An Intelligent Algorithm To Prevent The Creation Of Zombie Nodes For DDoS Attack In Client Server Architecture

Line 1: Haritha.A[#1]
Line 2: Computer Science Department
Line 3: Panimalar Institute Of Technology
Line 4:Chennai,India

line 1: Kanimozhi.S[#2]
line 2:Computer Science Department
line 3:Panimalar Institute Of Technology
line 4:Chennai,India

line 1: KauvyaKrishnaKumar[#3]
line 2:Computer Science Department
line 3:Panimalar Institute Of Technology
line 4:Chennai,India

line 1: Dr.S.Hemalatha[#4]
line 2:Computer Science Department.
line 3:Panimalar Institute Of Technology
line 4:Chennai,India

Abstract:

DDOS have been discussed significantly in the computer security domain particularly due to detrimental effect it causes to the organizational assets. There are many methodologies to launch a DDOS attack such as UDP Flood attack, SQL Injection, Brute Force attack, Ping of death, SYN flood, Denial of sleep attack. Existing solutions are divided into categories of machine learning solutions, distributed system solution or their combinations, but all are concentrated on the server side. The server is now burdened to process legitimate requests as well as to tackle the DDOS attacks. This paper proposes an intelligent algorithm which will be programmed and hosted in every candidate client machine. This intelligent algorithm will be responsible for processing the different requests that its host machine is issuing. Various parameters like the type of the request issue and frequency of request issue is considered to classify whether the request is a part of DDOS or a normal request. Since processing is done in the client side itself, all computational overhead is decreased significantly. Moreover, the server is less- burdened.

Keywords — *DDoS Attack, SQL injection, Brute force attack.*

I. INTRODUCTION

Networking is the most commonly used technology of the various means of communication available. The information is shared by the methodology of sending and receiving the request and response respectively in a FIFO manner. In this case, the server performance can be degraded due to multiple requests sent to the server by the clients. It may also happen due to the attack of Hackers. Websites become accessible to large number of users through internet, it may sometimes lead to overload of the server due to the maximum utilization. The result is server performance goes down and the processing time becomes slow.Due to the overload of the server, network traffic will be increased to corrupt server bandwidth.

DoS attacks are pernicious, a type of attack which checks on the availability of services and resources. It is an attack whose intention is to interrupt the normal traffic of a targeted server or a service. The request processing abilities of a server is exploited to process non-legitimate or rather unwanted requests from the attacker machine. Thus, the server/service is unavailable for the legitimate or rather the intended users. DoS attacks when occurs in a distributed environment is called DDoS. In DDoS scenario, the flooding non-legitimate requests come from multiple machines or systems which are in a disguised form. Thus recognizing, identifying such attacker systems and preventing them from further carrying out DDoS attack becomes a hectic, infeasible, time consuming, difficult task.

The targeted server of the DDoS attack is now burdened with an additional overhead of detecting these attacker nodes and then avoiding processing of requests from such attacker nodes. This consumes the CPU cycle and bandwidth of the server which is meant to be used for serving the legitimate and intended users. The intended users or the admin will come to know that a DDoS attack has taken place only when one of the following occurs: (i) unusually slow network performance (response to requests, access to files, sites); (ii) unavailability of a particular web site. (ii) inability to

access any web site. In general, DoS attacks are characterized or rather classified on basis of how they deny services: either by crashing services or by flooding services. Evidently the most dangerous attacks are distributed.

The immunity towards DDoS attacks is very minimal in all cases ranging from independent websites to multinational banks. In fact, a 2017 report from CisCo found that the number of DDoS attacks exceeding 1gigabit per second of traffic will rise to 3.1 million by 2021, i.e. a 2.5 fold increase from 2016. In most cases, DDoS attacks are designed merely to distract the target servers from criminal activities like data theft or network infiltration. That is the target is busy fighting off the DDoS attack, it then when the attacker would easily sneak in a piece of malware to carry out the criminal activity. Several mechanisms are adopted by the large organizations and data centres. Still there cyber security efforts to mitigate the impact of these attacks are not up to the mark. The year of 2018 saw a slight decline in DDoS attacks.

However, 2019 saw an 84% increase when compared tp 2018. This increase accounts to both the size and frequency of the DDoS attack. GitHub, a popular online code management service faced the biggest recorded DDoS attack. Its servers weren't prepared for the huge 1.3Tbps of traffic that flooded with 126.9 billion packets of data each second. GitHub used a DDoS mitigation service that was able to sense the DDoS strategy called memcaching. Dyn, one of the major DNS provider and a contributor of crucial part of the nework infrastructure of several companies like Netflix, Paypal, Visa, Amazon, had to face the most dangerous DDoS attack in October 2016. The hackers created massive botnets using a malware called Mirai to execute the DDoS operation.

There are several other instances of many major companies which had to face the DDoS attack. The list includes BBC in Decemeber 2015, Spamhaus in March 2013, Bank of America in December 2012 and many more.The DDoS mitigation strategies used up by the organizations or servers are all implemented on the server side. This is an extra burden for the server. Thus, we propose an intelligent algorithm which is to be programmed on the client side itself which monitors the request sent from its respective system, monitors whether it is a DDoS attack or a legitimate request and then sends he request to the server accordingly if needed.

We propose that this algorithm should be incorporated at the OS level itself with hidden and unmodifiable properties such that the hackers or crackers will not be able to tamper on it. By client side, we mean each and every system that is being used by an individual which is capable of achieving services from a server. Various parameters like frequency of request issued, type of request issued will be considered by the intelligent algorithm to classify whether its host system is a DDoS attacker or a legitimate client.

II.A. Section Headings

1) Level-0:

A user tries to access a file or upload a file to the local memory or even tries to gain access to the local memory through an application.During this process the log file is generated at the client side.

2) Level-1:

The local server or the local memory will analyse the log file as either an legitimate file or an attacker file using big data analysis and then categorise the client .If the client is found to be an attacker then that respective id will blocked from gaining access.

Figures :

A.Level 0:

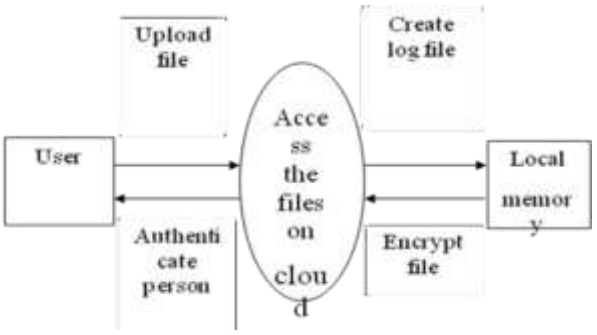

Fig 1: Client accessing the local memory

B.Level 1:

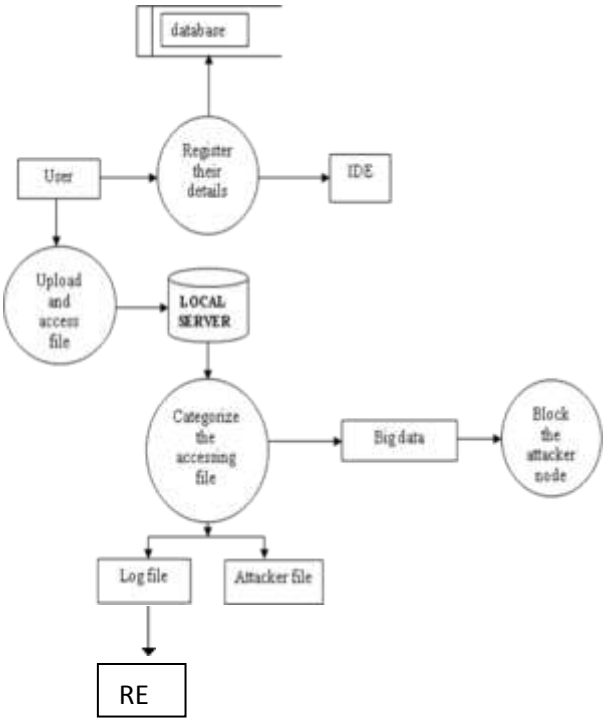

Fig 2:Categorizing the access

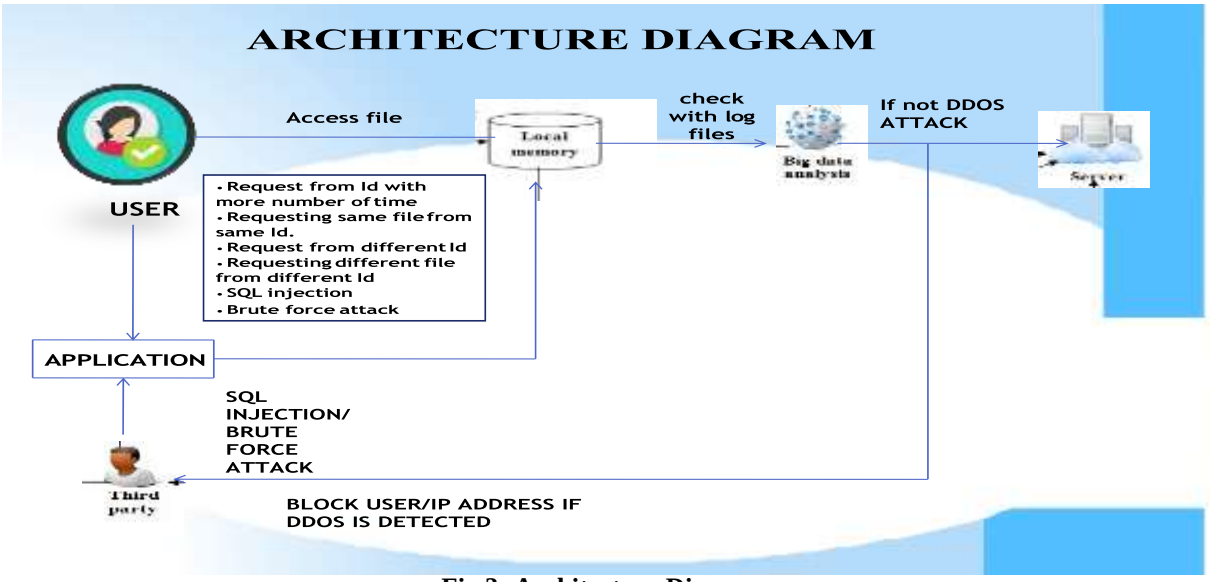

Fig 3: Architecture Diagram

II. PROPOSED SYSTEM

The proposed idea is that, an intelligent algorithm is employed at the zombie client itself which is capable of distinguishing between the legitimate and illegitimate requests from the client in which it is running. The six different attacks which is made by attacker:1) Same request from different Id within a time frame 2) Different request from same Id within a time frame 3) Multiple same request from one Id, within a time frame 4) Different request from different Id in a multiple time 5) SQL injection 6) Brute force attack After the classification of illegal requests from the legal ones the algorithm makes sure that those requests are never sent to the server.

III. MODULES

1. User Interface Design

In this Module User interface is created so that the user will send the request as a query to the server. The main objective of this module is to send requests to the server provided the request is forwarded to the server only if the request is genuine and out those attacks which are specified in the following modules.

2. Server

In this Module, server is deployed for fulfillment of request handling. Request is send to the server only if the request is genuine and out of DDOS, SQL injection and Brute Force Attack.

3. Big data Analysis

In this module, we deploy bigdata implementation to analyze the request . We deploy Big data implementation in the client side to analyze the user request. We implement DDOS attack, SQL Injection and Brute Force attack to analyze the request. When bigdata executes its job every slaves will assign for a job and finally it will shows the result as a output in the form of json or text file.

4. DDOS Attack Detection

User will raise a request to the server and the request is forwarded to the server only if the request is genuine. DDOS attack is detected 1. When same file is requested again and again from same user, 2. Different File request by same user within the time frame,3. Same file by different user from same IP, 4. Different File request by different user from same IP within the time frame.

5. SQL Injection Detection

In this module, SQL Injection attack is identified by requesting the server. This attack is deployed from the client end itself. An attacker who aims to execute SQL injection manipulates a standard SQL query to exploit non-validated input vulnerabilities in a database. There are many ways that this attack vector can be executed, several of which will be shown here to provide you with a general idea about how SQLI works.

6. Brute force attack Detection

In this module, Brute-force attack is identified by requesting the server. This attack is deployed from the client end itself. Generally, In Brute-force attack an attacker submits many passwords or passphrases with the hope of eventually guessing it correctly (the attacker might be someone who could easily guess the user data). The attacker consecutively checks all possible passwords and passphrases until the correct one is found, basically a trial and error method. Alternatively, the attacker can attempt to guess the key which is typically created from the password using a key derivation function.

IV. UML DIAGRAMS

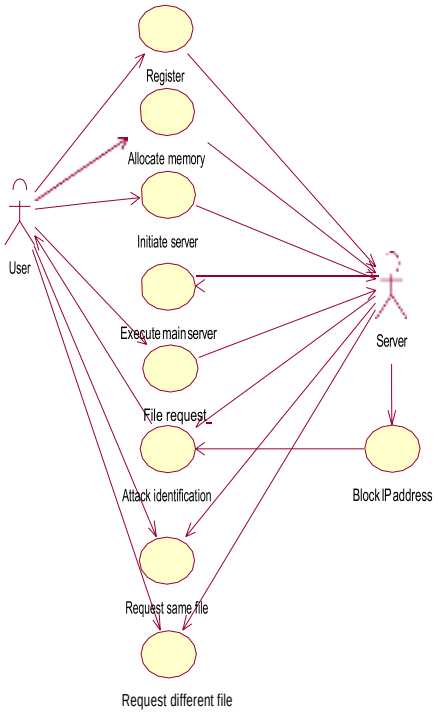

FIG 4: USE CASE DIAGRAM

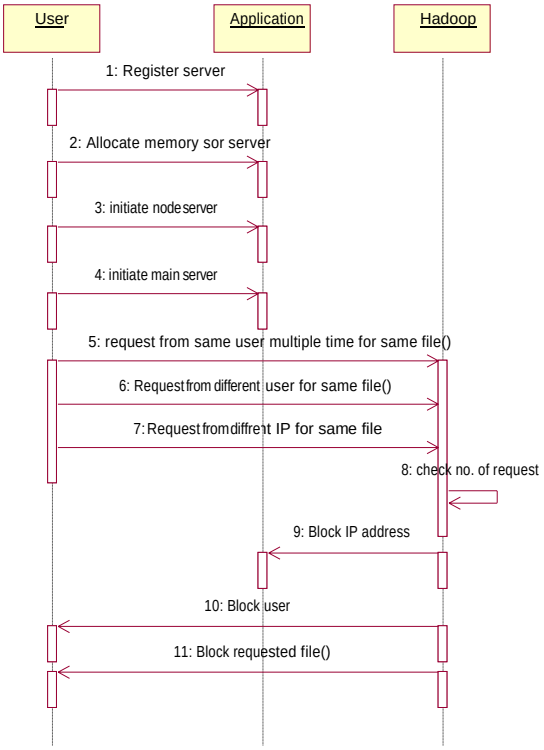

FIG 6: SEQUENCE DIAGRAM

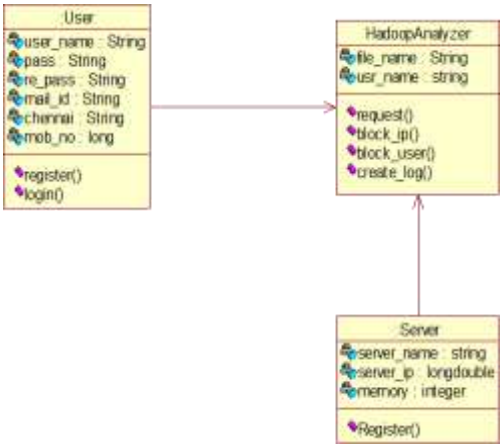

FIG 5: CLASS DIAGRAM

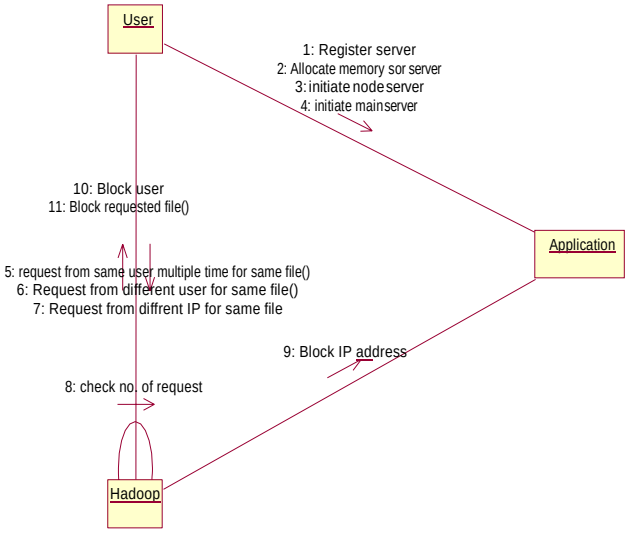

FIG 7: COLLABORATION DIAGRAM

V. CONCLUSION

Thus the paper conclude that through this system we identify the zombie clients, who try to gain access to the server to impart DDOS attack into the server ,their attacks and log file separately using big data analysis and categorize them into legitimate client file or an attacker file and provide access or block the ip accordingly from the client side itself.

REFERENCES

[1] A. Alsirhani et al "DDoS attack detection system: Utilizing classification algorithms with Apache Spark," inProc. 9th IFIP Int. Conf. New Technol. Mobility Security (NTMS),Feb. 2018.

[2] N. Miloslavskaya et al, "Application of big data, fast data, and data lake concepts to information security issues," in Future Internet of Things and Cloud Workshops (FiCloudW), IEEE International Conference on, pp. 148–153, 2016.

[3] E. Bertino, "Big data-security and privacy," in Big Data (BigData Congress), 2015 IEEE International Congress on, pp. 757–761, 2015.

[4] D. Rawat and K. Z. Ghafoor, Smart Cities Cybersecurity and Privacy. Elsevier, December 2018.

[5] S. M. T. Nezhad et al, "A novel DoS and DDoS attacks detection algorithm using ARIMA time series model and chaotic system in computer networks"

[6] B. Jia et al "A DDoS attack detection method based on hybrid heterogeneous multiclassifier ensemble learning".

[7] S. Lakavath et al, "A big data Hadoop architecture for online analysis," Int. J. Comput. Sci. Netw. Security, vol. 15, no. 11, pp. 58–62, 2015.

[8] Powerful Attack Cripples Majority of Key Internet Computers. Accessed: May 11, 2018. [Online].

[9] Mydoom Lesson: Take Proactive Steps to Prevent DDoS Attacks—Computerworld. Accessed: May 11, 2018. [Online].

[10] /mydoom-lesson–take-proactive-steps-to-prevent-ddos-attacks.html DDoS: Lessons From Phase 2 Attacks—BankInfoSecurity.

[11] 5 Biggest DDoS Attacks of the Past Decade.Available: https://www.abusix.com/blog/5-biggest-ddos-attacks-of-the-past-decade

[12] DDoS Attacks of 2017. Accessed: February 11, 2020. : /