Simulation & Prevention of DDoS attacks

[1]Jigar Panchal, [2]Jay Sethia, [3]Tanmay Bhoir, [4]Parag Mhatre

[1, 2, 3, 4] B.E. Computer Engineering, Mumbai University, Maharashtra, India

[1]jigarp08@gmail.com, [2]jaysethia31@gmail.com, [3]tanmaybhoir09@gmail.com, [4]paragmhatre1993@gmail.com

Abstract: A widespread issue that exists in today's connected digital world is an omnipresent threat of a complete and total DDoS attack. This could potentially wreck major processes and businesses, thereby creating a digital terror attack. This has become a common and increasingly popular choice of attack for hackers by far and large. DDoS attacks paralyze Internet systems by overwhelming servers, network links and network devices (routers, firewalls, etc.) with bogus traffic. Service-level agreements (SLAs) are violated, triggering costly service costs. Company reputations are damaged, sometimes permanently. Irrelevant to the type of DDoS attack, current techniques used to deal with them fall short of mitigation & fail to provide business continuity. The current incumbent systems are not equipped to deal with the most modern forms of DDoS attacks as effectively. They are based on an old foundation method of recognizing a DDoS attack based on reactive approach. For effective prevention, there is a strong need of a proactive approach to detecting and obfuscating a DDoS attack.

Our proposed system will be monitoring the incoming & outgoing traffic continuously & try to identify a potential threat to our system. It will do this by monitoring for various aspects of the Internet traffic such as the frequency of the packets, check for half-open connections at specific intervals, check for the finish bit, port number that the packet is sent to, etc. It is capable of detecting live DOS attacks and limiting the bandwidth being utilized by the attacker. Also additionally, our tool is capable of detecting attacks which were already done(using its logs) or are going on live. Thus, the expected outcome from our software is that the servers network links, and network devices to function seamlessly by nullifying the DDoS attacks.

Keywords – DDoS Attack, OpenBSD, PF Firewall, IDS, SYN, Packet Filtering, ICMP Packet.

I. INTRODUCTION

Simulation and prevention of a DDoS attack implies creation of a virtual network, a DDoS attack on the network, and active obfuscation of the network. The rapid growth and the increase in the use of the Internet have made the security of the Internet increasingly important. The bandwidth of the network has become one of the primary concerns in the present world.

Huge multinational companies use Physical Firewalls and expensive softwares which need huge resources to Block and Nullify the DDoS attacks. Such expensive techniques cannot be used by all the companies. Although firewalls have an important task in an organization's security structure, they are not built as a DDoS attack prevention devices. In fact, firewalls have certain inherent features that lead to dealy/obstruction to provide complete protection against today's most sophisticated DDoS attacks.

There are methods & softwares available to tackle DDoS attacks. But almost every software has shortcomings which limit the area of its usage. One such technique called “Cryptographic storage & network firewalls” studies the effect of different network anomaly-based attack detection techniques. This has methods of attacks stored in its memory. It then monitors the traffic for similar attempts after which it can block the attacks. But, if the attacking technique is new or even a modified version of an old technique, this tool will fail to thwart the attack[1].

Our tool follows an approach where the tool will look for indications of a packet being malicious depending on the parameters defined by us. It will log & monitor all the

 DOI : 10.18231/2454-9150.2017.0018

incoming / outgoing traffic on real-time basis. Once it detects a threat, as it looks for symptoms of an attack as programmed, it will take steps to nullify/limit the attack. The software will send a status report to the admin from time to time with a notification if a repeat offender IP is found so that the admin can blacklist it if needed.

II. PROPOSED SYSTEM

We are proposing a step-by-step approach to identify types of attacks & building modules to prevent them. A basic tool for detecting & identification of packets & their frequency will be implemented. Further improvements & additions will be done to this tool as the project reaches higher stages of research & implementation.

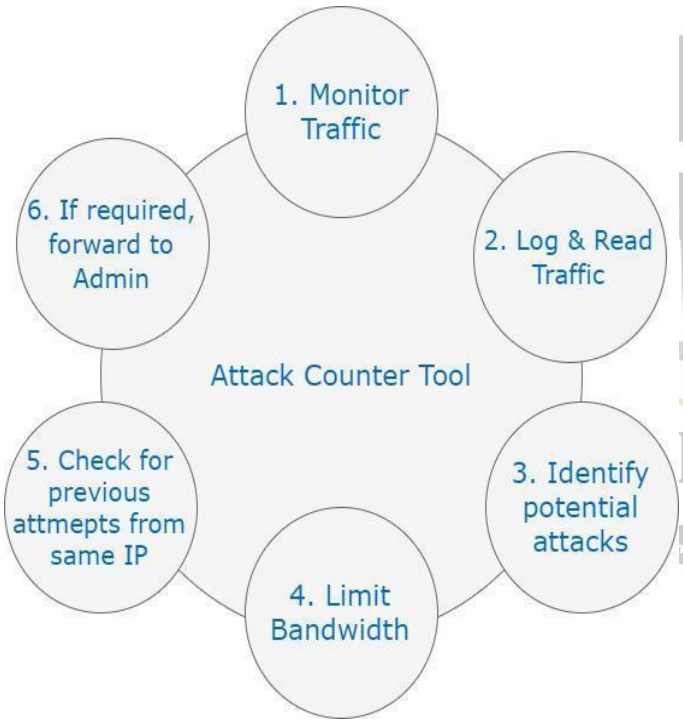

Fig 1: Attack Counter Life Cycle

Monitoring of different aspects of a packet will be done on a real time basis depending of the need of the software at that time. For example, the frequency, FIN bit of a packet, destination port number, etc. When the tool identifies a potential attack, it will limit the bandwidth allocated to that IP thereby stopping any DOS attacks from that IP[2]. Figure 1 above shows the process cycle of the tool. It is explained in detail in the algorithm below.

Algorithm

Step 1: Real-time logging & monitoring started.

Step 2: Check the header files of the incoming packets identify the type of that packets.

Step 3: If the packet type is ICMP, go to step 4. If it is TCP, go to step 5. If it is UDP, go to step 6.

Step 4: Check the frequency of the packets from that particular IP. If the frequency is more than the set limit, then log the IP, reduce the bandwidth to that IP & generate dynamic rules for firewall. Go to step 7

Step 5: Check if the FIN bit of the packet is SET. If yes, drop the packet. Otherwise, check if the acknowledgement signal is delayed beyond a period. If so, drop the packet. Check the log for the frequency of that packet & if it is beyond a certain limit, then log the IP, reduce the bandwidth to that IP & generate dynamic rules for firewall. Go to step 7

Step 6: Follow the process of Step4. Further, check the destination port of the packet. If it is an unopened port, then log the IP, reduce the bandwidth to that IP & generate dynamic rules for firewall[3]. Go to step 7

Step 7: Send report of throttled connections & logs to network administrator, if any. Block or Open connection to IPs if directed from the administrator.

Step 8: End

Tools Used

1. **Snort** - Intrusion Detection
2. **PF Firewall** - Creating queues; Application specific bandwidth allocation
3. **Python Script** - Core Programming
4. **QT Designer** - End User GUI

III. ARCHITECTURE

The system will be housed between the nodes & the internet. It will be placed before the firewall to avoid an attack on firewall like the Illegal flag flood. The system will monitor, identify & nullify possible DDoS attacks. We can use an inhouse script, third-party tool or an IDS to monitor the traffic. In this scenario, we are using a third party tool – PF (OpenBSD).

 DOI : 10.18231/2454-9150.2017.0018

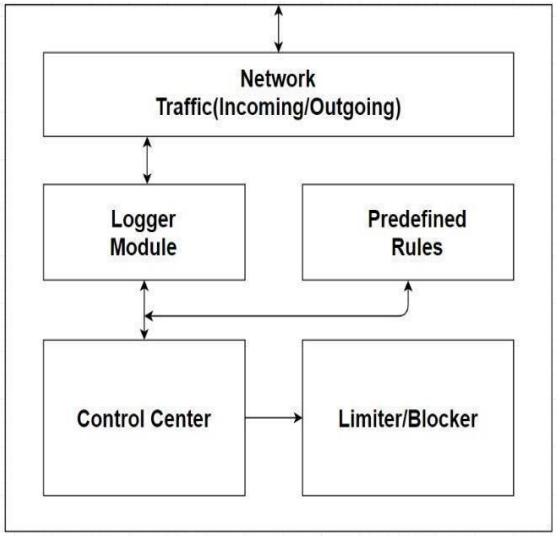

Fig 2: Proposed System Architecture

Network Traffic: This is all the incoming & outgoing network traffic that passes the firewall from in & out of the system. Logger: The Logger module will log all the incoming & outgoing traffic on the network in a text file on the desired location. It will read the header files of the packets & identify the packet & mention it in the logs as well. The logger can be anything from a self-compiled tool to a 3rd party application or a IDS[4].

Rules: This is a collection of modules that contain the rules that are set by us which have been programmed with the steps to nullify the different types of DDoS attacks[5] like SYN Flood attacks, Illegal TCP flag flood, ICMP flood, etc.

Control Center: This is the brain of the system. The Control center takes all the decisions in the system. The control center monitors the generated logs continuously. It tries to decide whether a packet is genuine or malicious and takes steps accordingly. If a malicious packet is detected[6], it uses the rules from the rules module to take action to nullify the attack & thereby maintaining proper network functioning. The Control center also checks for previous attack attempts from a particular source & if it finds similar previous attempts, the data is sent to the system administrator[7] for further actions that need to be taken.

Packet Limiter or Blocker: This module will block limit packets as directed by the Control Center. This may be a self-developed firewall[8] or a third party firewall like PF.

IV. OUTPUT

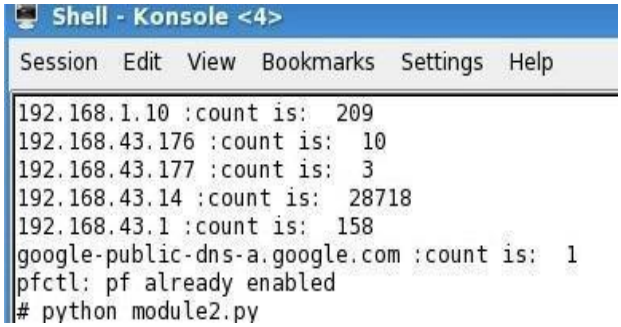

Fig 3: Output - Packet Count

The output performance of this tool can be measured in its ability to correctly detect malicious(non-genuine) packets from a series of genuine packets. To check this, we have employed the tool in ICMP mode & we simulated internet traffic with a DOS attack IP of 192.168.43.14. Below is the output that the tool gave in the mentioned scenario[9].

Figure 3 above shows the software which is employed in ICMP attack detect mode. Hence, it counts the frequency of the total number of packets received/sent with the IP. The output shows us that the tool was able to correctly identify & dynamically block the attacker IP resulting in the restoration of the network connection in under 10 seconds.

Type of Packet	Source IP	Packet Count	Expected Result	Actual Result
ICMP	192.168.1.10	209	Genuine	Genuine
ICMP	192.168.43.176	10	Genuine	Genuine
ICMP	192.168.43.177	3	Genuine	Genuine
ICMP	192.168.43.14	25718	Malacious	Malacious
ICMP	192.168.43.1	158	Genuine	Genuine
ICMP	google-public-dns-a.google.com	1	Genuine	Genuine

Table 1: Attack Counter Life Cycle

Now, according to the predefined rules in the tool, it concluded that the IP 192.168.43.14 is an attacker IP due to the large amount of packets sent in less than 10 seconds[10]. Thus, the tool will actively limit or block the traffic from that

 DOI : 10.18231/2454-9150.2017.0018

IP. Now, during the attack, the connection may temporarily disconnect for a short span of time. It will resume as soon as the attack is nullified.

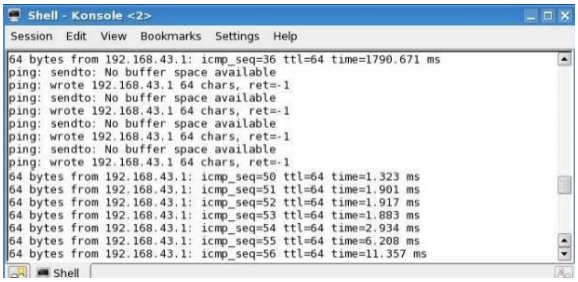

Fig 4: Re-establishing Connection

Figure 4 above shows the connection being reestablished after being temporarily disabled due to a simulated DDoS attack. To check the connection, we are continuously pinging the LAN server.

V. CONCLUSION & FUTURE WORK

As soon as the attack was initiated, the program detected it in a short span of time, and successfully blocked it. Results successfully showcase that the proposed tool is able to detect & block attacks. Not only was the attack blocked in a short span of time (in under 10 seconds), but the tool was able to restore the network immediately & further monitoring for more such attacks was resumed. In achieving this, a total throughput time of less than current existing systems was achieved. Given a high processing system, the throughput time can be even further reduced to as low as 1 second. Such results show a promising prospect of improved security over WAN networks.

India is a start-up oriented country. We have some small-scale firms/companies coming up every day. Most of these companies are dependent on the internet in one way or another. Such entities have limited resources & cannot afford expensive protection. Our system will be useful to all such small & large scale industries in defending their networks against DDoS attacks.

As we study more techniques of DDoS attacks, we can build further modules to nullify further attacks. The future work on this project includes adding all the other known types & techniques of DDOS attacks & constructing methods to detect such techniques & adding this functionality in the software.

VI. ACKNOWLEDGMENT

This project consumed a huge amount of work, research and dedication. It would not have been possible if we did not have a support of many individuals and organizations. Therefore we would like to extend our sincere gratitude to all of them.

The authors are thankful to our Project Guide – Milan Singh Thakur & our college professor - Govind Rao Mettu Sir along with our HOD – Dr. Ashok Kanthe Sir who all have given us crucial & valuable assistance & advice which enabled us to complete our project & come up with this idea to thwart DDoS attacks.

REFERENCES

[1] Bikram Khadka, Chandana Withana, Abeer Alsadoon, Amr Elchouemi, "*Distributed Denial of Service attack on cloud: Detection and prevention*", in International Conference and Workshop on Computing and Communication (IEMCON), 2015

[2] Akash Mittal, Prof. Ajit Kumar Shrivastava , Dr. Manish Manoria, "*A Review of DDOS Attack and its Countermeasures in TCP Based Networks*", in International Journal of Computer Science & Engineering Survey (IJCSES) Vol.2, No.4, Nov 2011

[3] Darshan Lal Meena, Dr. R. S. Jadon, "*Distributed Denial of Service Attacks and Their Suggested Defense Remedial Approaches*", in International Journal of Advance Research in Computer Science and Management Studies, Volume 2, Issue 4, April 2014

[4] "*Defeating DDOS attacks*", in Cisco Guard DDoS Mitigation Appliances, www.cisco.com

[5] John Ferrell, "*Firewall*" - Packet Queuing and Prioritization, on www.FreeBSD.org

[6] Peter N. M. Hansteen, "*Firewalling with OpenBSD's PF packet firewall*", at www.rlworkman.net, unpublished

[7] Snort User's Manual, at www.snort.org, unpublished

[8] OSI Model, at www.rstforum.net, unpublished

[9] Ipv4 header format, ICMP, TCP, etc packet header format, at www.cisco.com, unpublished

[10] Consultation with Milan Singh Thakur, APPSEC India Chair, International OWASP Speaker, Information Security Consultant, Security Blogger.

 DOI : 10.18231/2454-9150.2017.0018