International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 01, ISSN No. 2455-2143, Pages 261-267
Published Online May 2023 in IJEAST (http://www.ijeast.com)

DDOS ATTACK DETECTION AGAINST SDN CONTROLLER USING A SINGLE TRAFFIC FEATURE

Aishatu Abdullahi Wabi, Dr Ismail Idris, Prof Olayemi Mikail Olaniyi, Dr. Joseph A. Ojeniyi, Dr. Olawale Surajudeen Adebayo, Andrew Anogie Uduimoh.
Department of Cyber Security
Federal University of Technology Minna, Nigeria

Abstract— **The most widely used southbound API of the software-defined network is the Open Flow protocol. Each flow in Open Flow has a set of packet-forwarding rules, which are referred to as flow entries. The switch processes packets in the SDN operation that meet the flow entries. The Packet that doesn't match any entries is transmitted as a Packet_in message to the Controller. Therefore, sending a lot of Packet in messages in a short amount of time could bring down the controller, and as a result, the entire network consequently resulting in to distributed denial of Service Attacks (DDOS). This study uses the rate of Packet_in as a single feature, monitor, extract and utilize it to identify DDOS attacks in SDN using Random Fores classifier. The result shows 99.8% Accuracy which is slightly better than the work of [24] with 99.7%.**

Keyword- **Openflow, Packet_in, SDN, DDOS, Random Forest**

I. INTRODUCTION

Network technology faced difficulties as information communication improved. The restrictions of the conventional networks prevent quick network configuration and management. However, a potential tool that addresses some of these issues by guaranteeing programmability, manageability, and permitting innovation on a Network is the Software-defined network (SDN) [8]. The Control Plane and Data Plane are separated by SDN. The Control plane, Data plane, and Application plane make up the SDN architecture. The Southbound interface is used for communication between the Control and Data planes. The Open flow protocol is the Southbound API that is most frequently utilized. OpenFlow defines packet-forwarding rules for each flow, and these rules are referred to as flow entries. When SDN is in use, the switch processes packets that match the flow entries. The Packet that matches no entries is delivered to the Controller as a Packet_in message [5], whereas the other packets are not. The controller will install fresh flow entries into the switches as soon as it gets Packet-In messages and forward the packets contained therein. The controller needs some Packet-In messages to learn the network statuses from packets, such as the MAC address learning, the multicast source identification, the ARP, and the broadcast handling. Although installing the majority of the flow entries before receiving the packets is advised, this is not always possible. Overloading of the controller occurs when an excessive number of Packet _in messages are received in a short period of time [8]. Therefore, attackers take advantage of this flaw by sending numerous distinct flows to the switch that is sent to the Controller as Packet_in for processing and consequently overloading the controller and finally disrupting the network service of the SDN network.

A number of strategies have been put forth to identify and stop the Controller attack.

[23] suggested a Support vector machine-based classification for DDOS attacks. The 6-tuple features for DDOS detection were created by the authors by compiling features from the Switch flow table. Attack detection is made up of the flow state collection, which receives flow status information from the flow table switch and sends it to the module for character value extraction, which then sends it to the classifier judgment for SVM classification.

The use of five tuple characteristics in a DDOS attack Detection method that combines entropy and ensemble learning was proposed by [24]. [7] suggested a time series analysis and statistical DDOS attack Detection and Defense System. In order to detect instant changes in network behaviour, the study extracted two key features from the open flow switch flow table, such as a unique source IP address and a normalized unique destination IP address [11]. Based on the entropy variation of the destination IP addresses of the data flows, early DDoS attack detection against SDN controllers was designed. To identify a controller attack, the system keeps track of the destination IP of incoming packets and counts the number of packets originating from that IP. If multiple packets are arriving at the same location, the entropy is reduced, and the network anomaly is picked up. For their experimental setup, they utilized a Mininet network emulator along with the Scapy program to create traffic. [6] put forth a powerful Detection technique that is intended to both identify DDoS attacks and pinpoint the affected networks that the malicious attackers have linked to. The sequential probability ratio test (SPRT), which has bounded false negative and false positive

International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 01, ISSN No. 2455-2143, Pages 261-267
Published Online May 2023 in IJEAST (http://www.ijeast.com)

error rates, is chosen after the algorithm classifies the flow events connected to an interface. In order to identify packet in message attacks in an SDN Bayes network, the authors of [8] used Features such as Mean packet per flow, Percentage of pair flows. growth of foreign flows, expansion of several ports, deviations of both packet counts and bye counts.

Variety of detection methodologies have proposed using numerous detection features, however, the rate of Packet_in messages received by the Controller was the only feature employed in this research work to detect DDOS attacks. This will increase detection accuracy and decrease the overhead associated with computing the values of many characteristics.

II. BACKGROUND

A. Software Defined Network (SDN)

SDN is new network paradigm, that physically separates the Control Plane of network from the Data plane. In comparison with the traditional Network, SDN offers the advantages of Programmability, flexibility and, centralized control there by allowing the innovation and implementation of unique policy on a network. [12] The SDN Advantages is presented in Figure 1.1.

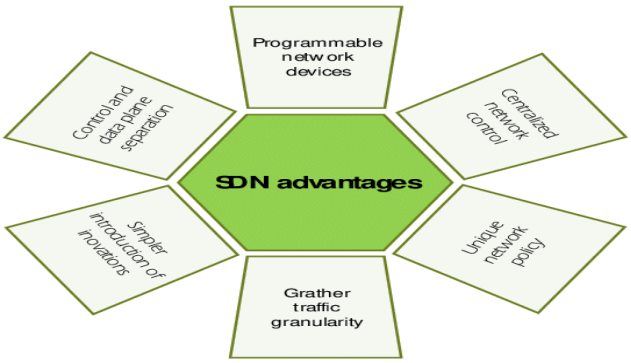

Figure 1.1: SDN Advantages

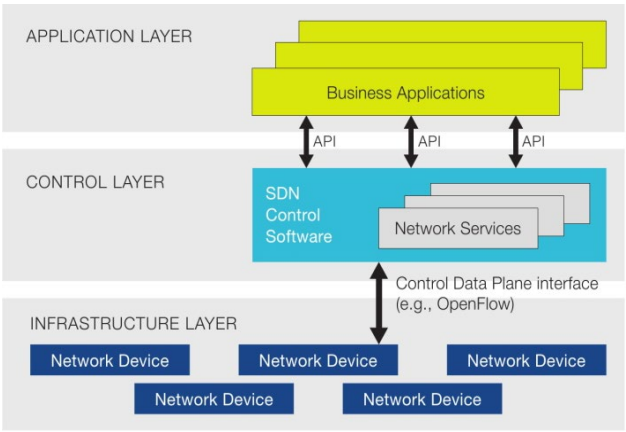

Figure 1.2 SDN Layers [24]

The Existing network architectures were not designed to meet the requirements of today's users, enterprises, and carriers; rather network designers are constrained by the limitations of current networks such as complexity, scalability and inconsistency policies. However, SDN is geared towards solving these challenges.

B. SDN Architecture: The architecture of SDN is made up of 3 layers and different interfaces as shown in figure 1.2. The layers are described as follows:

a. The Application Layer: An abstract representation of the network is offered by this layer. It permits the provision of application services like traffic engineering, security monitoring, and load balancing [19].

b. Control Layer: The SDN's centralized control functionality is located on the Control layer. It must decide how packets should be sent via one or more devices [14] through a Controller, which is the SDN's primary component. It is centralized logically and may also be distributed physically. The west and east-bound interfaces allow for communication between the distributed controllers. South-bound Application programming interface (API) like OpenFlow is used by this layer to connect with the infrastructure layer [18].

c. Infrastructure Layer: On this layer, packet forwarding devices reside and follow instructions from a logically centralized Controller. Consequently, this plane is known as the Data Plane [18]. This layer guarantees a respectable level of network virtualization, security accessibility, and quality preservation at the same time [14].

d. Northbound Interface: Communication between the application layer and the Control plane is made possible through the Northbound interface provided by the REST API. While guaranteeing that the internal workings of the network are concealed, this interface helps to achieve network programmability.[18]

e. Southbound interface: The interfaces enabling communication between the Control plane and Data plane are referred to as the southbound interface. The Southbound API offers Services including event notification, Capabilities advertisement, statistics reporting, programmatic control of all forwarding activities, and statistics reporting. The OpenFlow Protocol is the most used Southbound interface Protocol [18].

C. Open Flow Protocol

The Communication Channel between the Data plane and the Control plane is provided by the OpenFlow protocols. In order to communicate information and authenticate users, it created a secure connection between the planes. For the purpose of configuring and managing flows, the Protocol permits the Controller to have access to the OpenFlow switch's flow table. When a new packet-in message is received, the OpenFlow Protocol is used by the Controller to determine the best path. The Controller then informs the switch of the path by sending a packet-out message, and the switch then changes its flow table. The flow table entries, which can forward the packet to

International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 01, ISSN No. 2455-2143, Pages 261-267
Published Online May 2023 in IJEAST (http://www.ijeast.com)

one or more interfaces, are searched by the SDN to handle the pertinent network traffic. Each entry has a header field as well as counters and actions. The flow table serves as the foundation for the switch's packet routing. Multiple flow entries make up each flow table. The rules for data forwarding are formed by the flow table entries. The flow table entry structure diagram is shown in Figure 1.3.

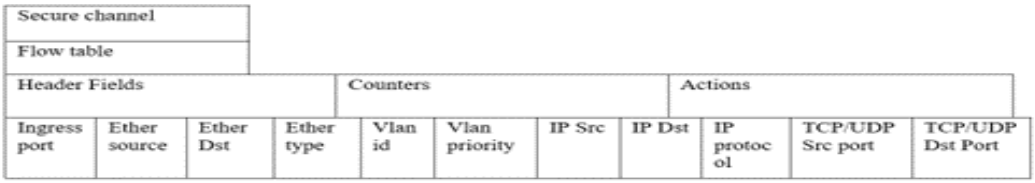

Figure: 1.3 OpenFlow Table diagram [22].

D. DDOS ATTACKS IN SDN

i. DDOS attack on Data Plane: In [4], it is amply demonstrated how DDOS attacks impact the Data plane. Attackers take advantage of the Switch's limited memory storage, as reported in [22]), [21], [13]), [2] and the SDN Switch's idle time-out mechanism [22]. When a matching flow is missing from the flow table, the Switch will often send some or all of a Switch packet's headers to the Controller while the packet is being stored in the Switch's nodes while waiting for the controller to respond [9]. As a result, an attacker keeps sending fresh, unknown packets before the idle timeout, taking into account the switch's limited storage, which causes switch buffer saturation and flow table overflow, finally draining the data plane infrastructure resources. [22], [21]

ii. DDOS attack on Control Plane: The control plane is where SDN's centralized control functions are located. An attacker might exploit the controller as a single point of failure [16] to launch DDOS attacks by flooding it with Packet In messages, depleting its resources, and blocking legitimate users from accessing it. The attacker might even knock down the entire network. [15] [2], [13] [4], [16].

ii. DDOS attack in Application Plane: Different apps offer the controller various services. To access network resources, some apps might cover their tracks under others. In order to deplete network resources or bring down networks, rogue applications can use network resources whilst another program is running [9]. [18] [16], [9]. Additionally, this attack results from a lack of Standard permission. and authentication mechanism for checking the validity of the Applications. [9].

iv. DDOS Attack on Control-Data Plane: The South Bound Interface (SBI), a channel or bandwidth used by the data plane to connect with the controller, occasionally transmits messages to and from the data and control plane [2]. The communication connection between the Data plane and the Control plane may be subject to DDOS attacks, as shown by [10]. Attackers may transmit more and more traffic in an effort to saturate the available bandwidth between the two planes, which would bring down the controller and the entire network. [15], [2], [4], [13].

v. DDOS Attack on Control – Application Plane: The communication route between the Control and Application, known as the Northbound API, was described in [9] as being vulnerable to DDOS attacks due to the lack of a common protocol for the North-bound API that would provide communication between the two planes. [4].

III. DETECTION APPROACH

The traffic generation, Feature Collection, Feature Extraction and Random Forest Classifier module constitute the detection approach.

i. **Traffic Generation**: On an SDN network, traffic generation entails producing both regular and attack traffic. TCP Syn flooding attack, UDP flooding attack, and ICMP flood attacks were all launched to achieve these goals.

ii. **Feature Collection**: In this phase, during a predefined interval, the Controller sends Flow_stat_request through OFPFLOWSTATREQUEST handler to individual switch on the network and in turn the switch responds to the Controller with a reply using OFPFLOWSTATREPLY handler with the flow statistics.

iii. **Feature Extraction Module**: This module utilizes the flow statistics and used it to extract the required features which will be used to distinguish attacks by the Next Modules follows. The feature extracted is described as follows:

Rate of packet in: The Controller of the SDN could be overwhelmed by sending multiple Packet_in message to Controller there by consuming the Controller resource and denying legitimate service. Hence to distinguish Normal traffic from legitimate traffic we used the number packet in message that were sent to the controller for processing within a specific time interval.

$$rate\_pkt\_in = \frac{\sum No\ of\ packet\_in}{T}$$

iv. **Random Forest Classifier module**: To categorize the traffic as either normal or attack traffic, Random Forest (RF) is used as the classifier to divide the traffic into normal and attack categories. For the classification process, there are

International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 01, ISSN No. 2455-2143, Pages 261-267
Published Online May 2023 in IJEAST (http://www.ijeast.com)

numerous decision trees used. A decision made wrongly by one decision tree can be corrected by another. Each decision tree provides a classification outcome, and the recommendation for categorization is based on the majority of votes. [1] Implementing Random Forest [17] involves the following steps:

i. Use N as the sample size for training data instances
ii. Assume that the incoming dataset's M characteristics.
iii. Assume that m is the number of parameters in the input that influence which attribute will be selected at each node of the tree in the future (m must be less than M).
iv. The training samples are collected, and a replacement tree is built for each sample.
v. Select m attributes from a tree node's attributes at random.

IV. EXPERIMENT

The experiment was conducted on Hp Laptop 4gram 1TB Intel Corei5 with a 64bit processor. A single network topology was used consisting of 5 host which is connected to one Open Virtual Switch connected to a single RYU Controller on Mininet. Hping3 was used to implement the attack, while Iperf and Ping was used to implement normal traffic. A near real time Dataset based on the flow table information was collected and used to compute the rate of Packet_in feature for both Normal attack and attack traffic. The traffic for both the Normal and attack consist of UDP, ICMP and TCP traffic. The generated traffic was collected and saved in a CSV file.
During the traffic generation, the number of Packet_in generated within a specific interval for the attack traffic kept growing in large numbers as shown in Figure 3.2. Whereas, during the normal traffic, the number of Packet_in was increasing but at a lower rate as shown Figure 3.1.

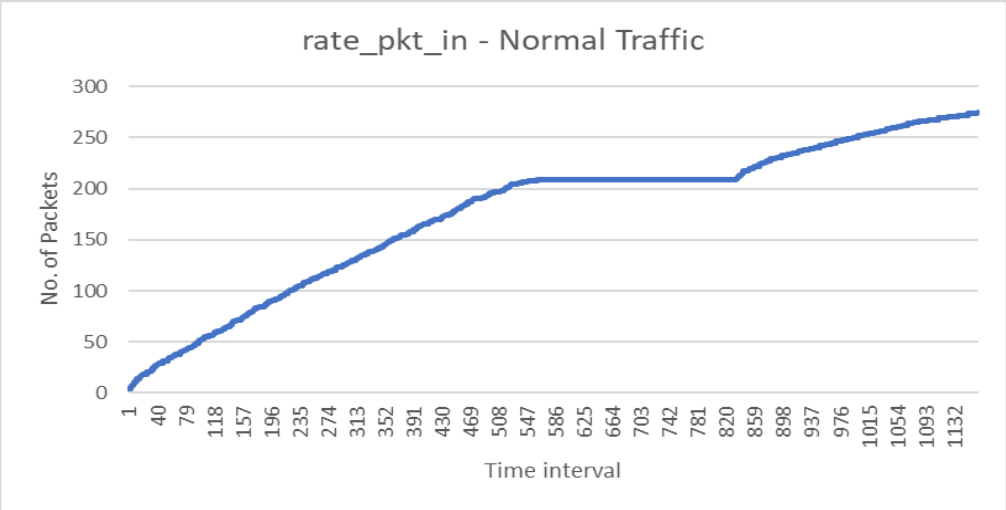

Figure 3.1: Rate of Packet_in during the Normal Traffic

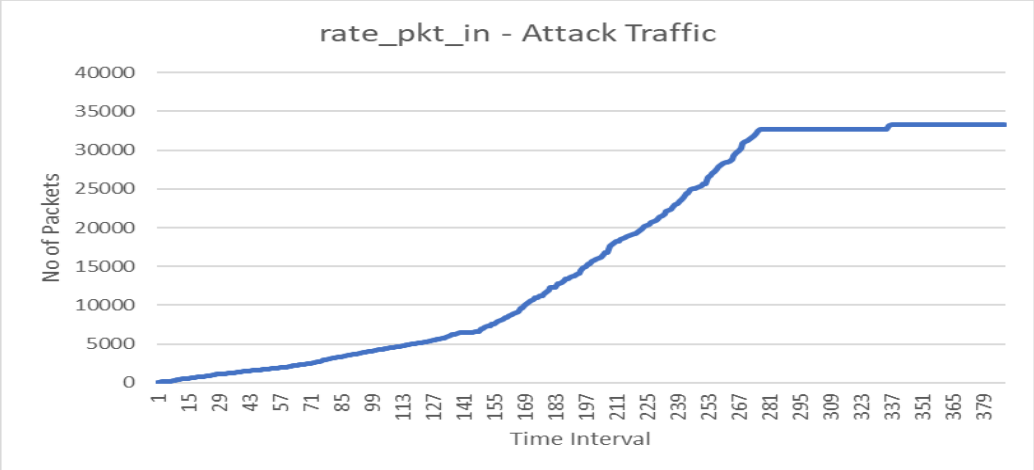

Figure 3.2: Rate of Packet_in during Attack traffic

International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 01, ISSN No. 2455-2143, Pages 261-267
Published Online May 2023 in IJEAST (http://www.ijeast.com)

V. EVALUATION

Accuracy: The accuracy rate is used as an evaluation index to evaluate the detection performance of the model [20]

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

FN: False Negative; FP: False Positive; TN: True Negative; TP: True Positive

The Random Forest (RF) algorithm was applied on the Datasets with different 70% training and 30% test set. The Accuracy rate was 99.78% with a with Area under curve of (AUC) of 0.995. The ROC Curve is shown in Figure 4.1.

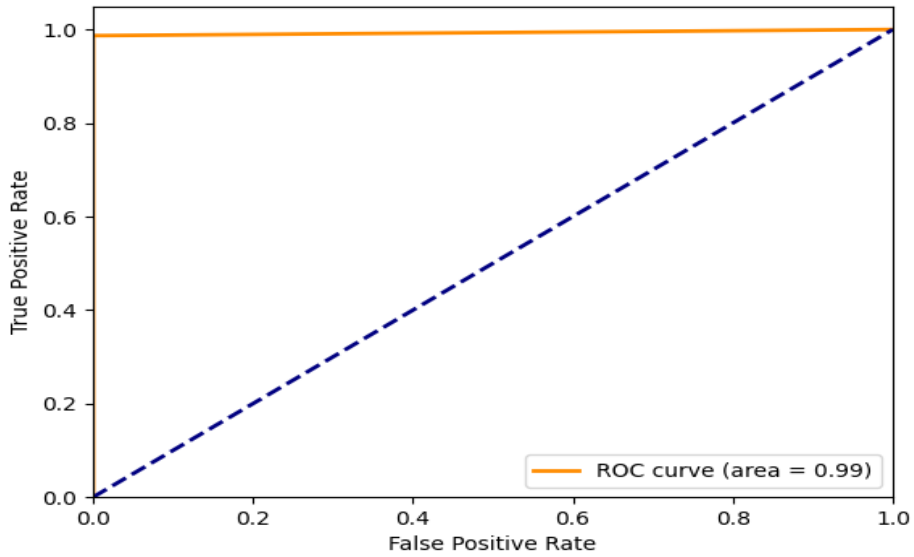

Figure :4.1: The Area under Curve for the RF classifier

TABLE 1: COMPARISON OF ACCURACY WITH OTHER STUDIES

SN	Author	Number of Features	Accuracy
1	[23]	6 tuple features	95.24%
2	[24]	5tuple features	99.7%
3	This work	Single Feature	99.8%

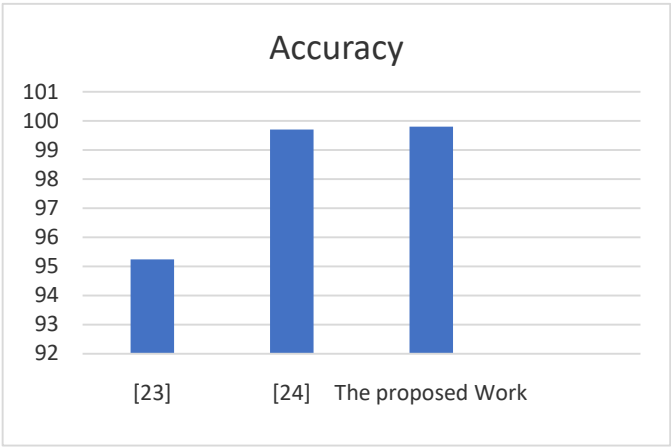

Figure 4.2: Comparison of Detection Accuracy with work of other Authors

In general Machine Learning, selecting the required feature for classification is always a great challenge. However, this study was able to utilize a single feature to design a detection approach for DDOS attacks based on RF classifier. The Accuracy rate shows that, most of the traffic were correctly classified which depicts the effectiveness of the feature used. The proposed work was compared with other existing research work in terms of the Accuracy. The result shows our proposed work has a slightly higher accuracy rate than the work of [24]. This connotes that, the Rate of Packet_in feature has potential to aid and improve detection accuracy.

VI. CONCLUSION

This work has been able to show how Packet_in affects the SDN Controller and have successfully used the rate of Packet_in as a single feature to detect DDOS attack against SDN Controller.

Random forest was used for the detection in a near real time. The result shows an accuracy of 99.8%. This means that, the rate of Packet_in single feature has great potential in detecting the DDOS attacks in Software defined Network, even though the single feature may not provide a good representation of traffic on the Network. Hence, for future work, the Packet_in feature could be used alongside other traffic features for DDOS attacks detection to maximize performance.

REFERENCES

[1] Ahuja, N., Singal, G., Mukhopadhyay, D., & Kumar, N. (2021). Journal of Network and Computer Applications Automated DDOS attack detection in software defined networking. Journal of Network and Computer Applications, 187(November 2020), 103108. https://doi.org/10.1016/j.jnca.2021.103108

[2] Conti, M., & Gangwal, A. (2019). A Comprehensive and Effective Mechanism for DDoS Detection in SDN. October 2017. https://doi.org/10.1109/WiMOB.2017.8115796

[3] Dargahi, T., Caponi, A., Ambrosin, M., Bianchi, G., & Conti, M. (2017). A Survey on the Security of Stateful SDN Data Planes. IEEE Communications Surveys and Tutorials, 19(3), 1701–1725. https://doi.org/10.1109/COMST.2017.2689819

[4] Dayal, N. (2017). Analyzing behavior of DDoS attacks to identify DDoS detection features in SDN. 274–281.

[5] Dayal, N., Maity, P., Srivastava, S., & Khondoker, R. (2016). Research Trends in Security and DDoS in SDN. Security and Communication Networks, 9(18), 6386–6411. https://doi.org/10.1002/sec.1759

[6] Dong, P., Du, X., Zhang, H., & Xu, T. (2016). A detection method for a novel DDoS attack against SDN controllers by vast new low-traffic flows. 2016 IEEE International Conference on Communications, ICC 2016. https://doi.org/10.1109/ICC.2016.7510992

[7] Fouladi, R. F., Ermiş, O., & Anarim, E. (2020). Journal of Information Security and Applications A DDoS attack detection and defense scheme using time-series analysis for SDN. Journal of Information Security and Applications, 54(August), 102587. https://doi.org/10.1016/j.jisa.2020.102587

[8] Gao, D., Liu, Z., Liu, Y., Heng, C., Ting, F., & Chao, Z. H. (2018). Defending against Packet-In messages flooding attack under SDN context. Soft Computing. https://doi.org/10.1007/s00500-018-3407-3

[9] Hafizah, S., Ariffin, S., Muazzah, N., Latiff, A., Khairi, M. H. H., Ariffin, S. H. S., Latiff, N. M. A., Abdullah, A. S., & Hassan, M. K. (2018). A Review of Anomaly Detection Techniques and Distributed Denial of Service (DDoS) on Software Defined Network (SDN). Technology & Applied Science Research, 8(2), 2724–2730. https://www.researchgate.net/publication/324830666

[10] Mladenov, B. (2019). Studying the DDoS Attack Effect over SDN Controller Southbound Channel. 2019 X National Conference with International Participation (ELECTRONICA), 1–4.

[11] Mousavi, S. M., & St-Hilaire, M. (2015). Early detection of DDoS attacks against SDN controllers. 2015 International Conference on Computing, Networking and Communications, ICNC 2015, 77–81. https://doi.org/10.1109/ICCNC.2015.7069319

[12] Open Networking Foundation. (2012). Software-Defined Networking: The New Norm for Networks [white paper]. ONF White Paper, 1–12.

[13] Polat, H., & Polat, O. (2020). Detecting DDoS Attacks in Software-Defined Networks Through Feature Selection Methods and Machine Learning Models.

[14] Prajapati, A., Sakadasariya, A., & Patel, J. (2018). Software defined network: Future of networking. Proceedings of the 2nd International Conference on Inventive Systems and Control, ICISC 2018, Icisc, 1351–1354. https://doi.org/10.1109/ICISC.2018.8399028

[15] Raghunath, K., & Krishnan, P. (2018). Towards A Secure SDN Architecture. 2018 9th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2018, 1–7. https://doi.org/10.1109/ICCCNT.2018.8494043

[16] Santos, R., & Moreno, E. (2019). Machine learning algorithms to detect DDoS attacks in SDN. April, 1–14. https://doi.org/10.1002/cpe.5402

[17] Shaik, A. B., & Srinivasan, S. (2019). A brief survey on random forest ensembles in classification model. In Lecture Notes in Networks and Systems (Vol. 56). Springer Singapore. https://doi.org/10.1007/978-981-13-2354-6_27

[18] Singh, J., & Behal, S. (2020). Detection and mitigation of DDoS attacks in SDN : A comprehensive review , research challenges and future directions. Computer Science Review, 37, 100279.

https://doi.org/10.1016/j.cosrev.2020.100279

[19] Singh, S., & Kumar, R. (2016). A Survey on Software Defined Networking : Architecture for Next Generation Network. Journal of Network and Systems Management. https://doi.org/10.1007/s10922-016-9393-9

[20] Sun, W. (2019). An Improved Method of DDoS Attack Detection for Controller of SDN. 249–253.

[21] Ubale, T., & Jain, A. K. (2019). Taxonomy of DDoS Attacks in Software-De fi ned Networking Environment. Springer Singapore. https://doi.org/10.1007/978-981-13-3804-5

[22] Wu, X., Liu, M., Dou, W., & Yu, S. (2016). DDoS attacks on data plane of software-defined network : are they possible ? https://doi.org/10.1002/sec

[23] Ye, J., Cheng, X., Zhu, J., Feng, L., & Song, L. (2018). A DDoS Attack Detection Method Based on SVM in Software Defined Network. Security and Communication Networks, 2018. https://doi.org/10.1155/2018/9804061

[24] Yu, S., Zhang, J., Liu, J., Zhang, X., Li, Y., & Xu, T. (2021). A cooperative DDoS attack detection scheme based on entropy and ensemble learning in SDN. EURASIP Journal on Wireless Communications and Networking. https://doi.org/10.1186/s13638-021-01957-9