Journal of
Cyber Security in Computer System

e-ISSN: 2583-4703
Volume-2, Issue-1 (January-April, 2023)

www.matjournals.com

Impact Analysis and Features for DDOS Attacks Detection in SDN

Abdullahi Aishatu Wabi[1*], Ismaila Idris[2], Olayemi Mikail Olaniyi[3], Joseph A. Ojeniyi[4]
[1]Research Scholar, Department of Cyber Security Science, School of Information and Communication Technology, Federal University of Technology, Minna, Niger State, Nigeria
[2,4]Professor, Department of Cyber Security Science, School of Information and Communication Technology, Federal University of Technology, Minna, Niger State, Nigeria
[3]Professor, Department of Computer Engineering, School of Electrical Engineering, Federal University of Technology, Minna, Niger State, Nigeria

[*]Corresponding Author: aishatuwabi@gmail.com

Received Date: February 27,2023 *Published Date: March 14,2023*

ABSTRACT

The goal of Distributed Denial of Service (DDOS) attacks is to make the network resources of its victim inaccessible and ultimately unusable. DDOS attacks in Software Defined Networks (SDN) are growing in strength and sophistication trying to exploit the programmability and centralized control features in the SDN Architecture among its other advantages. SDN can be viewed as a single point of failure by the attackers to carry out their malicious activities due to the centralized control nature of the SDN which was achieved by separating the Control plane from the Data plane. However, it also facilitates network monitoring, management and also anomalies detection which could be used by data centers, IT Facilities and businesses to deploy a robust, cost-effective and more secure network. Hence, SDN has gained wide attention from researchers, academia and businesses.

In this study, behavioral analysis of DDOS attacks in SDN is conducted to show how the various forms of DDOs attacks affect the SDN layers as well as ascertain features that could aid in its detection. This was achieved by launching various forms of traditional DDOS attacks on the SDN environment, monitoring and collecting the network flow and port statistics. The collected statistics were used to analyze trends of the attacks in SDN, and some of the network parameters that were evident during attacks were shown. A similar pattern of behavior was also seen among all the DDOS attacks and hence has similar features that could be used to design various techniques of detection for SDN.

Keywords- Distributed denial of service (DDOS) attacks, Network, Software-defined network (SDN), Sflow monitoring, Parameters

INTRODUCTION

IT infrastructure growth has brought about the challenges of ensuring the Integrity, confidentiality, Authentication, and Availability of information [1]. The emergence of SDN has brought about a reduction in the network complexity of the traditional network as it is loosely coupled. the Control plane is separated from the Data plane thereby ensuring adequate network management, programmability, and centralized control of the entire Network. Despite all these good features, it is faced with several security challenges. Among the prevailing SDN security issues is the Distributed Denial of Service of Attacks (DDOS). These attacks attempt to make the resource of a system or network unusable or inaccessible and are carried out due to certain reasons such as financial gains, political gains and disruption of Service [2]. The architecture of SDN is vulnerable to DDOS attacks due to its attribute of programmability together with its logically centralized control features. This is because the unavailability of an SDN controller can break up the service of an entire network [3]. The SDN architecture consists of three layers, the Application layer, the Data layer and the Control layer which could likely be affected by DDOS attacks.

This work aims to carry out a behavioral analysis of DDOS attacks by launching different forms of DDOS attacks in SDN such as TCP SYN flooding attacks, ICMP flooding, Smurf attack, HTTP flooding, and UDP flooding

attacks. This is to ascertain features that are peculiar to different forms of DDOS attacks, how they behave and how they affect the different layers of SDN

This work extends the work of [4] which considered only flow-based statistics by including port statistics in the analysis

RELATED WORK

N.Dayal & S. Srivastava (2017) [4], carried out a similar analysis of DDOS attacks in SDN, however, attention was paid to flow-based statistic features. X Wu, et al (2016) [5] discussed the possibility of a DDOS attack in the Data plane by thoroughly analyzing the flow table size and miss rate, it was discovered that attackers can inflict significant performance degradation over the system with a limited volume of attack resources. The focus was on the Data plane only. B Mladenov (2019) [6] studied the effect of Distributed Denial of Service attacks over the data-to-controller management southbound channel

Attack Strategies Across SDN Layers

DDOS Attack on Data Plane: In the general operation of the Switch, a part or whole of a Switch packet headers are sent to the Controller when a matching flow doesn't exist in the flow table while the packet is stored in the nodes of the Switch awaiting the response from the controller [7]. Hence, the attack was launched by sending new unknown packets before the *idle timeout* bearing in mind the limited storage of the switch and consequently leading to Switch Buffer Saturation, and Flow table overflow ultimately exhausting the data plan infrastructure resources.

DDOS Attack on Control Plane: The centralized control features of SDN lies at the control plane, the study leverage on the controller as a single point of failure [8] to launch DDOS attacks by sending different *Packet-In* messages to overwhelm the controller thereby Saturating the controller resources and making it unavailable to the legitimate user or even bring down the entire Network.

DDOS Attack on Control-Data Plane: The data plane communicates with Controller through a channel or Bandwidth called the South Bound Interface (SBI) which carries messages from time to time to and from the Data and control plane [9]. the increasing number of traffic was sent to overwhelm the Bandwidth between the two planes leading to Bandwidth Saturation and consequently bringing down the Controller and the entire network.

Attacks: Various forms of DDOS attacks launched include: SYN Flooding, UDP Flooding, ICMP Flooding, HTTP Flooding attack, and Smurf attacks.

Attackers and Victims: The attacks emanate from Hosts, Switches while victims of the attacks are the Host, Switch, Controller, and Bandwidth.

IMPLEMENTATION

In implementing the attacks the study considers **t**he Various forms of DDOS attacks, this includes SYN Flooding, UDP Flooding, ICMP Flooding, HTTP Flooding attack, and Smurf attacks. And the attacks emanate from Hosts, Switches while victims of the attacks are the Host, Switch, Controller, and Bandwidth. This is depicted in Fig. 1.

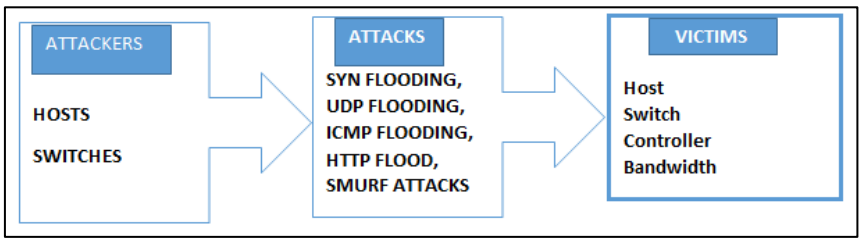

Figure 1: *Attack techniques.*

Tools: To conduct this analysis, an SDN Environment was set up using the Mininet Emulator. The Controller uses Ryu Controller, Sflow-rt is used for monitoring flows, Hping3, and APache Bench tool launching an attack.

Topology: A network topology consisting of 5 switches, 20hosts and one controller created using the following command: *Sudo mn --custom flow-rt/extras/sflow.py --switch ovsk --topo tree,depth=2, fanout=4 --*

controller=remote,127.0.0.1,port=6653. The attacks were launched from Host *h2* connected to switch *S with Data path ID and host h4 connected to Switch S with datapath ID.* All attacks were targeted to *host 8* connected to Switch *S with datapath ID and MAC address.* Sflow monitoring was enabled in the entire switch on the network as shown in Fig. 2.

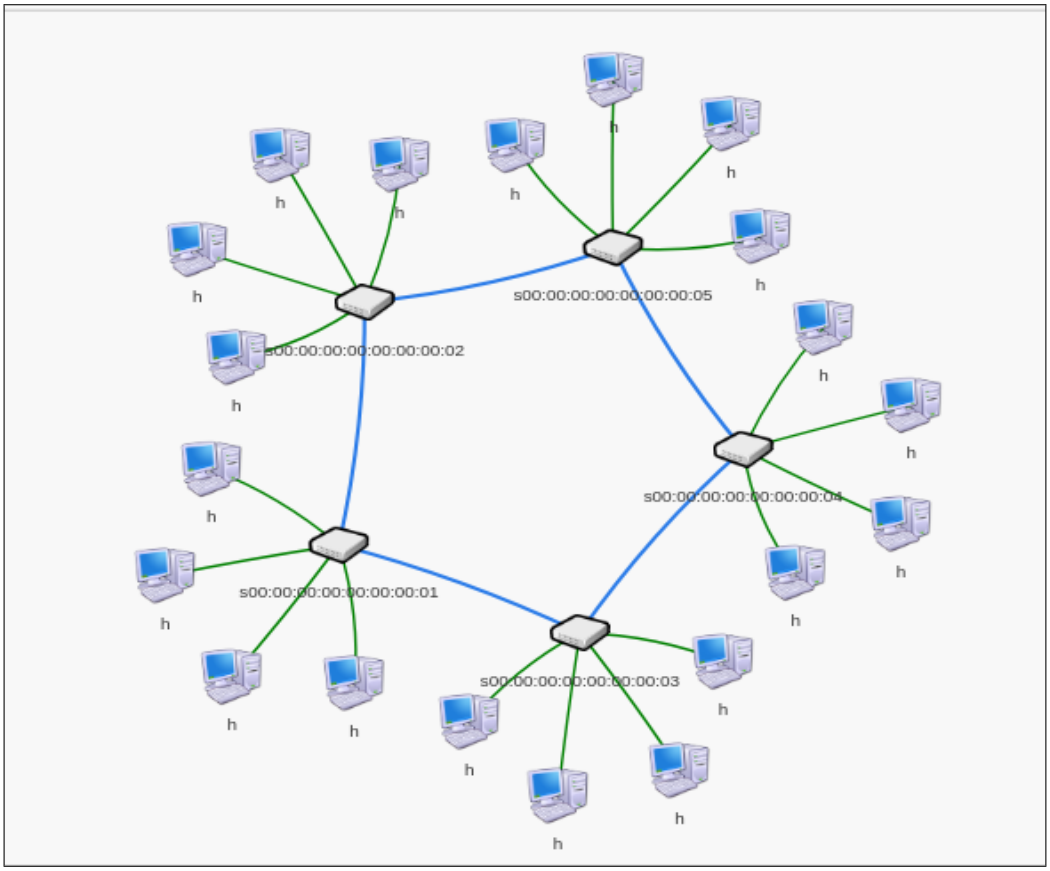

Figure 2: *The network topology*.

The attacks were made as follows:

SYN Flood Attack: In this type of attack. Three-way handshakes are exploited and the victim is flooded with SYN requests from random sources, hence the victim is left waiting for responses for an SYN/ACK packet which is never returned. So, when a large number of SYN packets are sent, the ports of the Victim system are blocked preventing legitimate users from accessing the network. To launch this attack, the Hping3 tool was used to generate Syn packets from two hosts h2 and h4. However, the Flows were generated for the victim from random sources thereby hiding the real IP addresses of the attack hosts. The command used is as follows:

Hping3 --rand-source -i u20000 -d 64 -S -c100000 10.0.0.8

Hping3 --rand-source -I u35000 -d 64 -S -c10000 10.0.0.8

--rand source means to attack from random sources, *i* is the inter-, packet delay *d* is the size of the window, *and c* is the packet count.

After launching the attack, flows were generated and emanated from different sources' IP addresses. The flow continues to grow until the Switch port was blocked which prevented normal communication. During the attack, the *pingall* command was used to test the network connectivity. A packet drop of almost 100% was noticed during the attack, and when the attack was stopped, the connectivity was not restored. Consequently, it can be deduced that, though the traffic was low, the effect on Controller was high. The transmitted bits per second for the flow and switch ports are shown in Fig. 3. The Fig. number shows, random source addresses targeting a single destination address, with a specific protocol and random source port. All the Switch ports are also flooded with flows.

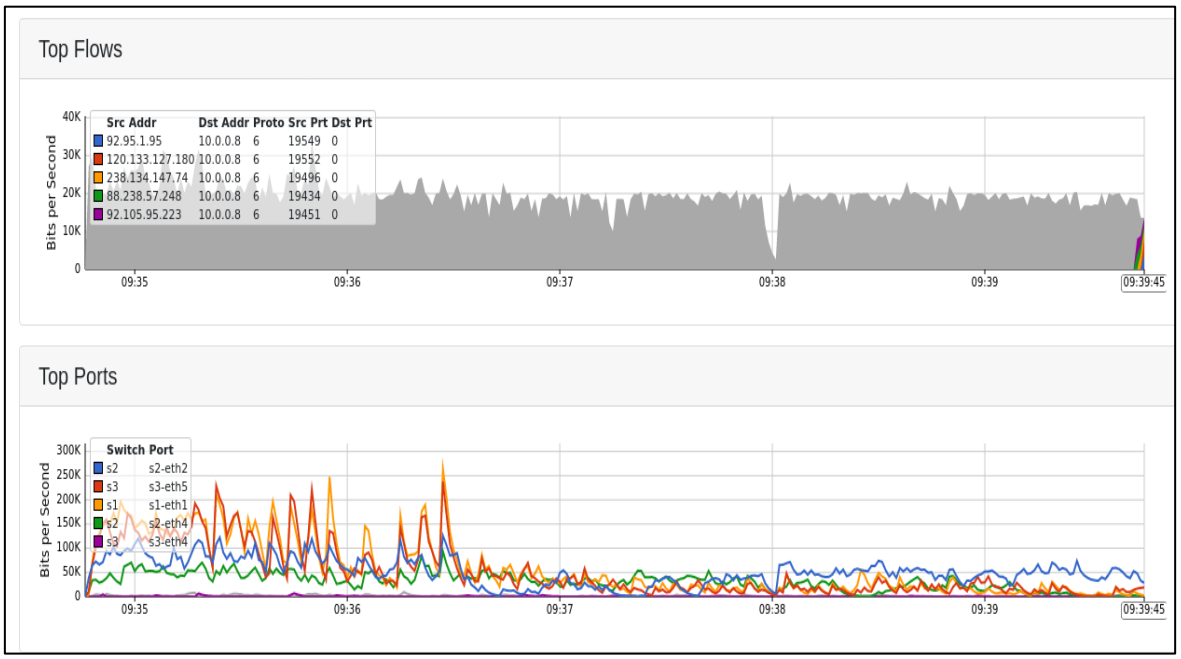

Figure 3: *Transmitted bits per second during SYN flooding attack for both port and flow.*

***HTTP Flooding*:** This entails sending HTTP requests to a network's server which is never completed thereby keeping fake connections with the server for a long period open. Consequently, the server will be unable to provide services to other users either legitimate or new. Apache bench tool was used to achieve this attack. A custom python-based HTTP server was first started on the Victim host terminal: *python -m SimpleHTTPServer 80%.* This was followed by a large number of GET requests sent from four different hosts *h2, h6, h10, and h4* to Victim host *h8.* The command used is as follows:

ab -n 1000 -c 500 http://10.0.0.8/

Ab - Apache bench tool, *n* is the total number of requests and *c* is the number of requests per second. Each of these commands was executed from the four attacking hosts to the victim simultaneously at the same. The packet bytes per second were large and requests were received by the server simultaneously thereby sending the Controller with new requests. The server soon becomes overloaded as shown in Fig. 4.

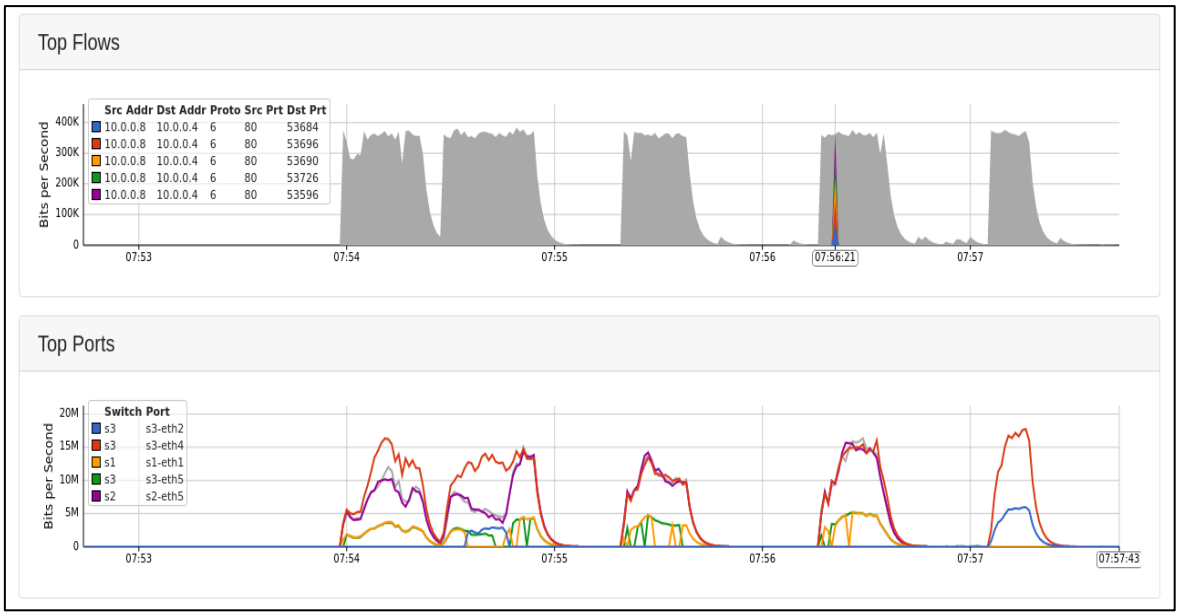

Figure 4: *Transmitted bits per second during http flooding attack for both switch port and flow.*

***ICMP*:** This attack scenario tries to overwhelm its victims with an ICMP echo request that affects the services used by other applications on the victim. This attack was implemented by sending a very fast echo request to the target victim using the following commands:

Hping3 -1 --rand-source -i u20000 -d 64 -c100000 10.0.0.8

Hping3 -1 --rand-source -i u35000 -d 64 -c10000 10.0.0.8

The victim host was flooded with a huge amount of echo requests for random sources. It was observed that as the random sources were used by the attacking host to attack the victim host, new Packet-In were generated each time and consequently directing huge traffic to the Controller.

During the attack, some percentage of packet drop was noticed. however, communication was still enabled. Therefore, when the attack was terminated, the Controller was able to recover and immediately resumed normal communication with the switches. A huge number of packet drops were recorded in the data path during the ICMP flood attack. However, the number of packet drops was reduced when the attack was stopped. This type of attack is a volumetric attack and at a High rate. Though, the controller was able to resume its normal communication with network devices as shown in Fig. 5.

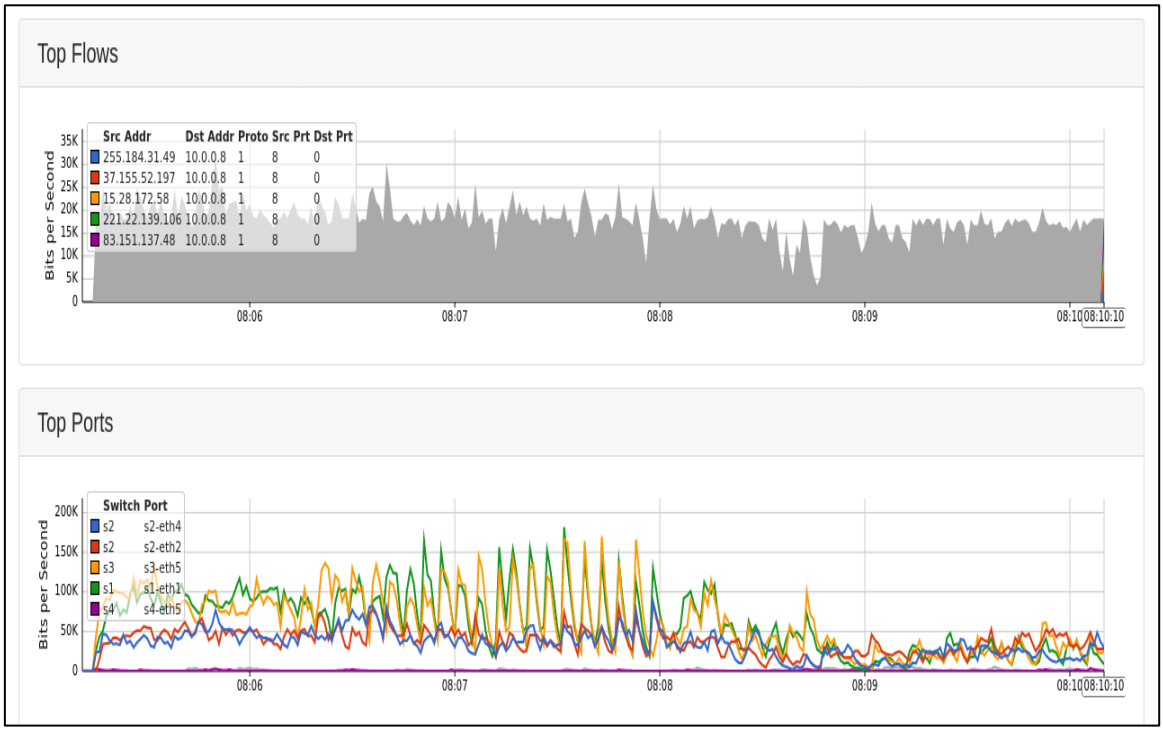

Figure 5: *Transmitted bit per second for flow and port during ICMP flooding.*

***UDP Flooding*:** UDP flood attack tries to send traffic containing UDP datagrams to the victim. A huge number of datagrams arriving at the victim results in buffer overflow. To implement the attack, Hping3 was used as follows:

Hping3 --rand-source -i u20000 -d 64 -c100000 10.0.0.8

Hping3 --rand-source -i u35000 -d 64 -c10000 10.0.0.8

During the attack scenario, a large volume of huge number UDP datagrams that looked like it is coming from different hosts was sent to the victim host. The large volume of traffic with huge Bytes flows was observed to be generated. After a particular period, the controller started dropping some packets. But, the Connection between the switches and the controller was not disturbed completely. After stopping the attack, normal communication between the controller and switches was resumed. Such attacks are not harmful to the Controller. However, the Datapath was flooded. Even though there was packet loss during and after the UDP flood, the network was able to recover quickly as shown in Fig. 6.

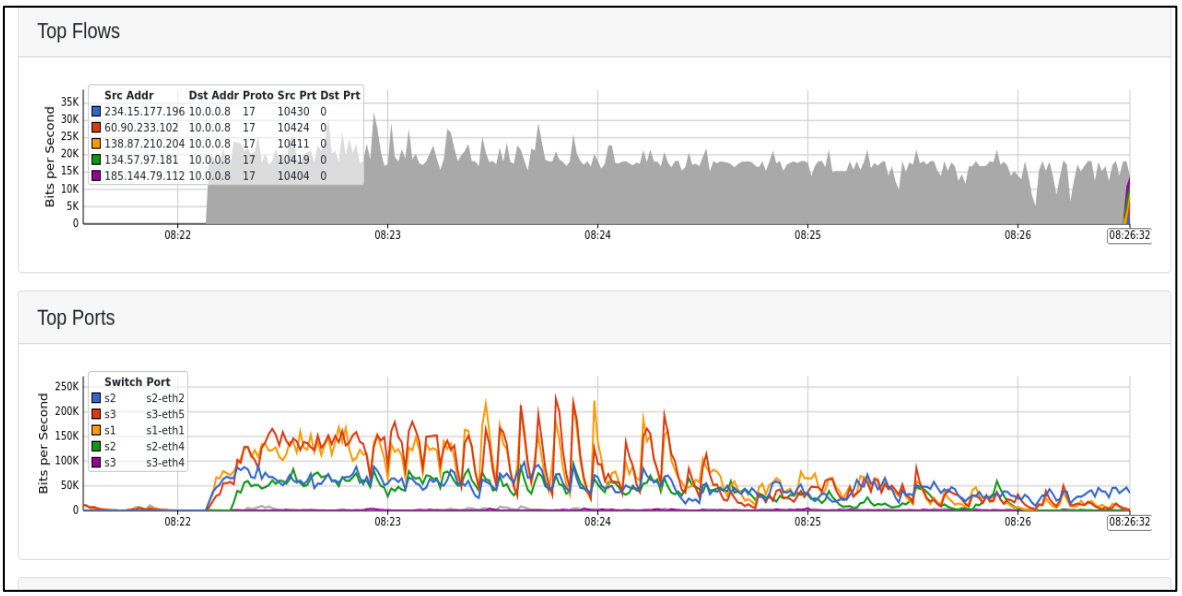

Figure 6: *Transmitted bits per second from the switch port and flows during UDP attack.*

Smurf Attacks: Smurf attack is another form of ICMP ping flood attack. Although in the case of the Smurf attack, the attacker sends ICMP echo requests to different machines with the victim's IP addresses as source IP, that results in the victim overflowing with a huge number of ICMP echo replies from random hosts. Hping3 was used to achieve this attack Using the following command:

Hping3 -1 --flood -a 10.0.0.8 10.255.255.255

The source IP address used for sending the ICMP echo request messages was changed to the victim's IP address together with the broadcast address. Hence, echo requests are broadcasted to a range of IP addresses on the network. In response to these ICMP echo requests, a huge amount of ICMP echo replies to the victim arrived, resulting in victim buffer overflow. A huge volume of bits is being transmitted, in which most of the traffic is incoming. Although the controller received certain error messages the connection was not completely lost, hence the impact on overall performance was also low. The controller's buffer was overwhelmed but the connection between the controller and switches was never lost completely. As soon as the attack was stopped, the controller's normal communication was resumed as shown in Fig. 7.

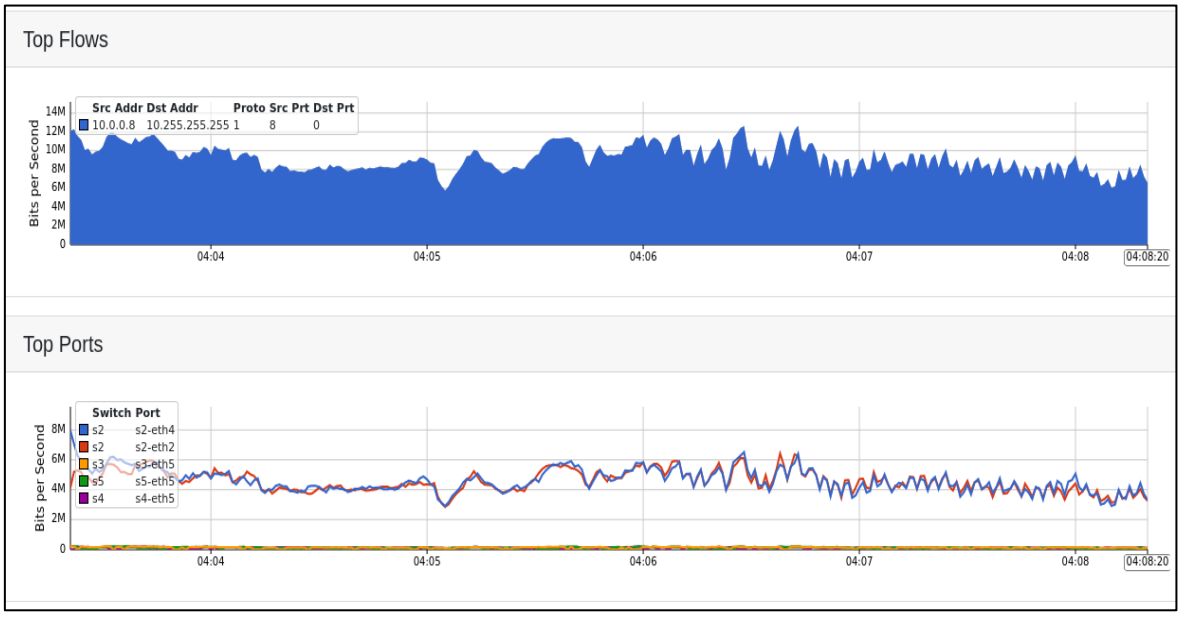

Figure 7: *Transmitted bit per second at the switch port and flows during smurf attacks.*

Table 1: *Summary of the impact of DDOS attacks in SDN.*

Attacks	Attack type	Features Monitored	Effect on Controller	Effect on Data Plane	Bandwidth
High rate UDP flooding	Volumetric attack	IP addresses, Source port, destination ip address, flow/sec, the randomness of Destination IP, Bits per second, flowpacket count, Flow byte count, Flows with the same destination, rate of Packet_in Protocol, flow Duration, Data rate (rx_byte, tx_byte,) rx_packet, tx_packet rate transmitted byte, rate of the transmitted packet	Controller buffer saturation, Buffer flow	Buffer Overflows	Bandwidth Exhaustion
Smurf Attacks					
ICMP Flooding					
Low Rate HTTP Flooding	Protocol exploitation attack		Controller unavailable	Switch buffer saturation,	Bandwidth Exhaustion
TCP SYN flood					

SUMMARY OF IMPACTS OF DDOS ATTACKS ON SDN

After launching the attack, *Sflow-rt* were used to monitor network traffic statistic *and the OpenFlow table* was queried to collect flow and switch port statistic during the attack such as the number of bytes transmitted per second, Number of bits per second, Number of packets per second, the packet rate, Number of flows, packet duration, rx_packet, rx_byte, tx_byte, tx_packet. Since an attack comes from multiple IP addresses, the randomness of IP addresses and protocols was monitored. The Controller was also observed, to ascertain how long it took to recover after the termination of the attack. The CPU was also monitored to check consumption. The summary of the impact is shown in Table 1. Both high rate and low DDOS were launched. High-rate DDOS attacks are volumetric attacks such as UDP flooding attacks, ICMP attacks, and Smurf attacks launched to exhaust the Bandwidth of the target victim and flood it with a huge amount of traffic while **Low rate** DDOS attacks such as HTTPS and SYN Flooding attacks on the hand, use less traffic and consumes the destination port of the victim thereby making it unavailable. The following were observed during the attacks:

- The attacks transmitted a huge number of Bytes and packets.
- The destination data path was flooded with the huge number of traffic resulting in Bandwidth exhaustion and buffer saturation.
- The controller was made unavailable during the attacks. However, for the high-rate attack, the controller recovered quickly as soon as the attack was terminated. However, for a low rate, it took a long to recover.
- The randomness of the Source IP address was higher than the destination IP, and protocol.
- Some of the notable features during the attacks include: Random and growing IP

addresses, protocols, unchanging destination IP, rate of packet Drops, no. Bytes per sec, no of packets per second, no of flows, the randomness of Destination IP.

- Although a large number of flows were generated with plenty of packets, most of the packets were observed to be empty.
- Since IP spoofing was adopted, the rate of packet_in was high.
- A small number of traffics were transmitted in bits during the low-rate attacks
- The number of open connections was high.
- Some flows delay in the network for a long time resulting in TCAM memory consumption, buffer overflow and Bandwidth Exhaustion.

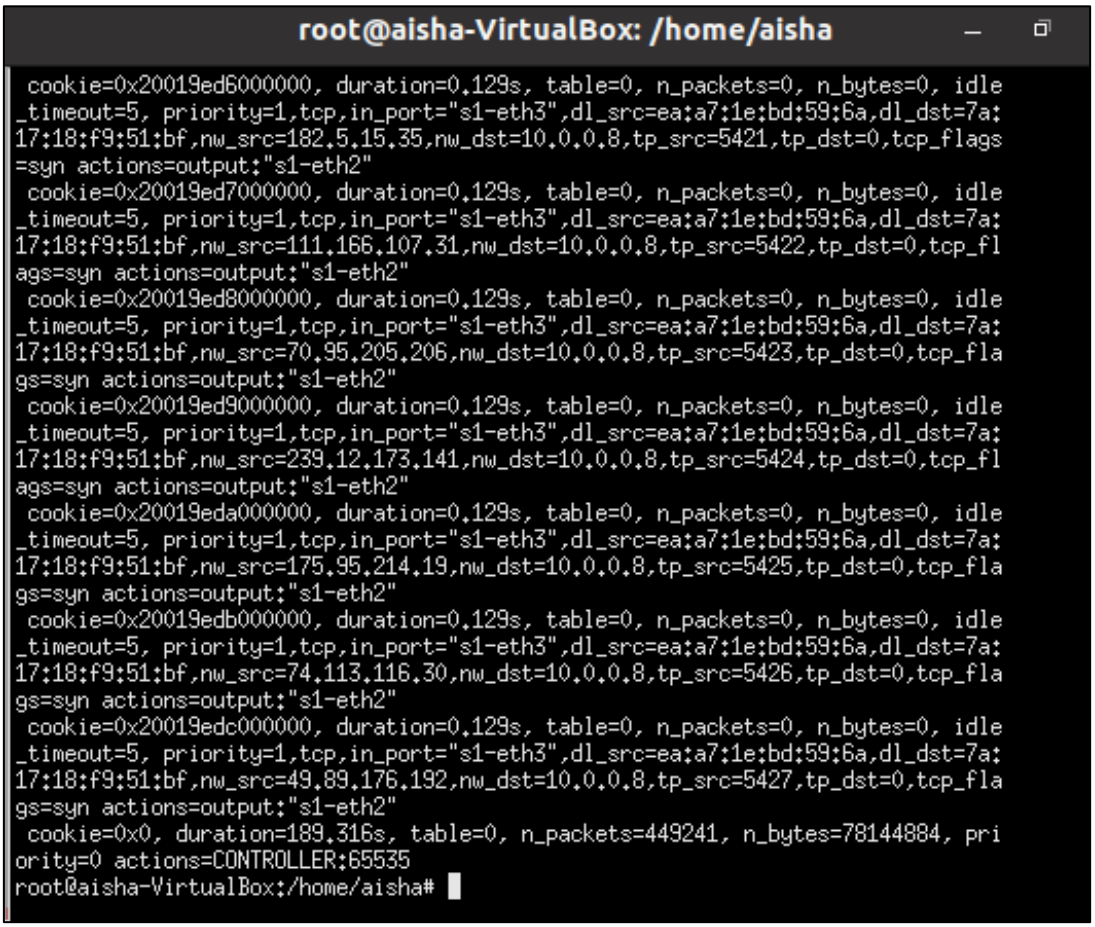

Figure 8: *Statistics from the switch flow table.*

PARAMETER TRENDS

From the experiment, it was observed that almost all the parameters have the same behavioural pattern during the DDOS attacks. A sample of flow statistics is shown in Fig. 8. Hence, we used the identified eigenvalues to monitor some set parameters. This includes the number of Growing Source IP, Speed of Flow entry, Packet count, Byte count, flow/sec Rate of Packet in, Bandwidth, the rate of transmitted bytes, and the rate of the incoming packet. Continuous increase in the number of flows per sec, average flows with the same destination and the rate of Packet as shown in Fig. 9-11. The Speed of flow entry, the growth of the Source IP address, and the growth of Source Port parameters exhibit similar behaviour as shown in Fig. 12, and Fig. 15-16. Since the attacks emanate from random sources, the number of flows, source IP addresses and source ports continue to grow.

At the switch port, the bandwidth parameter continues to increase which shows the level of consumption of the port bandwidth as depicted in Fig. 17. Similar trend is also shown in Fig. 18-20 for the received byte, rate of transmitted packet, and rate of the transmitted

byte. The continuous increase of the parameters depicts the consumption of the network resources. The packet count and byte count parameter as shown in Fig. 13 -14 is very low. This is because the attacks were launched to flood the network from random sources. hence, the packets and packet bytes are relatively small and sometimes empty.

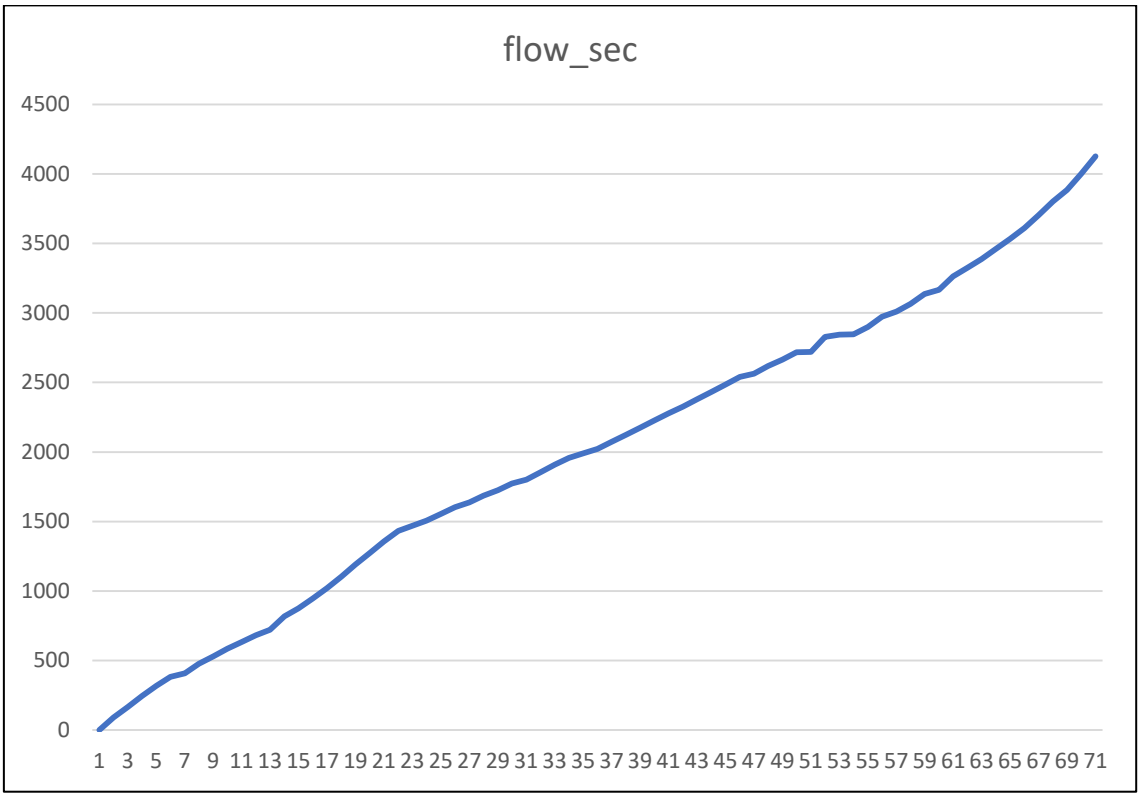

Figure 9: *The number of flows per sec.*

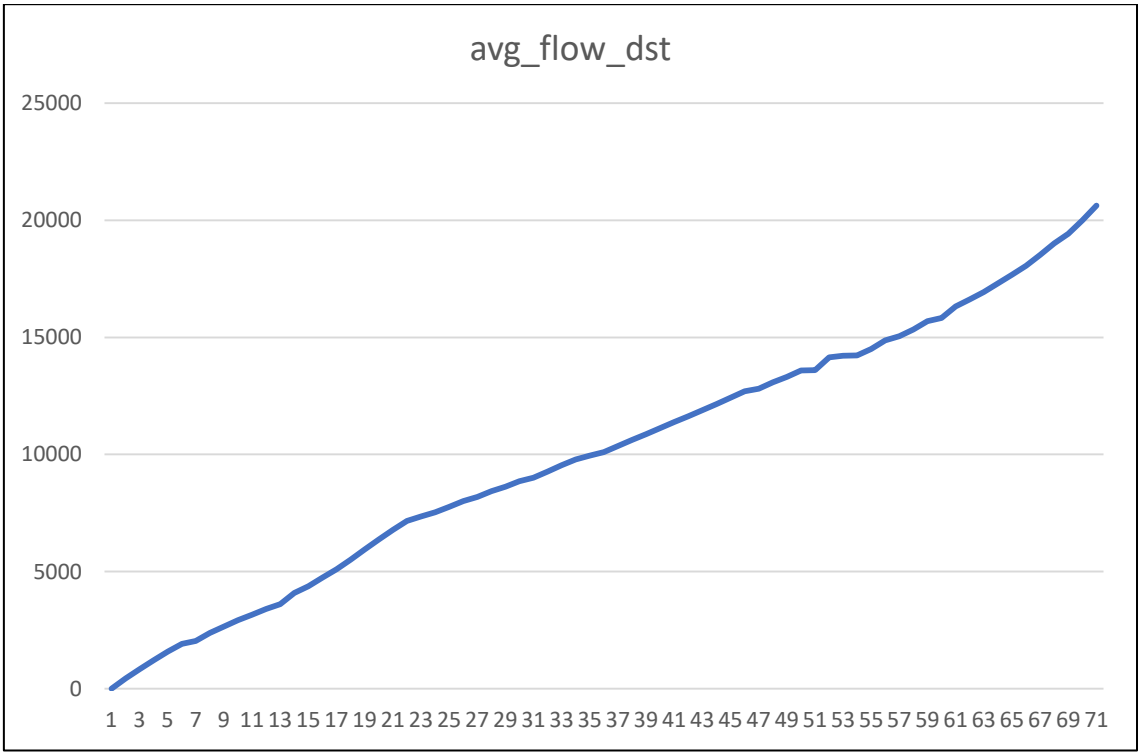

Figure 10: *Average flow with the same destination.*

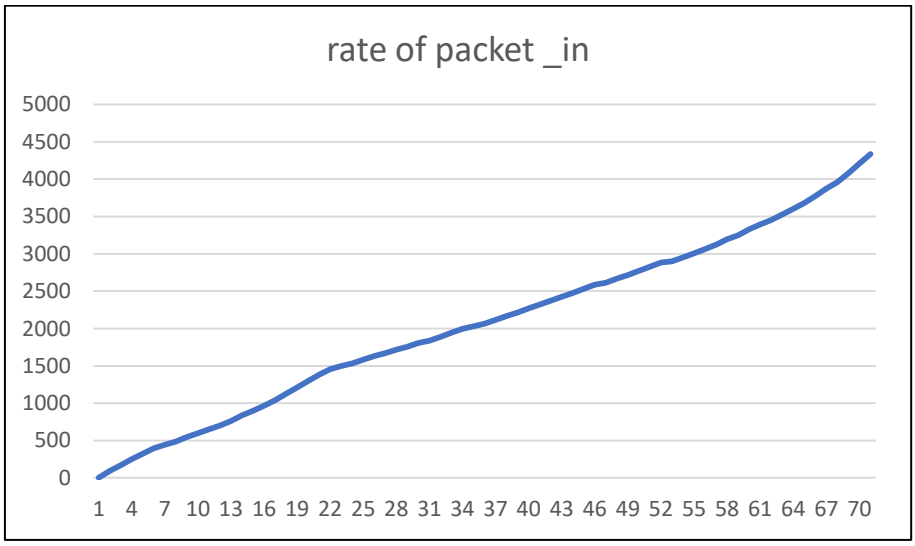

Figure 11: *The rate of packet _ in.*

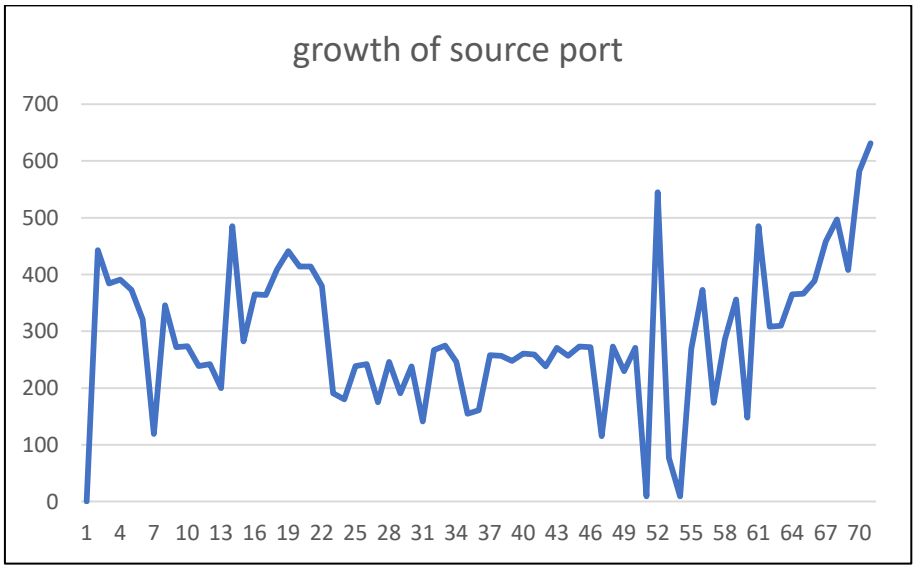

Figure 12: *The growth of the source port.*

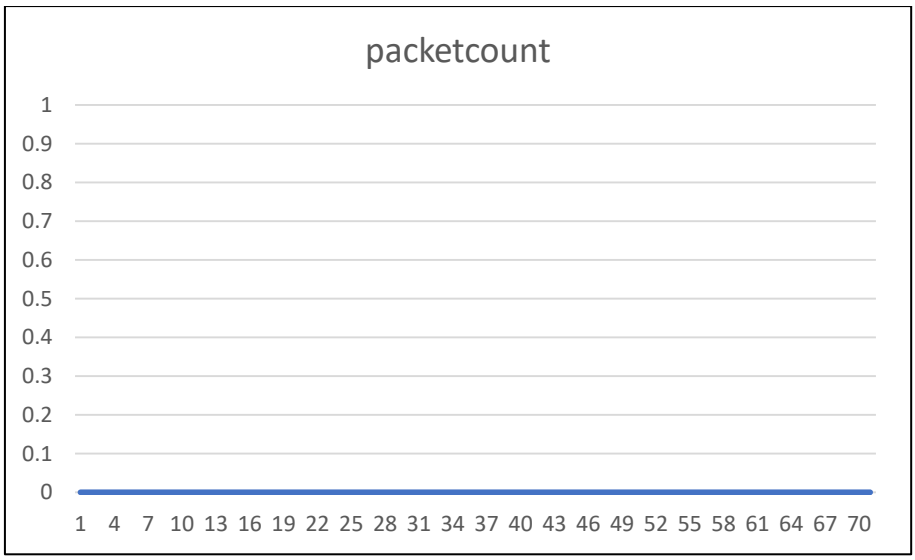

Figure 13: *Packet count.*

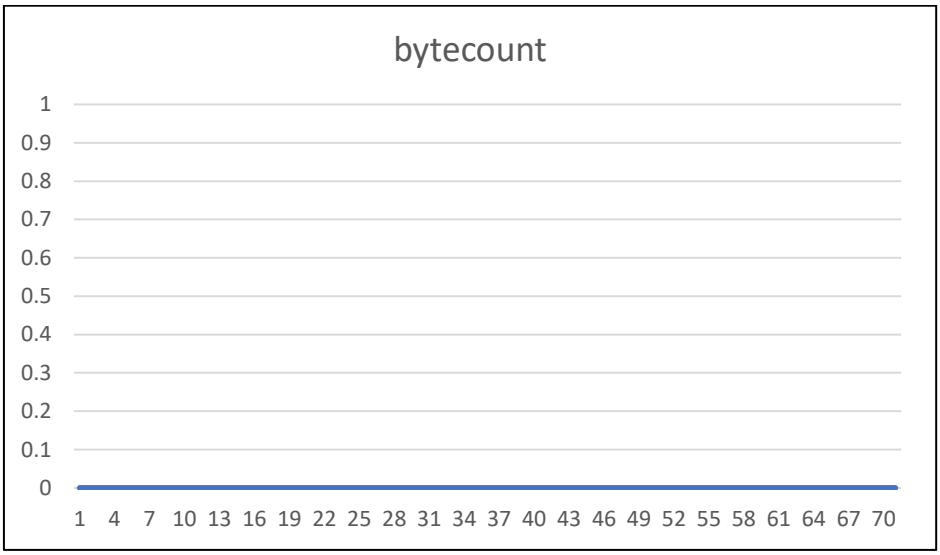

Figure 14: *Byte count within sampling interval.*

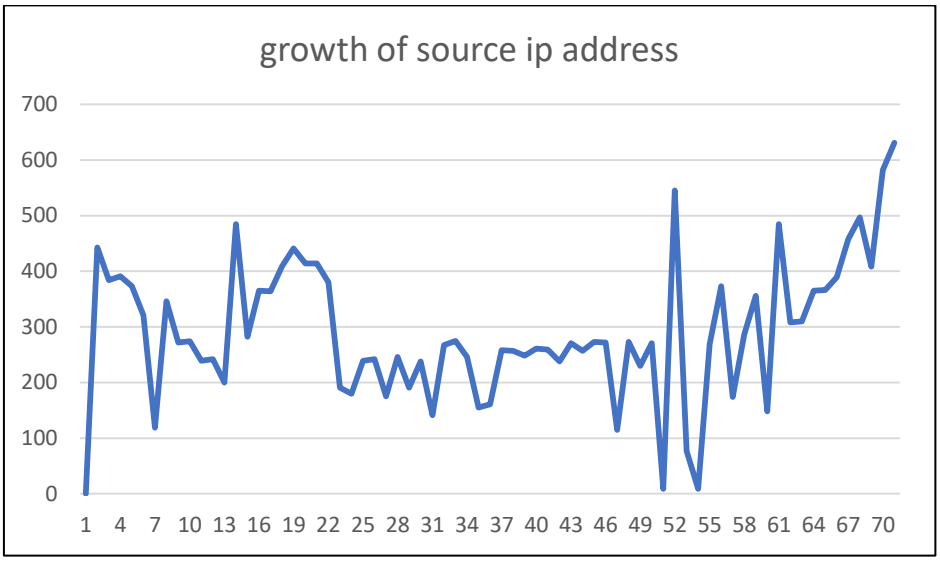

Figure 15: *The growth of the source IP address within the sampling interval.*

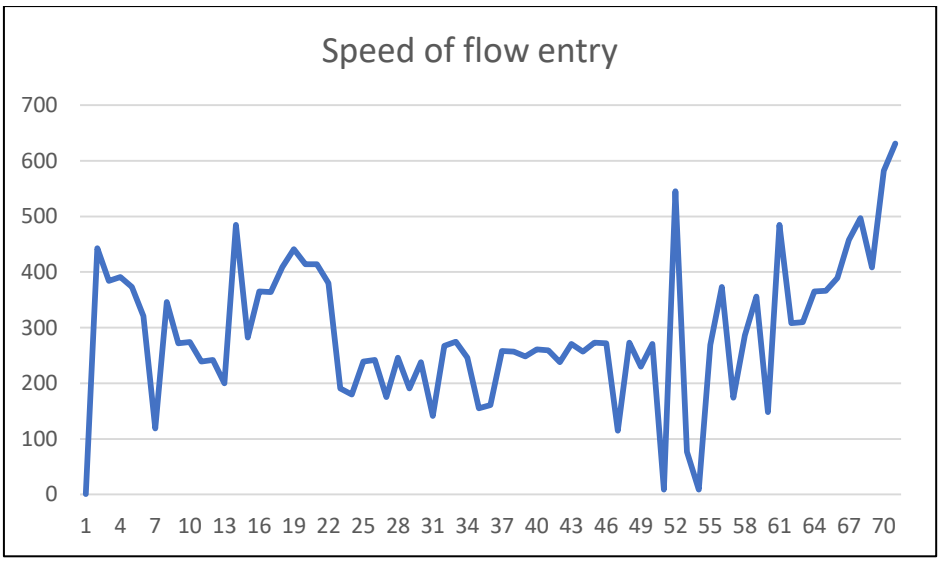

Figure 16: *The speed of flow entry within the sampling interval.*

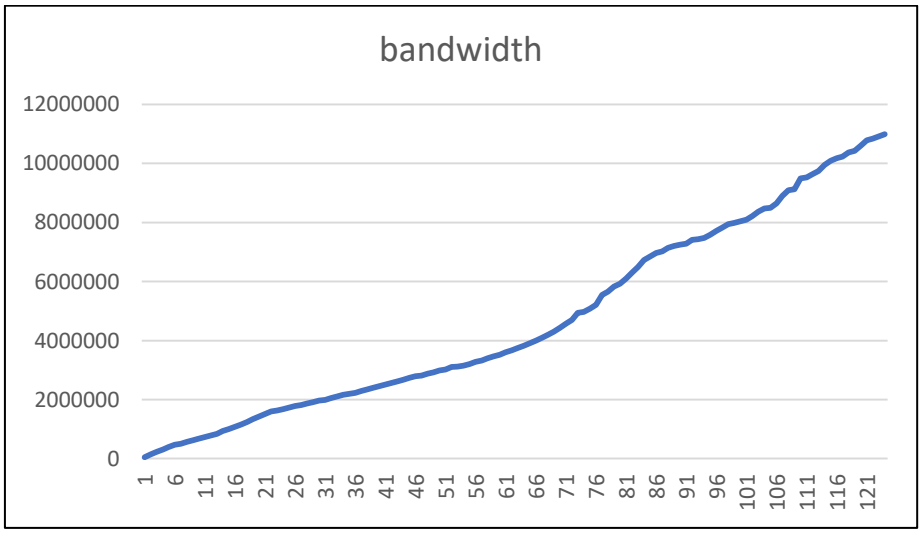

Figure 17: *The bandwidth occupied within the sampling interval.*

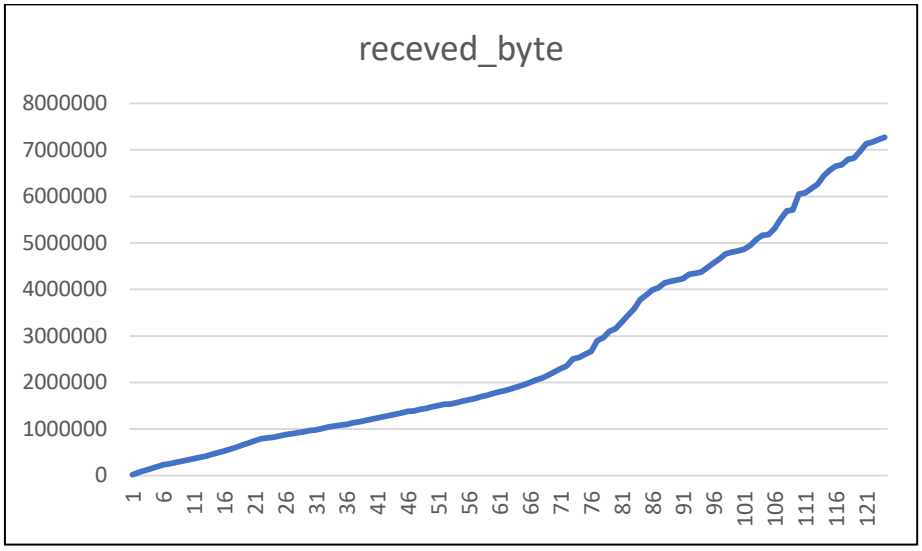

Figure 18: *The received byte within a sampling interval.*

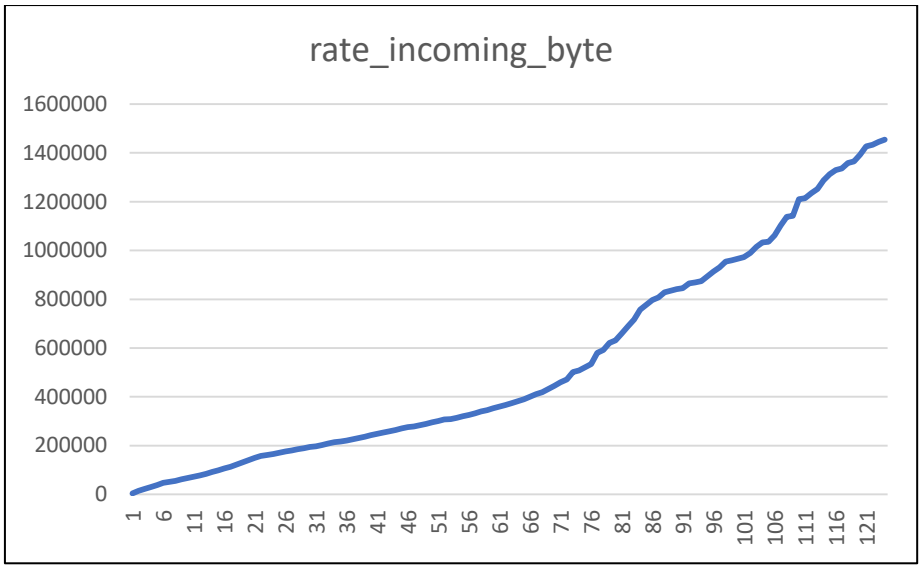

Figure 19: *The rate of the incoming byte.*

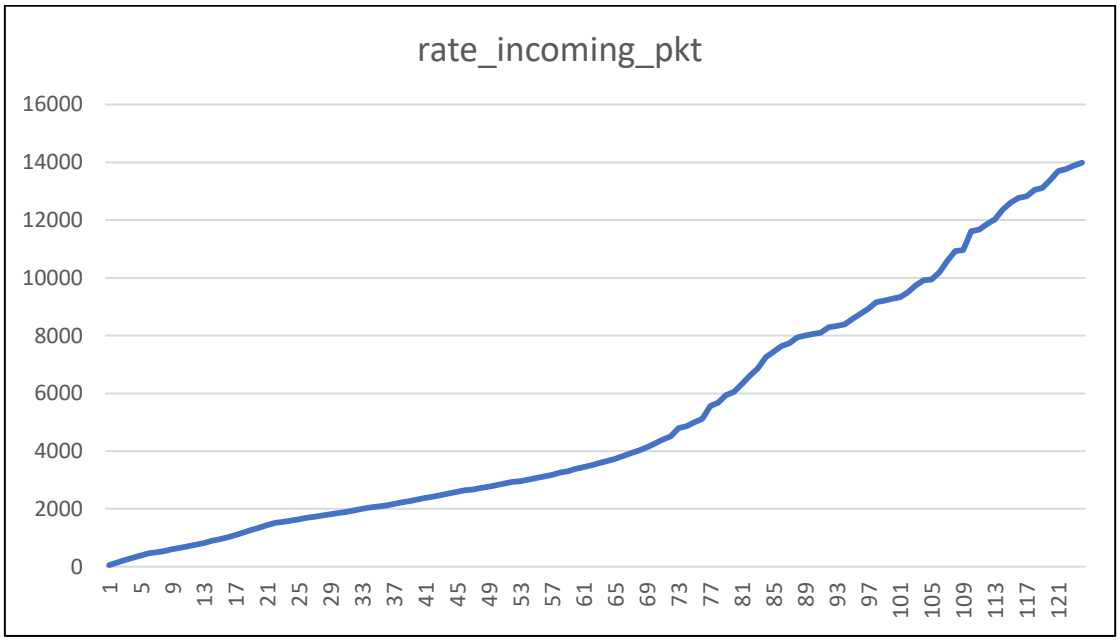

Figure 20: *The rate of the incoming packet.*

CONCLUSION

The behavioural analysis of DDOS Attacks has been carried out in SDN planes specifically the Data and Control plane and Bandwidth. It was discovered that High-rate attacks consume the bandwidth of the Control Data plane with little effect on the Controller as the service returned to normal when the attack was stopped. Low-rate attacks happen to be more dangerous as it gradually consumes network resources and eventually bring down the controller which is the core of the Network.

This study has been able to establish the possibility of a DDOS attack in the Data plane, Control Plane as well as Control - data plane Bandwidth. An SDN environment was set up using Mininet, and various forms of traditional DDOS attacks were launched namely UDP flooding, ICMP Flooding, SYN Flooding, Smurf Attack, and HTTP flooding attack. The Switch was queried to collect both flow and port-based statistics and the network connectivity was tested to check the response of the SDN Controller. Furthermore, the trends of some of the parameters during DDOS attacks were analyzed. From the analysis, it was discovered that all the DDOS attacks launched behave in a similar pattern and hence have similar features that could be used for detection. Further work is required to ascertain features that a peculiar to all layers in SDN including the application plane.

REFERENCES

1. J Singh and S Behal (2020). Detection and mitigation of DDoS attacks in SDN: A comprehensive review, research challenges and future directions, *Computer Science Review*, 37, Available at: https://doi.org/10.1016/j.cosrev.2020.100279.
2. N Zakaria Bawany, JA. Shamsi and K Salah (2017). DDoS attack detection and mitigation using SDN: Methods, practices, and solutions, *Arabian Journal for Science and Engineering*, 42, 425-441, Available at: https://link.springer.com/article/10.1007/s13369-017-2414-5#citeas:~:text=DOI-,https%3A//doi.org/10.1007/s13369%2D017%2D2414%2D5,-Keywords.
3. K Benzekki, A El Fergougui and A E Elalaoui (2016). Software-defined networking (SDN): A survey, *Security and Communication Networks*, 9(18), 5803-5833, Available at: https://onlinelibrary.wiley.com/doi/epdf/10.1002/sec.1737.
4. N Dayal and S Srivastava (2017). Analyzing behavior of DDoS attacks to identify DDoS detection features in SDN. *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*. IEEE, Available at: https://doi.org/10.1109/COMSNETS.2017.7945387.

5. X Wu, M Liu, W Dou and S Yu (2016). DDoS attacks on data plane of software-defined network : are they possible ?, *Security and Communication Networks*, 9, 5444-5459, Available at: https://onlinelibrary.wiley.com/doi/epdf/10.1002/sec.1709.
6. B Mladenov (2019). Studying the DDoS attack effect over SDN controller southbound channel. *2019 X National Conference with International Participation (ELECTRONICA)*. IEEE, Available at: https://doi.org/10.1109/ELECTRONICA.2019.8825601.
7. M. H. H. Khairi, S. H. S. Ariffin, N. M. Abdul Latiff, et al (2018). A review of anomaly detection techniques and distributed denial of service (DDoS) on software defined network (SDN), *Engineering, Technology & Applied Science Research*, 8(2), 2724-2730, Available at: https://doi.org/10.48084/etasr.1840.
8. R Santos, D Souza, W Santo, et al (2019). Machine learning algorithms to detect DDoS attacks in SDN, *Concurrency and Computation: Practice and Experience*, 32(16), Available at: https://doi.org/10.1002/cpe.5402.
9. M Conti, C Lal, R Mohammadi and U Rawat (2019). Lightweight solutions to counter DDoS attacks in software defined networking, *Wireless Networks*, 25, 2751-2768, Available at: https://link.springer.com/article/10.1007/s11276-019-01991-y#citeas:~:text=DOI-,https%3A//doi.org/10.1007/s11276%2D019%2D01991%2Dy,-Keywords.

CITE THIS ARTICLE

Abdullahi Aishatu Wabi et al.(2023). Impact Analysis and Features for DDOS Attacks Detection in SDN, *Journal of Cyber Security in Computer System*, 2(1), 29-42.