DDoS attack and Defense Techniques

Table of Contents

Table of Figures

I. ABSTRACT

Distributed Denial of Service (DDoS) is an extended version of the Denial of Services. In DDoS, multiple devices are flooded with the malicious traffic. In this literature review, firstly DDoS is explained in detail with block diagram, then types of DDoS attacks are explained which occur in different layers of the OSI layer, then different detection techniques are explained with the working architecture, and at last it is ended with conclusion and future research.

II. INTRODUCTION

A Distributed Denial of Service (DDoS) [1] attack is a malicious attempt to disrupt a targeted server, service, or network's normal traffic by flooding the target or its surrounding infrastructure with Internet traffic. DDoS assaults are effective because they use numerous compromised computer systems as attack traffic sources. Computers and other networked resources, such as IoT devices, are examples of exploited machinery. A DDoS assault is analogous to an unanticipated traffic congestion obstructing the roadway, preventing ordinary traffic from reaching its destination. DDoS assaults are carried out via networks of machines that are linked to the Internet. These networks are made up of malware-infected PCs and other devices (such as IoT devices), which may be manipulated remotely by an attacker. Individual devices are known as bots, while a botnet is a collection of bots. The attacker can direct an attack once a botnet has been established by sending remote instructions to each bot. When a botnet targets a victim's server or network, each bot sends requests to the target's IP address, potentially overloading the server or network and causing a denial-of-service to normal traffic. Separating the attack traffic from the genuine Internet traffic is difficult because each bot is a legitimate Internet device. Figure 1 shows the basic working of DDoS attack.

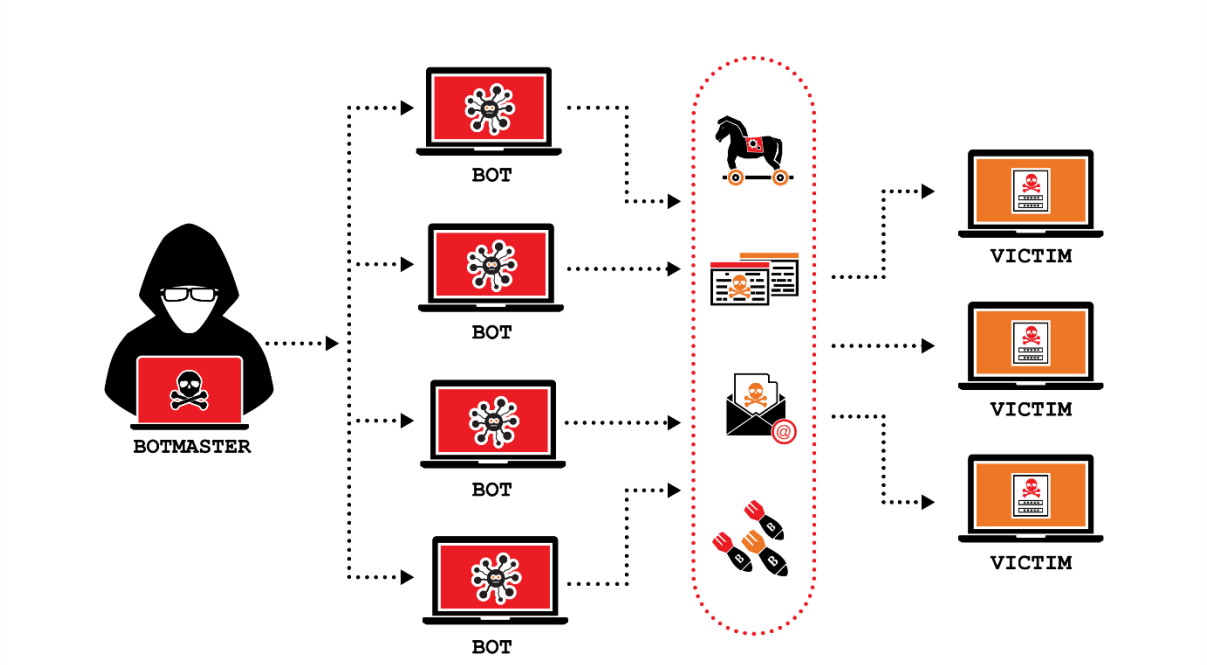

Figure 1: DDoS attack

III. TYPES OF DDoS ATTACKS

There are different types of DDoS attacks which can be taken place on different networking components from application layer to network layer of the Open System Interconnection (OSI) [2] model. HTTP flood [3] is an example of the attack on the application layer, Protocol attack [4] is the example of the attack on the network and transport layer. SYN flood [5] is also an example of the protocol attack.

A. HTTP Flood Attack: HTTP-flooding is a typical DDoS, which exhausts the target web-server resources by sending many legitimate-like HTTP-GET requests. Unlike the mostly used DDoS in the past, HTTP-flooding is much stealthier in attack strategy. Figure 2 shows how the HTTP flood attack works. First, compared with the tremendous traffic of Bandwidth-flooding, the low traffic of HTTP-flooding usually cannot cause traffic anomaly. Second, unlike the bogus TCP connections of SYN-flooding, the true TCP connections of HTTP-flooding do not significantly change the statistics of TCPSYN packets. Thus, it is much harder to detect HTTP-flooding than another DDoS.

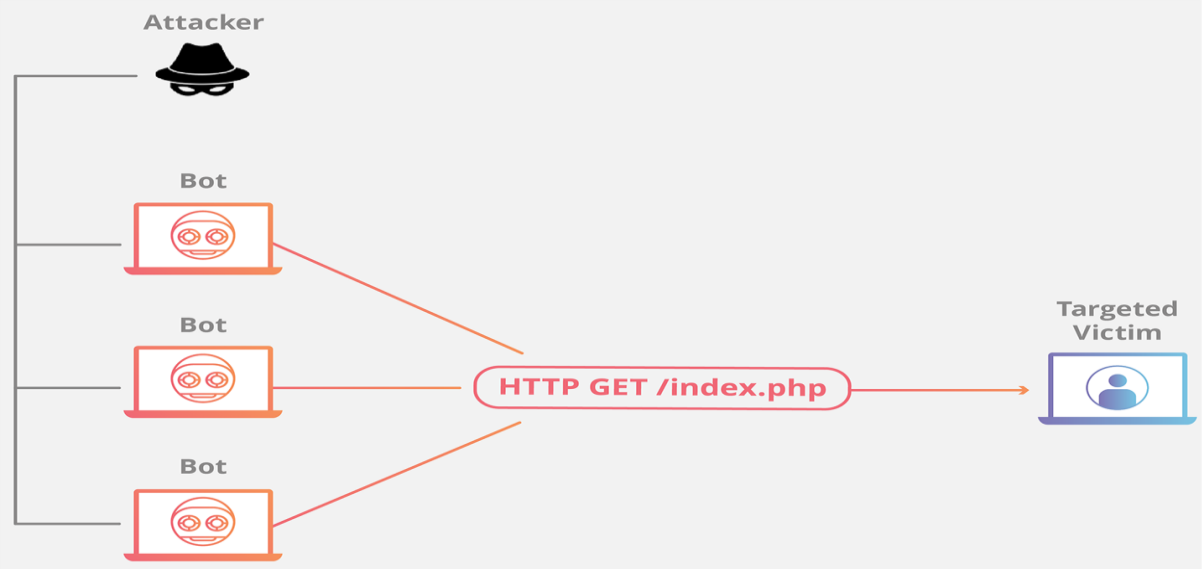

Figure 2: HTTP flood attack

B. Protocol attack: Protocol attacks, also known as a state-exhaustion attacks, cause a service disruption by over-consuming server resources and/or the resources of network equipment like firewalls and load balancers. Protocol attacks utilize weaknesses in layer 3 and layer 4 of the protocol stack to render the target inaccessible. Example, SYN flood. Figure 3 shows how the protocol attack works

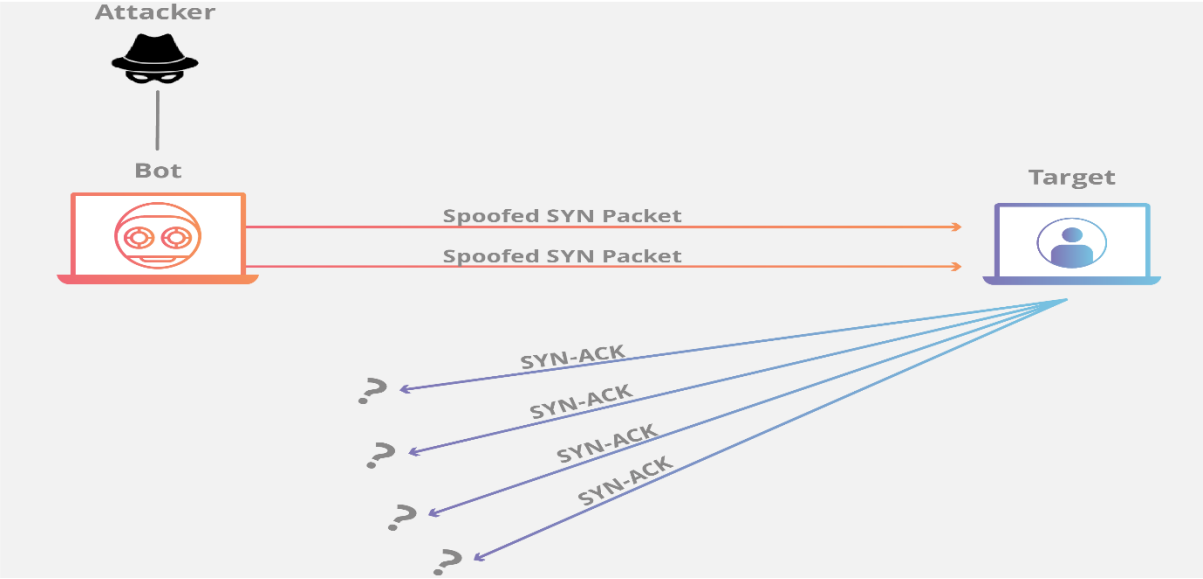

Figure 3: Protocol attack

IV. DDoS ATTACK DETECTION TECHNIQUES

Since DDoS are most widely used cyber-attacks, design of DDoS detection mechanisms has attracted attention of researchers. Design of these mechanisms involves building statistical and machine learning models. Most of the work in design of mechanisms is focussed on improving the accuracy of the model. However, due to large volume of network traffic, scalability and performance of these techniques is an important research issue. Following are some detection techniques taken in practice.

A. Using Apache Spark Framework

The Figure 4 illustrates the process of DDoS detection using Apache Spark framework. It consists of Spark RDD [6] streaming for pre-processing of features and MLlib for training and detection engine.

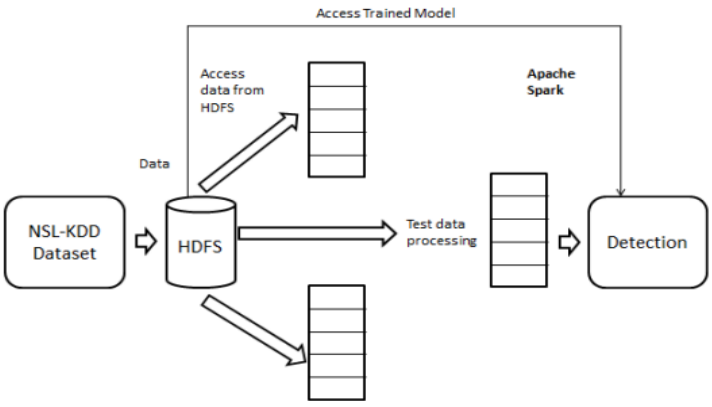

Figure 4: DDoS Detection using Apache Spark

The steps involved in the proposed system are described below.

- NSL-KDD is used to train the system to detect DDoS attack and store it in HDFS.
- To build the models, the classification algorithms are trained.
- The Classified models are used to test the future data.
- Different classification algorithms are applied to the model.
- The training delay and the accuracy, precision, recall is obtained.

Apache Spark is built on the YARN infrastructure. Spark is a distributed processing framework that supports in-memory iterative processing to perform machine learning algorithms effectively in addition to distributed batch job processing in the previous version of Hadoop. Although the Spark distributed processing framework is written in Scala, Spark also has a set of bindings for Python. As we are working in spark environment data is stored in RDD which is immutable, and it stores data in string format to apply classification algorithms we need have data in integer, so we have converted the dataset to integer type as all the values in the dataset are integer.

B. TCP Based Detection Technique

The overall architecture of our proposed TCP-based DDoS detection system [7] is shown in Figure 5. It consists of four main phases:

1. Data Collection Phase: In the Data Collection phase, we use a packet sniffer to capture every packet from TCP traffic flows. After extracting TCP/IP header from the captured packets, the proposed system partitions them according to every pair of IP addresses (local IP, the address of the local host, and remote IP, the address of the remote host that communicates with the local host) and counts the number of inbound (remote IP to local IP) packets of each IP pair every second.
2. Sample Generation and Feature Selection Phase: According to the two attack modes, we design different sample generation method and select different features.
    - Sample generation: To develop a practical real-time detection system, we begin by detecting abnormal traffic flows.
    - Feature Selection: depending on fixed source IP attacks (FSIA) [8] and random source IP attacks (RSIA), [9] and Chi-squared test [10].

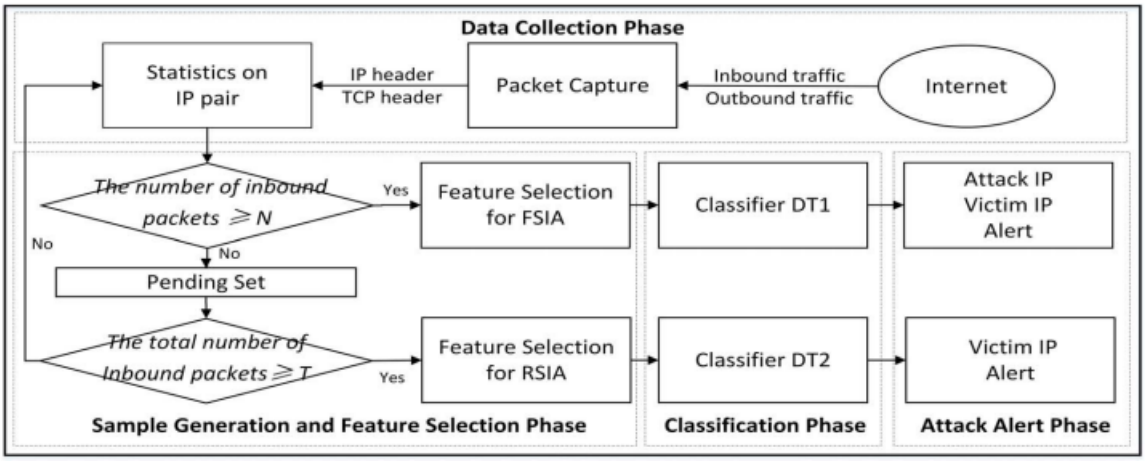

Figure 5: Architecture of the proposed TCP-based DDoS detection system

3. Classification Phase: In the Classification phase, we also provide two decision tree classifiers which are trained with our experimental data. One is designed for FSIA, another for RSIA. They can be used to label traffic flow as normal or attack.
4. Attack Alert Phase: During FSIA detection, the IP-pair method enables us to raise an alert, giving the fixed-source IP address, which is the malicious user. This enables the operator to react with an appropriate defence mechanism.

C. Collaborative Source-Side Detection Technique

In the source-side network, the amount of traffic observed is relatively small compared to a victim side network, so attack traffic can easily mix with normal traffic. Because of difference in the usage of normal traffic for each different time zone, the performance of source-side attack detection module located in different time zone may be different. The collaborative source-side DDoS attack detection method determines the result using the detected results and statistical weights of the source-side attack detection modules located in in different time zones. Currently, the detection result d_i^{ti} of the i^{th} source-side attack detection module in the time window t_i. Figure 6 represents the architecture of collaborative source-side DDoS Detection.

The time window t_i is composed of 1 minute interval. As a result of detection, the value of d_i^{ti} has the value of 1 when an attack is detected and has a value of 0 when no attack detected. The collaborative attack detection module uses the weighted arithmetic mean A^t to determine the final result by using the detection result d_i^{ti} and statistics weight $W_{ti,}$ shared by each site of the source-side attack module. The weighted arithmetic mean formula is shown below.

$$A^t = \sum_{i=1}^{L} \frac{W_{t_i} * d_i^{t_i}}{W_{t_i}}$$

If the weighted arithmetic average is greater than specific threshold, the attack is finally determined to be detected at that time window t.

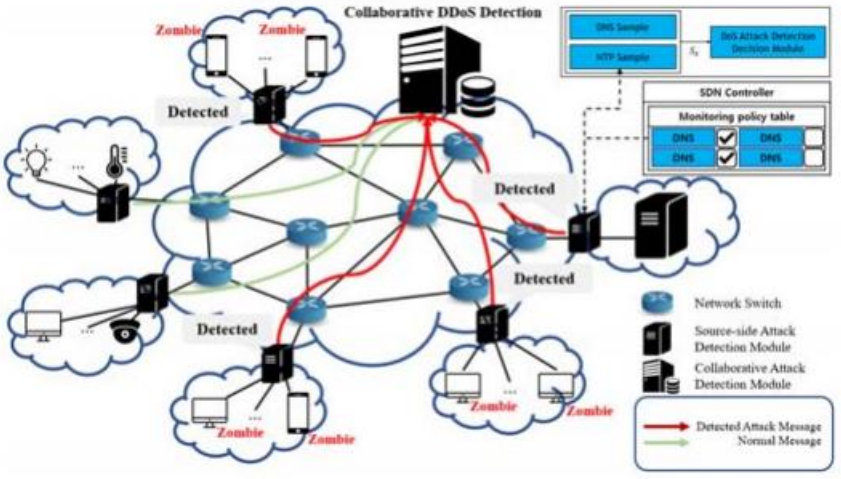

Figure 6: Architecture of collaborative source-side DDoS Detection

V. Conclusion and Future research

To conclude this literature review, we looked at the basic principle and working of DDos, types of attacks and their detection techniques. These techniques are experimentally performed and proved to be beneficial. Future research can be in the direction of method that dynamically modifies the margin according to observed traffic volume and features according to the error rate of the source-side attack detection module located in different time zones, plan to explore parameter tuning of the Spark framework to improve DDoS attack detection efficiency in big data condition. Also, we plan to use deep learning techniques.

REFERENCES:

[1] https://www.cloudflare.com/en-in/learning/ddos/what-is-a-ddos-attack/

[2] C. T. Nguyen, M. . -C. Vialatte and C. Rieu, "OSI application layer standards analysis for a distributed application implementation," [1989] Proceedings. 14th Conference on Local Computer Networks, 1989, pp. 225-233, doi: 10.1109/LCN.1989.65266.

[3] J. Wang, M. Zhang, X. Yang, K. Long and J. Xu, "HTTP-sCAN: Detecting HTTP-flooding attack by modeling multi-features of web browsing behavior from noisy web-logs," in China Communications, vol. 12, no. 2, pp. 118-128, Feb. 2015, doi: 10.1109/CC.2015.7084407.

[4] C. Pu, "Spam DIS Attack Against Routing Protocol in the Internet of Things," 2019 International Conference on Computing, Networking and Communications (ICNC), 2019, pp. 73-77, doi: 10.1109/ICCNC.2019.8685628.

[5] Pi-E Liu and Zhong-Hua Sheng, "Defending against tcp syn flooding with a new kind of syn-agent," 2008 International Conference on Machine Learning and Cybernetics, 2008, pp. 1218-1221, doi: 10.1109/ICMLC.2008.4620589.

[6] https://spark.apache.org/docs/latest/rdd-programming-guide.html

[7] J. Jiao et al., "Detecting TCP-Based DDoS Attacks in Baidu Cloud Computing Data Centers," 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS), 2017, pp. 256-258, doi: 10.1109/SRDS.2017.37.

[8] X. He, J. Liu, C. -T. Huang, D. Wang and B. Meng, "A Security Analysis Method of Security Protocol Implementation Based on Unpurified Security Protocol Trace and Security Protocol Implementation Ontology," in IEEE Access, vol. 7, pp. 131050-131067, 2019, doi: 10.1109/ACCESS.2019.2940512.

[9] https://javapipe.com/blog/ddos-types/

[10] Y. Li, "Applications of Chi-Square Test and Contingency Table Analysis in Customer Satisfaction and Empirical Analyses," 2009 International Conference on Innovation Management, 2009, pp. 105-107, doi: 10.1109/ICIM.2009.31.

[11] S. Yeom and K. Kim, "Improving Performance of Collaborative Source-Side DDoS Attack Detection," 2020 21st Asia-Pacific Network Operations and Management Symposium (APNOMS), 2020, pp. 239-242, doi: 10.23919/APNOMS50412.2020.9237014.

[12] ShwetaGumaste, Narayan D. G., SumedhaShinde, Amit K, “Detection of DDoS Attacks in OpenStack-based Private Cloud using Apache Spark”, Journal of Telecommunication and Information Technology, Vol. 3, pp. 62-71, 2020.