SCIS 2022 2022 Symposium on
Cryptography and Information Security
Osaka, Japan & Online, Jan. 18 – 21, 2022
The Institute of Electronics,
Information and Communication Engineers

SDN-based Detection Method against DoS/DDoS attacks in an IoT Environment

Abdul Adhim * Satoshi Okada * Takuho Mitsunaga *

Abstract: The number of IoT devices has been increasing over the years, and it is expected to only keep increasing in the next few years. Keeping these devices secured is always important. One of the ways to keep them more secure is to introduce Software Defined Network (SDN) architecture into the IoT network. SDN enables software-based network management without any physical changes. This technique is well suited for the management and incident response of IoT networks, which continue to grow in size these days. Some of the potential security threats to IoT devices are Denial of Service (DoS), and its other variant Distributed Denial of Service (DDoS). IoT devices' nature to be limited in computation, storage, and network capacity, make them more vulnerable to be compromised. SDN is a promising technology that could help detect and mitigate DoS/DDoS attacks within the IoT network. In this paper, a solution using an entropy-based detection method to detect an incoming DoS/DDoS attack in an SDN-based IoT network is proposed. A statistical approach to distinguish normal traffic from anomalous traffic.

Keywords: IoT, SDN (Software Defined Network), DoS, DDoS, Entropy-based detection

1 Introduction

In recent years, we have seen a growing number of network-connected devices all over the world. Different kinds of devices are network-connected. Thus the size of network traffic is increasing too. This increasing number of devices causes network traffic, which is expected that there will be 29.3 billion networked devices by 2023 [2].

Amongst these connected devices, a type of device called Internet of Things (IoT) is also expected to increase. IoT has and still plays an important role in realizing the future's advancement in technology. Especially in realizing the up and coming advancement of society, or also known as Society 5.0 [13]. Just in 2021, the number of IoT devices alone is recorded to be 31 billion [1]. The drastic rise of these IoT devices could cause several noteworthy security issues such as incorrect access control, overly large attack surface and lack of encryption [3]. This especially is threatening to the industrial IoT and potentially damaging an industrial scale work.

Considering these threats, to fully realize the potential of IoT in the future it is necessary to take into account the security aspect of it. Both its threats and countermeasures. The detection and mitigation of security threats is an important measure to be taken. Especially for threats like Denial of Service (DoS) and Distributed Denial of Service (DDoS) given how detrimental the cause of these attacks can be when devices are exploited. These attacks can cause disruptions for manufacturing facilities and utility services in industrial IoT [10].

Emerging technologies such as Software-Defined Network (SDN) or Network Function Virtualization (NFV) have a promising future for overcoming these problems. These technologies provide flexibility and scalability to the IoT networks, and a more secure networking by making it easy for software updates. More on SDN and its practicality in securing IoT devices will be described later in this paper.

In this paper, a solution to securing IoT devices from DoS/DDoS attack in an SDN framework is designed. By implementing the function of detection and mitigation, IoT devices can be protected from such attacks. Here, an entropy-based detection is utilized to measure the distribution of the network traffic when the attacks happen. By measuring the distribution, it is possible to differentiate anomalous traffic from normal traffic. Entropy-based detection is also a lightweight solution for detecting DoS/DDoS attack, suitable for the IoT network. In the IoT network, it is practical to use a statistical DDoS detection method to have the necessary computational complexity.

The key contributions in this paper are listed in the following:

- A real-time approach to detect DoS/DDoS attacks by calculating the entropy of the network traffic in the SDN architecture. A practical way to use entropy-based detection with the consideration of fast-response while the network traffic is being loaded.
- The calculation of entropy is done with the help of sFlow-RT. (sFlow Monitoring Technology). Num-

* Toyo University, 1-7-11 Akabanedai, Kita-ku, Tokyo 115-8650

ber of packets being analyzed do not effect the computational power of the main controller of SDN.

2 Related Work

In [7], the paper suggested the use of fast-entropy to detect an incoming DDoS attack on flow-based network. The author used an adaptive threshold algorithm for setting the threshold of the detection, due to the unpredictable nature of the network activities. This proposal however did not discuss about the mitigation approach to the attack.

Another paper [11], discusses about the method for an early detection of DDoS attack against SDN controllers. The authors used entropy calculation for the first 500 packets of the traffic, to determined whether they are anomalous or not. Although the method has proven to be successful, the threshold set for the detection was calculated based on attack rate of 25%, 50%, and 75% only. This set threshold is not adaptive of the traffic condition and might vary depending on the traffic condition. A way of mitigating the attack was also not proposed in the paper.

[8] discusses the detection and mitigation of DoS/DDoS attacks in an IoT-based stateful SDN. The method suggests the use of entropy-based detection in a stateful SDN architecture. It uses limit calculation for setting the detection threshold, with the consideration of the mean and standard deviation of previously calculated entropy values. As for its mitigation process, packets that are determined to be malicious are dropped. The time of mitigation here rose if the time window size is increased.

In this paper, attack traffic will be mitigated by blocking the source of attack. The window size will not effect the mitigation time much because blocking the source of attack does not need much calculation.

3 Preliminaries/Background

This section describes the technology, tools, and approach necessary for implementing the proposed method of detecting and mitigating DoS/ DDoS attacks in the network. Which include SDN, sFlow-RT, hping3, and entropy-based detection. What DoS/DDoS are and how they can affect IoT devices is also briefed here.

3.1 Software-Defined Network (SDN)

Software-Defined Network or SDN, is a network architecture that is realized by virtualizing its components. SDN is structured in a way that its network is centralized, and can be centrally controlled with software applications. Although physical network can still be connected with SDN. Apart from its advantage of making a network more flexible can scalable, SDN enables the control of the whole network more consistent as updating software can be easily done. This type of network architecture is especially suitable for the high-bandwidth and dynamic nature of applications nowadays [3].

As shown in Figure 1, the components of SDN are made up of 3 layers: i.e Application Plane, Control Plane, and Data Plane. The separation of network packets and forwarding functions (Data Plane), and routing process (Control Plane) is meant for enabling direct programmability of the network applications and services [14]. The controller(s) on the Control Plane are considered to be the brain of the SDN network where the whole network is managed.

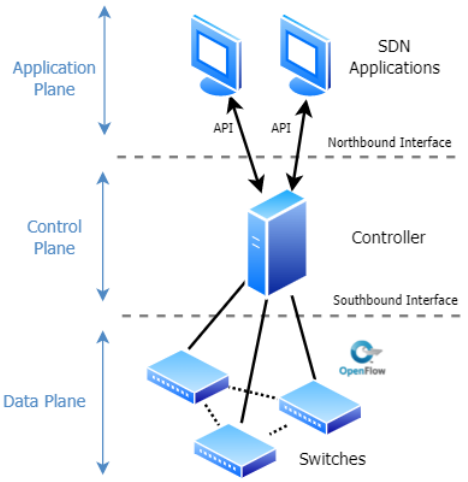


Figure 1: An SDN network.

Each layers communicates by sending and requesting API messages. The Application Place communicates with Control Plane through the Northbound Interface [Figure 1], and vice versa. As for Control Plane and Data Plane, Southbound Interface is used. To date, one of the most well-known protocol of the communication between a controller and a switch in the SDN network are OpenFlow protocol [15].

3.2 sFlow-RT

sFlow-RT is an asynchronous analytics technology that delivers real-time visibility to SDN [12]. It allows the creation of new performance-aware applications such as load balancing, DDoS mitigation and workload placement.

sFlow-RT receives continuous stream from network devices and applications, and convert these measurements into actionable metrics which can be accessed with REST API. This allows the configuration of measurements and control of the network flow.

In this paper, sFlow-RT is used to calculate the entropy of network traffic in the SDN network and judge the state of the traffic.

3.3 hping3

To simulate traffic in the experiment, hping3 is used. hping3 is a network tool used to send network packets

such as UDP, TCP, or ICMP [5]. It can perform different kinds of test like port scanning, network performance tih different protocols and such. Here, hping3 is used to simulate normal UDP traffic and anomalous traffic such as UDP amplification in the network.

3.4 DoS and DDoS

Denial-of-service (DoS) attack is a type of cyberattack in which the attacker intends to take down a machine by making unavailable. Typically, DoS is done by flooding a targeted machine with redundancy of request messages, more than it can handle, in attempt to overload it and prevent all legitimate request messages from entering. When the attack is carried, the machine is either slowed temporarily or completely down.

On the other hand, distributed denial-of-service (DDoS) attack is a DoS attack but the attacking machines come from multiple sources at a time. There can be thousands attacking machines conducted and these machines have been previously infected with malicious codes, and can be controlled by an attacker. Collectively, these attacking machines are known as "botnets".

3.4.1 DoS/DDoS in an IoT network

DDoS attacks in the IoT networks are considered one of the growing challenges in recent years [6]. The use of the limited resources (e.g. storage limitation and network capacity) in IoT devices make them more vulnerable to flooding attack.

3.5 Entropy-Based Detection

Entropy-based detection is one of the common ways of detecting DoS/DDoS traffic. In information theory, entropy can be used to measure the uncertainty of information. In the context of detecting a heavy incoming traffic, entropy would measure the randomness in the incoming packets in the network. This calculation can be done with Shannon entropy. Shannon entropy can be defined as :

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i \tag{1}$$

Where there is an information source, n is the independent symbols, p_i is the probability of each n, and H would be entropy value.

During a DoS/DDoS attack, the changes in network traffic distribution fluctuates drastically. When this happens, the entropy calculation could measure that and trigger alert about the state of the traffic. Anomalous traffic are usually created from different spoofed source IPs. The higher the randomness of the traffic, the higher the entropy. Conversely, the lower the randomness of the traffic (with the redundant appearance of single source IP) the lower the entropy.

To judge whether the calculated entropy is anomalous' or not, it is necessary to set a *threshold*. If the calculate entropy drops below the threshold, then the traffic can be considered as an attack. Aside from setting a threshold, a *window size* should also be set. A window size could either be a number of a packets or a time period. The traffic's entropy will be calculated per window size at every interval.

4 Proposed Method

This method proposes the monitoring of traffic flow using sFlow-RT. With sFlow-RT, it will get an estimated number of packets travelling from each host in the network [9]. Here, estimated number of packets and each connections (source IP and destination IP) are being analyzed. sFlow-RT is being implemented in the Control Plane and it works together with the controller to analyze any anomalous packets. sFlow-RT will then instruct the controller to block the connections that have been determined to be sending anomalous packets.

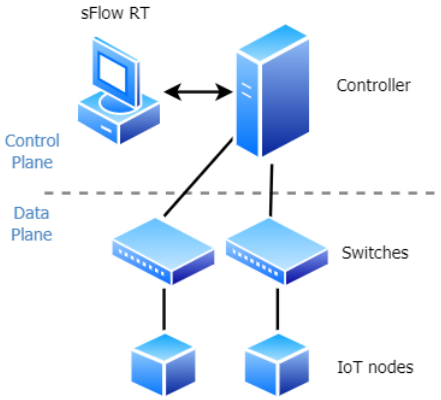


Figure 2: An SDN network with sFlow-RT.

4.1 Entropy-Based Detection in sFlow-RT

Packets travelling in the network will be monitored with sFlow-RT. Derived from Equation (1), the entropy calculation for the detection of anomalous packets can be written as follows:

$$H = -\sum_{i=1}^{c} \left(\frac{x_i}{n}\right) \log_2 \left(\frac{x_i}{n}\right) \tag{2}$$

Where c is the total number of hosts, x_i is the number of travelling packets from each i^{th} connected source IP, and n is the total number of travelling packets in the network.

To analyze the travelling packets, the time window is set to be 5 seconds. Therefore, entropy of the network is being calculated every 5 seconds. It is worth noting that to get the value of x_i and n, sFlow-RT uses sampling to estimate the number of travelling packets. This is why sFlow-RT is applicable to high speed networks and suitable for handling large flows [13].

During a steady network the entropy value will fluctuate with the reasonable amount of packets travelling throughout the network. When DoS/DDoS attack occurs, a large flow is sent from one or more sources. The source IPs of these attackers appears more frequently

than the rest of the source IPs in the entropy calculation. Thus making them less unique and decreases the entropy value.

4.1.1 Flow Aggregation

For every time window, the flow of the network is aggregated by their source IP and destination IP. For each connection established in the network, every travelling packets is monitored and each of the packet that share the same source/destination IP will be categorized as the number of x_i. Here is an example of one aggregated flow:

```
{ "value": 5.715389170949964,
  "key": "10.0.0.1,10.0.0.6,50" }
```

Where `value` is the estimated number of packets from `10.0.0.1` (source IP) to `10.0.0.6` (destination IP) through port `50`.

4.1.2 Threshold Setting Method

As explained in Section 3.5, setting a threshold is needed for entropy detection. The threshold setting method used here is derived from [7]. This mitigation method was chosen because of its adaptive threshold algorithm. Where the threshold of the detection is updated according to the state of the traffic. It is suitable for detecting small and stealthy attack.

Basically, the method would calculate the mean and standard deviation of previous entropy values for a period of time and set the threshold base on it. That way, the practice of highly set threshold can be avoided, and detection would work in sensitive situation.

The flow of the threshold setting method can be summarized as below:

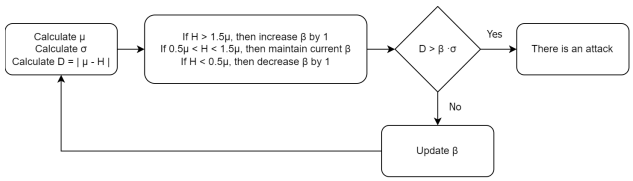


Figure 3: A flowchart of adaptive threshold algorithm [7].

Where H is the current entropy value, μ and σ are the mean and standard deviation of flow count during a particular time interval , D is the difference between the mean value μ and the entropy, and β the threshold multiplication factor.

4.2 Mitigation Method

The mitigation of the attack would be triggered if the network traffic goes higher then the set threshold. The mitigation is done by blocking the source IP and source port of the attack. This can be implemented by sending a POST request from sFlow-rt to the controller (Ryu), instructing it to stop the flow of the malicious traffic.

There are different methods of mitigation for DoS/DDoS in SDN. For example, instructing the switch to drop suspicious packets, or null routing (blackholing) where all malicious traffic is directed to a non-existent IP address. But here mitigation is done by blocking attack source instead of dropping the packets.

Here is an example of the display of a successful block from a source attacker on Ryu controller:

```
(14314) accepted ('127.0.0.1', 41124)
127.0.0.1 - - [30/Dec/2021 14:32:40] "POST /
    stats/flowentry/add HTTP/1.1" 200 134
0.002230
```

4.3 Flowchart

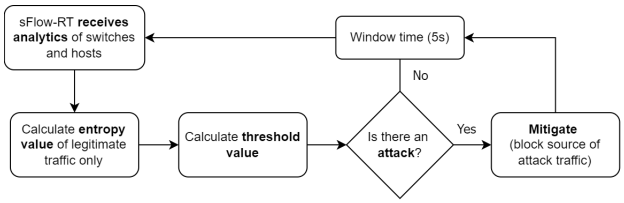


Figure 4: A flowchart of the proposed method.

sFlow-RT will first analyze the analytics of every switch and host in the network. The information being analyzed include, number of packets travelling in the network in accordance to their source/destination IP for statistical measure. Then, their entropy is calculated to determine whether the traffic is a threat or not. An attack is considered to be conducted if the traffic escalates beyond the threshold. The window time is set to be 5 seconds, to decide if any attack is being conducted. Otherwise, if there is an attack, mitigation action should be taken. This process is being repeated at every interval.

5 Experimental Method

To conduct the experiment, an SDN emulator was used to simulate network traffic in an SDN setting. Here, for simplicity only UDP packets are used and DoS/DDoS traffics are UDP amplification attack. The attacking devices in the network are assumed to have been infected by malware and could conduct DoS/DDoS attack. Although in a more realistic SDN-IoT network, edge devices and gateways are used, here such devices are not set up.

5.1 Experiment Setup

The tools for the conducting the experiment include; 1) Mininet, a network emulator that can create virtual hosts, switches, controllers, and links, in an SDN framework. 2) hping3, the tool to simulate network traffic. 3) sFlow-RT, the tool used to analyze and monitor network packets.

5.2 DoS Scenario

To test the detection in a DoS scenario, a network like in Figure 5 was set up. The red node (h1) represents the attacker and the green node (h4) represents the target. The bandwitdh of each connection is set to be 1Mbits/sec. So any flow that goes beyond 1Mbits/sec would make the connected device inaccessible. The traffic is being monitored from the Control Plane. The attacking machines are sending packets as fast as possible to the target machine until it goes beyond the bandwidth.

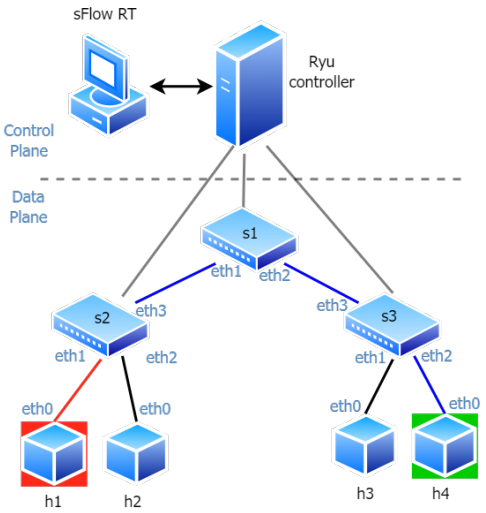
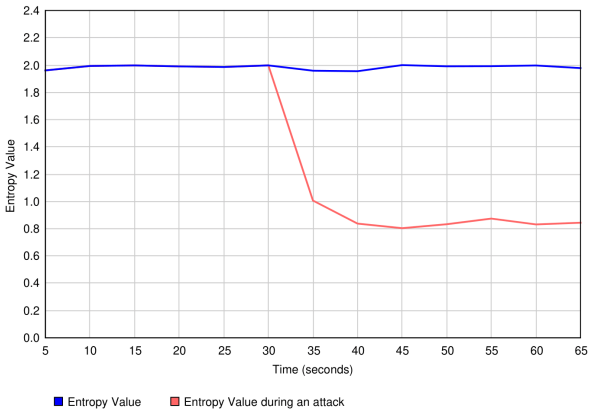


Figure 5: Dos scenario.

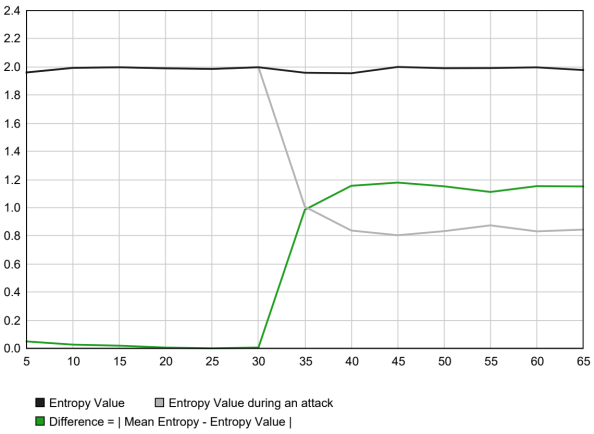
In the graph below, the entropy value of the traffic is plotted during an attack. The attack occurred between second 30 and 35, and the entropy value dropped from 1.998 to 1.006 within that time window. This significant drop alerts the threshold described in Section 4.1.2, and the large flow is then considered to be an attack. The sudden increase in the difference between *mean entropy* and *current entropy value* is shown in Figure 6. The attack flow is considered be malicious as long as this difference is huge (Figure 3).

The entropy value in Figure 6 shown in blue did not drop together with the entropy value shown in red line because the entropy value of the attacking flow is not calculated after it is detected. Thus only calculating the entropy of legitimate traffic.

This significant drop alerts the detection that an attack is occurring. Only within one time window (5s), the attack was detected. sFlow-RT immediately instructs the controller to block the flow that has been detected as malicious, so that it will not affect the targeted host. As shown in Figure 7, only port s2-eth1 was affected from this attack. Referring from Figure 5, the attack could have affected the ports that are in blue line. Instead, only one port in which the attack was from is affected. Port s1-eth1, s3-eth3, and h4-eth0 could have been affected if source of attack was not blocked.



(a) Entropy value before and after the attack.



(b) Difference in mean entropy and current entropy

Figure 6: Analysis of entropy values during a DoS attack.

Figure 7 plotted a slight increase in the flow of the ports (e.g. port s1-eth1) that are almost affected by the attack before it is mitigated (shown in the red arrow).

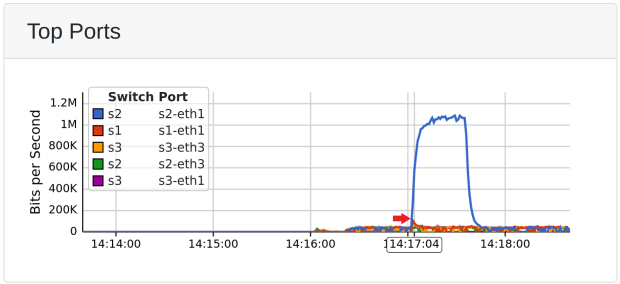


Figure 7: Affected ports during the DoS attack.

5.3 DDoS Scenario

In the DDoS scenario, the network architecture that was set up is similar to the one in DoS scenario. Instead this time 9 hosts and 4 switches were used connected in a tree topology. Multiple attacking hosts (h1-h4) are set to attack a single target (h9). The attacking

machines are also sending packets as fast as possible to the target machine.

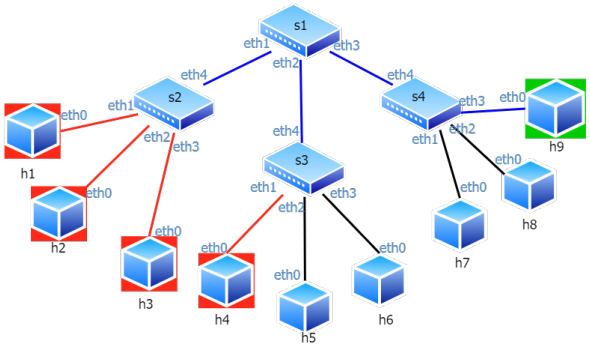
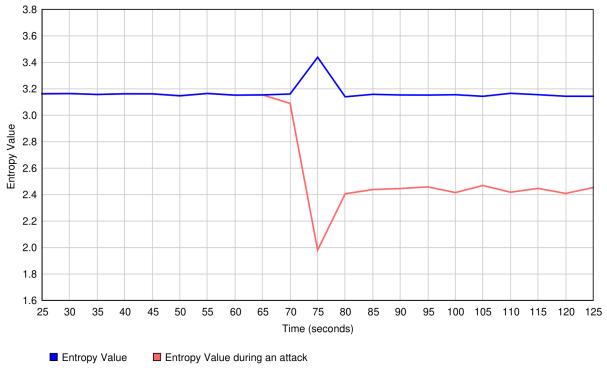
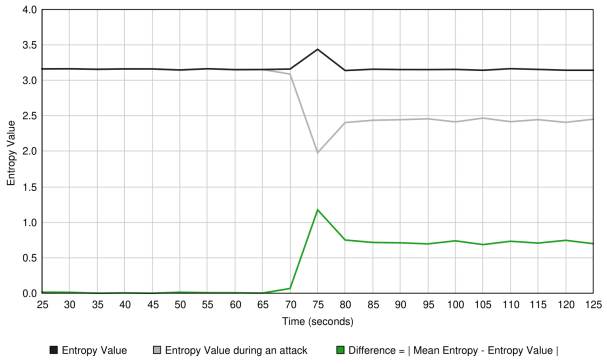


Figure 8: DDoS scenario.

In Figure 9 below, the entropy value of the traffic is graphed during the attack. The attack in Figure Figure 9(a) occurred between second 75 and 125, and the entropy value dropped from 3.1540 to 1.980 within time window of 70 to 75. This significant drop again alerts the threshold set the flow from that connections are then classified to be an attack. This time multiple connections are being blocked. The sudden surge in the difference between mean entropy and current entropy value is shown in Figure 9(b).



(a) Entropy value before and after the attack.



(b) Difference in mean entropy and current entropy

Figure 9: Analysis of entropy values during a DDoS attack

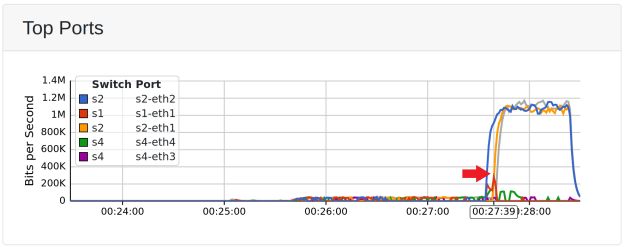


Figure 10: Affected ports during the DDoS attack

Table 1: Result comparison of detecting DoS and DDoS attack.

Type of Attack	DoS (1 source of attack)	DDoS (4 sources of attack)
No. of connections made	4	9
Detection speed (no. of time window)	1	1
Difference in Entropy at the start of the attack	0.985	1.175

Similar to DoS scenario, the controller blocked the malicious flow. Figure 10 shows that only port s2-eth1, s2-eth2, s2-eth3, and s3-eth1 were affected from the attack. The blue lines in Figure 8 (s1-eth1, s1-eth2, s1-eth4, h9-eth0) could have been affected but mitigated instead. The red arrow in Figure 10 points out how s1-eth1 was almost affected by the slight increase in the beginning.

5.4 Overall Comparison

After conducting the experiments, the overall results of both attacks can be tabulated as below:

The number of attacking source in DoS was 1 and in DDoS was 9, sending packets as quickly as possible to the target (Table 1). It took both the scenarios 1 time window (5 seconds) to detect and the incoming attacks. With the calculated difference in entropy at the start of the attack of 0.985 for DoS, and 1.175 for DDoS.

6 Conclusion

This paper presented a solution to detecting an incoming DoS/DDoS attack in an SDN-based IoT network. Blocking the source of attack upon detection is observed to be a quick solution for mitigating large flow attack. Adaptive threshold setting was used so that the threshold varies depending on the normal state of the traffic. As for future work a larger scale of the network with a more realistic topology could be conducted.

References

[1] C. Petrov, "49 internet of things statistics to show how big it is in 2021," *TechJury*, 06-Dec-2021. [Online]. Available: https://techjury.net/blog/internet-of-things-statistics/gref. [Accessed: 04-Jan-2022].

[2] "Cisco annual internet Report - Cisco Annual Internet Report (2018–2023) White Paper," *Cisco*, 10-Mar-2020. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html. [Accessed: 04-Jan-2022].

[3] D. Oh, "Getting started with software-defined networking," *Enable Sysadmin*, 07-Jun-2019. [Online]. Available: https://www.redhat.com/sysadmin/getting-started-sdn. [Accessed: 04-Jan-2022].

[4] E. Jasinska, "sFlow I can feel your traffic," *Amsterdam Internet Exchange*, 2006.

[5] "HPING3: Kali linux tools," *Kali Linux*, 26-Nov-2021. [Online]. Available: https://www.kali.org/tools/hping3/. [Accessed: 04-Jan-2022].

[6] "Implementation status of alerting users of vulnerable IOT devices and IOT devices infected with malware (FY2019): Press release," *MIC ICT Policy*, 15-May-2020. [Online]. Available: https://www.soumu.go.jp/main_sosiki/joho_tsusin/eng/pressrelease/2020/5/15_1.html. [Accessed: 04-Jan-2022].

[7] J. David and C. Thomas, "DDoS attack detection using fast entropy approach on FLOW- based network traffic," *Procedia Computer Science*, vol. 50, pp. 30–36, 2015.

[8] J. Galeano-Brajones, J. Carmona-Murillo, J. F. Valenzuela-Valdés, and F. Luna-Valero, "Detection and mitigation of DOS and ddos attacks in IOT-based stateful SDN: An experimental approach," *Sensors*, vol. 20, no. 3, p. 816, 2020.

[9] P. Phaal and S. Panchen, "Packet Sampling Basics," *Packet sampling basics*. [Online]. Available: https://sflow.org/packetSamplingBasics/. [Accessed: 04-Jan-2022].

[10] S. Langkemper, "The most important security problems with IOT devices." [Online]. Available: https://www.eurofins-cybersecurity.com/news/security-problems-iot-devices/. [Accessed: 04-Jan-2022].

[11] S. M. Mousavi and M. St-Hilaire, "Early detection of ddos attacks against Sdn Controllers," *2015 International Conference on Computing, Networking and Communications (ICNC)*, 2015.

[12] "sFlow-RT," sFlow. [Online]. Available: https://sflow-rt.com/. [Accessed: 04-Jan-2022].

[13] *Society 5.0.* [Online]. Available: https://www8.cao.go.jp/cstp/english/society5_0/index.html. [Accessed: 04-Jan-2022].

[14] "Software-defined networking (SDN) definition," *Open Networking Foundation*, 03-Jun-2020. [Online]. Available: https://opennetworking.org/sdn-definition/. [Accessed: 04-Jan-2022].

[15] "What is an SDN controller? definition - sdxcentral." [Online]. Available: https://www.sdxcentral.com/networking/sdn/definitions/what-is-sdn-controller/. [Accessed: 04-Jan-2022].