ISSN 2395-1621

Implementation of Detection and Protection Mechanisms against Client Based HTTP Attacks

[#1]Shweta Shelke, [#2]Prof Santosh Shelke

[1]shweta.shelke28@gmail.com,
[2]santo.shelke@gmail.com

[#1]ME Scholar, CSE,
[#2]Asst. Prof., CSE

SAE, Kondhwa Pune, Savitribai Phule Pune University, India, 411048.

ABSTRACT

Nowadays web applications have tight cyber security policy and used active measures like firewalls vendor patches to protect application from various Cyber-attacks. The intent of security measures to protect confidentiality, integrity and availability of resources. The widely used WWW environment provides rich set of target for motivated attackers. Many prominent web sites face so called Distributed Denial of Service (DDoS) attacks. Former security measures failed to provide completely satisfying protection against DDoS attack. In this paper categorize different forms of attacks like SQL injection, URL injection, Cross site Scripting attack and Brute force attack and will propose a system which will provide protection against these attacks.

Keywords: Cyber-attack, Cyber security, DDoS, SQL injection, Brute Force, XSS attack.

ARTICLE INFO

Article History

Received: 18th June 2017

Received in revised form :

18th June 2017

Accepted: 21st June 2017

Published online :

21st June 2017

I. INTRODUCTION

Cyber-attacks to internet application are as old as internet itself. Attacking tools are evolving and are easily available in cracking community. In today's world where monetary transaction have increased tremendously so any sort of compromise with security is not acceptable.

Cyber-attacks are deliberate exploitation of computer systems, technology-dependent enterprises and networks. Cyber-attacks use malicious code to alter computer code, logic or data, resulting in disruptive consequences that can compromise data and lead to cybercrimes, such as information and identity theft.
Cyber-attacks may include following consequences:

- Phishing, spamming, spoofing, spyware, Trojans and viruses
- DoS and DDoS
- Password sniffing
- System infiltration
- Unauthorized accesses
- Brand and reputation damage
- Breach of contract and violations of service level agreement

If you're favourite website is down. There's a chance of it's suffering from a DoS attack. This is more likely if the Site is an online shop or another site that relies financially on being online at all. Prominent web applications like Ebay, Amazon or Buy.com were victims of such attacks [1]. A DoS attack tries to make a web resource unavailable to its users by flooding the target URL with more requests than the server can handle. That means during the attack period, regular traffic on the website will be either slow down or completely interrupted. The architecture or DDoS attack is as shown in Fig1.

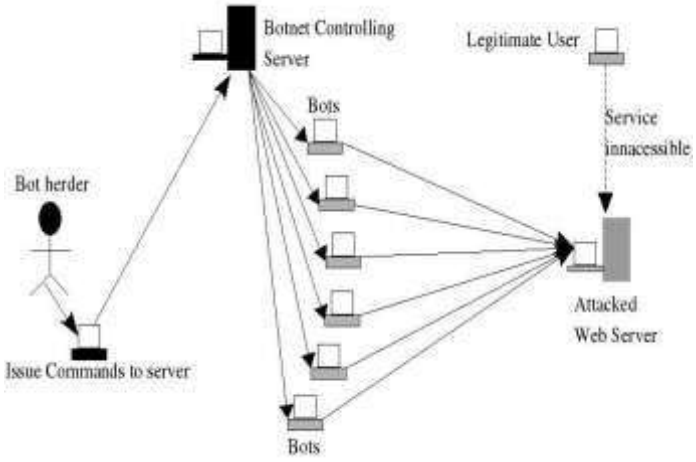

Fig 1: Architecture of DDoS

In other words DoS attacker sends huge number of fake requests to victims, which consumes the entire available resources as a result legitimate user's requests, will be denied. For example, the web site of European anti-spam company Spambhaus suffered the maximum attack traffic which reached 300GBps in 2013[2]. And traffic of DRDOS (Distributed reflection Denial of Service) which was launched via the network time protocol (NTP) reached 400GBps in 2014.

DDoS attack is a DoS attack that comes from more than one source at the same time. DDoS attack is typically generated using thousands of unsuspecting zombie machines. The machines used in these attacks are collectively known as "bot nets" and will have previously been infected with malicious software, so they can be remotely controlled by the attacker.

Cyber criminals combine DoS attacks and phishing to target online bank customers. They use a DoS attack to take down the banks website and then send out phishing emails to direct customers to a fake emergency site instead.

The motivation behind DDoS attack is financial fraud, extortion and competitive rivalry. Enterprises must pay attention to this threat and properly assess their environment and monitoring capability to protect and defend against these aggressive attacks. As DDoS attacks continue to evolve, it is critical not to underestimate the threat.

II. DIFFERENT ATTACKS

1. HTTP Flood Attack

It is type of DDoS attack http flood attack is layer 7 attacks that targets web application and service. Attacker exploits HTTP GET and POST request sent when an
HTTP client, web browser "talks" to an application server.

Attacker sends the victim server a large volume of GET and POST request to overwhelm the server's capabilities. The victim's server becomes busy in attempting to answer each request from the botnet which forces it to allocate maximum resources to handle the traffic. This causes DDoS means legitimate users cannot request to server.

Http flood attacks are volumetric attacks, often using a botnet "Zombie"- group of internet-connected computers, each of which has been maliciously taken over, usually with the assistance of Trojan horse. It does not use spoofing, malformed packet. It requires less bandwidth than other attacks. The HTTP attacks are very difficult to differentiate from valid traffic because they use standard URL request.

2.Brute Force Attack

In cryptography, a brute-force attack is a cryptanalytic attack that can be used against any encrypted data. It consists of systematically checking all possible keys or passwords until the correct one is found. Means attacker tries many passwords with hope of eventually guessing correctly.

When password guessing, this method is very fast when used to check all short passwords, but for longer passwords other methods such as the dictionary attack are used because of the time a brute-force search takes.Brute force Attack is a method of breaking a cipher, cracking a Password by trying every possible key. The Brute force attack is a method of defeating a cryptographic scheme by systematically trying a large number of possibilities. The Brute force attack is a large amount of time-consuming method due to the number of likely arrangement of letters, numbers and special characters [6, 7]. To come across the right Password additional complex brute force attacks entail trying all key combinations in an attempt. The attacker may have a list of words or regularly used Passwords and tries all sequences of words from beginning to end to achieve the access to a login account or system. Possibility of brute force attack depends on (i) the length of the Password (ii) the complexity of the Password (iii) the strength of the Password (iv) the amount of computational power being utilized to carry out the attack (v) how good enough the attacker knows the target. Even though a brute-force attack might be capable to achieve illegal admittance to an account in the long run, these attacks can take quite a lot of hours, days, months, and even years to run. With an increase in the length of the Password the resources vital for brute force attack scale raises exponentially this would not be linear [8].

To conquer from this brute force attack user can create their Passwords with the following criteria. (i) Length of
the Password is at least 10 characters long (ii) Must not be a dictionary word (iii) Repetition of Characters is to be avoided (iv) Password doesn't enclose frequent proper noun (v) Password with a mixture of all upper case, Lower case, Special characters and numbers [9,10].

3. SQL Injection

The concept of injection attacks is to inject (or insert) malicious code into a program so as to change structure of SQL query [11]. Such an attack may be performed by adding strings of malicious characters into data values in the form or argument values in the URL. Injection attacks generally take advantages of improper validation over input/output data. SQL Injection Attack or SQLIA is a type of code injection attacks which consist of injection of malicious SQL commands by means of input data from the client to the application that are later passed to the instance of the database for execution and aim to affect the execution of predefined SQL commands. There are a number of ways to prevent attacks made on the systems. In these ways a programmer uses different techniques in development cycle of application which contains uses parameterized queries, least privilege, different account, customized error message and etc.

SQL Injection Attack affects confidentiality, integrity and availability of information. SQL injection vulnerability is a type of attacks adds Structured Query Language code to a web form input box to gain access or make changes to data. By using this vulnerability an attacker could send his commands directly to web application's underlying

database or confidentiality.
Example:

*Set OK= execute("select * from Users where user=' "& form("user")&"' AND password=' "&form("password")&" ' ") ;*

If not OK.EOF

Login success

Else fail

In above example suppose user gives input as user=" 'OR 1=1--' "

Then script does

OK=execute(select……

Where user='OR 1=1-- …..')

In above example *"--" causes rest of line to be ignored.*

OK.EOF is always false and login succeed.

Suppose user=" ' ; DROP TABLE Users--' "

Then script does:

OK= execute(select…..

where user=' ; DROP TABLE Users….)

Else fail

In above example suppose user gives input as *user=" 'OR 1=1--' "*
Then script does
OK=execute(select……
Where user='OR 1=1-- …..')
In above example "--" causes rest of line to be ignored.
OK.EOF is always false and login succeed.

Suppose *user=" ' ; DROP TABLE Users--' "*
Then script does:
OK= execute(select…..
where user=' ; DROP TABLE Users….)

Query deletes Users table in database.

- Information Leakage and Improper Error Handling Broken Authentication and Session Management
- Insecure Cryptographic Storage Insecure Communications
- Failure to Restrict URL Access

XSS facilitates the hacker to insert some malicious script to the web application that may cause any kind of harm to legitimate user.

Example [13]
<%out.println("welcome"+request.getParameter("name")
); %> (Example of poorly –written code on Web server-saving it as test.jsp)
There are primarily three types of cross-site scripting attacks as follows [13]:

1. Stored or Persistent attacks
2. Reflected or Non – persistent attacks
3. DOM (Document Object Model)-based XSS attacks:

This exists within a site and can be used in a reflected manner. In this case, the malicious data exists solely in the browser and is not sent to the server. A brief example of a SQLIA classification based on Vulnerability is as shown below [11].

1. Bypassing Web Application Authentication:

This is the most common usage adopted by the attackers to bypass authentication pages, used in web applications.
In this category of attack, an attacker exploits an input field that is used in a query's 'where' condition part.

2. Getting Knowledge of Database Fingerprinting:

This attack is considered as pre-attack preparation by an attacker. This category of attack is performed by entering some inputs by which it generates an illegal or the logically incorrect queries. The error messages reveal the names of the tables and the columns that cause error. The attacker also comes to know about the application database used in the backend server.

3. Injection with UNION query:

In such an attack, an attacker extracts data from a table which is different from the one that was intended in the web application by the developer. An attacker exploits a vulnerable parameter to change the data set returned for a given query.

4. **Cross Site-Scripting(XSS)**

SQL Injection and XSS are vulnerability or loopholes which are being exploited by the hacker for malicious purposes.

The OWASP Top 10 report [12] lists the following as the ten most critical web application security vulnerabilities that are been exploited:

- Cross Site Scripting (XSS)
- Injection Flaws (SQL Injection, XPath Injection, LDAP Injection etc)
- Malicious File Execution
- Insecure Direct Object Reference
- Cross Site Request Forgery (CSRF)

DOM-based XSS attacks would be a modifying the web history of a user.

III.RELATED wORK

1. SDSNM: Software-Defined Security Networking Mechanism to Defend against DDoS Attacks[2]

In this paper the defects of IP network architecture are the fundamental reason and presented the necessary conditions of DDoS attacks, including connectivity, concealment and aggressiveness. SDSNM was proposed from removing or limiting these necessary conditions. SDSNM was able to restrict DDoS attacks under strict access control and attacker with the host in botnet can be found out under loose access control. But this doesn't provide communication efficiency and it is not tested in real network environment.

Distributed Capabilities-Based DDoS Defense[3] The system uses improved technique for capabilities based traffic differentiation and scheduling based rate limiting. They proposed a novel approach for predictions of attack to determine the prospective attackers.

Rate limiting scheme is based on pre-emptive scheduling. For this maintenance of blacklist, whitelist, graylist may impose memory overhead. Also resource availability needs continuous attention.

2. Feasible Method to Combat Against DDoS Attack in SDN Network[4]

In this paper they propose a feasible method to combat against DDoS attack. They can decrease the attack impact but not enough when the amount of attack traffic is very huge.

4. DDoS Attack and Countermeasures in Cyberspace

In this paper they present a detailed study of DDoS attacks on the internet, their counter measures and various DDoS attack mechanisms. Different defence Mechanism, Types of Defense techniques, their Strengths and Limitations are mentioned below.

Attack Prevention:

Types of Defense techniques

1. Ingress/Egress Filtering
2. Router-Based Packet Filtering
3. Source Address Validity
4. Enforcement (SAVE) Protocol
5. Hop count packet filtering (HCF)

Strengths

- Prevents source IP address spoofing of Internet traffic
- Prevents IP spoofing for static routes
- Filters attack traffic before it reaches the target, therefore, reduces collateral damage
- Handles flooding attacks without causing serious

Weaknesses

- Attacks with unspoofed source IP cannot be prevented
- Difficult to deploy RPFs globally to make it effective
- Cannot prevent DDoS attack with valid source IP address
- HCF cannot prevent bandwidth flooding
- DDoS attacks

Attack Detection:

- Defense technique: Conventional (High rate) DoS attack detection
- Signature Based Detection
  MULTOPS
  Spectral Analysis
- Anomaly-Based DoS Detection

Strengths

- The exact attack source or destination
- Addresses can be detected with accuracy
- Reduce false positives
- Minimizes the number of affected TCP flows

Weaknesses

- Data structure used for monitoring packet rates vulnerable to memory exhausting attack
- Only valid for TCP flows
- Need to trade-off between processing Speed and detection accuracy
- Defense technique: Low Rate Dos Attack Detection (LDoS)
- Dynamic Detection Method
- Periodic attack detection (PAD) and Modelled attack Detection Method (MAD)
- Packet and threshold percentage at target router's cache queue.
- Detection of attacks at the Edge Routers
- RTO randomization
- Detection Based on Self-Similarity

Strengths

- Simple and effective against spoofed IP packets
- Easy to implement and requires a very small storage space
- Easily deployable
- Improves TCP throughput in case of attacks
- With proper threshold set easy to differentiate between high rate and low rate DDoS attack

Weaknesses

- Fails against Distributed LDOS Attack
- Depends on accuracy of fuzzy controller designer
- Depends on Threshold value set
- Depends on known patterns matched
- Fails against Distributed LDOS Attack
- Depends on known patterns matched

IV. IMPLEMENTATION DETAILS

The main perspective of the Implemented system is to detect Hackers attacks. This is very helpful in Transaction based systems like Banking, online shopping, also in applications where we are storing personal data.

If the attack is conducted on a sufficiently large scale, entire geographical regions of Internet connectivity can be compromised without the attacker's knowledge or intent by incorrectly configured or flimsy network infrastructure equipment, to avoid such problems we are implementing this system. This system will give protection against DDoS attacks. We are using honeypot system to detect the attacks. Other module maintains the server and attack log. Using this attack log we will block the clients whose activities are malicious.

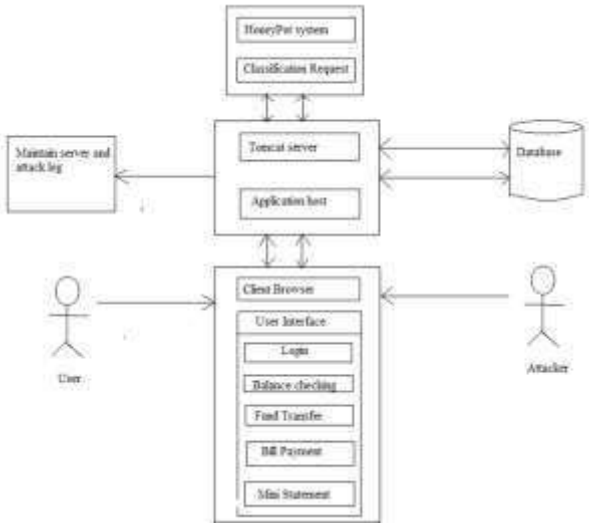

Fig 2: System Architecture

Some techniques we are going to use in system are as follows:

Honeypot system:

Honeypot system is a computer security mechanism set to detect, deflect or some manner counter act attempt at unauthorized use of information system. Honeypots are

- Pure Honeypots: Are full-fledged production systems. The activities of attacker are monitored by using a casual tap that has been installed on the honeypots link to network.
- High-Interactions Honeypots: Imitate the activities of the production systems that host a variety of services and, therefore, an attacker may be allowed a lot of services to waste his time.
- Low-Interactions Honeypots: Simulate only the services frequently requested by attackers.

AES 128 bit Encryption Decryption Algorithm:

The advanced Encryption Standard or AES is a symmetric block cipher used by the U.S. government to protect classified information and is implemented in software and hardware throughout the world to encrypt sensitive data.

The features of AES are as follows:

- Symmetric key symmetric block cypher
- 128 bit data, 128 / 192 / 256 bit keys
- Stronger and faster than Triple DES
- Provide full specification and design details
- Software implementable in C and Java

V. OBJECTIVE

Because of DDOS (Distributed Denial of Service attack) following things are happens with web server.

- Unusually slow network performance (opening files or accessing web sites)
- Unavailability of a particular web site
- Inability to access any web site.
- Dramatic increase in the number of spam emails received.
- Disconnection of a wireless or wired internet connection.
- Long-term denial of access to the web or any Internet services.

Main objective of Implemented system is to reduce such problems.

VI. IMPLEMENTATION OUTCOME

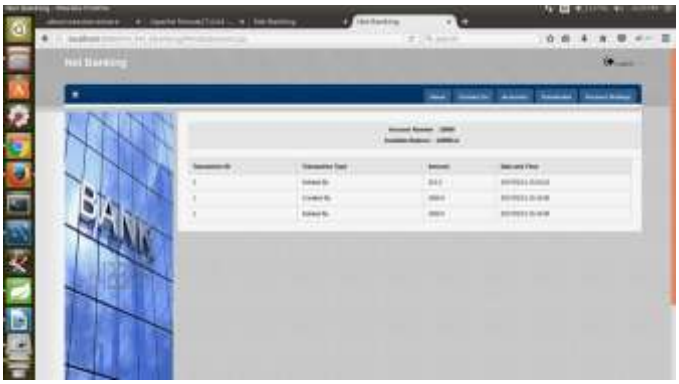

Fig 3. Transaction Details

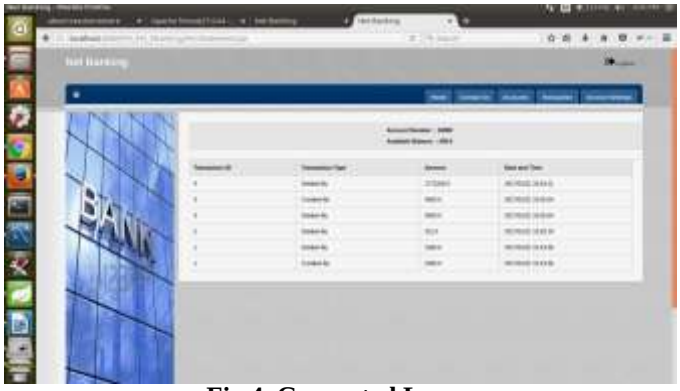

Fig 4. Generated Logs

Fig 5. Login Block

VII. Results Analysis

In the existing system, there is no provision for identifying the client intruder. It detects the innocent web proxy server. Here we proposed a new system that detects particular attacker node. It identifies the particular attacker node and has provision to notify the web proxy that a particular node is an attacker. There is provision for blocking the attacker node. The proposed system also has a technique to identify SQL injection, URL injection, Cross-site Script Attacks. We utilized TBAD algorithm which is highly accurate. Thus our proposed system provides mechanism which is capable of detecting DDoS attack along with identifying particular attack creating node. Thus the case of blocking innocent web proxy in other systems is avoided in this system and enables efficient attack management. Our Implemented system tested on the various aspects which proven the efficiency & reliability of the system.

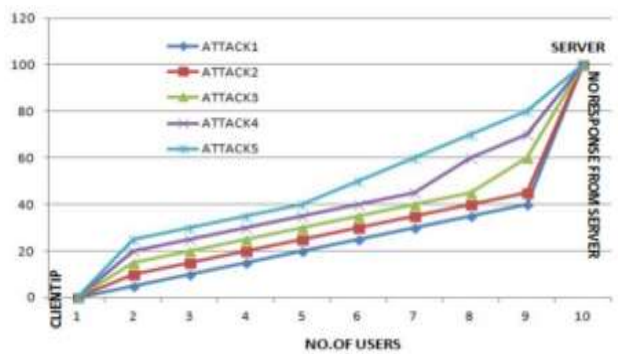

Fig 6. Result Analysis

The above Fig.1 illustrates the fact that there is almost four attacks which are all DDOS attacks occurs with number of users plotted against the time duration of the usage of the system by the users measured in milliseconds. As a result of these attacks, even though the clients expects and waits for the response from the cloud server the server does not register its response to the clients according to their requests. This increases the infra-structure response time.

VIII. CONCLUSION

In this paper, we present a brief survey of DDoS attacks, and some of the methods that have been developed for the detection and prevention of these attacks. The necessary factor in inhibiting a DoS attack is to enhance the reliability of global network infrastructure. We observed that most of the approaches deal with attack traffic detection and filtering near the target. We presented an architecture where we use an AES 128 bit encryption decryption Algorithm and Honeypot system. We also discussed various potential optimizations for improving performance. We believe that even at its current level, the overhead imposed is acceptable in many critical environments and applications.

REFERENCES

[1] Frank Kargl, Joern Maier, Michael Weber, "Protecting web servers from Distribute Denial of Service Attacks", in ACM, 1-58113-348-0/01/0005.

[2] Xiulei Wang, Ming Chen, Changyou Xing, "SDSNM: Software-Defined Security Networking Mechanism to Defend against DDoS Attacks", in IEEE FCST, 978-1-4673-9295-2, 2015.

[3] Manjiri Jog, Maitreya Natu, Sushma Shelke, "Distributed Capabilities-Based DDoS Defense", in IEEE ICPC, 978-1-4799-6272-3, 2015.

[4] Nhu-Ngoc Dao, Junho Park, Minho Park, Sungrae Cho, "Feasible Method to Combat Against DDoS Attack in SDN Network", in IEEE ICOIN, 978-1-4799-8342-1, 2015.

[5] Khan Zeb, Owais Baig, Muhammad Kamran Asif, "DDoS Attack and Countermeasures in Cyberspace", in IEEE, 978-1-4799-8172-4, 2015.

[6] Jim Owens and Jeanna Matthews "A Study of Passwords and Methods Used in Brute force SSH attack" In USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET), 2008

[7]Richard Clayton "Brute force attack on cryptographic keys"- file:////H:/brute force attack / brute.html, Oct 2001

[8] S. Vaithyasubramanian, A. Christy, D. Saravanan, "An Analysis of Markov Password against Brute Force Attack for Effective Web Applications", in Applied Mathematical Sciences, Vol. 8, no. 117, 5823 – 5830, 2014

[9]Carlisle Adams, Guy-Vincent Jourdan, Jean-Pierre Levac and Francois Prevost, "Lightweight Protection against brute force login attacks on web applications"PST, 181– 188, IEEE – 2010.

[10]http://www.infosecpro.com/applicationsecurity/a11.ht m

[11] Sayyed Mohammad Sadegh Sajjadi, Bahare Tajalli Pour," Study of SQL Injection Attacks and Countermeasures", in International Journal of Computer and Communication Engineering, Vol. 2, No. 5, September 2013

[12]OWASP Top 10, The Ten Most Critical Application Security Vulnerabilities, http://www.owasp.org/images /e/e8/ OWASP_Top_10_200 7.pdf

[13] Gurvinder Kaur," Study of Cross-Site Scripting Attacks and Their Countermeasures", International Journal of Computer Applications Technology and Research Volume 3– Issue 10, 604 - 609, 2014, ISSN: 2319–8