SDN-based Intrusion Detection System for Mitigating Automated Malicious Traffic Flows

PROJECT GUIDE

Dr. V. Vetriselvi
Professor,
Department of Computer Science and Engineering,
College of Engineering Guindy,
Anna University.

TEAM 12

Srinath S - 2018103070
Sobika S - 2018103067
Lekha Shanthini R - 2018103558

INTRODUCTION

One of the most hazardous exploits capable of harming an organization's reputation and operation is a Distributed Denial of Service (DDoS) attack. The ease with which an attack can be launched has increased dramatically in recent years. The purpose of the DDoS attack is to deny legitimate users the access to services by exhausting the hardware resource or bandwidth which means servers lose their availability. This is an important part of any service's security. DDoS attacks launched at the server are difficult to detect because the attack request packet looks identical to the normal request packet. For decades, this Distributed Denial of Service (DDoS) attack has severely affected network availability, and there is still no effective protection mechanism in place. However, the emerging Software Defined Networking (SDN) technology offers a new way to rethink the defense against DDoS.

SDN differs from traditional networking in that it is software-based, whereas traditional networking is typically hardware-based. Because SDN is software-based, it provides users with more flexibility and simplicity in managing resources virtually throughout the control plane. Traditional networks, on the other hand, use switches, routers, and other physical infrastructure to link and run the network. APIs are communicated via a northbound interface on SDN controllers. Instead of using the protocols required by traditional networking, application developers can directly programme the network using this communication.
OpenFlow controller is the switch control plane that is implemented in different machines. It is used as a communication channel between the switch and the controller. The switch traffic control system acts according to the flows installed by the controller.

OVERALL OBJECTIVES

To tackle the DDoS flow to our server, we propose to:

- Integrate a feature extraction and classification application to detect any malicious flow entering the system.
- Make the system sensitive to suspicious flows, along with reducing the false positive rate as low as possible.
- Implement a defense mechanism to mitigate the attack by redirecting the malicious flow

to a CAPTCHA server.

LITERATURE SURVEY

1. **A cooperative DDoS attack detection scheme based on entropy and ensemble learning in SDN**
   Yu, S., Zhang, J., Liu, J. et al. , J Wireless Com Network 2021, 90 (2021).
   https://doi.org/10.1186/s13638-021-01957-9
   This paper employs a coarse grained entropy detection and fine grained classification using random forests and CART trained using the generated attack and normal traffic using scapy and nettich-ng. Mitigation measure is very primitive where false positives might block legit users.

2. **A New Framework for DDoS Attack Detection and Defense in SDN Environment**
   L. Tan, Y. Pan, J. Wu, J. Zhou, H. Jiang and Y. Deng, in IEEE Access, vol. 8, pp. 161908-161919, 2020, doi: 10.1109/ACCESS.2020.3021435
   This paper provides an effective classification method combining KNN and K-Means Algorithm and provides a defensive mechanism using NSL-KDD dataset but places huge burden on controller and does not scale to large scale traffic"

3. **Smart detection: an online approach for DoS/DDoS attack detection using machine learning**
   Lima Filho Francisco Sales de, Silveira Frederico A. F., de Medeiros Brito Junior Agostinho, Vargas-Solar Genoveva, Silveira Luiz F., Security and Communication Networks Hindawi, 2019
   This used a sampled dataset sourced from various benchmark datasets to make an efficient online classification system using the random forest algorithm on CIC-Dos, CICIDS2017, CSE-CIC-IDS2018 and customised dataset but does not deal with a mitigation that could deal with the malicious traffic

4. **Denial of Services Attack Detection using Random Forest Classifier with Information Gain**
   Reena Singh Rajput, Dr. Sanjay Agrawal,, International Journal for Scientific Research & Development| Vol. 4, Issue 06, 2016 | ISSN (online): 2321-0613
   This research Achieves high F-Score and accuracy using Random Forest on NSL-KDD and better feature selection using Information Gain but uses Outdated dataset and was not real time.

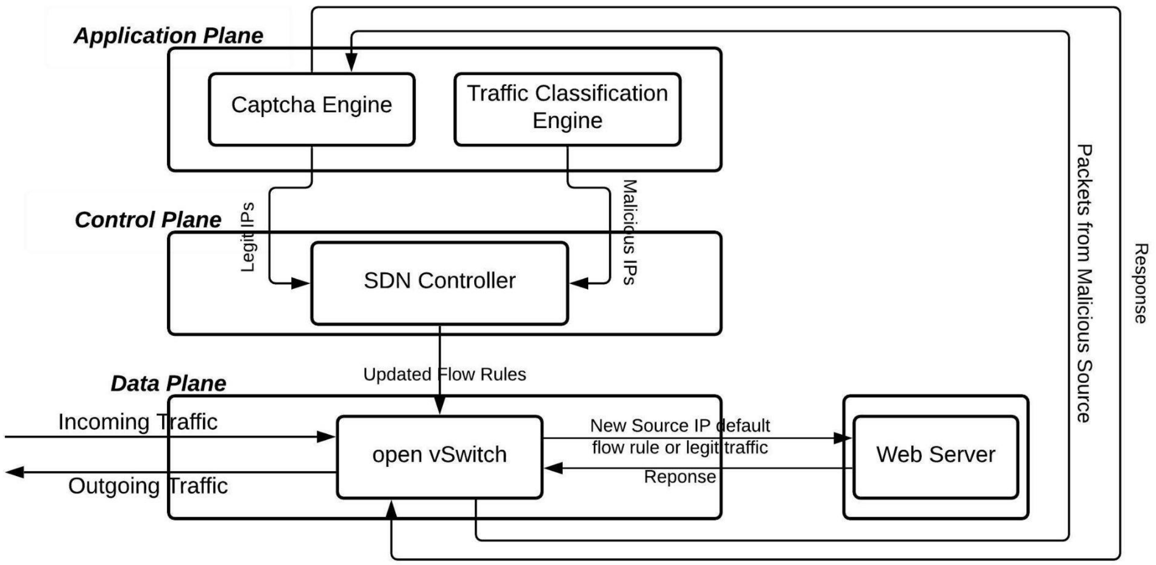
5. **Detection of DNS DDoS Attacks with Random Forest Algorithm on Spark**
   Liguo Chen, Yuedong Zhang, Qi Zhao, Guanggang Geng, ZhiWei Yan, Procedia Computer Science, Volume 134, 2018, Pages 310-315, ISSN 1877-0509
   "A novel method to reduce the DDoS traffic on TLD servers is introduced using Random Forests on Query logs collected during Water Torture attack of .CN site in Hong Kong in 2015, but the technique is limited to DNS traffic and cannot be extended to other types of traffic.

6. **Detection of Distributed Denial of Service Attacks using Machine Learning Algorithms in Software Defined Networks**
   N. Meti, D. G. Narayan and V. P. Baligar, 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 1366-1371, doi: 10.1109/ICACCI.2017.8126031.
   Compared the performance of Bayes, support vector machine and neural network methods in detecting DDoS attacks, and found the performance of SVM is the best on Real time traffic collected in Lawrence Berkeley Laboratory, but focuses only on the TCP traffic and cannot be extended to other traffic types.

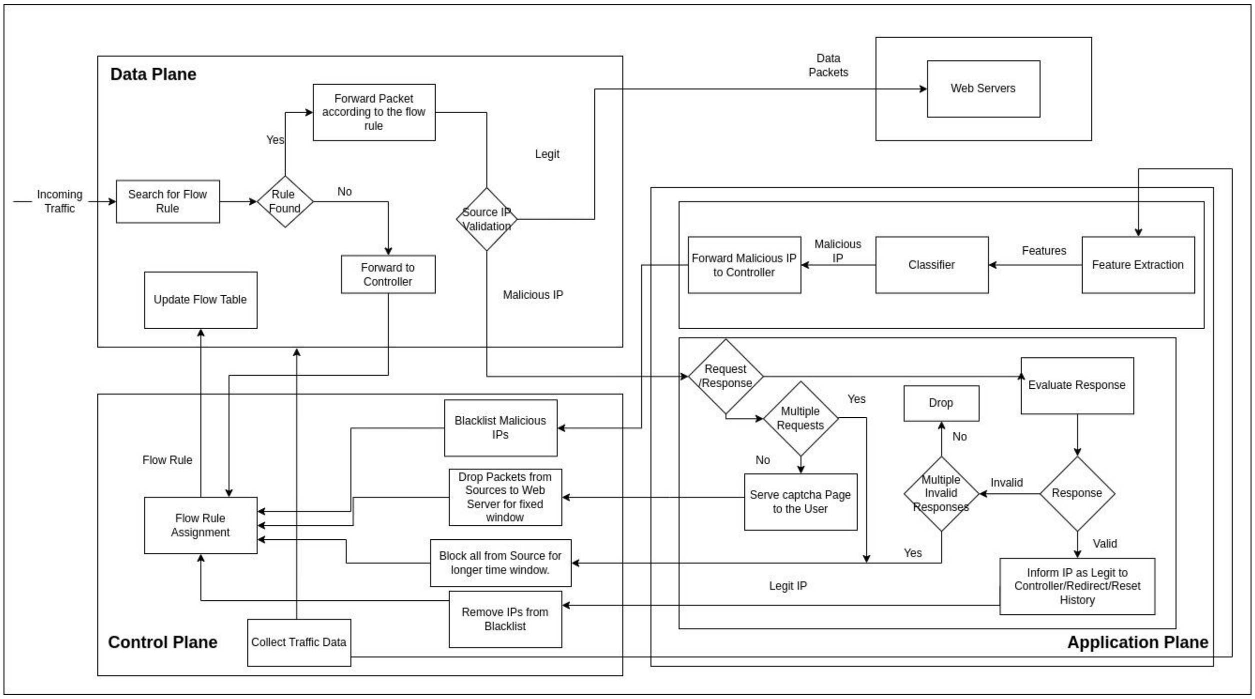
7. **Prevention of DDoS Attacks and Detection on Cloud Environment—A Review and Survey**
   Allam S.K., Prasad G.S. (2021) Intelligent System Design. Advances in Intelligent Systems and Computing, vol 1171. Springer, Singapore.
   https://doi.org/10.1007/978-981-15-5400-1_47
   This paper discusses the DDoS attack to cloud environments, and ways to tackle it and one of the most important approaches is CAPTCHA.

8. **A Survey on Secure Network: Intrusion Detection & Prevention Approaches. American Journal of Information Systems**
   Manu, B. (2016). American Journal of Information Systems, Vol. 4, No. 3,Science & Education Publishing, pp 69-88.
   This survey discusses the various attacks and intrusions an system can face and proposes CAPTCHA as one of the main mitigation measures.

ARCHITECTURE DIAGRAM



BLOCK DIAGRAM



MODULES

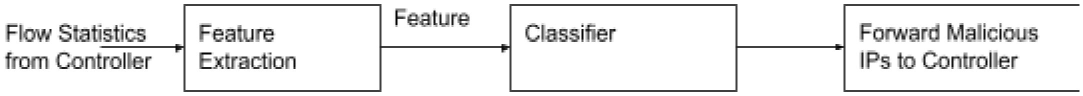
1. Feature Extraction, Training and Classification
2. CAPTCHA Engine
3. Deploying Northbound APIs
4. Flow Rule Assignment

MODULE DESCRIPTIONS

1. **Feature Extraction, Training and Classification**

Input : Flow Statistics from Controller.
Output : Malicious IP addresses

We collect the traces from the Controller's traffic monitor. Features such as packet count, byte count, flow duration are extracted by grouping the data with respect to 5-tuples (IP source, IP destination, source port, destination port and protocol type) for every ten seconds.


Flow Statistics from Controller
Feature Extraction
Feature
Classifier
Forward Malicious IPs to Controller

From these features, we have to calculate features like,

1. Packet count
2. Byte count
3. Duration of flow
4. Packet per flow
5. Bytes per flow
6. Packet rate

We propose the usage of these attributes as these exhibit higher information gain in random forest algorithms as shown in [4].

We train a random forest classifier, using the dataset put together from [17]. Then we put this classifier inside the classification engine that detects the anomalies in the flow and gives the list of malicious IPs. The list of malicious IPs are then sent out to the controller, which then installs flow rules to forward the packets to the captcha engine.

2. CAPTCHA Engine

CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) is a type of security measure known as challenge-response authentication. This is effective in differentiating actual humans from automated scripts and bots. The captcha engine in our system poses challenges to the users and evaluates their response. It also maintains a list of users who continuously fail at the challenge multiple times.

CASE - I -

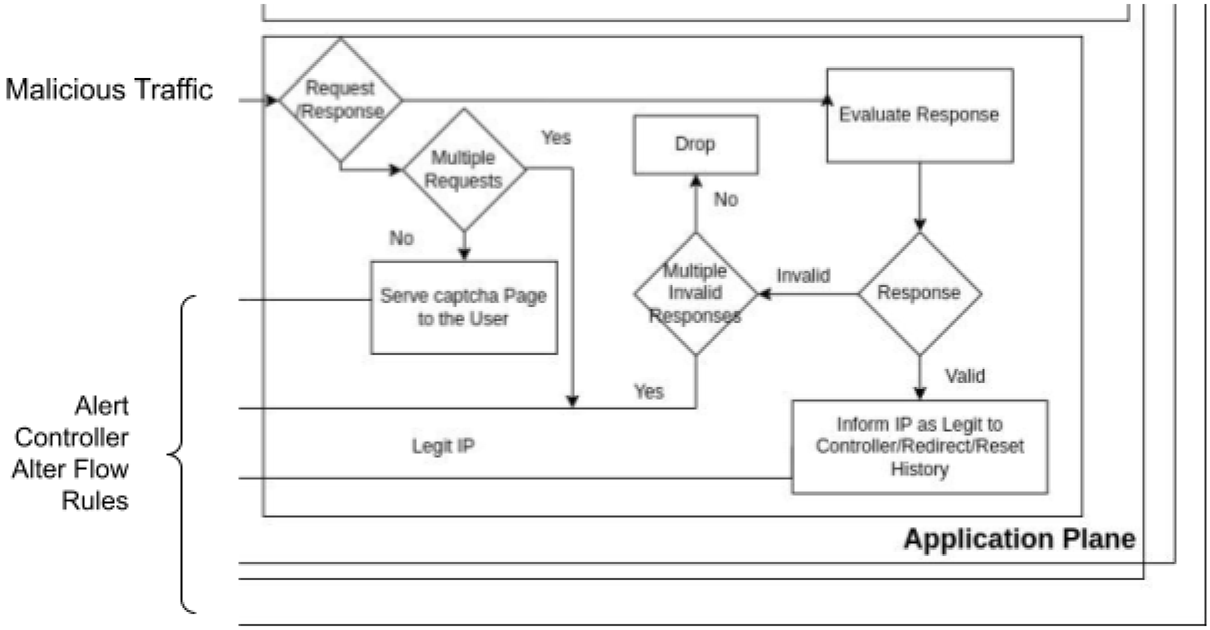
Input : Redirected source packet of malicious traffic
Output : captcha Challenge to the malicious

When a malicious user is redirected to the captcha engine first, the engine sends out a challenge to the user and informs the controller of the source IP, such that the source is unable to access the web server for a short time window. If the number of requests flood for longer time window the source is block for a longer time window

CASE - II

Input : Captcha response
Output : API call to Controller to flag IP as legit and redirect to server.

When a user submits the captcha response the engine evaluates the response and informs the controller about its genuinity and redirects it to the web server. If the user submit challenges and fails multiple times,


Malicious Traffic
Request /Response
Evaluate Response
Multiple Requests
Yes
Drop
No
No
Serve captcha Page to the User
Multiple Invalid Responses
Invalid
Response
Alert Controller Alter Flow Rules
Yes
Valid
Legit IP
Inform IP as Legit to Controller/Redirect/Reset History
Application Plane

3. Deploying Northbound APIs

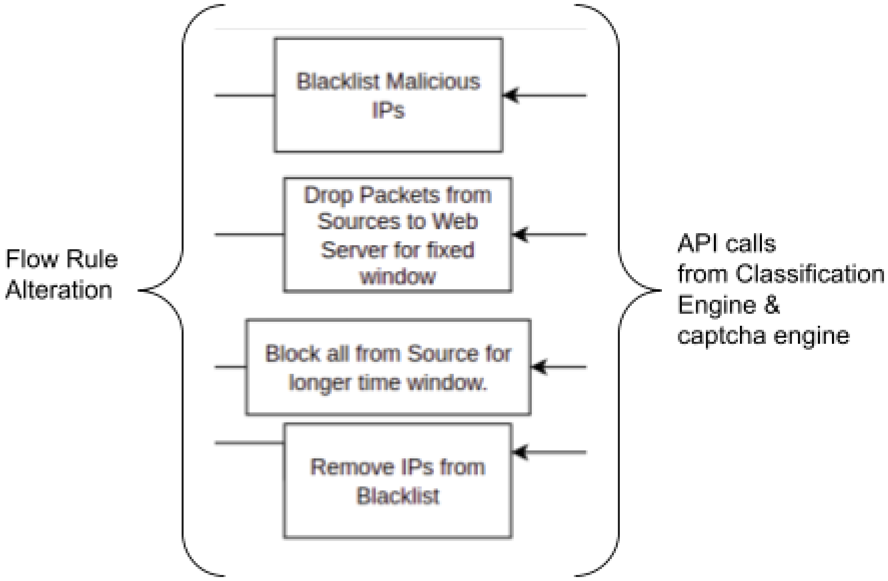
THIS IS THE COMMUNICATION MIDDLEWARE FOR CONTROL PLANE-APPLICATION PLANE COMMUNICATION
Output : REST API endpoints for application layer to communicate with the Controller

Software-defined northbound application program interfaces (SDN northbound APIs) are usually SDN RESTful APIs used to communicate between the SDN Controller and the services and applications running over the network. These APIs can be used to facilitate efficient orchestration and automation of the network to align with the needs of different applications via SDN network programmability.

We integrate such APIs for the following purposes into the controller for the application layer communication:

I. The classification engine must be able to alert the controller of a malicious IP address accessing the web server.
II. The captcha engine must be able to alert the controller of a benign source IP after it has completed the captcha challenge
III. The captcha engine must communicate about the IP address to be temporarily blocked from the server access, before it can complete the captcha challenge sent out by it.

IV. The captcha engine must be able to block malicious IPs for longer time windows that have failed the captcha challenge multiple times.


Blacklist Malicious IPs
Drop Packets from Sources to Web Server for fixed window
Block all from Source for longer time window.
Remove IPs from Blacklist
Flow Rule Alteration
API calls from Classification Engine & captcha engine

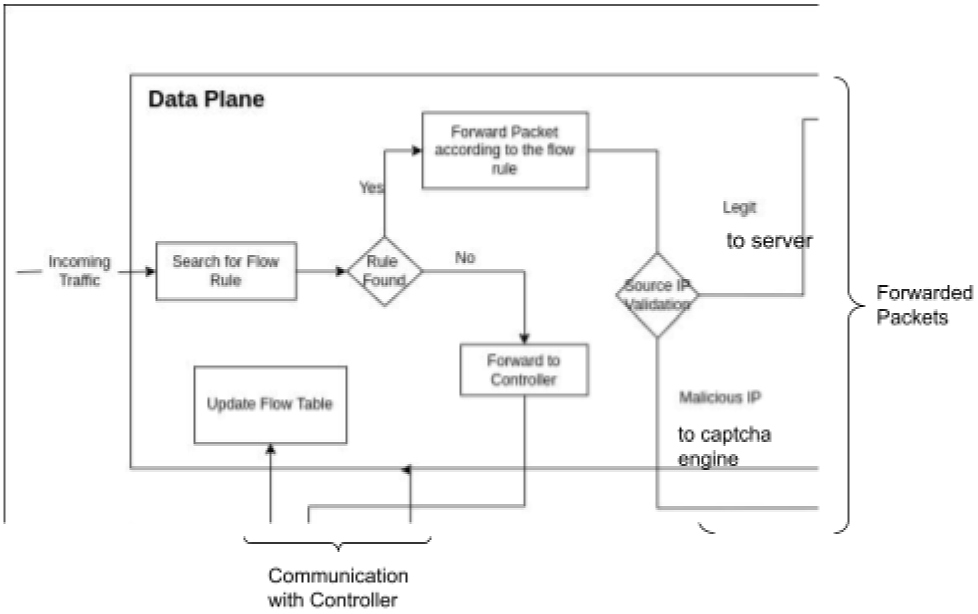
4. Flow Rule Assignment

Input : Malicious IP from Classification Engine/Legit IP from captcha engine
Output : Flow rule to the switch

Flow Rule Assignment module is an application on the SDN controller that continuously updates the flow table for the switch to forward and handle packets accordingly.

I. New packet from a new source IP arrives with the destination as a web server, default flow rule is installed, which forwards the packet to the web server(flow rule request).
II. When the source IP is classified as malicious and destination is web server, the packet is redirected to the captcha engine.
III. When source IP is classified as malicious and destination is captcha, the packet is forwarded to the captcha engine.
IV. When the source IP is legit and destination is the web server, the packet is forwarded to the web server.
V. When the source IP is legit and the destination is the captcha engine, the packet is dropped.
VI. When the captcha engine sends the captcha challenge to the user, the source IP is blocked from the web server for a short window of time.
VII. When the captcha engine records multiple incorrect requests/tries for the captcha challenge, the packets from the source IP are blocked for longer periods

of time.



DATASET DESCRIPTION

We use the datasets produced by the Canadian Institute for Cybersecurity. We collect samples for malicious and benign traffic from three different datasets, produced in three different environments to build a random forest classifier [3].

1. **CSE-CIC-IDS2018 on AWS**
   In CSE-CIC-IDS2018 dataset, we use the notion of profiles to generate datasets in a systematic manner, which will contain detailed descriptions of intrusions and abstract distribution models for applications, protocols, or lower level network entities.
   Link: https://www.unb.ca/cic/datasets/ids-2018.html
2. **Intrusion Detection Evaluation Dataset (CIC-IDS2017)**
   CICIDS2017 dataset contains benign and the most up-to-date common attacks, which resemble the true real-world data (PCAPs).
   Link - https://www.unb.ca/cic/datasets/ids-2017.html
3. **CIC DoS dataset (2017)**
   Generated application layer DoS attacks were intermixed with the attack-free traces from the ISCX-IDS dataset.
   Link - https://www.unb.ca/cic/datasets/ids-2018.html

The nature of files extracted from datasets is ‘.pcap’ files, which contain raw attack traffic captured at the network level for 4 days. Each day of data comprises around 500 .pcap files. Each file is converted into a .csv file and all the files of a single day are merged and labeled with the name of a type of attack. The type of attack traffic is identified based on Source IP of attack and victim systems. By following the same procedure, all 4 days of data are labeled and merged into corresponding files. Finally, the three aggregated multiclass labeled files are combined into a single file, with bidirectional flows amounted to 6.4 Million.

The purpose of combining multiple datasets is to simulate the near real-time DDoS traffic by introducing diversity, as each dataset was captured in different years (2016,2017,2018) using various attack tools in a different network and system configuration. The extracted DDOS flows are combined with "Benign " flows which are extracted separately from the same base dataset to form a single larger dataset.

PERFORMANCE MEASURES

1. ANOMALY DETECTION PERFORMANCE

Accuracy is the ratio of the total number of correct predictions and the total number of predictions.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + False\ Positive + True\ Negative + False\ Negative}$$

Precision is the ratio between the True Positives and all the Positives.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

The recall is the measure of our model correctly identifying True Positives.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

F1-score is the Harmonic mean of the Precision and Recall.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

2. MITIGATION

File Transfer Time is the elapsed time between the end of an inquiry or demand on a computer system and the beginning of a response. We are going to compare the file transfer time with and without mitigation of malicious traffic.

REFERENCES

1. Yu, S., Zhang, J., Liu, J. et al. A cooperative DDoS attack detection scheme based on entropy and ensemble learning in SDN. J Wireless Com Network 2021, 90 (2021). https://doi.org/10.1186/s13638-021-01957-9
2. L. Tan, Y. Pan, J. Wu, J. Zhou, H. Jiang and Y. Deng, "A New Framework for DDoS Attack Detection and Defense in SDN Environment," in IEEE Access, vol. 8, pp. 161908-161919, 2020, doi: 10.1109/ACCESS.2020.3021435.
3. Lima Filho Francisco Sales de, Silveira Frederico A. F., de Medeiros Brito Junior Agostinho, Vargas-Solar Genoveva, Silveira Luiz F., “Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning”, Security and Communication Networks Hindawi, 2019
4. Reena Singh Rajput, Dr. Sanjay Agrawal, A Survey on Denial of Service Attack Detection Techniques in Cloud Computing, International Journal for Scientific Research & Development| Vol. 4, Issue 06, 2016 | ISSN (online): 2321-0613
5. Research, Mahadev & Kumar, Vinod & Sharma, Himani. (2019). Detection and Analysis of DDOS Attack at Application Layer Using Naïve Bayes Classifier. 208-217.
6. Liguo Chen, Yuedong Zhang, Qi Zhao, Guanggang Geng, ZhiWei Yan, Detection of DNS DDoS Attacks with Random Forest Algorithm on Spark, Procedia Computer Science, Volume 134, 2018, Pages 310-315, ISSN 1877-0509
7. N. Meti, D. G. Narayan and V. P. Baligar, "Detection of distributed denial of service attacks using machine learning algorithms in software defined networks," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2017, pp. 1366-1371, doi: 10.1109/ICACCI.2017.8126031.
8. Allam S.K., Prasad G.S. (2021) Prevention of DDoS Attacks and Detection on Cloud Environment—A Review and Survey. In: Satapathy S., Bhateja V., Janakiramaiah B., Chen YW. (eds) Intelligent System Design. Advances in Intelligent Systems and Computing, vol 1171. Springer, Singapore. https://doi.org/10.1007/978-981-15-5400-1_47
9. Boukari Souley, Hauwa Abubakar, A CAPTCHA – BASED INTRUSION DETECTION MODEL,International Journal of Software Engineering & Applications (IJSEA), Vol.9, No.1, January 2018
10. Manu, B. (2016). A Survey on Secure Network: Intrusion Detection & Prevention Approaches. American Journal of Information Systems, Vol. 4, No. 3,Science & Education Publishing, pp 69-88.

11. http://mininet.org/
12. https://www.openvswitch.org/
13. https://www.google.com/recaptcha/about/
14. Iman Sharafaldin, Arash Habibi Lashkari, and Ali A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization”, 4th International Conference on Information Systems Security and Privacy (ICISSP), Portugal, January 2018
15. Hossein Hadian Jazi, Hugo Gonzalez, Natalia Stakhanova, and Ali A. Ghorbani. "Detecting HTTP-based Application Layer DoS attacks on Web Servers in the presence of sampling." Computer Networks, 2017
16. Ali Shiravi, Hadi Shiravi, Mahbod Tavallaee, Ali A. Ghorbani, Toward developing a systematic approach to generate benchmark datasets for intrusion detection, Computers & Security, Volume 31, Issue 3, May 2012, Pages 357-374, ISSN 0167-4048, 10.1016/j.cose.2011.12.012.
17. M Devendra Prasad, Prasanta Babu V, C Amarnath, “Machine Learning DDoS Detection Using Stochastic Gradient Boosting,” International Journal of Computer Sciences and Engineering, Vol.7, Issue.4, pp.157-166, 2019.