A Survey on Distributed Denial-of-service Attacks and Defense Mechanisms

[1]Vipul Chidri, [2]V.S Balasubramani, [3]Syed Sadath Ali, [4]Shrikrishna S Hegde, [5]Pradeep Sadanand
[5]Assistant Professor, [1,2,3,4]U.G Student,
B.M.S College of Engineering, Bangalore, India

***Abstract*-Distributed Denial-of-service (DDoS) attack is one of the most dangerous threats that could cause devastating effects on the Internet. DDoS attacks started in the late 1990s but the influence of it was realized by people only when the big establishments and corporations were hit by these attacks. Numerous tools are available that can perform DDoS attacks from thousands of compromised hosts and can take down any connection, any link on the Internet by just a few command keystrokes. Distributed Denial of Service (DDoS) attacks are a virulent attack on the availability of Internet services and resources. DDoS attackers intrude huge number of computers by exploiting software vulnerabilities and set up DDoS attacks. These computers are then invoked to initiate corresponding, large-scale attack against one or more victim systems. It is desirable to develop all-inclusive DDoS solutions that defend against known and imminent DDoS attack variants. However, this requires a complete understanding of the scope and techniques used in different DDoS attacks. This paper proposes new taxonomies to categorize DDoS attack networks, to classify the diverse techniques used in a DDoS attack. It proposes classes of countermeasures that target the DDoS problem. This work is intended to stimulate research into effective and efficient defenses and detection mechanisms for DDoS attacks, and to assist in creating comprehensive solutions that will provide generic and effective approach to countering known and derivative DDoS attacks.**

Index Terms- Introduction, Defenses architectures, A preventive measure to resource inflated DDoS attacks, Cracking algorithm to prevent DDoS attacks, Mitigating application layer DDoS attacks, General rules to help mitigate DDoS attacks, Conclusion.

I. INTRODUCTION

Denial of service through server flooding can be thought of as simply filling up a pipe with enough material to prevent anything else from getting through. Denial of service may occur reluctantly if a server receives more traffic than it was designed to handle. This happens many times, such as when a low-trafficked website becomes famous. Numerous tools could initiate DDoS attacks from thousands of compromised hosts. Denial of service attack programs are out and about for several years from now. The sources of single source attacks are answered easily by numerous prevailing defense mechanisms. These can be easily deactivated with improved tracking techniques.

According to the CERT/CC, the primary DDoS attacks occurred in 1999[1]. In January 2000, one of the first major attacks was waged against Yahoo, EBay and Amazon. This attack kept them off the web for about 1 hour and causing them a loss of 1.7 Billion Dollars. Another DDoS attack occurred in October 2003 against the 13 root servers that provide the DNS service to internet users around the world. If all 13 root servers were not operational, there would have been unfortunate problems accessing the web. Even though the attack only lasted for an hour and the effects were hardly noticeable to the typical Internet user. If unchecked, more powerful DDoS attacks might probably disable essential Internet services in minutes.

DDoS attacks are two kinds of architectures: the Agent Handler architecture and the Internet Relay Chat Architecture

1. The Agent-Handler architecture is comprised of clients, handlers, and agents. First, the attacker builds a network of computers by discovering weak hosts and uses them to yield the volume of traffic needed. Attacker installs attack tools on the cooperating hosts of the attack network. Machines running these attack tools are recognized as Handlers which operate under the control of the invader. The hosts that have been infected by the attack tools look for other defenseless hosts and install on them the same attack tool.

2. In the IRC-type design, an IRC channel is used to connect the clients to the agents. IRC ports can be utilized to send commands to the agents. The main advantage of this architecture is that an attacker can hide his presence. Low Orbit Ion Cannon (LOIC) is an example [3].

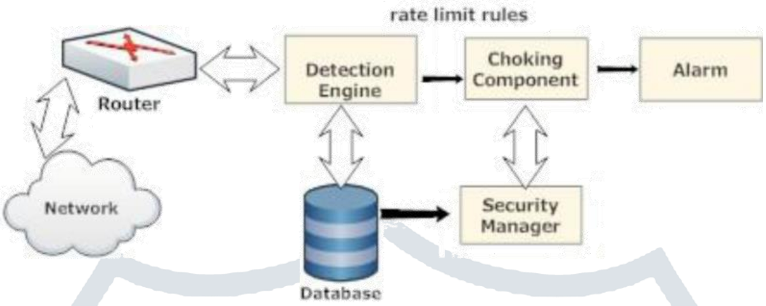
Among these two architectures, the Agent Handler architecture is frequently found in use in the literature.

II. DEFENCE ARCHITECTURES

Introduction of generic architectures of DDoS defense mechanisms are classified based on vicinity of deployment. DDoS defense mechanisms can be classified into 3 classes based on the locality of deployment: victim-end, source-end, and intermediate network protection mechanisms [4].

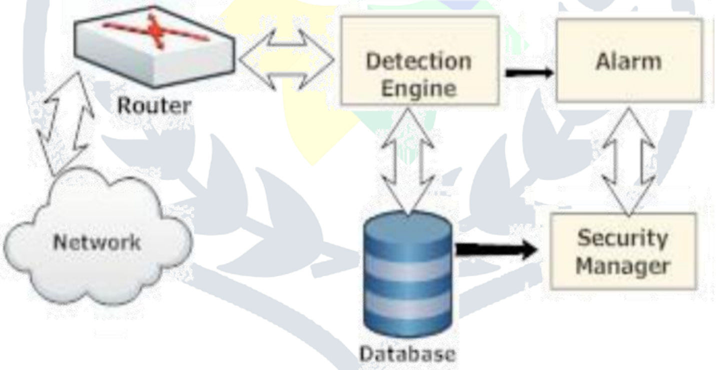
1) *Source-end defense mechanisms*

It is the best defense to detect and end a DDoS attack at the source, which avoids the chance of flooding on the victim side and also in the complete network. This method has two demerits:
i.) It is very difficult to identify DDoS attack at the source end because the sources are distributed and a single source behaves similar to the normal traffic.
ii.) The difficulty to install a system at the source end.



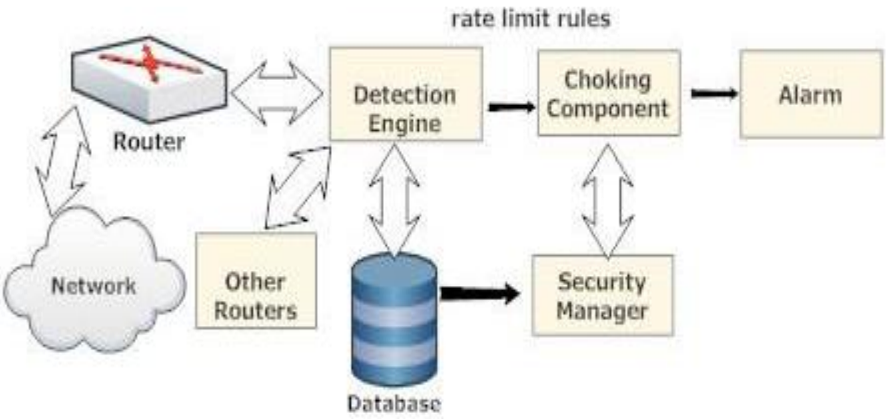
2) *Victim-end defense mechanisms:*

It is relatively simple to detect DDoS attacks in victim routers. But it is important to secure the network resources which are used by Web Servers which provides services to the network users. This approach has two weaknesses:
i.) Victim resources typically get weak and the flow can’t be stopped on the far side victim routers.
ii.) Attacks can be identified if and only if it reaches the victim end.



3) *Intermediate network defense mechanisms*

In this approach, finding and trace back of attack sources are simple due to cooperative operation. Routers form an overlay network to split their interpretations. One main disadvantage of this approach is its deploying ability.
Routers on the network need to employ this detection scheme in order to achieve full detection accuracy. Obviously, full practical application of this scheme is tough due to the need for reconfiguration of all the routers on the Internet.



III. A PREVENTIVE MEASURE TO RESOURCE-INFLATED DDoS ATTACKS

Software puzzle scheme is proposed to defeat GPU-inflated DoS attack [7]. It mainly adopts software protection technologies to ensure challenges for data confidentiality and code security for an appropriate time period, *e.g.*, 1-2 seconds. Hence, it has different security requirements from the conventional cipher which demands long-term privacy and code protection which focuses on long-term robustness against reverse-engineering only. The paper focuses on GPU-inflation attack. Its idea can be extended to thwart DoS attackers which exploit inflation resources such as Cloud Computing. For example, suppose the server inserts some anti-debugging code for detection in the cloud platform using the software puzzle. When the puzzle is running, the software puzzle will not carry on the puzzle-solving process on the cloud environment when the Cloud-inflated DoS attack fails. In the present software puzzle, the server has to spend time to build the puzzle. In other words, the present puzzle is generated at the server side. An open problem is how to construct the client-side software puzzle so as to save the server time for better performance.

IV. CRACKING ALGORITHM TO PREVENT DDoS

Due to increase of users on the Internet, large numbers of people want to attack other system resources. Many competitors desire to make their web site more famous than others. As a result they want to hack the other web site's services. They keep on logging in to a particular web site more times, and then performance of the service provided by this web server decreases. To avoid that, this program maintains a record of status. In that it keeps the IP addresses of current users and their status. If the particular IP address has been signed on for a first time, then the status is made as that of a Genuine user. For 2, 3 and 4 it marks them as a Normal user. For the fifth time it makes the particular IP address status as an Attacker. User wishes the server to increase the time depending on the application. After that, the user is not allowed to get the service of that particular web site. The service is not given to that IP address [8].

Algorithmic steps

i. Packet Filter

Packet filters act by checking the "packets" which transfer between systems on the Internet. If any packet matches with the packet filter's set of regulations then the packet filter will drop (silently discard) the packet, or discard it (and send "error responses" to the source).

ii. MAC Generator

It distinguishes the packets which have a genuine source IP address from those that have a spoofed address. Once the first TCP/SYN packet of a client gets through, the proposed system redirects the client to a pseudo-IP address at once (still belonging to the website) and port number pair, through a standard HTTP URL redirect message. Few bits from this IP address and the port number pair will act as the Message Authentication code (MAC) for this client's IP address

iii. IP Handler

When an attacker is using genuine addresses, the proxy server makes use of the Deficit Round Robin algorithm to collect the address of the request sent by a client. If an attacker sends packets much faster than its fair share, the scheduling policy stops its excess traffic. Also for every authentic IP address, the system does an accounting on the number of packets that reach the firewall but are dropped by the scheduler; its IP address will be blacklisted.

V. MITIGATING APPLICATION-LAYER DDoS ATTACKS

Zombie machines attack the victim through legitimate packets in such a way that packets have legitimate format and are sent through normal TCP connections [9]. In this system, connections get ratings based on their behavior. It is highly probable that the connections which get lesser Scores are harmful connections; thus the server retakes bottleneck resources from them. The experimental results on Emu lab environment show that the Connection Score schemes can efficiently handle application-layer DDoS attacks. According to them the administrators of websites need not annoy their users by forcing them to solve CAPTCHA tests. Also CAPTCHA tests are not very reliable. Hence the connection Score scheme can be effectively considered as an alternative method to handle application-layer DDoS attacks.

VI. GENERAL RULES TO HELP MITIGATE DDoS ATTACKS

There are some general rules to help defend against a DDoS attack. These are to be used as a guide, as they cannot stop all attacks, especially some of the complex types.

1. Turn down all needless ports and protocols. If you are running a web server, and utilizing the TCP protocol over port 80, then implement Access Control List entries to obstruct all other ports and protocols from entering the network [10].
2. Implement an IP blacklist: Become familiar with trusted security related websites that have lists of IP addresses known for bringing malicious traffic. These IP addresses can be added to an IP blacklist so their traffic will never reach your infrastructure.
3. Block invalid and abnormal packets: Consider blocking invalid and malformed packets from entering your network. If you have a custom or proprietary application that sends genuine deformed packets over the network, then you may need to take other options to consideration to handle this traffic, like outsourcing your security protection to a DDoS mitigation specialist.
4. Configure and harden network equipment: Recommended configuration settings, like those from the Center for Information Security (CIS), can protect your devices and network better [12]. Consider implementing them.

VII. CONCLUSION

In this paper, we have presented an overview of DDoS defense schemes. Practically designing and implementing a DDoS defense is very difficult. During the development of a DDoS defense scheme, the issues mentioned in this paper should be deliberated and considered with due seriousness. The comparison of the existing detection mechanisms shows that the majority schemes are not capable of fulfilling all the requirements for real time network defense. Special performance parameters have to be balanced against each other finely and fitly.

VIII. ACKNOWLEDGMENT

The work reported in this paper is supported by the college through the TECHNICAL EDUCATION QUALITY IMPROVEMENT PROGRAMME [TEQIP-II] of the MHRD, Government of India.

REFERENCES

[1] CERT Coordination Center, Denial of Service attacks, Available from http://wwww.cert.org/techtips/denial_of_service.html

[2] Specht, S. M. and Lee, R. B. (2014) Distributed denial of service: Taxonomies of attacks, tools, and countermeasures. Proceedings of the ISCA 17th International Conference on Parallel and Distributed Computing Systems, San Francisco, California, USA, 15-17 September, pp. 543–550.

[3] Batishchev, A. M. (2004). LOIC (Low Orbit Ion Cannon). http://sourceforge.net/projects/loic/.

[4] Mirkoviac, J., Prier, G., and Reiher, P. (2012) Attacking DDoS at the source. Proceedings of the 10th IEEE International Conference on Network Protocols, Paris, France, 12-15 November, pp. 1092–1648. IEEE CS.

[5] Chen, C. L. (2009) a new detection method for distributed denial of- service attack traffic based on statistical test. Journal of Universal Computer Science, 15, 488–504.

[6] Akella, A., Bharambe, A., Reiter, M., and Seshan, S. (2003) Detecting DDoS attacks on ISP networks. Proceedings of the Workshop on Management and Processing of Data Streams, San Diego, CA, 8 June, pp. 1–2. ACM.

[7] Software Puzzle: A Countermeasure to Resource-Inflated Denial-of-Service Attacks by Yongdong Wu, Zhigang Zhao, Feng Bao, and Robert H. Deng. IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, VOL. 10, NO. 1, JANUARY 2015

[8] V.Priyadharshini, Dr.K. Kuppusamy / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, May-Jun 2012, pp.2263-2267

[9] Tackling Application-layer DDoS Attacks HakemBeitollahi and Geert DeconinckProcedia Computer Science 10 (2012) 432 – 441

[10] Peng, T., Leckie, C., and Ramamohanarao, K. (2012) Survey of network-based defense mechanisms countering the DoS and DDoS problems. ACM Computing Survey, 39, 3:1–3:42.

[11] Lin, S. and Chiueh, T. C. (2013) A survey on solutions to distributed denial of service attacks. Technical Report TR201, Department of Computer Science, State University of New York, http://www.ecsl.cs.sunysb.edu/tr/TR201.pdf.

[12] Gil, T. M. and Poletto, M. (2011) MULTOPS: a data structure for bandwidth attack detection. Proceedings of the 10th conference on USENIX Security Symposium - Volume 10, Berkeley, CA, USA, 13-17 August 3.USENIX Association Berkeley.

[13] Thomas, R., Mark, B., Johnson, T., and Croall, J. (2013) NetBouncer: Client-legitimacy-based high performance DDoS filtering. Proceedings of the 3rd DARPA Information Survivability Conference and Exposition, Washington, DC, 22-24 April, pp. 111–113. IEEE CS, USA. Volume 10, Berkeley, CA, USA, 13-17 August3.USENIX Association Berkeley.