Comparative Evaluation of Machine Learning Efficacy in DoS Attack Detection

I.A. Oshin, O.A. Odeniyi
Federal University of Technology, Akure, Nigeria

Abstract

Software-Defined Networking (SDN) presents a novel approach to network architecture, characterized by the separation of control and data planes, enabling centralized management and agile resource allocation. However, this newfound flexibility also exposes networks to significant security vulnerabilities, particularly concerning Denial-of-Service (DoS) attacks, which disrupt service availability by overwhelming network resources. This study explores the application of machine learning techniques to detect and mitigate DoS attacks in SDN environments. Utilizing the InSDN dataset, a benchmark specifically designed for SDN-based intrusion detection, two widely used machine learning models—Support Vector Machine (SVM) and Random Forest (RF)—were trained to differentiate between normal and attack traffic. The experimental results indicate that the Random Forest model outperforms SVM, achieving a remarkable accuracy of 98%, while SVM follows closely with 97%. In order to resolve class imbalances present in the dataset, the research evaluates the models under other situations, such as under and over-sampling. The results demonstrate the Random Forest model's better performance across a range of data handling techniques, highlighting its resilience and adaptability. This research contributes to the growing body of research on using machine learning to improve SDN's security posture and makes a compelling argument for incorporating these advanced algorithms into existing network security frameworks. This study highlights the significance of choosing suitable models and techniques based on particular data distributions and security requirements by clarifying the advantages and disadvantages of both models. This helps practitioners strengthen SDN infrastructures against changing cyber threats.

Keywords: Software-Defined Networking (SDN), InSDN dataset, Support Vector Machine (SVM), Random Forest (RF), Denial-of-Service (DoS), Distributed Denial-of-Service (DDoS)

1. Introduction.

The explosion of internet connected devices has led to an increased dependence on digital infrastructure, making network security an urgent priority for organizations. One of the most persistent security threats today is the Denial-of-Service (DoS) and of distributed DoS (DDoS) attack, which aims to overwhelm network resources with malicious traffic, rendering them inaccessible to legitimate users [1], these attacks are now a major worry for people, companies, and governments due to the growing dependence on online services [27] as the post devastating outcomes such as financial losses, harm to one's reputation, and interruption of necessary services. [24]

Despite various strategies to detect and prevent DoS attacks, they remain a significant challenge, therefore finding effective defenses against DoS attacks is crucial for maintaining the availability and reliability of vital services [2]. DoS attacks have become more sophisticated over time, initially a straightforward attack involving a single source flooding a target with excessive traffic. With improvements in defense against these attacks, Distributed Denial of Service (DDoS) attacks, more complex strategies among others has been devised by these attackers where numerous compromised systems attack a target simultaneously. Machine Learning (ML) has been employed by the cybersecurity field to combat these evolving threats as a promising solution as traditional security measures such as firewalls and intrusion detection systems (IDS), often struggle to effectively counter these sophisticated attacks, highlighting the need for more advanced solutions [3]. ML algorithms are capable of detecting threats, by learning from historical attack data and adapting to new threats, has the ability to analyze vast amounts of data and detect patterns, thereby, offering enhanced accuracy and timeliness in identifying abnormal activities. This makes ML algorithms highly suitable for the dynamic nature of network security [4]. Despite the potential of machine learning to enhance network security, challenges remain. The success of ML models relies on the quality and quantity of training data, the choice of features, and their ability to generalize from training data to real-world scenarios [5].

Software-Defined Networking (SDN) has emerged as an innovative network management

 | DOI: 10.20533/ijisr.2042.4639.2024.0125

Framework. SDN separates the network's control and data planes, offering centralized control and easier management. However, this centralization also creates potential security vulnerabilities, particularly against DoS attacks [6] This research explores the application of machine learning (ML) techniques to enhance DoS detection in SDN environments. By using data from the InSDN dataset, we compare the performance of two widely-used machine learning models: Support Vector Machine (SVM) and Random Forest (RF).

2. Literature Review

Software-defined networking (SDN) is a family of technologies that launched an era of radical innovation and transformation in the telecommunications industry, reminiscent of the early days of the Internet. SDN is a new network paradigm that is backed by many well-known companies, including Google [7] and Cisco Systems [8]. SDN is a new networking paradigm that enables the breaking of the ossification barrier. The SDN concept centralizes network control logically, instead of distributing it among each network device, enabling software to manage and program the network. Its foundation is the physical separation of network forwarding devices, such as switches, hubs, routers, and the like, from network intelligence, or the control plane [9]. Using SDN, network management becomes simpler and helps to remove network rigidity [10].

[28] assessed different intrusion detection methods using the KDD Cup '99 dataset. More recent studies have made use of more realistic and extensive datasets like CICDDoS2019 [29], UNSW-NB15 t[31], and NSL-KDD [30]. More realistic assessments are made possible by these datasets, which offer a greater variety of attack scenarios and network traffic characteristics.

The use of machine learning for DoS attack detection has been investigated in a number of research. [26] used the NSL-KDD dataset to test various machine learning techniques. Their results demonstrated how RF and SVM can be used to achieve excellent detection accuracy. Likewise, [25] looked at the application of deep learning methods for DoS detection and found encouraging outcomes. [11] suggested a new way to find and stop network intrusions by using the NSL-KDD dataset to make a machine-learning algorithm that can spot DoS and port scanning attacks. Using One Hot Encoding to transform the categorical data to non-categorical data to perform the data preprocessing and the ANOVA test for feature selection, the Naive Bayes model produced the best results with an accuracy of 86.9% for DoS attacks and 93.5% for port scanning attacks. [12] also used a public dataset to test SVM, Naive Bayes, J48, and Random Forest ML methods for finding DOS and DDoS attacks. When compared, J48's accuracy was the greatest at 80%. [13] used C4.5 (Decision Tree), Bayesian Network, Naive Bayes, and Decision Table for benchmarking with the Longtail Project 19 data. The attack was predicted using historical network data, and the Bayesian network had an accuracy of 91.68%. [14] assessed various machine learning techniques utilizing feature selection through Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), and Naive Bayes, emphasizing the accuracy of these methods.

The precision, F1 score, and accuracy all differed significantly when the feature selection approach was used. [15] conducted an experiment in a simulated environment to validate that Software-Defined Networking (SDN) enhances the implementation of Machine Learning (ML) for Intrusion Detection Systems (IDS) in several ways. SVM achieved 97.5% accuracy, while three of the four algorithms tested had over 96% accuracy. Furthermore, a DDoS attack was not combined with excessive bandwidth utilization.

[16] discussed how SDN helps network engineers monitor, control, and identify malicious traffic and link failure effortlessly and efficiently in a network. Datasets were created and the machine learning algorithms were trained on the resulting SDN dataset. Their findings show that the hybrid support vector classifier with random forest (SVC-RF) model is good at finding malicious packets in a mininet emulator-simulated data set. Moreso, [17] discussed the cyber-security threats faced in deployments of IoT. Large-scale IoT device networks are notably susceptible to hijacking and the execution of distributed denial of service (DDoS) attacks, as seen by the notorious Mirai incident and other persistent threats. The utility of a dataset derived from an urban IoT deployment comprising 4060 nodes, alongside a synthetic DDoS attack generator that introduces attack activity into the dataset based on adjustable parameters such as the number of nodes targeted and the attack duration, was demonstrated through the training and evaluation of a basic multi-label feed-forward neural network.

In order to discuss the novel security and privacy risks that software-defined networking (SDN) presents, [18] use a variety of classification methods, such as support vector machines (SVMs), K-nearest neighbours (KNNs), decision trees (DTs), multiple layer perceptrons (MLP), and convolutional neural networks (CNNs), and compare their performance. CNN has the lowest prediction accuracy (90.08%) and the highest train accuracy (97.808%) whereas the SVM-based DDoS detection model performs better, with the greatest prediction accuracy of 95.5%.

[19] stated that cyberattacks can lead to power outages, problems with military equipment, and breaches of confidential information, such as the theft of medical records if they fall into the wrong hands. Support vector machines (SVM), K-Nearest Neighbors (KNN), and random forest algorithms (RF)

 | DOI: 10.20533/ijisr.2042.4639.2024.0125

are a few of the classification models that were used to predict distributed denial-of-service (DDOS) assaults on financial organizations using the Banking Dataset. When it comes to detecting DDoS attacks, the SVM has an accuracy of 99.5%, while KNN and RF received scores of 97.5% and 98.74%, respectively. It was determined that, in terms of robustness, the SVM performs better than KNN, RF, and current machine learning (ML) and deep learning (DL) techniques.

[20] discussed how well SVMs can analyse network flow features to quickly find and stop DoS attacks. Their study reported that SVMs can handle real-time data streams well, which makes them a good choice for use in network security applications. According to [21] algorithms like isolation forests and one-class SVMs have demonstrated the potential to detect new attack patterns and zero-day vulnerabilities. Moreover, adaptive network security has discovered a new frontier in reinforcement learning (RL), where algorithms can adjust and improve defence plans in response to continuous interactions with network environments. Because RL can dynamically adjust to evolving attack strategies, it is a powerful tool for creating robust DoS mitigation systems.

Machine learning models have proven to be extremely effective tools in improving network security for real-time detection and prevention of Denial of Service (DoS) attacks. Random Forest Classifier, Decision Tree Classifier, Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), and others are some of the machine learning algorithms that can accurately recognising and classifying network attacks. Despite machine learning's potential to enhance network security, real-world applications require the resolution of several issues. Since machine learning models must manage high network traffic volumes effectively while preserving low latency, scalability is still a major challenge [22.]

Real-time processing skills are also required to ensure prompt detection and mitigation of denial-of-service (DoS) attacks and reduce their influence on network functions. [23] explained that integration of ML with SDN systems can generate the potential to develop dynamic and policy-driven defence mechanisms. Machine learning models, which can analyse network traffic patterns in SDN systems in real-time, can enable automated reaction mechanisms to combat DoS attacks, ensuring optimal network performance and resource allocation.

3. Methodology

This study employs the InSDN dataset, a specialized dataset for SDN environments. The dataset includes both normal traffic and various attack types, including DoS.

3.1. Dataset

In SDN is a new dataset specifically designed to address the leakages in the SDN networks. The dataset contains all the attacks targeted at the three layers of SDN (data, control and application). The dataset, a standard benchmark for SDN network was introduced to address the challenges of existing traditional networks datasets, which are seen to be outdated and ineffective in handling SDN based security challenges. For this study, 48-SDN-based features whose labels are DoS attack and normal were extracted from the InSDN dataset (see Figure 1).

	precision	recall	f1-score	support
0	0.99	0.95	0.97	252
1	0.95	0.99	0.97	248
accuracy			0.97	500
macro avg	0.97	0.97	0.97	500
weighted avg	0.97	0.97	0.97	500

Figure 1. Removing duplicate records with under-sampling using the SVM Model

Duplicate records total of 4988 (4675 normal traffic and 313 DoS records) were removed to eliminate irrelevant records and to prevent redundancy. Under-sampling technique was applied to resolve the dataset imbalance which after implementation resulted to a normal traffic and DoS having 832 records each from a normal class of 63749, and DoS class of 832 (see Figure 2).

	precision	recall	f1-score
0	0.97	0.99	0.98
1	0.99	0.97	0.98
accuracy			0.98
ıacro avg	0.98	0.98	0.98
;hted avg	0.98	0.98	0.98

Figure 2. Removing duplicate records with over-sampling using the SVM Model

The total record used for the study is 69569, in whi.ch normal traffic is 68424 while the DoS record is 1145.

3.2. Models for Classification

In this research, we uses two machine learning classifiers; Random Forest (RF) and Support Vector Machine (SVM)) to classify DoS and normal traffic. By using the bagging strategy during the training phase, RF, a tree-based supervised learning model frequently employed for regression and classification tasks, lowers the high variance of those models. Due to its random selection of data tuples, consideration of packet length, inter-arrival time, and average bulk

 | DOI: 10.20533/ijisr.2042.4639.2024.0125

rate, and reduction of its influence on network capacity, the RF model is particularly useful for servers with higher traffic. The following equations express the RF algorithm is expressed with equation 1-4. RF is schematically represented in Figure 2

$$y = \sum_{k=1}^{n} f(x) \quad (1)$$

$$In\frac{p}{1-p} = a + by \quad (2)$$

$$\frac{p}{1-p} = e^{a+by} \quad (3)$$

$$P = \frac{e^{a+by}}{1+e^{a+by}} \quad (4)$$

One kind of supervised learning method that can assist with classification, regression, and outlier detection is support vector machines (SVMs). Support vector machines are helpful in high-dimensional settings, and the technique works even when there are more dimensions than samples. The general architecture of SVM is shown in Figure 3.

	precision	recall	f1-score
0	0.99	1.00	1.00
1	0.94	0.59	0.73
accuracy			0.99
ıacro avg	0.97	0.79	0.86
;hted avg	0.99	0.99	0.99

Figure 3. Removing duplicate records with no sampling using the SVM Model

The vector input signal (x) is contained in the input layer. The difference between the input signal vector (x) and support vector (s) is calculated by the hidden layer (y). The linear outputs O of the hidden layer neurons are added together by the output neuron. Equation 5 shows the algorithm of SVM:

$$O = \sum W_i k((x_i s_i) \quad (5)$$

3.3. Performance Metrics

The Models performance was evaluated based on several performance metrics, including accuracy, precision, recall, and F1-score. Equation 6- 9 represents the Accuracy, Precision, Recall and F1-score respectively.

$$\text{Accuracy}=(TP+TN)/(TP+TN+FP+FN) \quad (6)$$

$$\text{Precision}=(TP)/(TP+FP \quad (7)$$

$$\text{Recall}=(TP)/(TP+FN) \quad (8)$$

$$\text{F1-score} = 2\times(\text{Precision}\times\text{Recall})/(\text{Precision}+\text{Recall}) \quad (9)$$

Where: TP is True Positive (Correctly identified attack), TN is True Negative (Correctly identified normal attack), FN is False Negative (Attack classified as normal traffic), and FP is False Positive (Normal traffic classified as attack)

4. Results

We compared the performance of two machine learning models, Support Vector Machine (SVM) and Random Forest (RF), in detecting Denial of Service (DoS) attacks in Software Defined Networking (SDN) environments using the InSDN dataset. The dataset contained both normal and attack traffic, with significant class imbalance.

4.1. Evaluating SVM Model

The model was evaluated by removing duplicate records under three conditions which are: Under-sampling, Over-sampling and No sampling. The model achieved 97% accuracy with well-balanced precision, recall, and F1-score for Under-sampling. With Over-sampling, the accuracy of the model increased to 98%, with consistent improvements in all metrics while the model performed best with 99% accuracy, showing its sensitivity to natural data distributions when evaluated under the No sampling condition. It achieved strong precision and recall, suggesting that SVM works optimally when data is left unaltered without additional sampling (see Figures 4, 5, and 6).

4.2. Evaluating RF Model

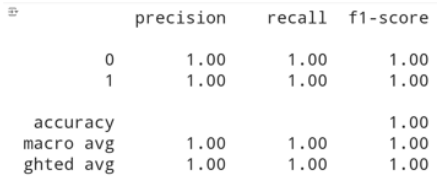
Similar tests were performed with SVM was also done with Random Forest under the same three conditions. The RF model achieved 98% accuracy, with balanced precision and recall across all metrics for Under-sampling. The model achieved 100% accuracy, precision, recall, and F1-score, showing its adaptability to Over-sampling while RF achieved 100% accuracy with perfect performance across all metrics, demonstrating its flexibility in handling imbalanced datasets when evaluated under the No sampling condition.

```
# classification report
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.99	0.98	0.98	252
1	0.98	0.99	0.98	248
accuracy			0.98	500
macro avg	0.98	0.98	0.98	500
weighted avg	0.98	0.98	0.98	500

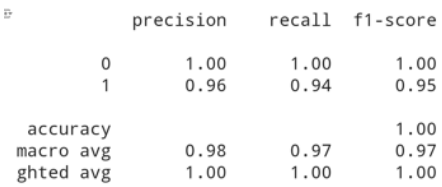
Figure 4. Removing duplicate records with under-sampling using the RF Model

 | DOI: 10.20533/ijisr.2042.4639.2024.0125



	precision	recall	f1-score
0	1.00	1.00	1.00
1	1.00	1.00	1.00
accuracy			1.00
macro avg	1.00	1.00	1.00
ghted avg	1.00	1.00	1.00

Figure 5 Removing duplicate records with over-sampling using the RF Model



	precision	recall	f1-score
0	1.00	1.00	1.00
1	0.96	0.94	0.95
accuracy			1.00
macro avg	0.98	0.97	0.97
ghted avg	1.00	1.00	1.00

Figure 6. Removing duplicate records with no sampling using the RF Model

4.3. Comparative Analysis of The Two Models

Assessing the performance of both models (the SVM and RF) was carried out across three scenarios: removing duplicate records with under-sampling, over-sampling, and without applying any sampling technique. Both models exhibited strong classification performance across all scenarios; however, key differences emerged in terms of accuracy, precision, recall, and F1 score.

Table 1. Performance of SVM and RF model of DoS detection

S/N	MODEL	ACCURACY (%)	PRECISION (%)	RECALL (%)	F1 SCORE(%)
1	Support vector machine	98	98	99	98
2	Random forest	97	95	99	97

SVM performed best when no sampling technique was applied, reaching 99% accuracy, which was slightly lower in both under- and over-sampling conditions. RF outperformed SVM overall, achieving 100% accuracy in both over-sampling and no-sampling scenarios, making it the more robust model. Even under-sampling did not significantly affect RF's performance (maintaining 98%), making it ideal for real-world applications where data distribution might vary. The results showed that both models performed well in classifying normal and DoS traffic. The Random Forest model achieved an accuracy of 98%, while the SVM model followed closely with 97%. The RF model performed best when no sampling techniques were applied, indicating its Resilience in handling imbalanced datasets. In contrast, the SVM model performed slightly better when no additional sampling techniques were applied, achieving 99% accuracy. This implies that SVM models may benefit from datasets that maintain their natural distribution.

Table 2. The comparison study of SVM, KNN, RF with existing ML

S/N	Reference	Model	Accuracy (%)	Dataset
1	This Study	SVM and RF	98 and 97	InSDN
2	(Tariq Emad Ali, Chong, and Manickam, 2023)	SVM and KNN	95.08 and 91.23	CICDDoS2019
3	(Islam et al., 2022)	KNN	97.5	InSDN

The results of this study indicate that machine learning models, particularly Random Forest, are effective tools for detecting DoS attacks in SDN environments. The Random Forest model's ability to achieve perfect classification performance, regardless of the sampling technique, makes it a robust choice for real-world applications. Meanwhile, the SVM model offers competitive performance, especially in scenarios where preserving the original data distribution is important. These findings highlight the potential of machine learning for enhancing SDN security. By comparing results of this study with previous studies (Table 2), the results of this study perform significantly well above the previous ones. While both models showed high classification performance, Random Forest proved to be more flexible and robust across different data handling techniques, making it the preferred choice for detecting DoS attacks in SDN environments. However, SVM remained competitive, particularly when the natural distribution of the data was preserved. This study highlights the importance of selecting the appropriate model and sampling technique based on the specific data distribution and task requirements.

5. Conclusion

This study compared the classification performance of the SVM and RF models under three different data preparation approaches: removing duplicates with under-sampling, over-sampling, and without applying any sampling technique.

The SVM model delivered its highest performance when duplicates were removed without any additional sampling indicating that SVM performs best on datasets that reflect the original, natural distribution of the data. Both under-sampling and over-sampling slightly decreased the model's performance, suggesting that SVM is somewhat sensitive to changes in dataset balance. In contrast, the RF model demonstrated exceptional flexibility. It achieved perfect classification metrics 100% accuracy, precision, recall, and F1-score when using either over-sampling or no sampling techniques. It maintained strong performance in the under-sampling scenario, with only a minor decrease to 98% highlighting RF's ability to handle imbalanced data while still delivering high performance across all sampling techniques. From a practical point of view, the RF model emerges as the superior choice for tasks requiring consistently

 | DOI: 10.20533/ijisr.2042.4639.2024.0125

high performance across varied data distributions, as it consistently outperformed the SVM model in this study. However, for datasets where maintaining the original data distribution is critical, the SVM model provides competitive performance, particularly when no additional sampling techniques are applied.

Conclusively, both models offer strong classification capabilities, but the RF model's ability to achieve flawless performance in diverse scenarios makes it the preferred option for many applications. The SVM model remains a solid alternative, particularly in cases where preserving the natural balance of the dataset is essential.

6. References

[1] Ramprasth J., Krishnaraj, N., and Seethalakshmi, V. (2022). Mitigatin services on SDN for distributed denial of service and denial of service attacks using machine learning techniques. IETE Journal of Research, 1–12. DOI: 10.1080 /03772063.2022.2142163.

[2] Aslan, Ö., Aktuğ, S. S., Ozkan-Okay, M., Yilmaz, A. A., and Akin, E. (2023). A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions. Electronics, 12(6), 1–42. DOI: 10.3390/electronics120613 33.

[3] Aljuhani, A. (2021). Machine Learning Approaches for Combating Distributed Denial of Service Attacks in Modern Networking Environments. IEEE Access, 9, 42236–42264. DOI: 10.1109/access.2021.3062909.

[4] Peneti, S., and Hemalatha, M. (2021). Enhancing network security: Machine learning models for real-time detection and prevention of DoS attacks.

[5] Shaghaghi, A., Kaafar, M. A., Buyya, R., and Jha, S. (2020). Software-Defined Network (SDN) Data Plane Security: Issues, Solutions, and Future Directions. In Handbook of Computer Networks and Cyber Security, 341–387. DOI: 10.1007/978-3-030-22277-2_14.

[6] Do, E. H., and Gadepally, V. N. (2020). Classifying anomalies for network security.

[7] Jain, S., Zhu, M., Zolla, J., Hölzle, U., Stuart, S., Vahdat, A., and Zhou, J. (2013). B4. ACM SIGCOMM Computer Communication Review, 43(4), 3–14. DOI: 10.1145/2534169.2486019.

[8] CISCO Systems. (2019). Software-Defined Networking Overview.

[9] Hussein, A., Chadad, L., Adalian, N., Chehab, A., Elhajj, I. H., and Kayssi, A. (2019). Software-defined networking (SDN): The security review. Journal of Cyber Security Technology, 1–66. DOI: 10.1080/23742917.2019.1629529

[10] Zhu, L., Karim, M. M., Sharif, K., Xu, C., Li, F., Du, X., and Guizani, M. (2021). SDN controllers. ACM Computing Surveys, 53(6), 1–40. DOI: 10.1145/3421764 11.

[11] Almasri, W., et al. (2022). Network intrusion detection using ML on the NSL-KDD dataset. Sustainability, 14(2), 2235.

[12] Elsayed, A., et al. (2020). Comparison of ML models for detecting DoS/DDoS attacks. IEEE Access, 8, 135233–135241.

[13] Nanda, S., Zafari, F., DeCusatis, C., Wedaa, E., and Yang, B. (2016). Predicting network attack patterns in SDN using machine learning approach. 2016 IEEE Conference on NFV-SDN. DOI: 10.1109/nfv-sdn.2016.7919493.

[14] Polat, H., Polat, O., and Cetin, A. (2020). Feature selection methods and ML models for DDoS detection in SDNs. Sustainability, 12(3), 1035. DOI: 10.3390/su120 31035.

[15] Ahmad, A., et al. (2020). Security in SDN using machine learning. IEEE Globecom Workshops. DOI: 10.1109/gcwkshps50303.2020.9367477.

[16] Ahuja, N., Singal, G., Mukhopadhyay, D., and Kumar, N. (2021). Automated DDOS attack detection in software-defined networking. Journal of Network and Computer Applications, 103108. DOI: 10.1016/j.jnca.2021.103108.

[17] Hekmati, A., Safaei, B., and Noori, S. (2021). Cybersecurity threats in IoT: A study of Mirai and other DDoS attacks. Journal of Cyber Security Technology, 5(1), 33–48. DOI: 10.1080/23742917.2020.1848580.

[18] Tariq, E. A., Chong, Y.-W., and Manickam, S. (2023). Comparison of ML/DL approaches for detecting DDoS attacks in SDN. Applied Sciences, 13(5), 3033–3033. DOI: 10.3390/app13053033.

[19] Islam, U., Muhammad, A., Mansoor, R., Hossain, M. S., Ahmad, I., Eldin, E. T., and Shafiq, M. (2022). Detection of distributed denial of service (DDoS) attacks in IoT-based monitoring system of the banking sector using machine learning models. Sustainability, 14(14), 8374. DOI: 10.3390/su14148374.

[20] Buczak, A. L., and Guven, E. (2016). A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Communications Surveys and Tutorials, 18(2), 1153-1176.

[21] Zhang, H., Cai, Z., Liu, Q., Xiao, Q., Li, Y., and Cheang, C. F. (2020). A survey on security-aware measurement in SDN.

[22] Yang, Y., and Zhao, R. (2022). ML scalability in real-time detection of DoS attacks.

[23] Chanhemo, K., et al. (2023). ML integration with SDN systems for dynamic defense mechanisms. Electronics, 12(4), 1045.

[24] Mirkovic, J., and Reiher, P. (2004). A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review, 34(2), 39–53.

[25] Hasan, M., Islam, M. M., Zarif, M. I. I., and Hashem, M. M. A. (2016). Attack and anomaly detection in IoT

sensors in IoT sites using machine learning approaches. Internet of Things, 1, 118-131.

[26] Idrees, S., Rajarajan, M., Conti, M., and Chen, L. (2019). A comparative study of machine learning algorithms for network intrusion detection. 2019 IEEE Symposium on Computers and Communications (ISCC), 1-6.

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

[28] Tavallaee, M., Bagheri, E., Lu, W., and Ghorbani, A. A. (2009). A comparative study of machine learning techniques for intrusion detection. Proceedings of the 2009 International Conference on Data Mining, DMIN 2009, 31–37.

[29] Sharafaldin, I., Lashkari, A. H., Ghorbani, A. A., and Claise, B. (2019). Toward generating a new intrusion detection dataset: Evaluating the performance of machine learning algorithms. 2018 14th International Conference on Network and Service Management (CNSM), 706–711.

[30] Revathi, S., and Malathi, A. (2013). A review on intrusion detection using neural network. International Journal of Computer Applications, 64(1).

[31] Moustafa, N., and Slay, J. (2015). UNSW-NB15: A comprehensive dataset for network intrusion detection systems (UNSW-NB15 network data set). 2015 Military Communications and Information Systems Conference (MilCIS), 1–6.

 | DOI: 10.20533/ijisr.2042.4639.2024.0125