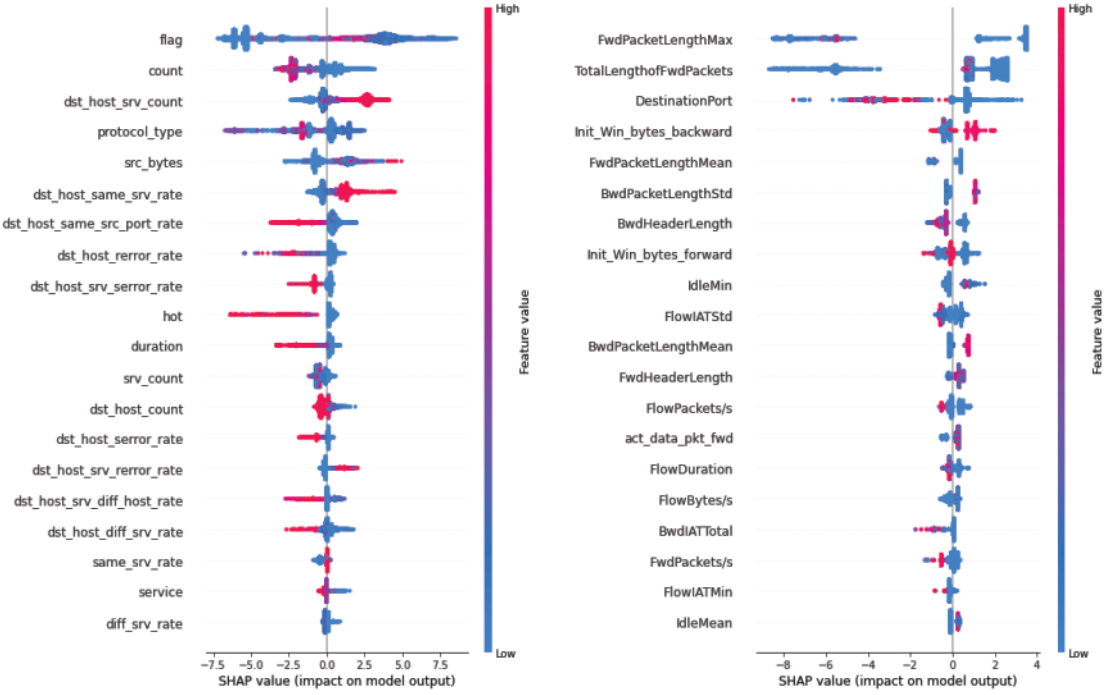
For both datasets, were used the SHapley Additive exPlanations (SHAP) method [43] to select essential features for the predictive output of the model. The classic SHAP method is based on cooperative game theory. For every set of training data $S(H_i)$, it is possible to explain the importance of each feature H_i by calculating its contribution to the output of the model through Shapley values Φ_i that are given by:

$$\Phi_i = \sum_{S \subseteq H \setminus \{i\}} \left\{ \frac{|S|!\,(|H| - |S|)!}{|H|!} \left[f_{SU\{i\}}\left(x_{SU\{i\}}\right) - f_S\left(x_s\right) \right] \right\} \tag{5}$$

where $f_{SU\{i\}}$ is the first model trained by feature $x_s \in S$ and f_S is the second model trained by selected feature. Fig. 4 shows the features selected from the NSL-KDD and CICIDS2017 datasets.

3.2 *Proposed Scheme*

The proposed MAIDS scheme is illustrated in Fig. 5. It comprises two machine-learning modules implemented for classifying data traffic and an SDN controller that installs the security-related rules in the data layer in case of an alert. In the first phase, the SDN controller requests flow-related statistics from the switches as input data for analysis by machine learning. The machine-learning algorithms 1 and 2 compare their results of prediction before sending the final result to the SDN controller. These algorithms are trained on different datasets to allow continuous and real-time comparison between their results. During regular operation, both algorithms predict the same results. If the results are different, the errors of classifications are compared to determine the final classification. This scheme uses the errors in threat detection in the data stream of the network to reduce the number of false alarms.



(a) Feature importance for NSL-KDD (b) Feature importance for CICIDS2017

Figure 4: Impact of features on the output of the model

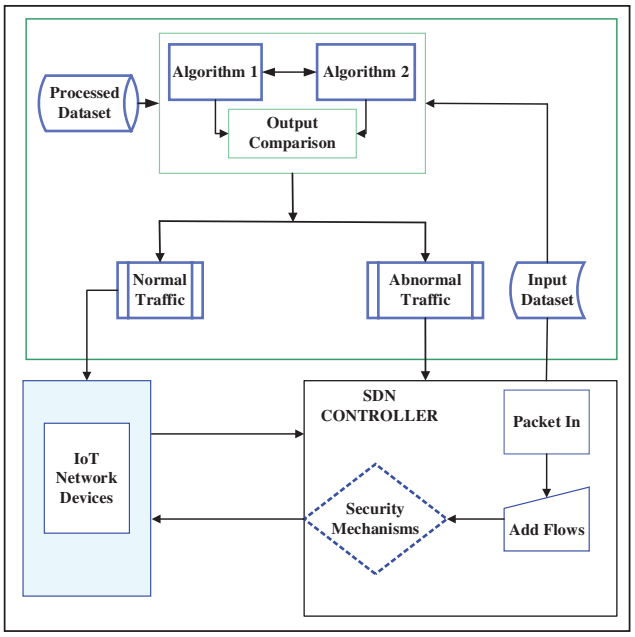


Figure 5: Multi-attack IDS scheme

As the IoT environment contains many connected devices, a potential attack on it can lead to various losses, such as the loss of information and material damage. A solution to this problem is provided in the form of various security-related proposals in the literature. However, many systems do not have a specific mechanism that guarantees reliable control over false alarms. An IDS should usually not penalize legitimate users on the network as a consequence of an FP because this compromises industrial production in the IoT. Likewise, the IDS must not allow attack-related traffic to flow into the network due to an FN. Thus, all security systems should include a mechanism to control these false alarms to prevent damage, and to ensure the safety and efficiency of connectivity of the IoT.

Many security-related methods have been designed to detect specific attacks. Some do not offer techniques to prevent the detected attacks, and depend on other resources of the network. The proposed scheme can detect and prevent multiple attacks with high precision due to a combination of the two machine-learning algorithms. In addition, it is simple and inexpensive because it does not require the use of more than two algorithms. A general overview of the SDN-IoT network is shown in Fig. 6. It consists of a controller that monitors the network and several connected devices that require security.

Let D_k be the real-time input dataset to the IDS scheme that is composed of k instances of data streams of the network. Algorithms A_1 and A_2 classify each k in D_k and form vectors of length y_{ij} for each group with the same results of prediction. If the result of prediction P_{ij} is the same for all k instances, where i corresponds to the algorithm number and j is the position of the given class, this means that y_{ij} is equal to the length of the input dataset D_k and that the algorithms have correctly classified the entire input dataset. Therefore, the model output is considered to be reliable. On the contrary, if the results of prediction P_{ij} for D_k contain some errors that can generate FP and FN, this means that two different results form the final output of the models. For a binary classifier, all

instances k in D_k classified as 0 (normal) or 1 (abnormal) are then grouped according to their length y_{ij}, as shown in the matrix of Eq. (6):

$$P_{ij} = \begin{matrix} \\ A_1 \\ A_2 \end{matrix} \overset{\begin{matrix} 0 & \quad 1 \end{matrix}}{\begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix}} \tag{6}$$

where the first row of the matrix contains y_{11} and y_{12}, which are the lengths of D_k with k instances classified as categories containing the values 0 and 1, respectively, by algorithm A_1. The second row of the matrix contains y_{21} and y_{22}, which are the lengths of D_k with k instances classified as categories containing the values 0 and 1, respectively, by algorithm A_2. The operation of this scheme for controlling and preventing FP and FN is detailed in Algorithm 1.

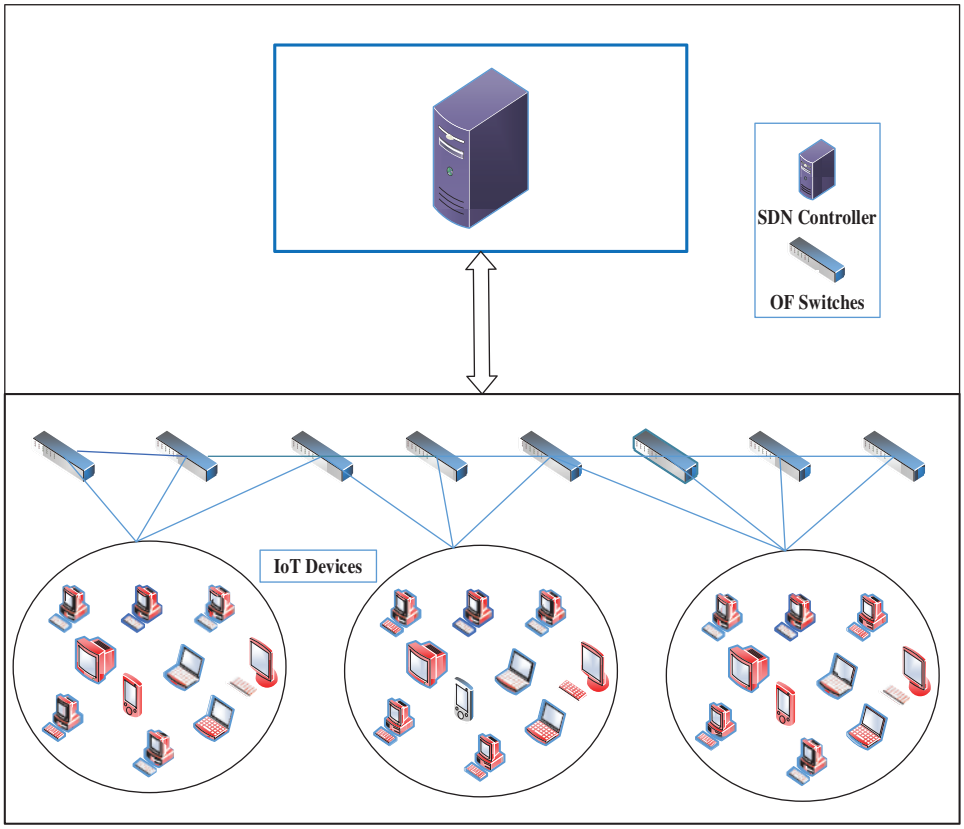
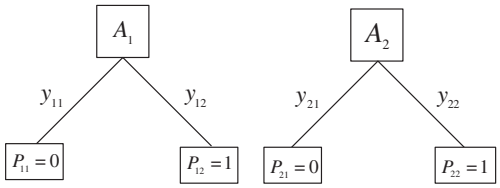


Figure 6: Overview of the network topology of SDN-IoT

Algorithm 1: Algorithm to control false alarms

0: **Start MAIDS**:
1: *Network Status* = [*Normal*, *Abnormal*]
2: *Predicted* = [P_1, P_2]
3: $A_i = [A_1, A_2]$
4: **for** P_{ij} *in* A_i:
5: **if** $P_{11} > P_{12}$ **and** $P_{21} > P_{22}$:
6: $P_1 = P_2$
7: *Network Status* = *Normal*
8: **end if**
9: **if** $P_{11} < P_{12}$ **and** $P_{21} < P_{22}$:
10: $P_1 = P_2$
11: *Network Status* = *Abnormal*
12: **end if**
13: **end for**
14: **while** $P_1 = P_2$:
15: *Network Status* = *Predicted*
16: **end while**
17: **while** $P_1 \neq P_2$:
18: **if** $y_{11} + y_{21} > y_{12} + y_{21}$:
19: *Network Status* = *Normal*
20: **end if**
21: **if** $y_{11} + y_{21} < y_{12} + y_{21}$:
22: *Network Status* = *Abnormal*
23: **end if**
24: **end while**
25: **end MAIDS**



3.3 Dataset Selection Method

The choice of a dataset is a fundamental step in modelling the IDS based on machine learning. Many researchers have evaluated such aspects as the size of the dataset, the number of attacks included, and when it was generated for selection. However, few rules have been established for this stage. For example, the authors in [44] justified the choice of the CICIDS2017 dataset because it had been created in an environment similar to real structures. On the contrary, others have preferred the NSL-KDD dataset because it contains data on different types of attacks, and its results are not biased [23,39,45]. The authors of [26,36,46] have chosen different datasets for their models. Thus, a suitable method is proposed to select the dataset for a given machine-learning algorithm in a universe containing several pre-selected datasets.

Consider a universe of datasets $D_i = \{D_1, D_2, D_3, \ldots, D_n\}$, where n is the number of datasets. It is assumed that these datasets are similar as they all are designed to be applied to network security. Their

features f_i may differ, but the types of attacks and the architecture or destination network are similar. The size of the datasets is reasonable. All datasets in the universe D_i have been recently developed or are adequate for forming an IDS. Thus, one can select the best dataset for a machine-learning algorithm A_i using the Algorithm 2, with m_i metrics selected to evaluate A_i. The set of important features for the output of the model on a dataset, can be denoted by $D_n[f_{in}]$. The dataset selection method offers the results of the evaluation metrics $D_n[m_i]$ obtained for each dataset.

Algorithm 2: Dataset selection algorithm

Step 1: **Definition of variables**
0: $A_i = \{A_1, A_2, A_3, \ldots, A_n\}$
1: $D_i = \{D_1, D_2, D_3, \ldots, D_n\}$
2: $m_i = \{m_1, m_2, m_3, \ldots, m_n\}$
3: $f_i = \{f_1, f_2, f_3, \ldots, f_n\}$

Step 2: **Start evaluation**
4: **for** f_i *in* D_i:
5: *Select best features*
6: *return*
7: $D_1[f_{1n}] = \{f_{11}, f_{12}, f_{13}, \ldots, f_{1n}\}$
8: $D_2[f_{2n}] = \{f_{21}, f_{22}, f_{23}, \ldots, f_{2n}\}$
9: $D_3[f_{3n}] = \{f_{31}, f_{32}, f_{33}, \ldots, f_{3n}\}$
10: ...
11: $D_n[f_{in}] = \{f_{11}, f_{12}, f_{13}, \ldots, f_{1n}\}$
12: **end for**

Step 3: **while** $A_1 = A_1$:
Step 4: **Select** $D_i = D_1$:
13: **Select** $m_i = \{m_1, m_2, m_3, \ldots, m_n\}$
14: **Train** A_1 **with** D_1
15: *return*
16: $D_1[m_{1n}] = \{m_{11}, m_{12}, m_{13}, \ldots, m_{1n}\}$
17: **end process**
18: *Repeat Step* 4: *for* $D_i = D_2, D_3, \ldots, D_n$
19: **end process**
20: **end while**

Step 5: **Dataset selection**
21: **with** $m_i = \{D_1[m_i], D_2[m_i], D_3[m_i], ..., D_n[m_i]\}$
22: **if** $D_1[m_i] > \{D_2[m_i], D_3[m_i], ..., D_n[m_i]\}$
23: *Select the best dataset* D_1
24: *else*:
25: *Select the best dataset*
D_n: $D_n[m_i] > \{D_n[m_i]\}$, $n = 1, 2, 3, \ldots, n-1$
26: **end with**

4 Results and Discussion

This section presents the results of each model as part of the first phase of the proposed scheme. In the second phase, were implemented the best models in the MAIDS scheme proposed in Fig. 5 to detect and mitigate attacks, obeying the mechanism to control false alarms developed in Algorithm 1. The FPR and the FNR were selected as evaluation metrics for dataset selection because the proposed security scheme needs to avoid false alarms. The dataset was selected based on the two best machine-learning algorithms using the proposed selection method.

4.1 Evaluation of Machine-Learning Algorithms

This subsection compared the accuracies of XGBoost, KNN, RF, SVM, and LR. The results on the NSL-KDD dataset are shown in Fig. 7 and on the CICIDS2017 dataset in Fig. 8. In both cases, XGBoost and RF reached a better accuracy. The RF had the highest accuracy of 99.81% on the NSL-KDD dataset, followed by XGBoost with 99.79% accuracy. However, XGBoost and RF had an

accuracy of 99.96% on the CICIDS2017 dataset. The KNN algorithm also showed promising results on the NSL-KDD and CICIDS2017 datasets, with accuracies of 99.40% and 98.58%, respectively.

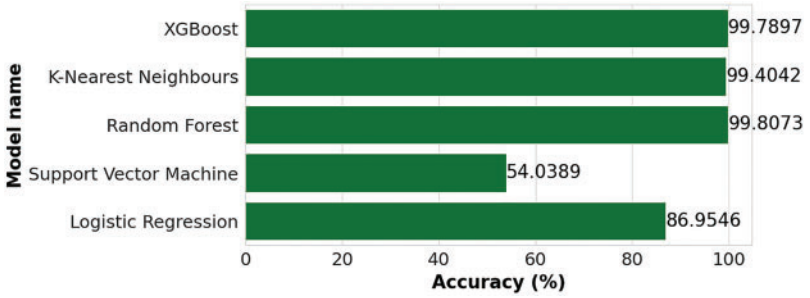


Figure 7: Comparison of accuracies on the NSL-KDD dataset

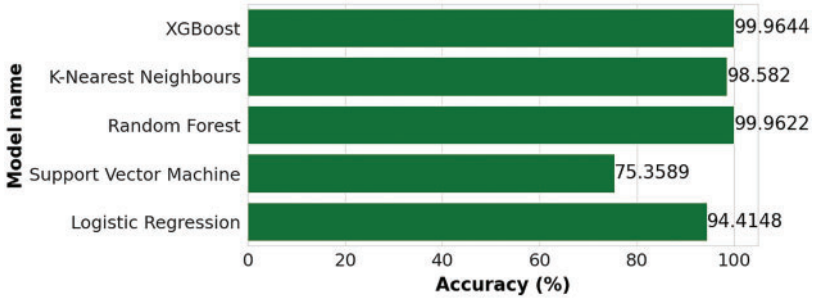


Figure 8: Comparison of accuracies on the CICIDS2017 dataset

The algorithms were also evaluated on other parameters. As shown in Fig. 9, XGBoost achieved the highest precision of 99.98%, followed by RF with 99.96% precision, on the NSL-KDD dataset. The Receiver Operating Characteristic (ROC) in Fig. 10 shows the performance of each model, relating the True Positive Rate (TPR) and the FPR. The RF and XGBoost algorithms had an Area Under the ROC Curve (AUC) of 99.99% on the NSL-KDD dataset. Furthermore, Fig. 11 shows that XGBoost achieved the highest precision of 99.99%, followed by RF with 99.98% on the CICIDS2017 dataset. The KNN had a precision of 98.67% on this dataset. Fig. 12 shows that XGBoost and RF reached an AUC of 100%, followed by the KNN with an AUC of 99.53% on the CICIDS2017 dataset. These results show that the RF and XGBoost algorithms are the best options for designing an IDS for SDN-IoT.

Another important parameter to consider when evaluating these algorithms is the processing time, which includes training and testing. Typically, the best machine-learning algorithm should take as little time as possible to detect a given network attack. Observing Table 2, it can be noted that the XGBoost model managed to obtain an ideal test time for the classification of attack traffic on both NSL-KDD and CICIDS2017 datasets. Training time is acceptable concerning the other models, as it outperforms them in terms of the other metrics discussed previously. LR presents acceptable test time for both datasets but failed to obtain better results in other metrics. Likewise, KNN managed to obtain the shortest training time but failed to provide a long test time compared to other algorithms. The RF is up to the same standard as XGBoost, although testing and training times are relatively long. On the other hand, SVM proved to be slow to be applied in IDS schemes, as its training and testing times are longer than other algorithms. Therefore, the model that can be applied in the proposed security

scheme is XGBoost with a training time of 4.7 s and a test time of 0.03 s on the NSL-KDD dataset. Additionally, RF can also be applied with a training time of 3.5 s and a test time of 0.09 s on the same dataset. While XGBoost achieved a training time of 1.54 s and a test time of 0.04 s on the CICIDS2017 dataset, RF achieved a training time of 1.09 s and a test time of 0.32 s on the same dataset.

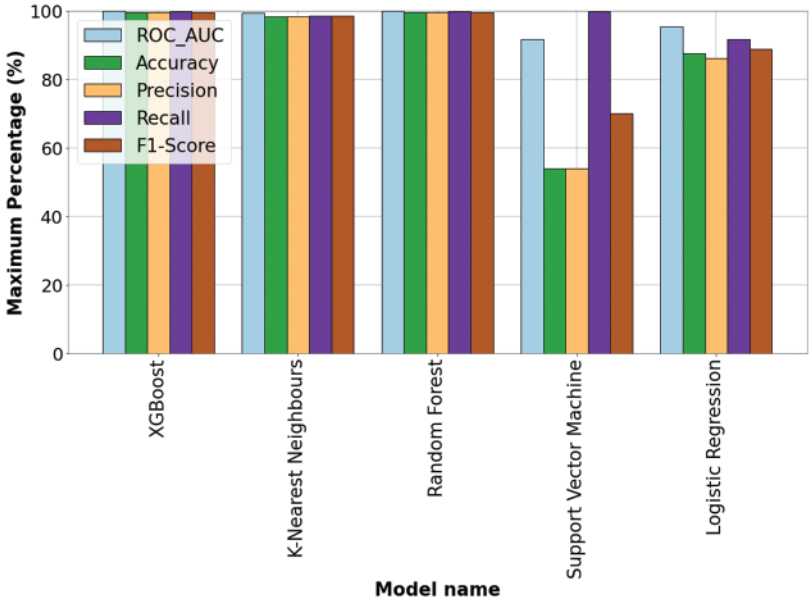


Figure 9: Comparison of performance on the NSL-KDD dataset

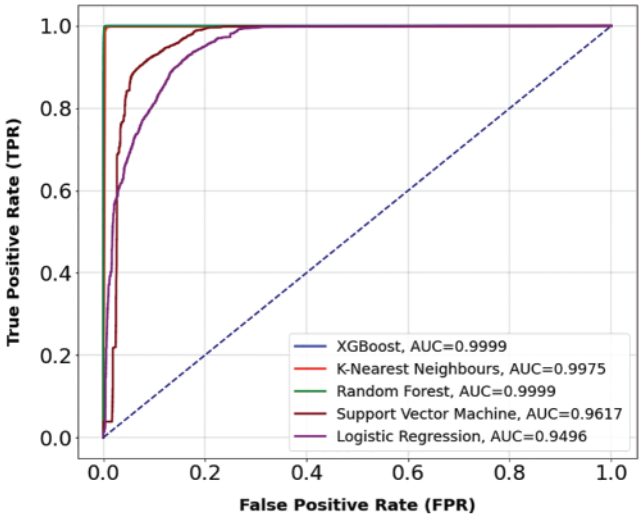


Figure 10: ROC curves on the NSL-KDD dataset

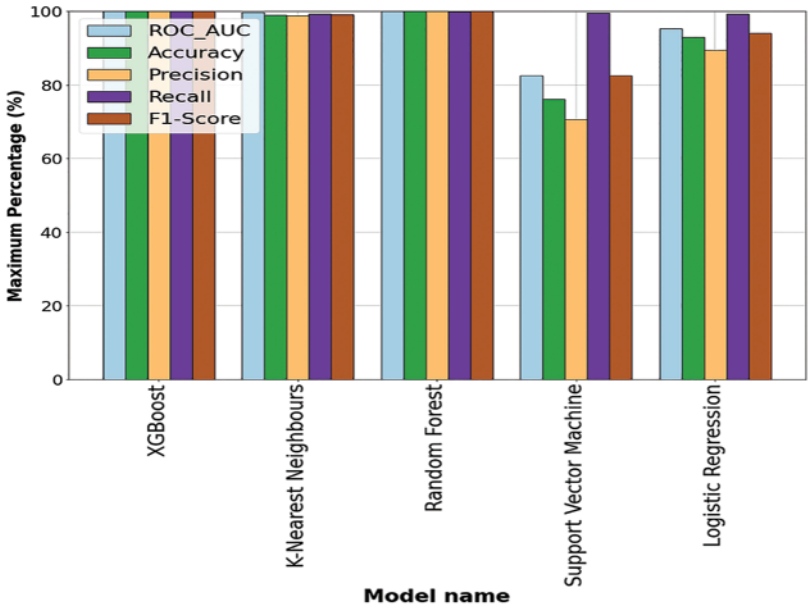


Figure 11: Comparison of performance on the CICIDS2017 dataset

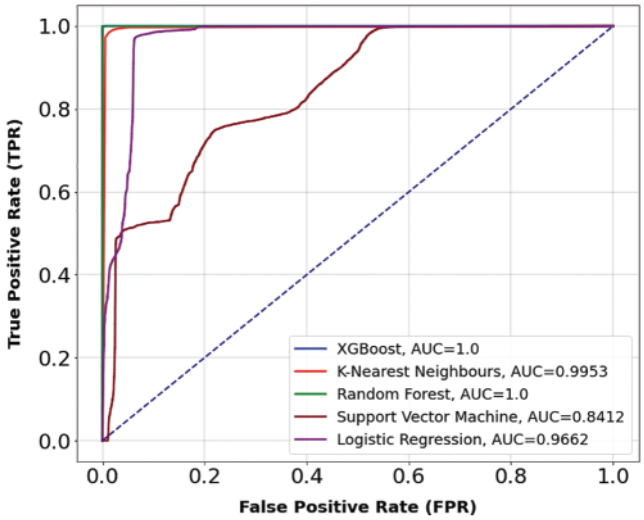


Figure 12: ROC curves on the CICIDS2017 dataset

Table 2: Processing time analysis

Results	NSL-KDD dataset					CICIDS2017 dataset				
	XGBoost	KNN	RF	SVM	LR	XGBoost	KNN	RF	SVM	LR
Training time (s)	4.70	0.01	3.50	1 342.94	0.74	1.54	0.005	1.09	55.85	0.20
Testing time (s)	0.03	2.69	0.09	26.41	0.003	0.04	3.10	0.32	22.63	0.21

The above results show that the most common algorithms used in the IDS, XGBoost, RF, and KNN, are the most suitable for ensuring the security of the IoT network. XGBoost and RF were selected to implement the MAIDS scheme, which means that all the following results were based on their application to the proposed IDS. The KNN can also be implemented in this scheme because it showed superior results than the SVM and LR.

Table 3 summarizes the results obtained for the XGBoost, KNN, RF, SVM, and LR algorithms. Another comparison of several results obtained in recent research is shown in Table 4. It can be noted that XGBoost and RF obtained higher accuracy than the other methods on different datasets.

Table 3: Summary of the results

Results	NSL-KDD dataset					CICIDS2017 dataset				
	XGBoost	KNN	RF	SVM	LR	XGBoost	KNN	RF	SVM	LR
Accuracy (%)	99.79	99.40	99.81	54.04	86.95	99.96	98.58	99.96	75.36	94.41
Precision (%)	99.98	99.51	99.96	54.04	84.80	99.99	98.43	99.98	72.23	89.44
Recall (%)	100	99.51	100	100	92.31	100	98.59	99.99	99.95	98.69
F1-score (%)	99.99	99.51	99.99	70.0	89.73	100	98.43	100	82.34	94.85

Table 4: Comparison of XGBoost and RF with other methods developed in recent research

References	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Description
[22]	99.70, 99.2	100	100	100	A DDoS attack detection system based on XGBoost algorithm for 5G networks, tested on the CICDDoS2019 [47] and NSL-KDD [37] datasets.

(Continued)

Table 4: Continued

References	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Description
[48]	100	100	100	100	A low-rate DDoS (LRDDoS) attack detection system that uses the RF algorithm on an independent dataset.
[49]	98.4	98	99	99	An intelligent intrusion detection system that applies machine learning, and was tested by using XGBoost on the UNSW-NB15 [29] dataset.
[50]	90.23, 99.43	98.74, 99.22	86.76, 99.30	92.36, 99.25	A model based on the logarithmic autoencoder (LogAE) and XGBoost, tested on the UNSW-NB15 [29] and CICIDS2017 [51] datasets.

4.2 Implementation of XGBoost and RF in MAIDS

The procedure to evaluate the proposed scheme consisted of calculating the FPR and FNR needed to apply Algorithm 1. To calculate these parameters was needed to plot the confusion matrix, where the generic form of which is shown in Fig. 13.

Network Status	**Normal**	**Abnormal**
Normal	**TP** (True Positives)	**FP** (False Positives)
Abnormal	**FN** (False Negatives)	**TN** (True Negatives)

Figure 13: Generic format of the confusion matrix

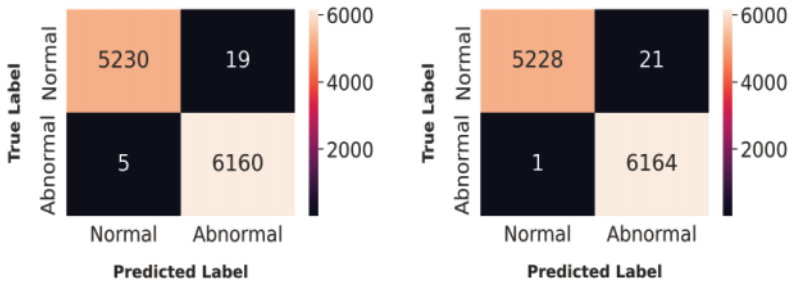
Starting from the generic confusion matrix in Fig. 13, can be written:

$$FPR = \frac{FP}{FP + TN} \tag{7}$$

$$FNR = \frac{FN}{FN + TP} \tag{8}$$

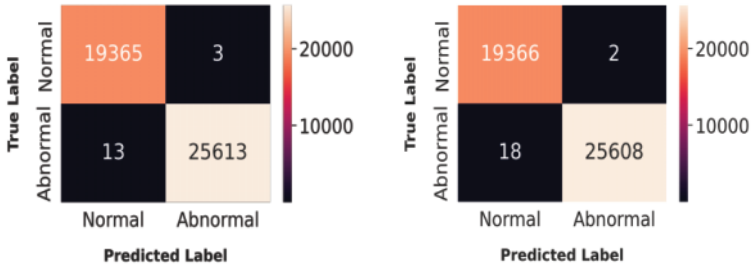
Figs. 14 and 15 show the confusion matrices of XGBoost and RF for the NSL-KDD and CICIDS2017 datasets, respectively, to calculate the FPR and FNR for these two algorithms. After

that, the mechanism to control false alarms was applied to evaluate the variation of these metrics. The results of the combination of XGBoost and RF on the NSL-KDD and the CICIDS2017 datasets are shown in Figs. 16 and 17. It can be concluded that the MAIDS scheme obtained lower FPR and FNR values on both datasets, compared to the values of the separate models, showing that the application of Algorithm 1 reduced the rate of false alarms. In addition, a comparison with other recent models, in Table 5, shows that the proposed scheme significantly reduced the false alarm rates, showing the effectiveness of the application of Algorithm 1.



(a) Confusion matrix of XGBoost (b) Confusion matrix of RF

Figure 14: Confusion matrices for the NSL-KDD dataset



(a) Confusion matrix of XGBoost (b) Confusion matrix of RF

Figure 15: Confusion matrices for the CICIDS2017 dataset

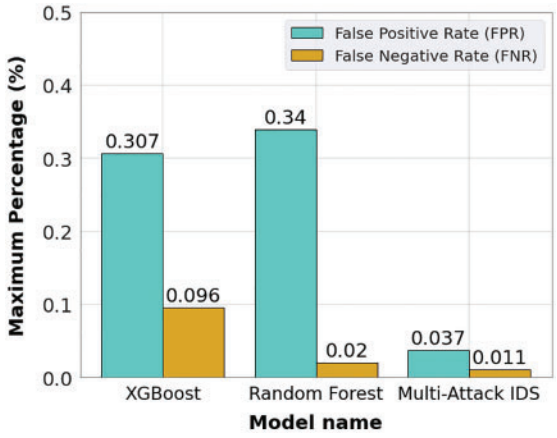


Figure 16: FPR and FNR for the NSL-KDD dataset

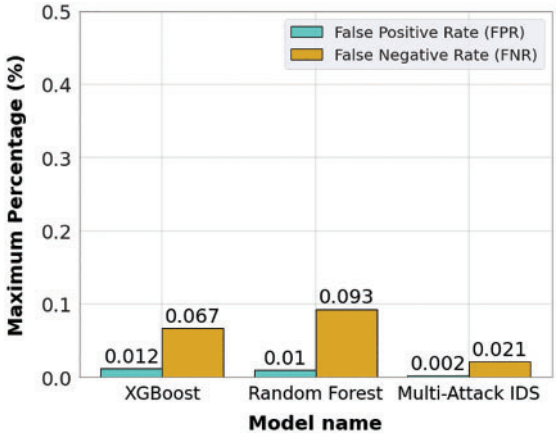


Figure 17: FPR and FNR for the CICIDS2017 dataset

Table 5: Comparison of the proposed model with others from recent literature

Model	FPR (%)	FNR (%)	Description
Proposed model	0.037, 0.002	0.011, 0.021	Multi-Attack IDS with NSL-KDD and CICIDS2017 datasets, respectively.
[26]	0.528	0.198	A hybrid deep learning technique for IDS with CICDDoS2019 dataset.
[40]	0.557	0.44	Combine two techniques to improve detection accuracy using NSL-KDD dataset.
[16]	6.82, 0.76	25.68, 8.34	Deep learning-based IDS for FNR reduction using NSL-KDD and UNSW-NB15 datasets, respectively.
[20]	0.21	0.12	Machine learning-based IDS for DDoS detection using a particular dataset.

The third phase of the proposed scheme consisted of selecting the best dataset that can be used with XGBoost and RF based on Algorithm 2. Both datasets were prepared to satisfy the assumptions of the application of this method, and the variables $m_i = \{FPR, FNP\}$ was defined as the evaluation metrics. The results show that MAIDS achieved an FPR of 0.037% and an FNR of 0.011% on the NSL-KDD dataset. Moreover, the scheme obtained an FPR of 0.002% and an FNR of 0.021% on the CICIDS2017 dataset. By using step 5 of Algorithm 2, it can be extracted $m_i = \{D_1[0.037\%, 0.011\%], D_2[0.002\%, 0.021\%]\}$. As the objective, was to minimize errors in detection, then, $D_2[0.002\%, 0.021\%] < D_1[0.037\%, 0.011\%]$. The CICIDS2017 dataset was, thus, chosen as the best for application in the proposed scheme. An analysis of the FPR and FNR of the XGBoost and RF algorithms showed that they achieved lower rates of false alarms on the CICIDS2017 dataset than on the NSL-KDD dataset.

5 Conclusion and Future Work

This paper proposed a dual machine learning-based IDS for SDN-IoT using XGBoost and RF. The scheme contains a mechanism to control and prevent false alarms to improve the security of IoT networks. The XGBoost and RF were selected through a comparative analysis of the main machine-learning algorithms used for the IDS in the SDN. The RF, XGBoost, and KNN achieved better results with respective accuracies of 99.89%, 99.88%, and 98.99% on both the NSL-KDD and the CICIDS2017 datasets. A comparison between XGBoost and RF with others models proposed in the recent literature showed that these algorithms can achieve better results on different datasets. Compared to separately implemented XGBoost and RF algorithms, the proposed scheme reduced the average FPR by 11.47% and the average FNR by 33.23% on the NSL-KDD dataset, and reduced the average FPR by 18.34% and the average FNR by 26.96% on the CICIDS2017 dataset. A method was also proposed to select the best dataset used in machine-learning algorithms to ensure network security. The method can help researchers to choose the best dataset among several for a given machine-learning algorithm. Tests of this method showed that the CICIDS2017 dataset was better for use with the proposed scheme than the NSL-KDD dataset, as it significantly reduced the FP and FN. In future work, researchers should examine other machine learning methods, such as deep learning algorithms, with the proposed security scheme and the mechanism to control false alarms. The proposed IDS can also be tested on datasets containing more recent attacks.

Funding Statement: This work was supported by the Pan African University Institute for Basic Sciences and Technology and Innovation.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding this study.

References

[1] Z. Shu, J. Wan, D. Li, J. Lin, A. Vasilakos *et al.,* "Security in software-defined networking: Threats and countermeasures," *Mobile Networks and Applications*, vol. 21, no. 5, pp. 764–776, 2016.

[2] N. Z. Bawany, J. A. Shamsi and K. Salah, "DDoS attack detection and mitigation using SDN: Methods, practices and solutions," *Arabian Journal for Science and Engineering*, vol. 42, no. 2, pp. 425–441, 2017.

[3] G. Pujolle, "SDN (Software-defined networking)," in *Software Networks Virtualization, SDN, 5G and Security*, 2nd ed., vol. 1. London, UK: ISTE Ltd and John Wiley & Sons, Inc., pp. 13–30, 2020.

[4] A. Pradhan and R. Mathew, "Solutions to vulnerabilities and threats in software defined networking (SDN)," in *Procedia Computer Science*, vol. 171. Kerala, India, pp. 2581–2589, 2020.

[5] O. Adekunle Okunade and E. GbengaDada, "Security algorithm for preventing malicious attacks in software defined network (SDN)," *Covenant Journal of Informatics & Communication Technology*, vol. 6, no. 2, pp. 51–63, 2018.

[6] R. Rietz, R. Cwalinski, H. König and A. Brinner, "An SDN-based approach to ward off LAN attacks," *Journal of Computer Networks and Communications*, vol. 18, no. 18, pp. 1–12, 2018.

[7] A. R. Choudhary, "OpenFlow switch controller as a policy-based system," *Issues in Information Systems*, vol. 22, no. 1, pp. 320–334, 2021.

[8] J. Yao, Z. Han, M. Sohail and L. Wang, "A robust security architecture for SDN-based 5G networks," *Future Internet*, vol. 11, no. 4, pp. 85–100, 2019.

[9] O. Yurekten and M. Demirci, "SDN-based cyber defense: A survey," *Future Generation Computer Systems*, vol. 115, no. 9, pp. 126–149, 2021.

[10] R. S. Alonso, I. Sittón-Candanedo, R. Casado-Vara, J. Prieto and J. M. Corchado, "Deep reinforcement learning for the management of software-defined networks and network function virtualization in an edge-iot architecture," *Sustainability (Switzerland)*, vol. 12, no. 14, pp. 5706–5729, 2020.

[11] K. Kalkan and S. Zeadally, "Securing internet of things with software defined networking (SDN)," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 186–192, 2018.
[12] M. Iqbal, F. Iqbal, F. Mohsin, M. Rizwan and F. Ahmad, "Security issues in software defined networking (SDN): Risks, challenges and potential solutions," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 10, pp. 298–303, 2019.
[13] P. Krishnan and J. S. Najeem, "A review of security, threats and mitigation approaches for SDN architecture," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 5, pp. 389–393, 2019.
[14] S. M. Tahsien, H. Karimipour and P. Spachos, "Machine learning based solutions for security of internet of things (IoT): A survey," *Journal of Network and Computer Applications*, vol. 161, no. 20, pp. 102630–102648, 2020.
[15] Milan, H. Sardana and K. Singh, "Reducing false alarms in intrusion detection systems–A survey," *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 2, pp. 9–12, 2018.
[16] J. Mijalkovic and A. Spognardi, "Reducing the false negative rate in deep learning based network intrusion detection systems," *Algorithms*, vol. 15, no. 8, pp. 258–286, 2022.
[17] W. Wei, B. Zhou, D. Połap and M. Woźniak, "A regional adaptive variational PDE model for computed tomography image reconstruction," *Pattern Recognition*, vol. 92, no. 19, pp. 64–81, 2019.
[18] X. Xia, M. Wozniak, X. Fan, R. Damaševičius, Y. Li *et al.,* "Multi-sink distributed power control algorithm for cyber-physical-systems in coal mine tunnels," *Computer Networks*, vol. 161, no. 19, pp. 210–219, 2019.
[19] W. Wei, H. Song, W. Li, P. Shen and A. Vasilakos, "Gradient-driven parking navigation using a continuous information potential field based on wireless sensor network," *Information Sciences*, vol. 408, no. 17, pp. 100–114, 2017.
[20] W. G. Gadallah, N. M. Omar and H. M. Ibrahim, "Machine learning-based distributed denial of service attacks detection technique using new features in software-defined networks," *International Journal of Computer Network and Information Security*, vol. 13, no. 3, pp. 15–27, 2021.
[21] R. Ahmed, S. Islam, S. Shatabda, A. Muzahidul and T. Robin, "Intrusion detection system in software-defined networks using machine learning and deep learning techniques–A comprehensive survey," *TechRxiv*, vol. 1, no. 21, pp. 1–56, 2021.
[22] H. A. Alamri, V. Thayananthan and J. Yazdani, "Machine learning for securing SDN based 5G network," *International Journal of Computer Applications*, vol. 174, no. 14, pp. 975–8887, 2021.
[23] P. Hadem, D. K. Saikia and S. Moulik, "An SDN-based intrusion detection system using SVM with selective logging for IP traceback," *Computer Networks*, vol. 191, no. 21, pp. 108015–108026, 2021.
[24] A. T. Kyaw, M. Zin Oo and C. S. Khin, "Machine-learning based DDOS attack classifier in software defined network," in *17th Int. Conf. on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, Phuket, Thailand, pp. 431–434, 2020.
[25] S. Y. Mehr and B. Ramamurthy, "An SVM based DDoS attack detection method for Ryu SDN controller," in *The 15th Int. Conf. on emerging Networking EXperiments and Technologies (CoNEXT'19 Companion)*, New York, NY, USA, pp. 72–73, 2019.
[26] D. Javeed, T. Gao and M. T. Khan, "SDN-enabled hybrid dl-driven framework for the detection of emerging cyber threats in IoT," *Electronics (Switzerland)*, vol. 10, no. 8, pp. 1–16, 2021.
[27] J. Cui, J. He, Y. Xu and H. Zhong, "TDDAD: Time-based detection and defense scheme against DDoS attack on SDN controller," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Cham, Switzerland, LNCS, vol. 10946, pp. 649–665, 2018.
[28] A. Victor Elijah, A. Abdullah, N. JhanJhi, M. Supramaniam, B. O. Abdullateef *et al.,* "Ensemble and deep-learning methods for Two-Class and multi-attack anomaly intrusion detection: An empirical study," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 9, pp. 520–528, 2019.
[29] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *MilCIS, 2015-Proc.*, Canberra, ACT, Australia, pp. 1–6, 2015.

[30] V. Ponnusamy, M. Humayun, N. Z. Jhanjhi, A. Yichiet and M. F. Almufareh, "Intrusion detection systems in internet of things and mobile Ad-Hoc networks," *Computer Systems Science and Engineering*, vol. 40, no. 3, pp. 1199–1215, 2021.
[31] D. Comaneci and C. Dobre, "Securing networks using SDN and machine learning," in *IEEE Int. Conf. on Computational Science and Engineering Securing*, Bucharest, Romania, pp. 194–200, 2018.
[32] A. Sebbar, K. Zkik, Y. Baddi, M. Boulmalf and M. Kettani, "MitM detection and defense mechanism cbna-rf based on machine learning for large-scale SDN context," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 12, pp. 5875–5894, 2020.
[33] H. Cheng, J. Liu, T. Xu, B. Ren, J. Mao *et al.,* "Machine learning based low-rate DDoS attack detection for SDN enabled IoT networks," *International Journal of Sensor Networks*, vol. 34, no. 1, pp. 56–69, 2020.
[34] F. Restuccia, S. D'Oro and T. Melodia, "Securing the internet of things in the age of machine learning and software-defined networking," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4829–4842, 2018.
[35] R. Santos, D. Souza, W. Santo, A. Ribeiro and E. Moreno, "Machine-learning algorithms to detect DDoS attacks in SDN," *Concurrency and Computation: Practice and Experience*, vol. 32, no. 16, pp. 1–14, 2020.
[36] Z. Chen, F. Jiang, Y. Cheng, X. Gu, W. Liu *et al.,* "XGBoost classifier for DDoS attack detection and analysis in SDN-based cloud," in *Proc.-2018 IEEE Int. Conf. on Big Data and Smart Computing*, Shanghai, China, pp. 251–256, 2018.
[37] Canadian Institute for Cybersecurity, "NSL-KDD dataset," *University of New Brunswick*, 2009. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html
[38] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *ICISSP, 2018-Proc. of the 4th Int. Conf. on Information Systems Security and Privacy*, Madeira, Portugal, pp. 108–116, 2018.
[39] M. W. Nadeem, H. G. Goh, V. Ponnusamy and Y. Aun, "DDoS detection in SDN using machine learning techniques," *Computers, Materials and Continua*, vol. 71, no. 1, pp. 771–789, 2022.
[40] M. R. Parsaei, S. M. Rostami and R. Javidan, "A hybrid data mining approach for intrusion detection on imbalanced NSL-KDD dataset," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 6, pp. 20–25, 2016.
[41] J. Note and M. Ali, "Comparative analysis of intrusion detection system using machine learning and deep learning algorithms," *Annals of Emerging Technologies in Computing (AETiC)*, vol. 6, no. 3, pp. 19–36, 2022.
[42] S. Prasath, K. Sethi, D. Mohanty, P. Bera and S. R. Samantaray, "Analysis of continual learning models for intrusion detection system," *IEEE Access*, vol. 10, no. 11, pp. 121444–121464, 2022.
[43] S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, vol. 17, no. 11, pp. 4766–4775, 2017.
[44] D. Stiawan, M. Idris, A. M. Bamhdi and R. Budiarto, "CICIDS-2017 dataset feature analysis with information gain for anomaly detection," *IEEE Access*, vol. 8, no. 1, pp. 132911–132921, 2020.
[45] M. Mehmood, T. Javed, J. Nebhen, S. Abbas, R. Abid *et al.,* "A hybrid approach for network intrusion detection," *Computers, Materials and Continua*, vol. 70, no. 1, pp. 91–107, 2021.
[46] N. N. Tuan, P. H. Hung, N. D. Nghia, N. Van Tho, T. Van Phan *et al.,* "A DDoS attack mitigation scheme in ISP networks using machine learning based on SDN," *Electronics (Switzerland)*, vol. 9, no. 3, pp. 413–432, 2020.
[47] I. Sharafaldin, A. H. Lashkari, S. Hakak and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *IEEE 53rd Int. Carnahan Conf. on Security Technology*, Chennai, Índia, pp. 1–8, 2019.
[48] W. D. Nanda and F. D. S. Sumadi, "LRDDoS attack detection on SD-IoT using random forest with logistic regression coefficient," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 2, pp. 220–226, 2022.
[49] N. Yadav, S. Pande, A. Khamparia and D. Gupta, "Intrusion detection system on IoT with 5G network using deep learning," *Wireless Communications and Mobile Computing*, vol. 22, no. 1, pp. 1–13, 2022.

[50] W. Xu and Y. Fan, "Intrusion detection model based on autoencoder and XGBoost," *Hindawi, Security and Communication Networks*, vol. 22, no. 1, pp. 1–7, 2022.
[51] C. I. for Cybersecurity, "Intrusion detection evaluation dataset (CIC-IDS2017)," *University of New Brunswick*, 2017. [Online]. Available: https://www.unb.ca/cic/datasets/ids-2017.html