# Analysis and Identification of Distributed Denial of Service Attacks Using Intra-Domain Messaging Schemes

Setiyo Budiyanto
*Department of Electrical Engineering*
*Universitas Mercu Buana*
Jakarta, Indonesia
sbudiyanto@mercubuana.ac.id

Lukman Medriavin Silalahi
*Department of Electrical Engineering*
*Universitas Mercu Buana*
Jakarta, Indonesia
lukman.medriavin@mercubuana.ac.id

Imelda Uli Vistalina Simanjuntak
*Department of Electrical Engineering*
*Universitas Mercu Buana*
Jakarta, Indonesia
imelda.simanjuntak@mercubuana.ac.id

Agus Dendi Rochendi
*Research Center for Oceanographic Physics*
*Badan Riset dan Inovasi Nasional*
Jakarta, Indonesia
agus105@brin.go.id

Septi Andryana
*Department of Informatic Engineering*
*Universitas Nasional*
Jakarta, Indonesia
septi.andryana@civitas.unas.ac.id

Abdul Hamid
*Industry Machining Tech Focus Group*
*Faculty of Technical and Vocational Education*
*Universiti Tun Hussein Onn Malaysia*
Johor, Parit Raja, Malaysia
abdulhamid@uthm.edu.my

Mochamad Ikhsan Mubarak
*Department of Electrical Engineering*
*Universitas Mercu Buana*
Jakarta, Indonesia
ikhsankatsu07@gmail.com

2024 IEEE 6th Symposium on Computers &amp; Informatics (ISCI) | 979-8-3503-5385-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/ISCI62787.2024.10668093

***Abstract*— *This research discusses security systems against Distributed Denial-of-Service (DDoS) attacks. The focus of domestic co-operation research is the resolution and identification of DDoS attacks using the Intra-Domain Messaging (I-DM) method. The proposed method has the advantage of a filtering method that requires the Internet Service Provider (ISP) to validate packets from its network using valid prefixes and filter out packets using false source addresses that are out of range of legitimate addresses in a Software Defined Network (SDN) network. This research uses the Mininet simulator as an SDN engineering for data collection of entropy values. Finally, the results show that the I-DM mitigation scheme with filtering method has been effective against IP-Spoofing attacks with an accuracy rate of 83.4%.***

***Keywords*— *I-DM, DDoS, Mininet, SDN, ISP***

## I. Introduction

Cyber Security [1], [2] is an interesting topic and of particular interest resulting in a novelty of cybersecurity systems to analyze security awareness in depth, and find factors such as socio-demographics, perceptions of cyber security, cybersecurity violations of IT (Information Technology) use [3].

The development of IT globalization has brought great changes to human life. According to [4]–[6] that communication relations between people and nations are getting easier and faster without being affected by space and time. Hardware threats are caused by certain activities within a system [7]–[9], as well as disruption to network systems and data/information threats. So, it becomes a threat caused by the spread of unauthorized data so that detection is needed so that the faster the attack is known, the better the mitigation.

In recent times, the threat of DDOS attacks has intensified, causing significant harm to Internet Service Providers (ISPs). Several high-profile incidents, such as those involving U.S. banks [10], [11] , have highlighted the increasing power and destructiveness of these attacks, resulting in silent disruptions to ISPs [12]. Consequently, major Internet services like Amazon and GitHub [13], [14] have experienced prolonged outages [15].

Hence, there is a pressing need for effective collaboration between domains to achieve two key objectives: reducing packet forwarding costs by efficiently handling amplified and irrelevant attack traffic and implementing DDOS mitigation strategies to counter attacks without relying on costly and inflexible centralized approaches.

In essence, relying on centralized solutions creates a single point of failure and exposes vulnerabilities to DDOS attacks, hindering effective information sharing and decision-making across various domains. Figure 1 illustrates the outcomes of research on SDN (Software Defined Network)-based DDOS attack mitigation, categorized into topic slices, highlighting prominent approaches and their associated constraints. Standard BCP 38 [16]–[18] introduces a filtering technique wherein each ISP is required to:

1. Authenticate that packets from its network employ a valid IP Address prefix.
2. Exclude packets using spoofed source addresses beyond the scope of legitimate addresses.

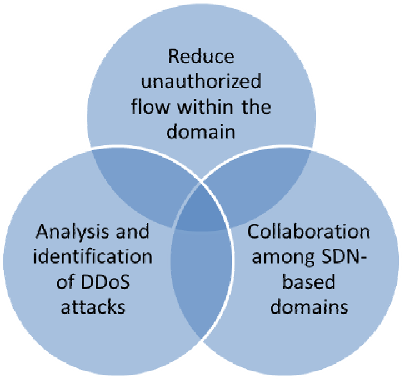

Reduce unauthorized flow within the domain
Analysis and identification of DDoS attacks
Collaboration among SDN-based domains

Fig. 1. Venn diagram

I-BS (Intra-Bayes) functions as a machine learning (ML) binary classifier, automatically discerning between legitimate and invalid flows based on stateful traffic characteristics. The I-DM (Intra Domain Messaging) mitigation strategy employs a filtering approach, mandating ISPs to validate packets' origins within their network using legitimate prefixes and discard packets with spoofed source addresses outside the authorized range. In [19] , a flow-based self-organizing maps scheme was proposed, utilizing the OpenFlow (OF) to gather traffic flow data, albeit without accounting for the overhead generated by the OF protocol's flow collection process [20].

Several previous studies were used as primary reference references discussing problems related to handling DDOS [21], management of unauthorized flow control on networks [22], application of DDOS prediction and bot detection [23], application of SDN [24] to identify and mitigate DDOS attacks and compared with other methods, as well as research references related to the discussion of DDOS mitigation methods with I-DM [19], [20]. Based on these references, a research objective was created to conduct DDOS analysis and identification using the DDOS I-DM mitigation method using SDN and control unauthorized flow on the compromised network. Mininet, as outlined in [25], [26], serves as a tool for efficiently testing extensive network prototypes with minimal resources. By employing Mininet as a simulator, users can execute code interactively on a laptop or virtual machine without necessitating code modifications, ensuring the simulation mirrors real network environments accurately.

The utilization of entropy in DoS (Denial of Service) detection is attributable to its capability to assess packet randomness upon network ingress. Entropy computation within a specified window serves to gauge the uncertainty of forthcoming packets. Detection of an attack necessitates the establishment of a threshold. Once the computed entropy surpasses or falls below this threshold (depending on the applied scheme), an attack is identified. The entropy (H) is determined based on the number of packets in the window (n) and the probability of each element within the window, denoted as "Pi" [27]–[29].

Building upon prior studies, this research posits that maximal entropy occurs when all elements exhibit equal probabilities, while a lower occurrence of an element correlates with decreased entropy. Rooted in Shannon's information theory, this fundamental concept underpins Intra-Entropy (I-ES). Operating atop the controller, I-ES utilizes the sFlow protocol. During a DDoS attack, the concentration of packets with the same IPDST increases, contrasting with a more evenly dispersed $IP_{DST}$ probability distribution during normal network operation. High entropy signifies a widespread $IP_{DST}$ probability distribution, whereas low entropy indicates $IP_{DST}$ concentration.

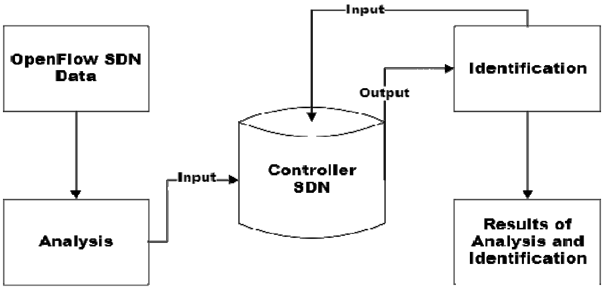


Fig. 2. Block diagram system

## II. Method

### A. Research Method

Figure 2 illustrates a research block diagram, commencing from OF protocol data and flow monitoring data and concluding with the outcomes of DDoS attack analysis and identification. Flow monitoring operates via the sFlow protocol, tracking the received packet count and corresponding flow duration. Upon receiving a feature request (*ofp_flow_stats_request*) from the SDN controller, the OF-switch responds by transmitting the flow table contents (*ofp_flow_stats_reply*).

During DDoS attacks with a high volume of flow entries, sFlow conducts necessary flow aggregation. Following this, the I-ES method calculates entropy values to assess data randomness within the domain via sFlow and extract network features using traffic flow network features. Simultaneously, the I-BS technique leverages entropy values to automatically classify unauthorized flows.

The analysis testing incorporates the DDoS-ML I-DM detection and mitigation module within the I-DM scheme. This research focuses on identifying unauthorized flows through I-ES, I-BS, and I-DM. I-ES measures data randomness within the victim's domain in real-time using network traffic flow features, while I-BS detects unauthorized flows based on stateful network traffic features. It operates atop the SDN Controller, where the application layer utilizes entropy values to gather traffic data and identify unauthorized flows.

The research data were gathered from the SDN controllers implementing the I-ES and I-BS schemes. Each scheme involves analyzing the randomness of incoming data, specifically the incoming flow over a defined period. I-ES operates as an application atop the controller, utilizing the sFlow protocol to collect traffic data and compute the entropy of each stream. During DDoS attacks, there's a concentration of packets with the same destination IP address (denoted as IPDST), while under normal conditions, IPDST exhibits a more dispersed probability distribution. High entropy values indicate a widely dispersed IPDST probability distribution, whereas low entropy values signify IPDST concentration. I-BS, functioning as a binary classifier in machine learning, employs stateful traffic features and assigns them false probabilities for classification.

### B. Intra DDoS Mitigation Scheme

DDoS mitigation schemes create mitigation modules, then analyze and identify using I-DM to obtain results based on the I-DM scheme using information collection methods based on packet flow samples, namely the I-ES and I-BS methods. Then at the time of simulation, a normal UDP packet is created from one of the hosts by sending all randomly generated packets to go to all hosts. To perform an attack against a single host and run manually by entering the destination IP Address of the target host. Table 1 shows an attack traffic rate of 25%, the interval of sending 1 packet for normal traffic is 0.1 s and for attack traffic is 0.025 seconds. By default, only packet headers are sent to the controller, so no payload is added when the package is created.

Table 1. Attack profile

| Type | Payload | Interval (s) | Traffic Rate (%) |
|---|---|---|---|
| UDP | - | 0,025 | 25 |

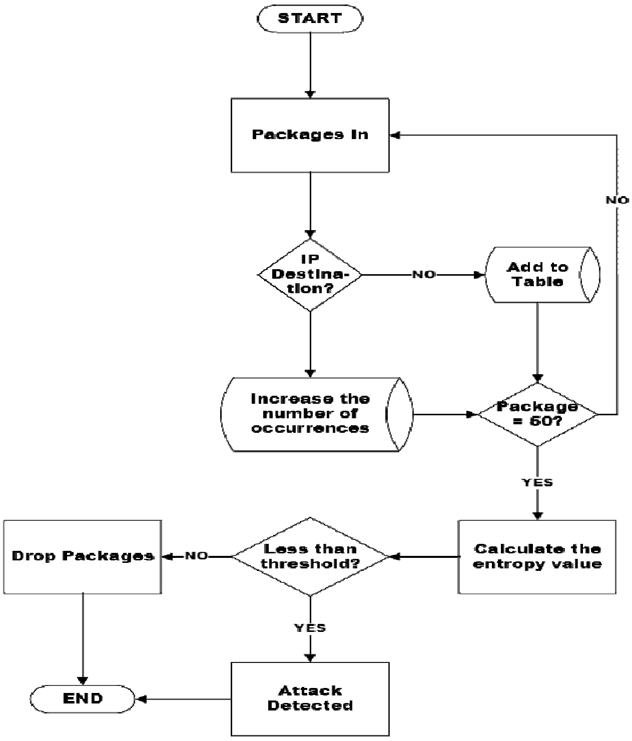


Fig. 3. Flowchart system

Figure 3 shows a flow chart that when a packet enters the controller, the destination IP address of the incoming packet will be seen in a table, if the destination IP has not been stored in the table it will be added to the table as one, but if the destination IP has been saved then the count will increase. When the packet has reached 50, the entropy value will be calculated, when the entropy value is below the threshold value then the packet will be considered an attack packet and dropped but if the entropy value is above the threshold value, then the packet will be forwarded.

## C. *Intra-Domain Method with SDN*

The I-DM approach serves as a means of countering DDoS attacks by safeguarding victims and efficiently mitigating illicit traffic. It actively monitors unauthorized flows identified by I-BS, deleting suspicious packets if their rate exceeds a set threshold. This strategy is geared towards identifying and addressing unauthorized streams, incorporating I-ES, I-BS, and I-DM, leveraging entropy values for real-time detection and mitigation. Furthermore, it utilizes REST APIs to manage SDN controllers and block unauthorized traffic, effectively curbing unauthorized traffic within the domain. While the OF protocol was not originally intended for QoS features, the introduction of OF 1.3 brought about support for such capabilities.

### *1) Determine the entropy value*

Entropy serves as a crucial metric for DDoS detection owing to its capability to assess the randomness of incoming packets within the network. Calculated within a specific window, entropy gauges the uncertainty regarding future packets. When each element within the window exhibits identical probabilities, the entropy value reaches its maximum. Conversely, if certain elements surpass others in frequency, the entropy diminishes. Detection of an attack necessitates the establishment of a threshold, with entropy exceeding or falling below it triggering an alert, depending on the applied scheme. In equation 1, representing the calculation of entropy (H), "n" denotes the packet count within the window, while "Pi" signifies the probability of each element.

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

### *2) Calculating the entropy value*

Table 2 presents the notation key utilized by I-ES. Implemented atop the controller via the sFlow protocol, I-ES functions to gauge alterations in traffic data within the victim domain over the monitoring interval, denoted as $\Delta T$. During a DDoS attack on the target domain, there's a concentration of packets with identical destination IP addresses (IPDST), contrasting with the dispersed IPDST probability distribution observed in normal network conditions. High entropy reflects a widely dispersed IPDST probability distribution, while low entropy indicates IPDST concentration, hence facilitating I-ES in traffic monitoring.

Table 2. Notation

| Notation | Definition |
|---|---|
| $MAC_{SRC}$ | The MAC source address of a packet. |
| $MAC_{DST}$ | The MAC destination address of a packet. |
| $IP_{SRC}$ | The IP source address of a packet. |
| $IP_{DST}$ | The IP destination address of a packet. |
| $PORT_{SRC}$ | Source PORT of a packet. |
| $PORT_{DST}$ | The destination PORT of a packet. |
| $IP_{PROTO}$ | Trasport Protocol Packet (UDP/TCP). |
| $S_J$ | Switch OF-Sj. |
| $\Delta T$ | Monitoring intervals. |
| $F_{I,J}$ | Fi flow on local OF switch Sj. |
| $P_{I,J}$ | Fi flow probability over all flows on the local OF switch Sj. |
| N | The total number of flows on the local OF switch Sj. |
| R | The set of natural numbers. |
| I | The set of positive integers. |

### *3) Testing of intra-domain entropy schemes*

The intra-domain approach suggests a filtering technique where ISPs:

1. Validate packets from their networks with legitimate IP address prefixes.
2. xclude packets with spoofed source addresses beyond valid ranges.

This research evaluates I-DM schemes for effective and scalable DDoS attack detection, integrating entropy computations via sFlow to prevent unauthorized traffic.

# III. Result and Discussion

## A. *DDoS Detection Results*

DDoS detection using the I-DM scheme is carried out to analyze and identify DDoS attacks based on the entropy value in the I-DM scheme by presenting the output of information obtained through the process of calculating the entropy value, through the following tests:

### *1) Normal Traffic Network Testing*

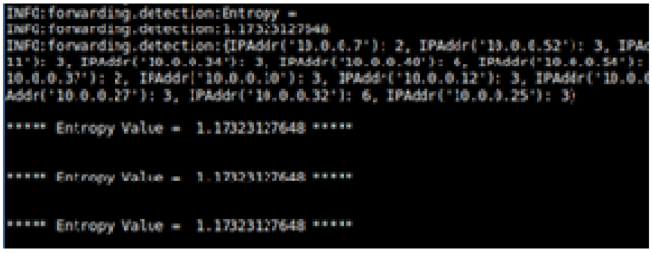
$IP_{SRc}$ uses the random function "randrange (1.256)" while for $IP_{DST}$ it will be specified. Normal traffic is executed by creating random packets that are sent randomly to all hosts. In normal traffic testing, the controller calculates the entropy value to determine whether the incoming packet is an attack packet or a normal packet. Figure 4 shows the results of testing normal network traffic that the entropy value is normal, this is because in the Mininet simulator when given the command "*python topo.py*" to activate the topology that has been created in the simulator automatically the controller and all functions that have been created will run on host 2 by giving the command "*h2 source gentraffic.sh*".

### *2) Attack Traffic Network Testing*

Figure 5 shows the results of attack traffic network testing when incoming packets first the entropy value is above the threshold value, then the entropy value drops and the entropy value is below the threshold value, then the entropy value rises above the threshold value and falls below the threshold until the end of the observation.

The test results in figure 5 occur when from 2 (two) hosts that have been run then manually to 1 (one) host that is the target of the attack. In an attack traffic test, the controller will do the same thing as in a normal traffic test. Attack traffic is run from 2 (two) hosts and carried out manually to 1 (one) host that is the target of the attack. In an attack traffic test, the controller will do the same thing as in a normal traffic test.

Figure 6 shows the output of the entropy value graph experiencing significant and unstable ups and downs. This shows that when the entropy value is below the threshold value, there is a DDoS attack when the packet arrives. In addition to getting a significantly changed entropy value, then, when the packet enters through the controller, the controller will also take an action on the packet that causes the entropy value to be below the threshold value, namely by blocking the packet that is considered an attack packet.



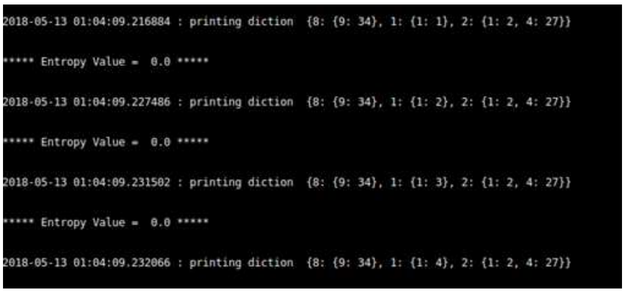
```
INFO:forwarding.detection:Entropy =
INFO:forwarding.detection:1.17323127648
INFO:forwarding.detection:{IPAddr('10.0.0.7'): 2, IPAddr('10.0.0.52'): 3, IPAc
11'): 3, IPAddr('10.0.0.34'): 3, IPAddr('10.0.0.40'): 4, IPAddr('10.0.0.54'):
10.0.0.37'): 2, IPAddr('10.0.0.10'): 3, IPAddr('10.0.0.12'): 3, IPAddr('10.0.0
Addr('10.0.0.27'): 3, IPAddr('10.0.0.32'): 6, IPAddr('10.0.0.25'): 3}

***** Entropy Value =  1.17323127648 *****


***** Entropy Value =  1.17323127648 *****


***** Entropy Value =  1.17323127648 *****
```

Fig. 4. Normal traffic network



```
2018-05-13 01:04:09.216884 : printing diction  {8: {9: 34}, 1: {1: 1}, 2: {1: 2, 4: 27}}

***** Entropy Value =  0.0 *****

2018-05-13 01:04:09.227486 : printing diction  {8: {9: 34}, 1: {1: 2}, 2: {1: 2, 4: 27}}

***** Entropy Value =  0.0 *****

2018-05-13 01:04:09.231502 : printing diction  {8: {9: 34}, 1: {1: 3}, 2: {1: 2, 4: 27}}

***** Entropy Value =  0.0 *****

2018-05-13 01:04:09.232066 : printing diction  {8: {9: 34}, 1: {1: 4}, 2: {1: 2, 4: 27}}
```

Fig. 5. Attack traffic network

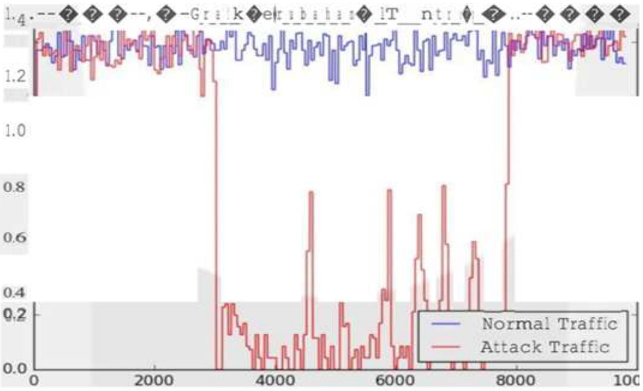


Fig. 6. Threshold status

## *B. Data Analysis*

This section discusses DDoS attack detection, which is when a module is added to the controller. Checking the incoming packets into the controller, the $IP_{DST}$ of incoming packets will be seen in a table, if the $IP_{DST}$ has not been saved it will be added, if it has been saved then the count will increase. Equation 2 shows the table of occurrences of destination IP addresses. When it is 50 packets, the entropy value will be calculated. To calculate the entropy value shown in equation 1, use equation 2 and equation 3, where W is the window size and Pi is the probability for each $IP_{DST}$.

$$W = \{(X_1, Y_1), (X_2, Y_2), (X_3, Y_3), \dots\} \tag{2}$$

$$p_i = \frac{p_i}{n} \tag{3}$$

If the IP address appears only once, then the entropy value will be maximum. But when there is an attack, many packets will flood the host and fill the windows size, reducing the entropy value. If the entropy value is below the threshold value, then the packet will be considered an attack. Equation 4 shows the formula for generating accuracy analysis and identification of DDoS attacks on the proposed network schema as follows.

$$Accuration = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{4}$$

Information:

1. True Positive (TP) is the number of attack records classified as attacks.
2. True Negative (TN) is the number of attack records classified as normal.
3. False Positives (FP) is the normal number of records classified as an attack.
4. False Negative (FN) is the actual number of attacks but is classified as normal.

This data is obtained from the traffic capture process that runs using the TCPDump tool using output with .pcap extension. Then, it is extracted using the CICFFlowmeter tool and converted into a .csv file so that it can be processed. Table 3 shows that this experiment used the dataset obtained at the time of testing.

Table 3. Dataset

| Time | Frame Number | Frame Length | $IP_{SRC}$ | $IP_{DST}$ | $PORT_{SRC}$ | $PORT_{DST}$ | Protocol |
|---|---|---|---|---|---|---|---|
| 0,00022 | 2 | 900 | 10.50.197.6 | 31.13.84.8 | 49218 | 443 | TCP |
| 0,00023 | 3 | 171 | 31.13.84.8 | 192.168.66.111 | 443 | 40991 | TCP |
| 0,00024 | 4 | 1500 | 192.168.68.148 | 54.225.245.82 | 54602 | 443 | TCP |
| 0,00047 | 5 | 126 | 192.168.79.128 | 64.15.113.173 | 55251 | 443 | TCP |
| 0,00086 | 6 | 126 | 192.168.79.128 | 64.15.113.173 | 55251 | 443 | TCP |
| 0,00111 | 7 | 2906 | 74.125.133.141 | 10.50.198.74 | 443 | 50936 | TCP |
| 0,0014 | 8 | 70 | 10.50.198.74 | 74.125.133.141 | 50936 | 443 | TCP |
| 0,00151 | 15 | 126 | 31.13.84.8 | 192.168.66.111 | 443 | 40991 | TCP |
| 0,00151 | 16 | 70 | 10.50.198.74 | 74.125.133.141 | 50936 | 443 | TCP |
| 0,00151 | 17 | 126 | 192.168.79.128 | 64.15.113.173 | 53611 | 443 | TCP |
| 0,00157 | 18 | 70 | 10.50.198.74 | 74.125.133.141 | 50936 | 443 | TCP |
| 0,00175 | 20 | 86 | 221.203.142.71 | 10.50.195.142 | 40136 | 22 | TCP |
| 0,00193 | 22 | 70 | 10.50.195.142 | 221.203.142.71 | 22 | 40136 | TCP |
| 0,00193 | 23 | 126 | 192.168.79.128 | 64.15.113.173 | 55251 | 443 | TCP |
| 0,00256 | 25 | 1500 | 192.168.68.148 | 54.225.245.82 | 54602 | 443 | TCP |
| 0,00256 | 26 | 126 | 54.225.245.82 | 192.168.68.148 | 443 | 54602 | TCP |
| 0,00284 | 27 | 70 | 192.168.4.10 | 199.16.156.72 | 64374 | 443 | TCP |
| 0,00285 | 28 | 70 | 192.168.4.10 | 199.16.156.72 | 64374 | 443 | TCP |

Figure 6 shows the results of DDOS detection with various network density threshold values to detection accuracy. Table 3 show threshold values of 0.1 to 0.9, DDOS attacks can be detected (depicted with true positive (TP) blue lines, true negative (TN) green lines, False Negatives and False Positives). TP=91, FP=10, TN=10, FN=10.

$$Accuration = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% = \frac{91 + 10}{91 + 10 + 10 + 10} \times 100\% = 83{,}4\%$$

The accuracy obtained is 83.4% from the simulation and different targets with a threshold value of 1.00. Because of the difference in determining the threshold value and the number of simulations carried out in this research. By applying entropy as a detection method, mitigation schemes using I-DM can be performed against attacks on a single host or subnet in a network.

## IV. Conclusions

Based on the results of testing, processing, and data analysis that has been carried out, this research concludes that the analysis and identification using the I-DM scheme in the SDN feature in managing, unifying, and programming networks using SDN controllers comprehensively from existing DDoS detection techniques and comparing them according to the criteria specified in the research shows an accuracy of 83.4%. Furthermore, this research is the first to classify several DDoS detection approaches based on the techniques and features used, the nature of thresholds and locations where the approach has been applied in the SDN environment, until finally this research shows success in identifying and handling DDoS attacks. Suggestions for the further research are to focus on protecting the filtering of attacks on the network so as to facilitate the handling then the development of advanced methods that can detect and mitigate when flooding occurs in one of the traffic.

## Acknowledgment

Thanks to Universiti Tun Hussein Onn Malaysia which has supported in foreign collaborative research and the second to Badan Riset dan Inovasi Nasional and third to Universitas Nasional for their assistance. Hopefully there will be the novelty of joint research in further research.

## References

[1] Ubedilah, S. Budiyanto, and L. M. Silalahi, "Analysis QoS VoIP using GRE + IPSec Tunnel and IPIP Based on Session Initiation Protocol," in *2022 5th International Conference of Computer and Informatics Engineering (IC2IE)*, 2022, pp. 47–54. doi: 10.1109/IC2IE56416.2022.9970120.

[2] E. Darmawan, S. Budiyanto, and L. M. Silalahi, "QoS Analysis on VoIP with VPN using SSL and L2TP IPSec Method," in *2022 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*, Nov. 2022, pp. 130–136. doi: 10.1109/COMNETSAT56033.2022.9994572.

[3] R. Muwardi, H. Gao, H. U. Ghifarsyam, M. Yunita, A. Arrizki, and J. Andika, "Network Security Monitoring System Via Notification Alert," *J. Integr. Adv. Eng.*, vol. 1, no. 2, pp. 113–122, 2021.

[4] S. Budiyanto, K. N. Nahampun, F. A. Silaban, L. M. Silalahi, and R. Fajar, "Optimalisasi Private Cloud Storage Berbasis Devstack Guna Meningkatkan Performansi Network Function Virtual," *TELKA-Jurnal Telekomun. Elektron. Komputasi dan Kontrol*, vol. 6, no. 1, pp. 1–9, 2020.

[5] S. Budiyanto, L. M. Silalahi, F. A. Silaban, R. Muwardi, and H. Gao, "Delivery of Data Digital High Frequency Radio Wave Using Advanced Encryption Standard Security Mechanism," *Proc. - 2021 Int. Semin. Intell. Technol. Its Appl. Intell. Syst. New Norm. Era, ISITIA 2021*, pp. 386–390, 2021, doi: 10.1109/ISITIA52817.2021.9502262.

[6] S. Budiyanto and I. Pratama, "Classification of Network Status in Academic Information Systems using Naive Bayes Algorithm Method," in *2020 2nd International Conference on Broadband Communications, Wireless Sensors and Powering (BCWSP)*, 2020, pp. 107–112. doi: 10.1109/BCWSP50066.2020.9249398.

[7] Rachit, S. Bhatt, and P. R. Ragiri, "Security trends in Internet of Things: a survey," *SN Appl. Sci.*, vol. 3, no. 1, p. 121, 2021, doi: 10.1007/s42452-021-04156-9.

[8] Z. D. A. and M. A. A. ugli, "Network Security Issues and Effective Protection Against Network Attacks," *Int. J. Integr. Educ.*, vol. 4, no. 2, pp. 79–85, 2021, doi: 10.31149/ijie.v4i2.1204.

[9] W. Hu, C.-H. Chang, A. Sengupta, S. Bhunia, R. Kastner, and H. Li, "An Overview of Hardware Security and Trust: Threats, Countermeasures, and Design Tools," *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 40, no. 6, pp. 1010–1038, Jun. 2021, doi: 10.1109/TCAD.2020.3047976.

[10] W. Zhao and L. Yi, "Research on the evolution of the innovation ecosystem of the Internet of Things: A case study of Xiaomi(China)," in *8th International Conference on Information Technology and Quantitative Management, ITQM 2020 and 2021*, L. Y., S. Y., S. Y., W. Y., E. D., B. D., T. J., L. J., and T. Y., Eds., XiDian University, No. 266, Xinglong Section, Xifeng Road, Xi'an, 710126, China: Elsevier B.V., 2021, pp. 56–62. doi: 10.1016/j.procs.2022.01.008.

[11] D. Tang, Y. Yan, S. Zhang, J. Chen, and Z. Qin, "Performance and Features: Mitigating the Low-Rate TCP-Targeted DoS Attack via SDN," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 428–444, Jan. 2022, doi: 10.1109/JSAC.2021.3126053.

[12] Y. Wang *et al.*, "Parallel Hospital: ACP-Based Hospital Smart Operating System," in *2021 IEEE 1st International Conference on Digital Twins and Parallel Intelligence (DTPI)*, Jul. 2021, pp. 474–477. doi: 10.1109/DTPI52967.2021.9540207.

[13] R. M. A. Ujjan, Z. Pervez, K. Dahal, A. K. Bashir, R. Mumtaz, and J. González, "Towards sFlow and adaptive polling sampling for deep learning based DDoS detection in SDN," *Futur. Gener. Comput. Syst.*, vol. 111, pp. 763–779, 2020, doi: https://doi.org/10.1016/j.future.2019.10.015.

[14] P. Wang, L. T. Yang, X. Nie, Z. Ren, J. Li, and L. Kuang, "Data-driven software defined network attack detection : State-of-the-art and perspectives," *Inf. Sci. (Ny).*, vol. 513, pp. 65–83, 2020, doi: https://doi.org/10.1016/j.ins.2019.08.047.

[15] Z. A. El Houda, A. Hafid, and L. Khoukhi, "BrainChain - A Machine learning Approach for protecting Blockchain applications using SDN," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Jun. 2020, pp. 1–6. doi: 10.1109/ICC40277.2020.9148808.

[16] L. M. Silalahi, V. Amaada, S. Budiyanto, I. U. V. Simanjuntak, and A. D. Rochendi, "Implementation of auto failover on SD-WAN technology with BGP routing method on Fortigate routers at XYZ company," *Int. J. Electron. Telecommun.*, pp. 5–11, 2024.

[17] I. U. V. Simanjuntak, Heryanto, A. D. Rochendi, and L. M. Silalahi, "Simulation and Analysis of Link Failover Using Routing Border Gateway Protocol (BGP) Multi- Protocol Label Switching (MPLS) Networks," in *2023 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)*, Nov. 2023, pp. 341–346. doi: 10.1109/ICRAMET60171.2023.10366652.

[18] L. M. Silalahi, S. Budiyanto, F. A. Silaban, H. B. H. Sitorus, A. D. Rochendi, and M. F. Ismail, "Analysis of the effectiveness of online electronic learning system using data traffic network performance management to succeed merdeka learning--Merdeka campus during the Covid-19 pandemic," *Int. J. Electron. Telecommun.*, vol. 67, no. 4, pp. 595–601, 2021.

[19] J. Cui, M. Wang, Y. Luo, and H. Zhong, "DDoS detection and defense mechanism based on cognitive-inspired computing in SDN," *Futur. Gener. Comput. Syst.*, vol. 97, pp. 275–283, 2019, doi: https://doi.org/10.1016/j.future.2019.02.037.

[20] Z. Abou El Houda, A. S. Hafid, and L. Khoukhi, "Cochain-SC: An Intra- and Inter-Domain Ddos Mitigation Scheme Based on Blockchain Using SDN and Smart Contract," *IEEE Access*, vol. 7, pp. 98893–98907, 2019, doi: 10.1109/ACCESS.2019.2930715.

[21] Y. Cao, H. Jiang, Y. Deng, J. Wu, P. Zhou, and W. Luo, "Detecting and Mitigating DDoS Attacks in SDN Using Spatial-Temporal Graph Convolutional Network," *IEEE Trans. Dependable Secur. Comput.*, vol. 19, no. 6, pp. 3855–3872, Nov. 2022, doi: 10.1109/TDSC.2021.3108782.

[22] B. M. Rahal, A. Santos, and M. Nogueira, "A Distributed Architecture for DDoS Prediction and Bot Detection," *IEEE Access*, vol. 8, pp. 159756–159772, 2020, doi: 10.1109/ACCESS.2020.3020507.

[23] I. A. Valdovinos, J. A. Pérez-Díaz, K.-K. R. Choo, and J. F. Botero, "Emerging DDoS attack detection and mitigation strategies in software-defined networks: Taxonomy, challenges and future directions," *J. Netw. Comput. Appl.*, vol. 187, p. 103093, 2021, doi: https://doi.org/10.1016/j.jnca.2021.103093.

[24] D. Erhan and E. Anarim, "Hybrid DDoS Detection Framework Using Matching Pursuit Algorithm," *IEEE Access*, vol. 8, pp. 118912–118923, 2020, doi: 10.1109/ACCESS.2020.3005781.

[25] R. F. Fouladi, O. Ermiş, and E. Anarim, "A Novel Approach for distributed denial of service defense using continuous wavelet transform and convolutional neural network for software-Defined network," *Comput. Secur.*, vol. 112, p. 102524, 2022, doi: https://doi.org/10.1016/j.cose.2021.102524.

[26] G. Atharvan, S. Koolikkara Madom Krishnamoorthy, A. Dua, and S. Gupta, "A way forward towards a technology-driven development of industry 4.0 using big data analytics in 5G-enabled IIoT," *Int. J. Commun. Syst.*, vol. 35, no. 1, 2022, doi: 10.1002/dac.5014.

[27] Q. Tian and S. Miyata, "A DDoS Attack Detection Method Using Conditional Entropy Based on SDN Traffic," *IoT*, vol. 4, no. 2, pp. 95–111, 2023, doi: 10.3390/iot4020006.

[28] A. Shahar, Y. Alfassi, and D. Keren, "Communication Efficient Algorithms for Bounding and Approximating the Empirical Entropy in Distributed Systems," *Entropy*, vol. 24, no. 11, 2022, doi: 10.3390/e24111611.

[29] S. J. Siriyapuraju, V. S. Gowri, S. Balla, M. K. Vanika, and A. Gandhi, "DoS and DDoS attack detection using Mathematical and Entropy Methods," in *2023 2nd International Conference on Paradigm Shifts in Communications Embedded Systems, Machine Learning and Signal Processing (PCEMS)*, Apr. 2023, pp. 1–6. doi: 10.1109/PCEMS58491.2023.10136042.