INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN COMPUTER SCIENCE AND ENGINEERING TECHNOLOGIES

ISSN : 2454-9924

EDDA: ENTROPY-BASED EARLIER DETECTION OF DISTRIBUTED DENIAL OF SERVICE ATTACK IN SDN IOT

[*1] Mrs.I.Varalakshmi, [2]Dr.M.Thenmozhi
[1] *Research Scholar, Puducherry Technological University, Puducherry*
[2] *Associate Professor, Puducherry Technological University, Puducherry*
[3]*B.Tech, Department of Computer Science and Engineering, Puducherry*
[1]*varalakshmi.mka@gmail.com*
[2]*thenmozhi@ptuniv.edu.in*

ABSTRACT

Internet of things is a booming technology in this digital world. Because of IoT devices communicate and share data smartly through the internet and achieve enormous accuracy of data exchanging and control the devices smartly. Because of the heterogeneous environment and wireless data transmission, security plays a major role in this cyber era. Huge numbers of ransomware, malware, attacks, and threats are affecting the regular transmission. Among these, DDoS attacks are the most well-known security issues in the current Cyber world. The intruder sends frequent flooding attacks to the server/machine in the IoT environment. DDoS attacks spread large numbers of agents to flood massive amounts of flooded requests to the targeted server through this, the attackers disrupt the genuine user requests. This is challenging in the IoT environment because of the high volume of traffic occurring in the network. The proposed algorithm Earlier DDoS Detection algorithm (EDDA) detects the DDoS attacks and their types like TCP, UDP, and ICMP SYN Flood using an entropy-based method to achieve high accuracy and reduce the computing power of the algorithm to offload part of the detection tasks in Southbound Interface of SDN. The proposed algorithm is incorporated in the SDN_RYU Controller; the controller blocks the abnormal traffic/packets based on parameters like SIP, DIP, time interval, and entropy. The attack-detected packets are dropped immediately by the Controller and the flow table is updated immediately for each entry. Simulation results of the detection of the DDoS SYN flood and ICMP attacks are discussed in this paper. This system effectively detects the abnormal traffic flow and reduces the communication overhead as well as the detection rate of attacks and the result comparison is also discussed.

Keywords: SDN-IoT, EDDA, UDP, TCP, Entropy, DDoS, RYU controller

A. INTRODUCTION

IoT technology helps to communicate smarter and transfer data between the sensors, and devices through the internet. Smart IoT devices make the industry smarter and faster data transfer in all organizations/industries. But as the number of devices grows, it becomes challenging to handle the attacks occurring in the IoT environment. Among the various attacks on IoT, DDoS attack is one of the prominent attacks that are prevailing. [6] In DDoS attacks, the victims are infected with larger

ISSN : 2454-9924

numbers by the attacker through the zombies. Sarah Coble, a North American news writer states that DDoS attacks is launched by 2.9 million in the first quarter of 2021 from NETSCOUT ATLAS [27] Security Engineering & Response Team (ASERT) research center. Not clear. Recent studies indicate that the use of Software Define Network (SDN) provides an efficient way to manage IoT devices. GitHub was hit by a DDoS attack on 28th February 2018 with a volume of 1.35Tbps and lasted around 20 minutes duration [28].

In 2020, The Google Threat Analysis Group (TAG) published a blog update on October 16 about how threats and threat actors adapt their strategies considering the 2020 presidential election [Cybercrime Report]. Massive DDoS attacks were launched against Amazon Web Services in February 2020, the 800-pound gorilla of cloud computing. Incredibly, the attack peaked at 2.3 gigabytes per second over three days.

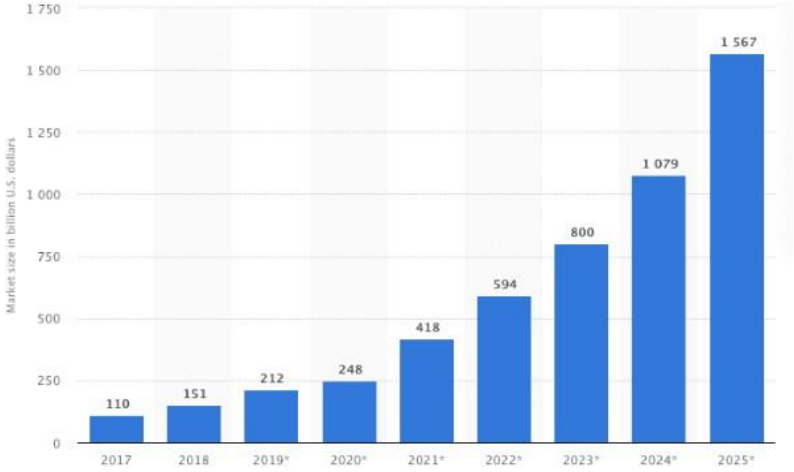


Fig.1. Global Market of IoT

The increase of IoT devices connected to the network is increasing year by year and shared data through the internet. The growth of IoT devices in the global market is shown in Fig.1. From 2017 to 2023 connected devices are increased drastically and it is expected to reach 1567 billion in 2025. The Fig.2. shows the increase in DDoS attack traffic in the global market. According to the report, the impact increased to 14% in the last quarter of 2023. The global market expects 25 million IoT devices to be affected by DDoS attacks in 2025.

Recently The Kremlin's cyberspace strategy has comprised a combination of denial-of-service attacks, and data wipes in the run-up to and during the initial stages of Russia's invasion of Ukraine. Several attempts to DDoS Russia in response were made this week, with varying degrees of success. [11] The websites of the Russian government, the Army, and bankers have all endured traffic tsunamis, although they generally seem to be holding firm.

ISSN : 2454-9924

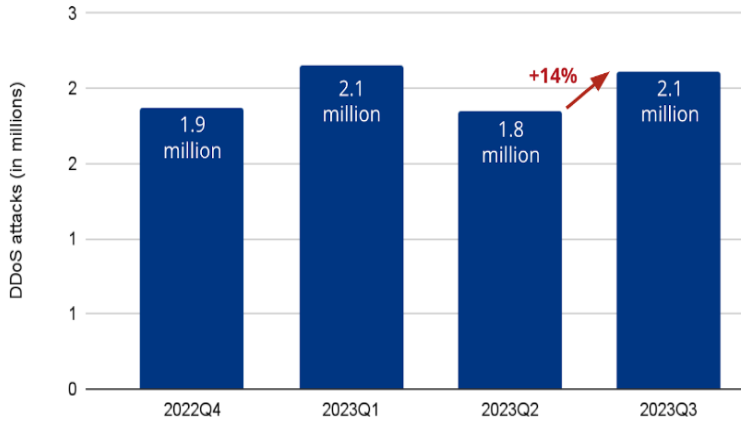


Fig.2.Impact of DDoS attack in the real world

To improve network programming and flexibility, SDN separates the two layers (control and data plane) from the traditional way. The SDN OpenFlow switches are presented in the data plane and are responsible for the data forwarding controlled by the control plane. The centralized controller present in SDN is a danger for DDoS attacks. The attackers flood a huge volume of requests to the victim and the SDN switch fails to match the flow of many packets and that information is sent to the controller as a packet. So, DDoS flow and original flow combination are run out through the controllers and switches [3] this will make the controller not responsible for the new packets and slow down the SDN framework.

B. RELATED WORKS

The low-rate DDoS attack against the data link layer was detected by using the FM algorithm [4]. The proposed structure is built with a C-DAD (Counter-based DDoS Attack detection) framework on the peak of the SDN WISE framework to detect and analyze DDoS attacks [5]. The counter-based function has a flow monitor and detects the DDoS attack in the SDN-IoT network. The C-DAD algorithm and framework are tested with different parameters and achieve a high detection rate.

In [6], they have used the KNN technique for DDoS attack detection in the low time with high accuracy. Since the possibility of flooding of bot into the network is high. KFNN technique is proposed to detect the DDoS attack in SDN [7]. Here high efficiency and accuracy are produced for limited networks and get reduced if the number of resources increases.

The [8] concentrated on a global controller using a cloud that is connected to a local controller. Each sub-network has a local controller. The LEDEM is divided into three parts: data capture, DDoS attack detection, and mitigation. The OpenFlow switch enables the passing of the entire IoT traffic.

ISSN : 2454-9924

The local controller ensures that the traffic is either forwarding or dropping. It extracts the necessary feature from the packet and gives it as an input for the detection of DDoS attacks. SDELM is used to detect abnormal traffic and the output of SDELM marks the malicious packet.

Detection [9] of DDoS attacks is achieved using multilevel auto-encoder-based feature learning. This technique employs multilevel features using multiple kernel learning (MKL) algorithms that exploit the deep auto-encoder and shallow learning in unsupervised behaviour. Here, the first learning is carried out at the multilevel of the deep autoencoder and shallow in an unsupervised manner from the training data. Next, features are generated for every training data by encoding them. Next, the feature is projected to the kernel space so it can automatically combine with the weighted fashion using an MKL algorithm. The final detection model was achieved using data classification between the infected and non-infected nodes during testing time.

In [10] four methods are used to detect the attack, for a new connection the controller checks the switch flow so that the incoming packet will reach the destination host. The controller collects the statistics from the table and monitors the existing flow and it is detached if it is inactive for more than a minute. The window size should be <= no. of hosts for calculating the accurate entropy. They also tested the entropy with three different window sizes to measure the CPU and memory usage. To detect the attack, the destination IP is monitored for the new incoming packet and a new hash table is created for the incoming packet.

C. PROPOSED METHODS

SYSTEM ARCHITECTURE USING SDN

Software Defined Network (SDN) is a promising architecture that allows groups to be wise and centrally programmable using software solutions. Fig[3] shows the three layers of SDN architecture. The application layer handles data and business applications, which get data through northbound interfaces from the control plane. The Control plane consists of controllers, a traffic analyzer, and a security provider carrying out the major works. The lower layer is called the Data plane/Infrastructure layer which contains various IoT devices and hosts that gather data from those devices and transfer them to the control plane using a southbound interface.

The proposed framework is designed using an SDN_RYU controller and OpenFlow switches with a huge number of hosts connected to it. OpenFlow Switches are administered by the single RYU controller shown in Fig. 3. SDN switches (OpenFlow switches) are equipped with minimum CPU utilization with less computational resources for intense traffic by utilizing the calculating capability

ISSN : 2454-9924

of switches to take part in the detection process.

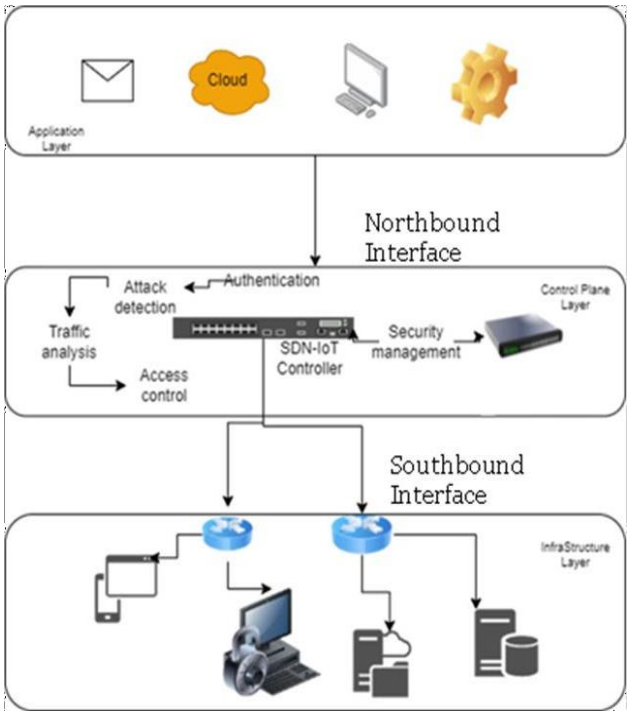


Fig.3.SDN Layered Architecture

The RYU controller is placed in the control plane layer. The proposed algorithm is incorporated in the controller to achieve earlier DDoS attack detection through a continuous update of flow entry and to verify with the last consecutive time interval. For each entry in the network, entropy is calculated for the Source_IP address along with the threshold value.

DETECTION OF DDOS ATTACK BASED ON ENTROPY IN CONTROL PLANE

In SDN_IoT, the controller concentrates on directing the switches to forward the packets and update the OpenFlow table which includes multiple entries. The detection algorithm should reduce the complexity of abnormal activity which is used to measure the traffic characteristics. The entropy method is proposed, because of the distribution of randomness of the flooding request, and earlier detection achieves high accuracy, by continuously monitoring the incoming packets Source_IP address and threshold parameter. The proposed algorithm EDDA calculates the entropy of each Source_IP and Destination_IPs randomness.

The proposed algorithm calculates the time interval for each packet and increases the packet counter in the flow_table. This will reduce the overload of the controller and achieve less computation. Here the entropy is initialized with the value of 0.6 and the threshold value is 0.2 and the proposed algorithm explains that Source_IP, Destn_IP, SRC_port, and DST_port, must maintain to

ISSN : 2454-9924

monitor the SYN requests directed to a destination counter.

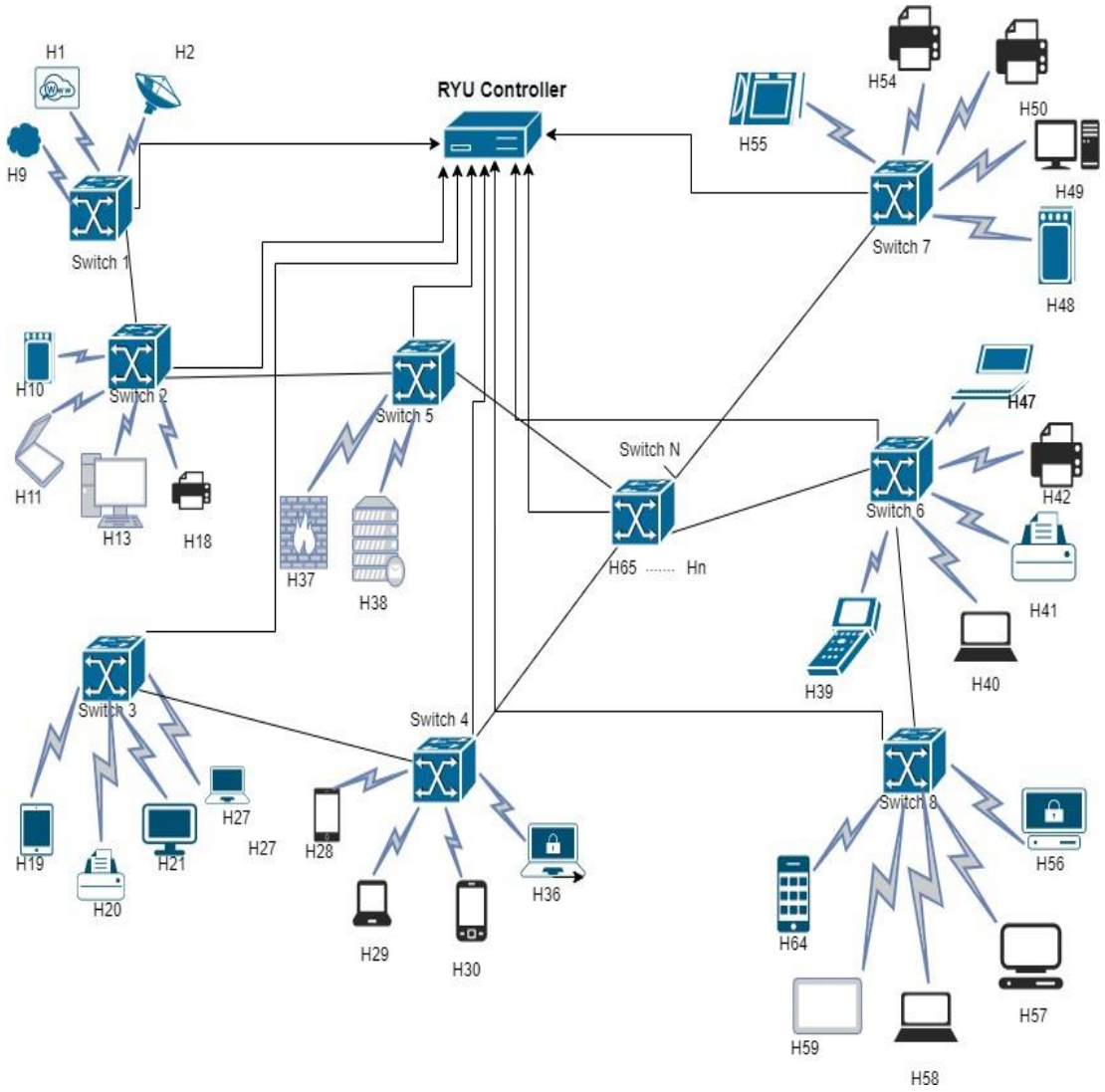


Fig.3. SDN_IoT Testbed Setup

After a certain time (Δt), the entropy of the network is calculated using Destn_IP and evaluate threshold value. If the measured entropy is less than the threshold, then a variable Counter will be incremented. If a violation of window entropy is found to be certain consecutive times ($H_t(x)$), it gives a message as an attack detected and drops the packet to the OpenFlow switch.

To compute the entropy, Eqn (1) illustrates a window, where x_i (x_1, x_2, x_3,…..x_n) is the random variable, and y_i (y_1, y_2, y_3,…..y_n) represents its frequency.

$$W=f\ \{(x_1, y_1), (x_2, y_2), (x_3, y_3),..(x_n, y_n)\}$$

ISSN : 2454-9924

Eqn (2) is used where $P(x_i)$ denotes the probability of occurrence of each random variable in the set. $P(x_i) = y_i/N$ (2)

where in Eqn (3) N represents the number of IPs entered the network and its outcome. The entropy of a discrete random variable (X) that is present in a system is defined as

$$N=y_1+y_2+y_3+\ldots\ldots + y_n \quad (3)$$

where n represents unique number of outcomes (i.e., unique destination IP). Since the entropy falls in the range of $[0, \log_2 n]$, it will vary based on the window lengths. Therefore, sets the entropy in the range of [0, 1], which is not dependent of the size of the window.

$$(X) = \sum_{i=0} (x_i)_2 P(x_i) \quad (4)$$

Algorithm for Earlier Detection of DDoS attack using Entropy method

Input: Source_IP, Destn_IP, Sport, Dport, Δt
Output: Abnormal traffic
Procedure: Detection of abnormal traffic in Switch
Initialize : H(X), Pc, δ
Start:
    Read Source_IP, Destn_IP,Sport,Dport
    Pc++
  Delay Δt
For i=1 : n do
    Calculate H(X), Pc
    goto start
End for
    If (Ht(X) –H(X)> δ)
        Report abnormal
        Update flow_tab
    else
      goto start
    end if
End procedure

In other case, greater the entropy (in case of normal traffic statistics) than the threshold, then the window entropy will consider as normal value and forward to entropy list H(X), where i is an instance of normal entropy. Fig.4 shows the structure of the openflow table that updates the flow entry and illegitimate packet entered the network. The packet enters the network in the data plane layer and the OpenFlow table updates the packet count and time interval for the forwarding packets and calculates

ISSN : 2454-9924

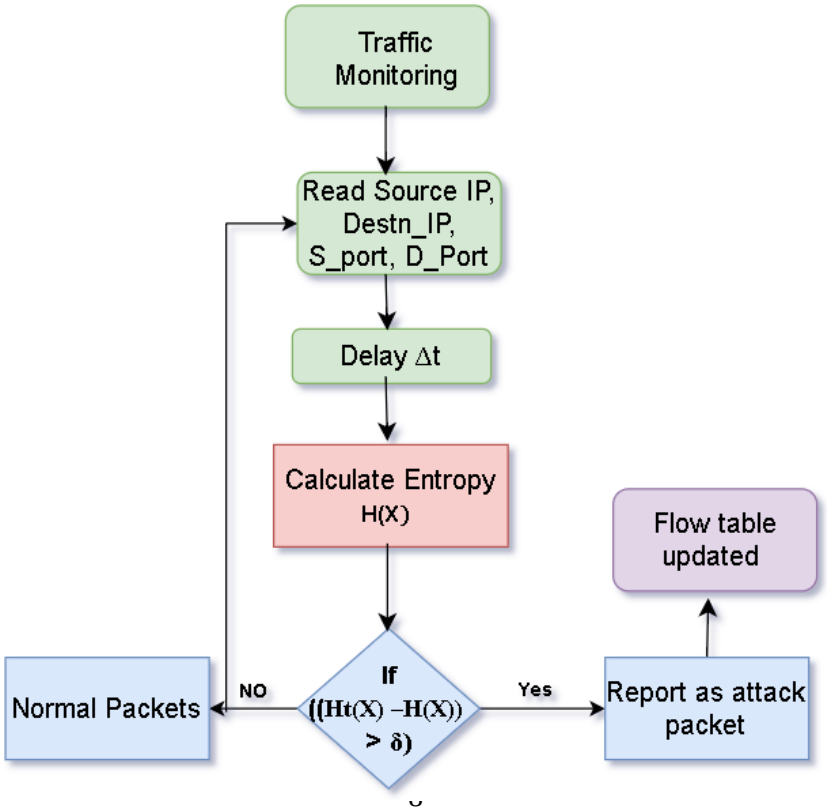
the entropy using the destination IP. It detects as abnormal when the same packets flow into the network multiple times using Δt and $H_t(X)$. The structure of the OpenFlow switch table is shown in Fig.4

Source_IP	Destn_IP	Packet_Count	Time_in	Time_out	No.of Occurrence

Figure 4. Structure of openflow table

D. EXPERIMENTAL SETUP

The performance of the detection framework is estimated based on SDN; NS 3.25 is used to build the SDN framework as shown in Fig.4. And the test bed setup for hardware parameters is as follows: RYU controller is used for emulating the links and switches on a single Linux kernel (Ubuntu 16.04 LTS 64-bit) using process-based virtualization. Openflow switches version 1.5.19 (s1 to s9) is increased based on network size for each simulation it forwards the packets in the SDN environment based on the flow table (depending on the traffic). OpenFlow switches make the communication to an external RYU controller. Here 150 hosts relate to 9 switches and the total number of packets is 49786. The detection algorithm is called and then topology is created using a simulator by specifying the number of nodes and switches, then the traffic is generated, which runs as an external command by providing (Linux Terminal ./waf –run filename) it simulates the attack and detection part in the terminal.



ISSN : 2454-9924

Fig.5 Flow Diagram for Abnormal Detection

Based on the literature statistics, the entropy value of the backdrop traffic is about 0.8 for normal traffic occur in network. Hence, our system set H(X) = 0.6 and threshold= 0.2 in the simulation. The experiment was carried out for the most common attacks of DDoS called UDP and SYN flood attacks. The above fig.5 shows the working flow of detection of DDoS attack using the proposed EDDA algorithm. Initially monitor the traffic entered the network and read the incoming packets Source_IP, Destn_IP address and forwards the packet into the network towards the destination along with Delay time of each packet. The entropy is calculated for each IP to measure randomness of the distribution and measure the entropy with previous entropy value, If the value is greater than the threshold abnormal packets flooded into the network and are reported as attack packets in the OpenFlow table. All detected and suspected packets are stored in the Openflow table. Otherwise, proceed the same for the remaining packets to flow into the network.

E. COMPARISON AMONG VARIOUS ALGORITHMS

The simulations and evaluations were performed on the SDN platform among various algorithms and compared with the proposed EDDA algorithm for normal and abnormal traffic flow. The goal is to determine the normal entropy values for the network consisting of 81 nodes. Traffic was launched on the SDN network with a traffic interval of 0.5 seconds. The traffic_rate is calculated by (1/.35=3packets/sec). Table 1. Shows the traffic rate for UDP, TCP, and ICMP SYN flood were attacked for multiple scenarios. Here, we simulated an attack rate of 25% and increased drastically to evaluate the network performance and CPU utilization.

Simulation Traffic	**Normal**	**Multiple Victims at 25% attack rate**	**Multiple Victimsat 50% attack rate**
Packet Type	***TCP***		
Time Interval	0.25 sec	0.06 sec	0.125-sec
Traffic Rate	4 packets/sec	25 Packets/sec	50 Packets/sec
Packet Type	***UDP***		
Time Interval	0.35sec	0.087 sec	0.175-sec
Traffic Rate	3 packets/sec	24 Packets/ sec	50 Packets/ sec

Table 1. Traffic Flow

ISSN : 2454-9924

The traffic rate of TCP and UDP is calculated based on the time and number of abnormal traffic flow into the network. The attack interval rate is defined by (abnormal = time * % of attacks) with a rate of 25 packets/sec for TCP and 24 packets/second for UDP with the time interval of 0.06 sec (TCP) and 0.087sec (UDP) and increased drastically through the proposed EDDA algorithm and achieves efficiency.

For attack traffic with multiple victims is set as .06sec. To generate the same count of attack traffic, the attack is detected at 1.007sec for 25% of attacks and increased to 1.25sec for 50% of attacks. It is safe to say the proposed Entropy-based DDoS detection (EDDA) algorithm performs very well in distinguishing between normal and abnormal attack traffic.

F. PERFORMANCE ANALYSIS

Here, we analyze the result of attack traffic in a simulated environment for various algorithms like FADA [2], DPPC [6], and DDA [10] along with the proposed EDDA algorithm and show the results in Fig.6. The FADA [2] algorithm produces 83% of accuracy with the window size of 80. The time interval between the packets is fixed as 1sec and entropy as 0.8 with threshold = 0.2 and the detection time of the attack is 1.07sec. DPPC algorithm produces 82% accuracy for the window size of 49. Here the time to detect an attack is monitored as 1.037sec. DDA [10] algorithm produces 75% accuracy for the window size of 33 and detects the attack with a time interval of 5.14sec.

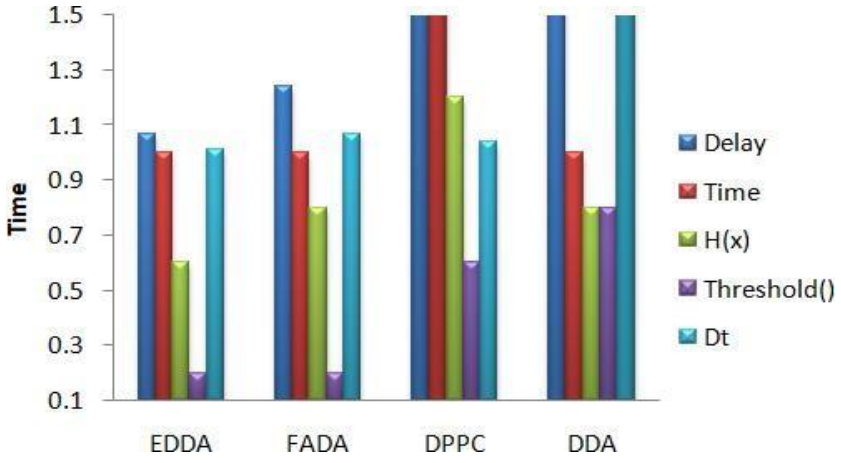


Fig.6. Comparison between Various Algorithms

The proposed EDDA algorithm detects the abnormal traffic in 1.011sec for the maximum window size of 100 and achieves the best accuracy of 83% with fewer delay Δ_t=1.07ms. Hence the proposed algorithm is well suited and produces the best performance analysis for the maximum window size of 100. The average detection rate of a DDoS attack is determined within one to two seconds shown in

ISSN : 2454-9924

Fig.7

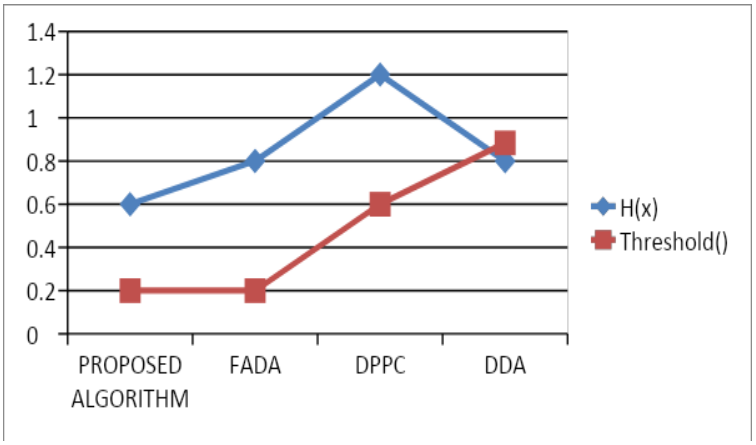


Fig.7. Comparison graph for threshold and entropy

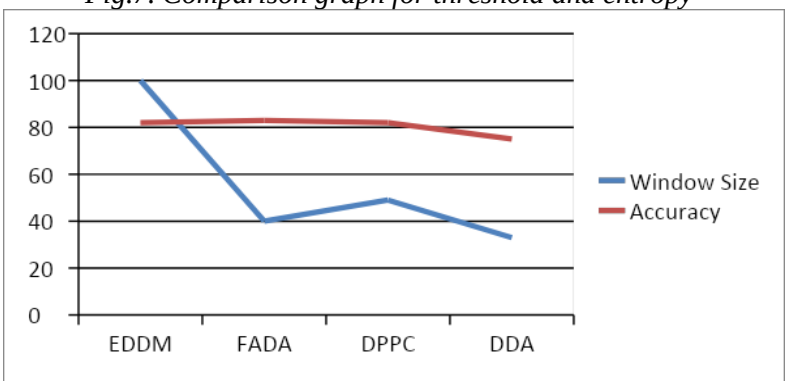
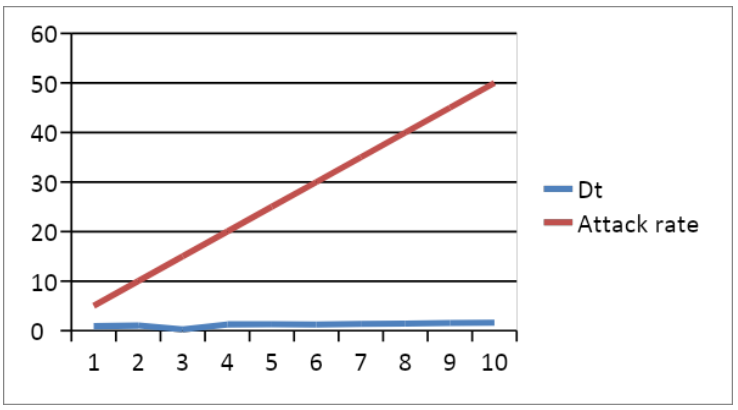


Fig.8. Accuracy Graph for Abnormal flow

Fig.9. shows the result analysis of attack detection of the various duration. The D_t =0.9 for 5 attacks in the window size of 50, increase the attack count to 50 with a delay of 8ms the proposed EDDA algorithm produces the detection time =1.23ms.



ISSN : 2454-9924

Figure.9. *Detection time of abnormal flow*

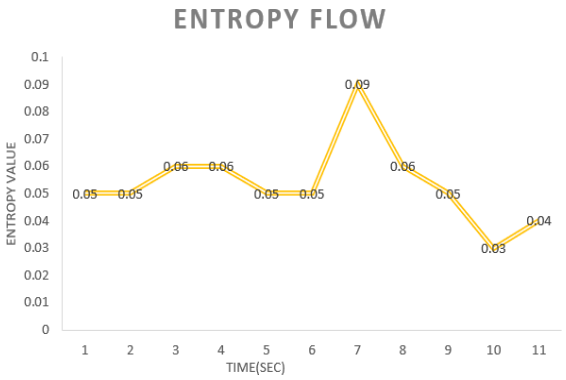


Fig.10. Graph of Proposed Entropy Calculation

The number of packets received by the RYU controller; the rate of detection rate for abnormal flow is less when compared and evaluated with existing algorithms shown in the graph. The proposed EDDA reduces the CPU load utilization from the range of 48% to 52% to 200 nodes. The rate ofattack flow is increased in each runtime by 30%.

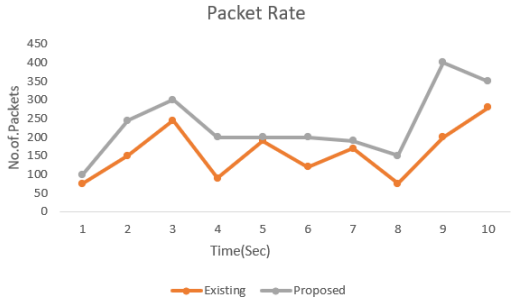


Figure.10 Graph for packet flow count

CONCLUSION

In this paper, we experiment detection of attacks by comparing with various algorithms, and the results along with the proposed algorithm are discussed. Our test bed setup consists of an RYU controller and OpenFlow switches with IoT terminal devices. The proposed algorithm detects the DDoS attacks in the SDN_IoT framework, and obtains entropy, and threshold value with the packet rate of all nodes in the OpenFlow switches; the threshold is used to determine the attack detection. The simulation results show that the proposed algorithm finds the illegitimate system/devices from

ISSN : 2454-9924

which a DDoS attack is launched within a minimum time. The proposed algorithm EDDA clattered based on: traffic features, Source_IP and Destn_IP together, along with TCP flags. And calculate the entropy efficiently by measuring the degree of randomness received by the SDN Controller. The existing system produced 83% accuracy using 3 switches and 50 hosts, this accuracy will be reduced when the number of hosts increases. The proposed system produces 90.6% accuracy for detecting TCP-SYN flood DDoS attacks with 9 switches and 64 hosts. Our Further work will increase the number of nodes and produce the same accuracy for detecting the DDoS (Distributed Denial of Service) SYN flood attacks like (TCP, UDP, ICMP SYN flood, etc.,) using the SDN_IoT experimental setup and will achieve a good detection rate in real setup environment.

REFERENCES

1. Aldaej, A. (2019). Enhancing cyber security in modern internet of things (IoT) using intrusion prevention algorithm for IoT (ipai). IEEE Access
2. Lianming Zhang 1, And Kun Yang 2, Da Yin1,(Senior Member, IEEE) 2018, A DDoS Attack Detection and Mitigation With Software-Defined Internet of Things Framework
3. Zekri, M., El Kafhali, S., Aboutabit, N., & Saadi, Y. (2017, October). DDoS attack detection using machine learning techniques in cloud computing environments. In 2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech) (pp. 1-7). IEEE.
4. Mocrii, D., Chen, Y., & Musilek, P. (2018). IoT-based smart homes: A review of system architecture, software, communications, privacy and security. Internet of Things, 1, 81-98.
5. Bhushan, K., & Gupta, B. B. (2018). Hypothesis test for low-rate DDoS attack detection in a cloud computing environment. Procedia computer science, 132, 947-955.
6. Galeano-Brajones, J., Carmona-Murillo, J., Valenzuela-Valdés, J. F., & Luna-Valero, F. (2020). Detection and Mitigation of DoS and DDoS Attacks in IoT-Based Stateful SDN: An Experimental Approach. Sensors, 20(3), 816.
7. Cvitić, I., Peraković, D., Periša, M., & Botica, M. (2019). A novel approach for detection of IoT-generated DDoS traffic. Wireless Networks, 1-14.
8. Lima Filho, F. S. D., Silveira, F. A., de Medeiros Brito Junior, A., Vargas-Solar, G., & Silveira, L. F. (2019). Smart detection: an online approach for DoS/DDoS attack detection using machine learning. Security and Communication Networks, 2019.
9. Jaafar, G. A., Abdullah, S. M., & Ismail, S. (2019). Review of recent detection methods for HTTP DDoS attack. Journal of Computer Networks and Communications, 2019.
10. Aamir, M., & Zaidi, S. M. A. (2019). DDoS attack detection with feature engineering and machine learning: the framework and performance evaluation. International Journal of Information Security, 18(6), 761-785.
11. Mikail Mohammed Salim1 · Shailendra Rathore1 · Jong Hyuk Park1(2019) Distributed denial of service attacks and its defenses in IoT: A survey Springer, The Journal of Supercomputing
12. Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P., & Sikdar, B. (2019). A survey on IoT security: application areas, security threats, and solution architectures. IEEE Access, 7, 82721-82743.
13. Velliangiri, S., Karthikeyan, P., & Vinoth Kumar, V. (2020). Detection of distributed denial of service attack in cloud computing using the optimization-based deep networks. Journal of Experimental & Theoretical Artificial

Intelligence, 1-20

14. Syed, N. F., Baig, Z., Ibrahim, A., & Valli, C. (2020). Denial of service attack detection through machine learning for the IoT. Journal of Information and Telecommunication, 1-22.

15. Meejoung Kim,(2019). Supervised learning-based DDoS attacks detection: Tuning Hyperparameters.Electronics and Telecommunications Research Institute (ETRI) 10.4218/etrij.2019-0156

16. Kejun Chen, Shuai Zhang, Zhikun Li, Yi Zhang, Qingxu Deng, Sandip Ray, Yier Jin(2018) Internet-of-Things Security and Vulnerabilities: Taxonomy, Challenges, and Practice. Journal of Hardware and Systems Security (2018) 2:97–110

17. Andrea Chiappetta. (2017)Hybrid ports: the role of IoT and Cyber Security in the next decade Journal of Sustainable Development of Transport and Logistics Hybrid ports: the role of IoT and Cyber Security in the next decade DOI: 10.14254/jsdtl.2017.2-2.4

18. N. Ravi, S.M. Shalinie, Learning-driven detection and mitigation of DDoS attack in IoT via SDN-cloud architecture.IEEE Internet Things J. 7(4), 3559–3570 (2020)

19. Dao, N.N.; Park, J.; Park, M.; Cho, S. A feasible method to combat against DDoS attack in SDN network. In Proceedings of the 2015 International Conference on Information Networking (ICOIN),Siem Reap, Cambodia, 12–14 January 2015; pp. 309–311

20. Mousavi, S.M.; St-Hilaire, M. Early detection of DDoS attacks against SDN controllers. In Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC), Anaheim, CA, USA, 16–19 February 2015; pp. 77–81.

21. I.Varalakshmi, Dr.M.Thenmozhi, “Mitigation of DDoS attack using Machine Learning Algorithms in SDN_IoT environment” published in Design Engineering (Toronto) ISSN: 0011-9342 | the Year 2021, Issue: 8 | Pages: 4381-4390 October 2021. (Scopus-Indexed)

22. X. Liu, S. Zhao, A. Liu, N. Xiong, and A. V. Vasilakos, ``Knowledge-aware proactive nodes selection approach for energy management in the Internet of Things," Future Generate. Comput. Syst., Aug. 2017, DOI: https://doi.org/10.1016/j.future.2017.07.022

ISSN : 2454-9924

23. K. Sonar and H. Upadhyay, ``A survey: DDoS attack on Internet of Things," Int. J. Eng. Res. Develop., vol. 10, no. 11, pp. 58_63, Nov. 2014.

24. Dong, P.; Du, X.; Zhang, H.; Xu, T. A detection method for a novel DDoS attack against SDN controllers by vast new low-traffic flows. In Proceedings of the 2016 IEEE International Conference on Communications (ICC), Kuala Lumpur, Malaysia, 23–27 May 2016; pp. 1–6.

25. U. Lindqvist and P. G. Neumann, ``The future of the Internet of Things," Commun. ACM, vol. 60, no. 2, pp. 26_30, Jan. 2017.R. Huoet al., ``Software de_ned networking, caching, and computing for green wireless networks," IEEE Commun. Mag., vol. 54, no. 11, pp. 185_193, Nov. 2016.

26. Z. Qin, G. Denker, C. Giannelli, P. Bellavista, andN. Venkatasubramanian, ``A software-defined networking architecture for the Internet-of-Things," in Proc. IEEE Netw. Oper. Manage.

27. Kanagavelu, R.; Aung, K.M.M. A Survey on SDN Based Security in the Internet of Things. In Advances in Information and Communication Networks; Arai, K., Kapoor, S., Bhatia, R., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 563–577.

28. Varalakshmi, I., M. Thenmozhi, and R. Sasi. "Detection of Distributed Denial of Service Attack in an Internet of Things Environment-A Review." 2021 international conference on system, computation, automation and networking(ICSCAN).IEEE,2021.