Grenze International Journal of Engineering and Technology, Jan Issue

Enhancing Network Security: A Novel Hybrid ML Approach for DDoS Attack Detection in SDN

Rakesh V S[1] and Vasanthakumar G U[2]
[1]Department of Computer Science and Engineering, Cambridge Institute of Technology, Bengaluru, Visvesvaraya Technological University, Belagavi, Karnataka, India.
Email: rakesh.tech102@gmail.com
[2]Department of Computer Science and Engineering, Nitte Meenakshi Institute of Technology, Bengaluru, Visvesvaraya Technological University, Belagavi, Karnataka, India.
Email: vasanth.gu@nmit.ac.in

***Abstract*— Software-defined networking represents a ground breaking advancement in network technology, characterized by its desirable attributes such as enhanced flexibility and manageability. Although ongoing, the issue of DDoS assaults in SDN is characterized by malicious and obtrusive network traffic that overwhelms SDN resources. Despite numerous security methodologies aimed at detecting DDoS attacks, the challenge of effectively addressing this issue continues to persist as an active area of research. The XG-Light Hybrid, a unique hybrid system, has been developed in this work as a solution to this problem. This discovery is significant because it has the potential to dramatically increase the reliability of DDoS attack detection in SDN environments, hence boosting network security and stability. Key findings reveal that the proposed hybrid approach outperforms individual machine learning algorithms with respect to DDoS detection.**

***Index Terms*— DDoS Detection, Machine Learning, Network Traffic, Light GBM, SDN, and XG Boost.**

I. INTRODUCTION

In the last ten years, the network infrastructure has undergone significant expansion, marked by a proliferation of network devices. This expansion has introduced greater complexity in network administration and posed potential barriers to future innovations in the realm of the Internet. Additionally, traditional network structures, characterized by their inflexibility, have limited adaptability and led to increased operational costs. As a result, these difficulties have slowed the development of emerging technologies like big data, cloud computing, and the Internet of Things, all of which require increased bandwidth, flexibility, and management. In response to these circumstances, Software-Defined Networking (SDN) has emerged as a revolutionary networking model. SDN is poised to meet the escalating demands of next-generation networks and emerging technologies, offering a promising solution to the administrative complexities and limitations associated with traditional network architectures [1].

The Software-Defined Networking (SDN) architecture constitutes the entire network infrastructure. It adopts a modular approach that outlines a hierarchy and sets standards for how components within the network interact with one another [2]. Fig. 1. provides a visual representation of the SDN architecture, comprising three main layers:

Grenze ID: 01.GIJET.10.1.351

the data/infrastructure, control, and application layers.

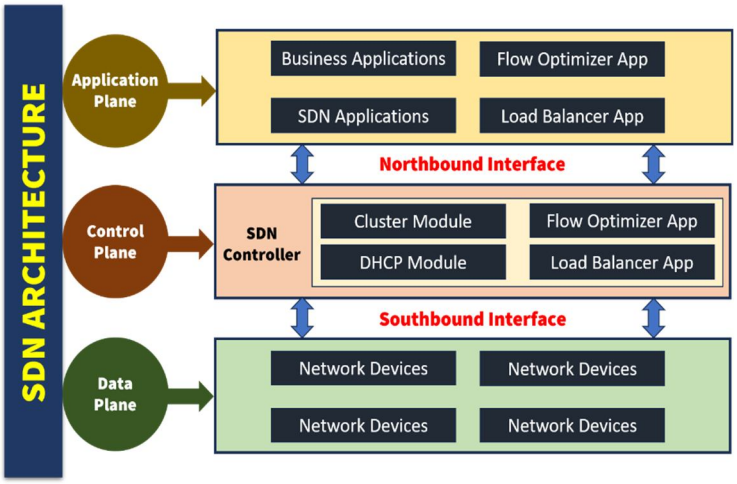

Fig. 1 SDN Architecture.

Within Fig 1. two primary elements stand out: controllers and forwarding devices. A forwarding device can be either a hardware or software component specialized in the task of packet forwarding. As opposed to that, the controller functions as the network's central intelligence, responsible for managing the network's state and operations [3]. As a revolutionary paradigm, SDN holds the promise of addressing the requirements for flexible, secure, dependable, and efficiently managed next-generation networks. It brings forth significant potential and optimism for meeting these demands head-on, ushering in a new era of network capabilities [4].

Due to the distinctive attributes of SDN architecture, conventional DDoS attack detection methods employed on the SDN controller often exhibit limited accuracy in identifying such attacks [5]. In order to discern the presence of a DDoS attack, the SDN controller must continuously amass network traffic data from switches, leading to an elevated workload for the controller. Consequently, there arises a necessity to establish a dedicated security framework to counter this threat [6].

A DDoS (Distributed_Denial_of_Service) attack is a common threat in which attackers take control of numerous devices, turning them into bots that work together to flood a system with fake traffic or requests. The goal is to overwhelm the system's resources or congest the victim's network as shown in Fig 2. Unlike some other network attacks, DDoS doesn't involve stealing sensitive information; instead, it aims to disrupt services temporarily. DDoS attacks are relatively simple to launch but challenging to trace back to their source.

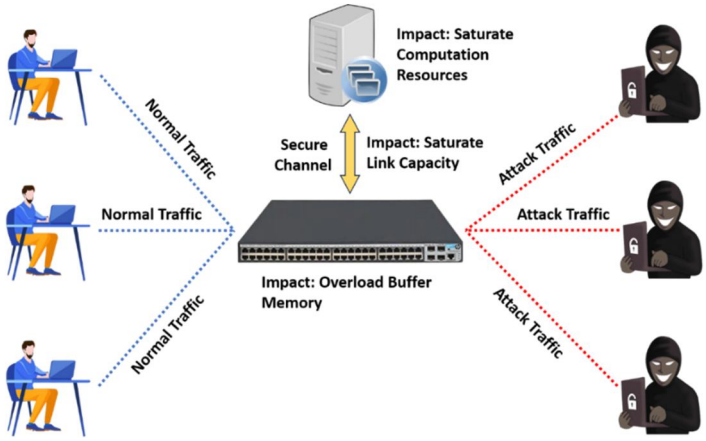

Fig 2. DDoS attack resulting from compromised nodes [7].

In a typical DDoS assault, multiple compromised devices inundate a target network with an overwhelming volume of traffic, exhausting its resources and rendering legitimate traffic inaccessible. Within SDN, the attack surface expands due to the centralization of control, making it essential to address DDoS threats comprehensively. Within SDN, the classification of DDoS is as shown in Fig 3.

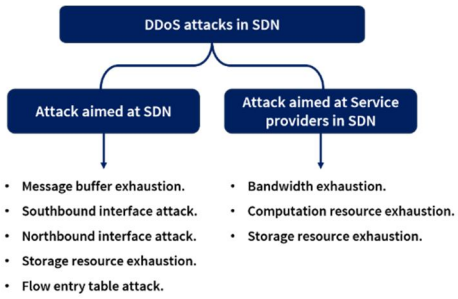

Fig 3. Overview of DDoS attacks in SDN.

II. Literature Review

It is because machine learning techniques have the capacity to make wise decisions and operate effectively automatically, they have gained a lot of traction in the field of cybersecurity [8]. The most successful machine learning algorithms developed in this domain are gathered in order to conduct a complete review of the techniques available. Machine learning approaches have demonstrated efficacy in a number of cybersecurity-related areas, including spam detection, malware detection, user identity verification, software vulnerability discovery, and DDoS attack detection [9].

In this section, a lot of research about how to stop, find, and deal with DDoS attacks using machine learning is examined. Machine learning-based Distributed Denial of Service (DDoS) detection approaches can be broadly classified into three main categories: Ensemble-based ML schemes, Hybrid-based ML schemes, and Single ML schemes methods. Table 1. also summarizes few schemes in the same context.

A. Ensemble based ML schemes

Maheshwari et. al., proposed an enhanced Weighted Voting Ensemble (OWVE) model designed for the detection and mitigation of Distributed Denial of Service (DDoS) attacks. This ensemble model leverages Support Vector Machines (SVM), Random Forests (RF), and Gradient-Boosted Machine classifiers, each with distinct hyperparameter configurations. Notably, the ensemble model exhibits remarkable classification accuracy, achieving 99.41% and 99.35% accuracy rates when tested on the CIC-DDoS-2019 and CAIDA-2007 datasets, respectively [10].

An intrusion detection system based on voting that is intended to protect SDN from DDoS assaults, has been introduced by swami et. al. This innovative voting model underwent training and testing utilizing three datasets: UNSW-NB15, CICIDS2017, and NSL-KDD, consistently demonstrating superior detection accuracy compared to alternative methods [11].

An ensemble machine learning technique, referred to as K-mean and RF, has been innovated to enhance the precision and effectiveness of identifying and detecting DDoS attacks. This advanced system underwent rigorous training and testing utilizing the InSDN dataset, resulting in flawless detection accuracy, achieving a remarkable 100% [12].

B. Hybrid based ML schemes

To distinguish between regular network traffic and traffic from DDoS attacks, Support Vector Machine (SVM) and Random Forest (RF) classification algorithms were combined. This method underwent rigorous testing and evaluation using a real-world Software-Defined Networking (SDN) dataset [13]. A study was conducted to explore the integration of P4 programmable technology and machine learning algorithms, including K-NN, RF, SVM, and ANN, for the development of a real-time detection system. An automated method for detecting DDoS attacks, specifically SYN flood attacks, at the local level on SDN switches was proposed [14].

In their study, Nadeem et al., utilized a range of machine learning algorithms, including Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbors (K-NN), Naïve Bayes (NB), and Decision Tree (DT). Their approach was rigorously assessed using the NSL-KDD dataset, resulting in notably high accuracy rates, particularly achieving an outstanding 99.97% accuracy for the Decision Tree (DT) algorithm [15]. In a separate research investigation, a diverse a variety of machine learning classification models, such as Logistic Regression (LR), Decision Trees (DT), Random Forest (RF), AdaBoost (AB), Multilayer Perceptron (MLP) were employed to analyze and identify TCP-SYN flood Distributed Denial of Service (DDoS) attacks targeting the SDN controller [16].

C. Single algorithm-based ML schemes

The approach proposed by Nurwaristo et. al., utilizes Random Forest (RF) to classify network traffic into normal and abnormal categories based on flow entries. If packets are classified as a DDoS attack, specific switch rules are applied for mitigation. This system achieves an impressive 98.38% overall detection accuracy while ensuring swift mitigation [17]. Oo et al. proposed an Advanced-SVM algorithm to detect UDP and SYN flood DDoS attacks in SDN networks, achieving strong overall evaluation performance: 87% for precision, 84% for recall, and 93% for F1-score, as demonstrated on SDN-TrafficsDS and KDDCUP99 datasets [18]. An approach centered around factorization machines (FM) was employed for improved accuracy in detecting low-rate Distributed Denial of Service (DDoS) attacks on SDN data planes. While it effectively identifies such attacks, the approach achieved a reasonably good detection accuracy of 95.80%. [19].

TABLE I. CONCISE SUMMARY OF OTHER ML-BASED DDOS DETECTION IN SDN NETWORKS

Scheme Reference	Approach	DDoS Techniques	ML Techniques	Research Gap
[20]	Ensemble	Detection	k-NN, NB, SVM & SOM	The proposed approach was assessed using datasets that did not accurately represent SDN network characteristics, resulting in suboptimal performance for the ensemble approach.
[21]	Hybrid	Detection	NB, SVM, XGBoost & RF	the proposed method's evaluation was conducted using a dataset that does not faithfully capture the SDN network's characteristics.
[22]	Hybrid	Detection & Mitigation	RF, SVM & MLP	The evaluation of the proposed model utilized an unrealistic dataset that does not mirror the characteristics of the SDN network environment.
[23]	Hybrid	Detection & Mitigation	XGBoost & Bandwidth Control	The proposed system's operation on the controller introduces excessive load and overhead. Additionally, the approach encounters significant processing and communication overhead.
[24]	Single	Detection & Mitigation	BPNN	The proposed approach exhibits a relatively low detection accuracy of 96.13% due to significant issues with false-positive and false-negative outcomes.

III. PROPOSED DDOS DETECTION MODEL

In predicting potential DDoS attacks on the SDN controller, a machine learning (ML) approach was employed, utilizing historical attack data. Fig 4 provides an overview of the proposed detection system.

In the proposed project, a DDoS attack detection technique is put into action within the SDN controller environment using machine learning. A combination of XGBoost and Light GBM models is employed to construct classification models as shown in Fig. 4. This approach leverages advanced traffic-related features to establish a model capable of categorizing SDN flow packets as either typical or suggestive of a DDoS attack. The feature data is sourced from an examination of the network traffic's headers and statistics.

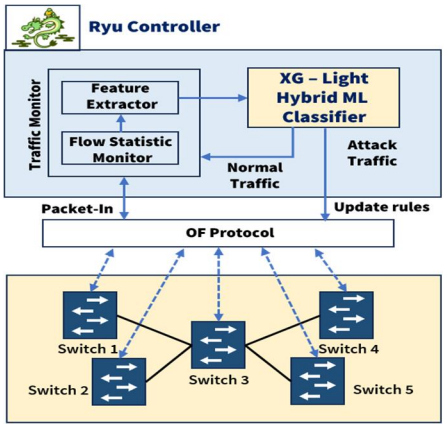

Fig 4. XG – Light Hybrid ML Classifier for DDoS detection

When the controller receives an OF (OpenFlow) packet from the OF switch, it initiates a series of actions. First, it extracts the source IP address from the incoming flow packet and checks if this address is listed in a blocklist. If it finds a match, the packet is dropped, and important statistics like the frequency of packets from that source address, timestamps, and other relevant data are recorded and considered. Conversely, if there's no match with any

blocked address, the packet's headers and associated statistics become input features for testing by a classifier. If the classifier determines that the incoming packet is normal, it's allowed to continue its journey as usual.
However, if the packet is classified as part of a DDoS attack, even if it's the first time that the source IP address is flagged as an attacker, that IP address is added to the blocklist. Additionally, all packet details are logged, ensuring that any suspicious packet from the same source IP address is discarded. Furthermore, various traffic flow statistics, including packet arrival timestamps and the number of packets sent by the same host, are recorded for analysis.

Algorithm: *Proposed Hybrid Algorithm*

```
1   while true do
2       Packets are sent by OF switch to the controller
3       Extract IP address of the source from each packet
4       if (IP matches with an entry in Blocklist) then
5           Collect all statistics from the packet
6           Update it in block list
7           Send to the switch to update flow table entry
8       else
9           Store: Time stamp, Destination IP, Port numbers, Transport Layer Protocol, Type of
                Service, and Payload length headers.
10          Obtain features by performing calculations that are necessary using the gained information.
11      Input the features into proposed hybrid model
12      if (The model detects a DDoS attack) then
13          Add the IP of the source into blocklist
14          Log all the necessary statistics
15      else
16          serve the packets normally and log the statistics for future use.
```

IV. Simulation And Result Analysis

In this section, we outline the setup for our experiments and detail the technologies we employed to assess the effectiveness of various machine learning techniques and the feasibility of our framework.

A. Experimental and Simulation setup

The experimental setup was executed on an HP laptop equipped with a Core i7 CPU and 16 GB of RAM, running the Linux Ubuntu 16.04 LTS operating system. The Mininet emulator and RYU controller were used to generate the test environment with Open vSwitches implementing the OpenFlow protocol version 1.3, forming an integral part of this configuration. The network topology is a linear topology consisting of 64 hosts, 8 switches, and one controller as shown in Fig 5.

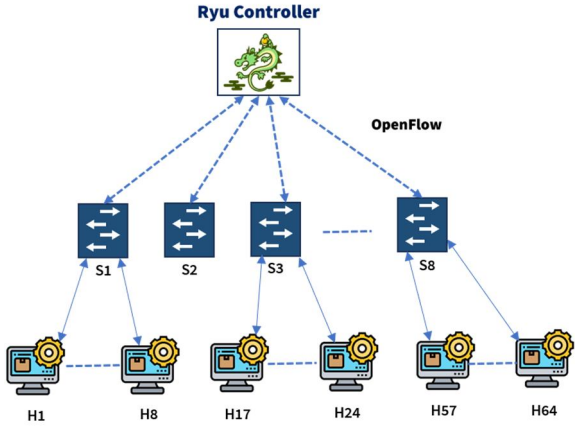

Fig 5. Simulation Network Topology

B. Dataset Generation

In order to evaluate the effectiveness of the proposed method, we established a virtual network and conducted simulations using Mininet with the Ryu controller. For packet generation, we employed Scapy. This virtual environment was configured with a Ryu controller, eight switches, and a total of 64 hosts, as depicted in Fig 5.

The CICDDoS2019 dataset, as outlined in [25], is a public dataset comprises both legitimate network traffic and contemporary DDoS attack patterns, making it a faithful representation of real-world data. It offers valuable insights into network traffic through the application of CICFlowMeter-V3, categorizing flows with timestamps, source and destination IPs, ports, protocols, and attack labels. This dataset served as the foundation for training and assessing the effectiveness of the proposed model in our research.

C. Performance Evaluation Criteria

The classifier's key performance metrics have been used to evaluate performance. derived from the confusion matrix presented in Table 2.

TABLE II. CONFUSION MATRIX FOR THE MODEL

		PREDICTED VALUE	
		POSITIVE	NEGATIVE
ACTUAL VALUE	POSITIVE	TP	TN
	NEGATIVE	FP	FN

In this context, True Positive (TP) indicates instances where the model accurately identifies the positive class. True Negative (TN) represents situations where the model correctly predicts the negative class. False Positive (FP) occurs when the model incorrectly forecasts the positive class, deviating from the actual classification criteria. False Negative (FN) arises when the model mistakenly anticipates the negative class, contradicting the established standards.

The evaluation of the approach's performance involved the use of various performance metrics, such as accuracy, precision, recall, and F-Measure, which gauges the correlation among traffic flows [26].

Accuracy is determined by calculating the ratio of correct predictions to the total number of predictions using Equation (1):

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$

Among all the expected positive cases, precision counts the number of genuine positive cases. It offers a value between 0 and 1, and Equation (2) can be used to determine precision:

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Recall quantifies the number of predicted positive cases as a percentage of all positive cases and is computed using Equation (3):

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

The F1-score is a single metric that takes into consideration both accuracy and recall; by accounting for erroneous positives and false negatives, it is useful for imbalanced datasets. Equation (4) is utilized in its calculation.

$$F1-score = \frac{2\ X\ (precision\ X\ recall)}{(precision+\ recall)} \quad (4)$$

D. Results and Discussions

The recommended strategy is examined in this section and illustrates the enhancements it achieves through an evaluation of different ML models, including Random Forest (RF), Naive Bayes (NB), Support Vector Machine (SVM), and k-Nearest Neighbor (k-NN). The analysis was independently performed on both the publicly available dataset and the dataset generated through simulation. In both scenarios, it was observed that the proposed model consistently outperformed existing ML models concerning the considered parameters. This highlights the effectiveness of proposed approach in achieving superior results compared to established machine learning models.

- *Analysis based on Public Dataset*

Following an analysis conducted using the public dataset, it has been noted that the proposed approach consistently demonstrates better results in comparison to existing methods as shown in Fig 6.

- *Analysis based on Generated Dataset*

An analysis was additionally carried out on the dataset generated within the simulation environment, as previously detailed in the preceding section. It's worth noting that due to the limited volume of data collected, the values for the considered parameters were relatively low. Nevertheless, even in this scenario, the proposed approach consistently outperformed alternative methods as shown in Fig 7, further affirming its effectiveness.

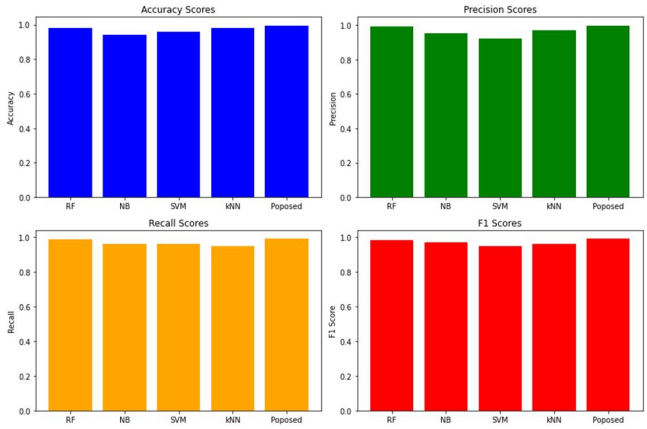

Fig 6. Comparison of various ML techniques for the public dataset.

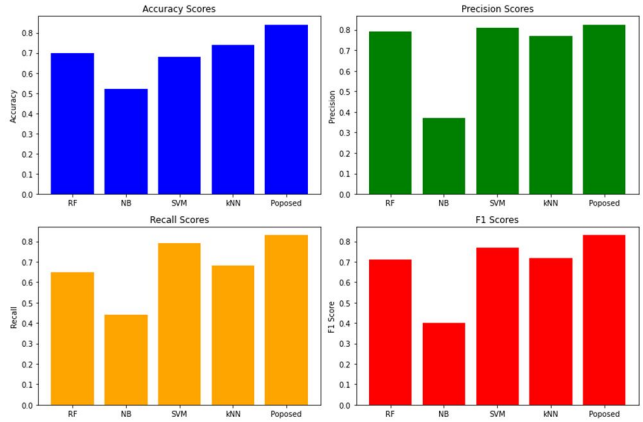

Fig 7. Comparison of various ML techniques for the generated dataset.

V. Conclusion

DDoS detection was carried out on two distinct datasets: the standard CICDDoS2019 dataset and a dataset generated within a virtual environment. In the methodology, feature selection was based on the type of attack, followed by the implementation of various machine learning procedures. The performance of various machine learning algorithms is evaluated in terms of accuracy, precision, recall, and F1-score. In the generated dataset, the overall model performance notably decreased due to the limited dataset size with a smaller number of features.
Overall, the analysis revealed thsat the proposed XG-Light hybrid algorithm outperforms other considered machine learning models in effectively identifying DDoS attacks. In the future, this research can be expanded by experimenting with multiple controllers in SDN environments. Additionally, conducting experiments with much larger datasets and comparing the outcomes with existing models can provide valuable insights.

References

[1] H. S. Abdulkarem and A. D. Alethawy, "DDOS attack detection and mitigation at SDN environment," Iraqi Journal of Information and Communication Technology, vol. 4, pp. 1-9, 2021
[2] Alashhab, A.A.; Zahid, M.S.M.; Azim, M.A.; Daha, M.Y.; Isyaku, B.; Ali, S. A Survey of Low Rate DDoS Detection Techniques Based on Machine Learning in Software-Defined Networks. Symmetry 2022, 14, 1563.
[3] Fan, N. M. Kaliyamurthy, S. Chen, H. Jiang, Y. Zhou, and C. Campbell, "Detection of DDoS attacks in software defined networking using entropy," Applied Sciences, vol. 12, p. 370, 2021.

[4] Yunhe Cui, Qing Qian, Chun Guo, Guowei Shen, Youliang Tian, Huanlai Xing, Lianshan Yan, Towards DDoS detection mechanisms in Software-Defined Networking, Journal of Network and Computer Applications, Volume 190, 2021, 103156, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2021.103156.
[5] 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 (IJCNIS), vol. 13, pp. 15-27, 2021.
[6] M. W. Nadeem, H. G. Goh, V. Ponnusamy, and Y. Aun, "DDoS Detection in SDN using Machine Learning Techniques," Comput. Mater. Contin., vol. 71, pp. 771-789, 2022.
[7] J. D. Gadze, A. A. Bamfo-Asante, J. O. Agyemang, H. Nunoo-Mensah, and K. A.-B. Opare, "An investigation into the application of deep learning in the detection and mitigation of DDOS attack on SDN controllers," Technologies, vol. 9, p. 14, 2021.
[8] Wang M, Lu Y, Qin J. A dynamic MLP-based DDoS attack detection method using feature selection and feedback. Comput Secur.2020;88:101645. doi:10.1016/j.cose.2019.101645
[9] Thomas T, Vijayaraghavan A, Emmanuel S. Machine Learning and Cybersecurity. Springer Singapore; 2020:37-47.
[10] Maheshwari, A.; Mehraj, B.; Khan, M.S.; Idrisi, M.S. An optimized weighted voting based ensemble model for DDoS attack detection and mitigation in SDN environment. Microprocess. Microsyst. 2022, 89, 104412.
[11] Swami, R.; Dave, M.; Ranga, V. Voting-based intrusion detection framework for securing software-defined networks. Concurr. Comput. Pract. Exp. 2020, 32, e5927.
[12] Firdaus, D.; Munadi, R.; Purwanto, Y. DDoS Attack Detection in Software Defined Network using Ensemble K-means++ and Random Forest. In Proceedings of the 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), Yogyakarta, Indonesia, 10–11 December 2020; pp. 164–169.
[13] Ahuja, N.; Singal, G.; Mukhopadhyay, D.; Kumar, N. Automated DDOS attack detection in software defined networking. J. Netw. Comput. Appl. 2021, 187, 103108.
[14] Musumeci, F.; Fidanci, A.C.; Paolucci, F.; Cugini, F.; Tornatore, M. Machine-Learning-enabled DDoS Attacks Detection in P4 Programmable Networks. J. Netw. Syst. Manag. 2022, 30, 1–27.
[15] Nadeem, M.W.; Goh, H.G.; Ponnusamy, V.; Aun, Y. DDoS Detection in SDN using Machine Learning Techniques. Comput. Mater. Contin. 2022, 71, 771–789.
[16] Swami, R.; Dave, M.; Ranga, V. Detection and Analysis of TCP-SYN DDoS Attack in Software-Defined Networking. Wirel. Pers. Commun. 2021, 118, 2295–2317.
[17] Nurwarsito, H.; Nadhif, M.F. DDoS Attack Early Detection and Mitigation System on SDN using Random Forest Algorithm and Ryu Framework. In Proceedings of the 2021 8th International Conference on Computer and Communication Engineering (ICCCE), Kuala Lumpur, Malaysia, 22–23 June 2021; pp. 178–183.
[18] Oo, M.M.; Kamolphiwong, S.; Kamolphiwong, T.; Vasupongayya, S. Analysis of Features Dataset for DDoS Detection by using ASVM Method on Software Defined Networking. Int. J. Netw. Distrib. Comput. 2020, 8, 86–93.
[19] Zhijun,W.; Qing, X.; Jingjie,W.; Meng, Y.; Liang, L. Low-Rate DDoS Attack Detection Based on Factorization Machine in Software Defined Network. IEEE Access 2020, 8, 17404–17418.
[20] Deepa, V.; Sudar, K.M.; Deepalakshmi, P. Design of Ensemble Learning Methods for DDoS Detection in SDN Environment. In Proceedings of the 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Vellore, India, 30–31 March 2019; pp. 1–6.
[21] Alamri, H.A.; Thayananthan, V. Analysis of Machine Learning for Securing Software-Defined Networking. Procedia Comput. Sci. 2021, 194, 229–236.
[22] Pérez-Díaz, J.A.; Valdovinos, I.A.; Choo, K.K.R.; Zhu, D. A Flexible SDN-Based Architecture for Identifying and Mitigating Low-Rate DDoS Attacks Using Machine Learning. IEEE Access 2020, 8, 155859–155872.
[23] Alamri, H.A.; Thayananthan, V. Bandwidth Control Mechanism and Extreme Gradient Boosting Algorithm for Protecting Software-Defined Networks Against DDoS Attacks. IEEE Access 2020, 8, 194269–194288.
[24] Hannache, O.; Batouche, M. Neural Network-Based Approach for Detection and Mitigation of DDoS Attacks in SDN Environments. Int. J. Inf. Secur. Priv. (IJISP) 2020, 14, 50–71.
[25] Noman, H.M.; Jasim, M.N. POX Controller and Open Flow Performance Evaluation in Software Defined Networks (SDN) Using Mininet Emulator. IOP Conf. Ser. Mater. Sci. Eng. 2020, 881, 012102.
[26] Alwabisi, Sulaiman, Ridha Ouni, and Kashif Saleem. 2022. "Using Machine Learning and Software-Defined Networking to Detect and Mitigate DDoS Attacks in Fiber-Optic Networks" Electronics 11, no. 23: 4065. https://doi.org/10.3390/electronics11234065