



INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X

Impact factor: 4.295

(Volume3, Issue4)

Available online at www.ijariit.com

Honeypot for Detecting Behaviour & Exposing Attacker's Identity for Dos and Ddos Attacks

Dilip Motwani

Associate Professor

Vidyalankar Institute of Technology

dmmotwani@gmail.com

Rachana Khorjuwekar

M.E Computers

Vidyalankar Institute of Technology

Rachana1994321@gmail.com

Abstract: Denial of Service attacks or Distributed Denial of service attacks are a big threat to the internet. Several methods, techniques, and proposals are introduced to deal with the attacks but none of it has given a successful result.

In this project, we aim to design a honey pot to deal with Denial of service attacks and Distributed Denial of service attack. A honey pot is a recent technology in the area of computer network security. It is a computer or network segment on the internet that is set up to attract and trap people who attempt to penetrate other people's computer system.

The project focuses on designing a honey pot which appears as the original network and traps the attacker by attracting it. The honey pot will identify the identity of the attacker using some browser exploitation technique and also record the pattern of attack done by the attacker. The advantages of the system are twofold: First, we can defend our operational network with a high probability against known Dos, DDoS and against new, future variants. Second, we trap the attacker so that recording of the compromise can help in a legal action against the attacker.

Keywords: Denial of Service, Distributed Denial of Service, Honeypot, Reload, Ping.

1. INTRODUCTION

Web applications often become the main target of attacks. A survey conducted by Open Web Application Security project (OWASP) has launched several common attacks aimed at the web application. [1]. the vulnerabilities in the web applications make some parties initiate the creation of a system that is specifically designed to observe the behavior of cracker. The system is then known as a honeypot. A honeypot is a system created to emulate service that runs on a server to observe the pattern of attacks. In general, a honeypot is divided into two main types based on the level of interaction with the attacker, namely high-interaction and low- interaction honeypot [2]. Low-interaction honeypot has a limited level of interaction because it only emulates a particular service on a system. In contrast, high –interaction honeypot has a high level of interaction because it uses the actual systems and services to be accessed by crackers. This leads high-interaction honeypot has a higher risk when compared with low-interaction one. By studying the patterns of attack, the protection of production systems can be formulated.

Based on the level of interaction, a honeypot can be categorized into three namely, low, medium and high-level interaction. Low-interaction honeypot gives attackers low interactions with the system and is used to detect and log attackers' connections to the honeypot. They are not suitable to collect detail and much amount of information about the attackers and are easy to configure, thus impose less risk on the network. Medium-interaction honeypot gives better interaction level to the attackers. They respond to a connection requests with emulated replays that resemble the real systems. Medium interaction honeypot imposes a medium risk to the network and also give medium depth of information. A high-interaction honeypot is constructed using real Operating Systems and real applications running on systems. Attackers are allowed to highly interact with the systems so that much deeper and detailed information about their tactics and methods can be obtained. The advantage of these kinds of honeypots is that it allows the attackers to do much more activities and this helps to fully understand the attackers' intent, tactics, and skills. The main disadvantage of this implementation, in addition to the difficulty to setup and maintain, is it's highly risky.

A honey pot is a security resource whose value lies in being probed, attacked or compromised.

Honeypot has several purposes such as:

- I. To distract attackers from valuable machines on a network.
- II. To provide early warning about new attack and exploitation trends.
- III. To allow in-depth examination of adversaries during and after exploitation of a honeypot.

1.1 Denial of service and distributed denial of service attack

A **denial-of-service attack (DoS attack)** is a cyber-attack where the perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely disrupting services of a host connected to the Internet. Denial of service is typically accomplished by flooding the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled.

A **distributed denial-of-service (DDoS)** attack occurs when multiple systems flood the bandwidth or resources of a targeted system, usually one or more web servers. Such an attack is often the result of multiple compromised systems (for example, a botnet) flooding the targeted system with traffic.

Both types of attacks have in common that they typically use a limited number of well-known attacks sometimes in different combinations. A DoS attack's main characteristics are that an attacker attempts to prevent one or more legitimate users of a service from the use of the required resources. Therefore, he attempts:

- (1) To inhibit legitimate network traffic by flooding the network with useless traffic.
- (2) To deny access to a service by disrupting connections between two parties,
- (3) To block the access of a particular individual to a service,
- (4) To disrupt a specific system or service itself. DDoS attacks follow the same path.

2. PROBLEM DEFINITION

The project is to design a network based high Interaction honeypot which is suitable for a single machine or a client server environment. The honeypot is to be designed to prevent the attackers to exploit invalidated and unsanitized user input with different attacking methods. It mainly focuses on dealing with the attacks such as Denial of service and Distributed denial of service attack.

The major aim is to design a honeypot to

- expose the identity of the attacker and
- Understand and analyze the behavior of the attacker.
- Record the pattern of the attack done by the attacker
- This system records usernames and passwords that are attempted by an intruder from the Internet. It also captures detail activities of the attackers while they are interacting inside the target honeypot.

The honeypot collects small sets of data it stores everything that it comes in contact with. The honeypot is designed to only have interaction with attackers. This way it collects smaller sets of data with very high value. Also, by capturing anything they come in contact with it, a honeypot can detect any new tools or technologies used by attackers. The most important and useful advantage is its simplicity resources required are minimal. This decreases cost in deploying a honeypot because an expensive, powerful computer is not necessary.

It not only focuses on identifying the identity of the attacker but by analyzing the behavior it can also overcome or avoids the attacks hence, becoming useful for the victim to protect the system and deal with the attacks.

The advantages of this system are two-fold: First, we can defend our operational network with a high probability against known DDoS and against new, future variants. Second, we trap the attacker so that recording of the compromise can help in a legal action against the attacker.

The project aims to identify the identity of the attacker, understand the behavior of the attacker, to analyze the pattern of the attacks, to determine the type of attack and then take appropriate measures to overcome and solve the problem.

2.1 Scope of the project

Honeypots are resources often used as decoys of IT assets. Honeypots draw the attention of attackers and inappropriate behaviors misaligned with operational policies, sometimes providing early insight into the very real and imminent threats to production resources. In the near future honeypot can be used on multiple websites to identify the attacker's pattern and type of attacks. The multiple attributes identified in the honeypot designed can be used to trap the attacker easily.

The DOS and DDOS attacks which we will implement in the project are reloaded attack and ping of death attacks.

Reload Attack: The reload attack is implemented in the attack by designing a login page using JavaScript. The attacker uses the page to send requests to the website continuously which creates traffic onto the site affects it.

Ping of Death Attack:

In this attack, the attacker sends a ping request that is larger than 65,536 bytes, which is the maximum size that IP allows. While a ping larger than 65,536 bytes is too large to fit in one packet that can be transmitted, it allows a packet to be fragmented, essentially splitting the packet into smaller segments that are eventually reassembled. Attacks took advantage of this flaw by fragmenting packets that when received would total more than the allowed number of bytes and would effectively cause a buffer overload on the operating system at the receiving end, crashing the system. Ping of death attacks is rare today as most operating systems have been fixed to prevent this type of attack from occurring.

2.2 Goal of the project

The main goal of the project is to identify the attacks done on the website. It majorly focuses on two attacks namely Dos and DDos attacks. The aim of the project is to identify the attackers

1. User Agent
2. User Host Agent
3. User Host Name
4. Time Stamp
5. Machine Name
6. Browser Name
7. Browser Version
8. Platform
9. Request Type
10. User Language

3. LITERATURE SURVEY

Aggressive Web Application Honeypot for exposing Attackers Identity

In this paper, they have built a web application honey pot that emulates XSS and SQL injection vulnerabilities found in web applications. In addition, it will dig up cracker's information using JavaScript code. If the request to a honeypot is a normal HTTP request, the honey pot will give a normal response anyway. However, if there is an indication of a threat, honey pot will then simulate these attacks and sends the response as if the attacks succeeded. For every request sent by attacker's browser, our proposed honeypot system will insert JavaScript codes into the response. These codes will be executed by the cracker's browser and collect certain information to be sent back to honey pot [8].

Implementing High Interaction Honeypot to study SSH Attacks

The system introduces in the paper implements high-interaction honeypot with Secure Shell installed to study common SSH attacks in Linux environment. This system records usernames and passwords that are attempted by an intruder from the Internet. It also captures detail activities of the attackers while they are interacting inside the target honeypot. Intruders attack SSH servers through dictionary and brute-force mechanism followed by the intrusion. This paper covers both dictionary attack and intrusion. Secure Shell (SSH), is an encrypted channel to communicate remotely which is used mainly in Linux and Unix-based operating systems. SSH uses port 22 to login into a remote machine using usernames and passwords. Even if, the username/password combination mechanism can be replaced with public key authentication, brute-force attacks against the SSH protocol become quite common. Attackers can create automated tools to attack SSH servers using either brute-force or dictionary-based attack methods. [9]

4. PROPOSED SYSTEM

The main point of the proposed system is building a web-based low interaction honeypot that can bite. Not only record attacker's request but also try to expose attacker's identity at the same time and prevent any further attacks from the same source by blocking its IP or MAC address and locating him geographically. The proposed system is aiming to classify the user by the request it sends to the server. If the user is a normal user it will be served normally. If it is an attacker, then send an emulated web page pretending as a real web page. Serve attackers attacks so as to give an effect as the attack is successful and send the desired output with attached JavaScript. The JavaScript runs on attacker's browser and sends the information like IP, MAC address, and attacker's social media account credentials. The information obtained is logged and the IP address is blocked for the further prevention of the attack.

3.2 System Design

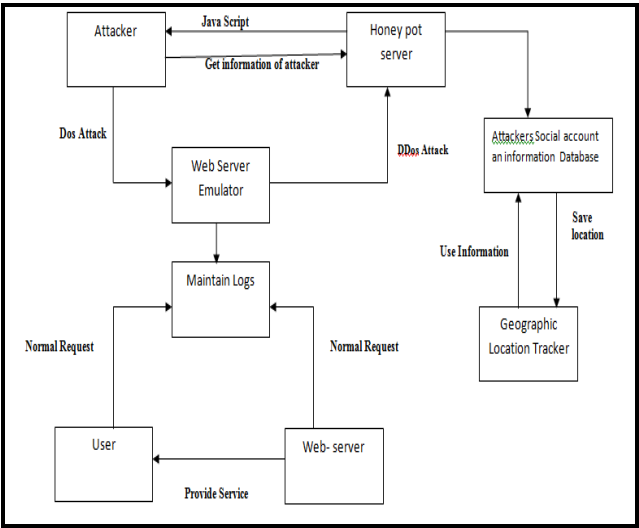


Fig.3.1. Proposed System

5. FLOW OF THE SYSTEM

The gym one website consists of two dashboards, the user dash board, and the admin dashboard. Home page visible to the user from where the user can register or login to the website is the user dashboard

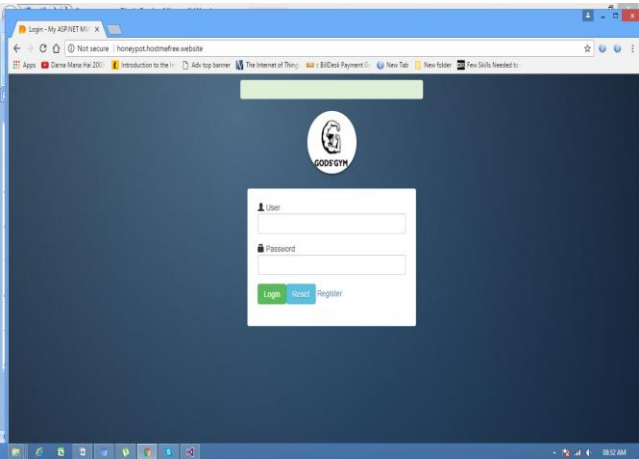


Fig.4.1.1. Home page

Once the user logs in to the website the home page consists of the options such as register member, payment renewal, payment listing, Receipt/candidate.

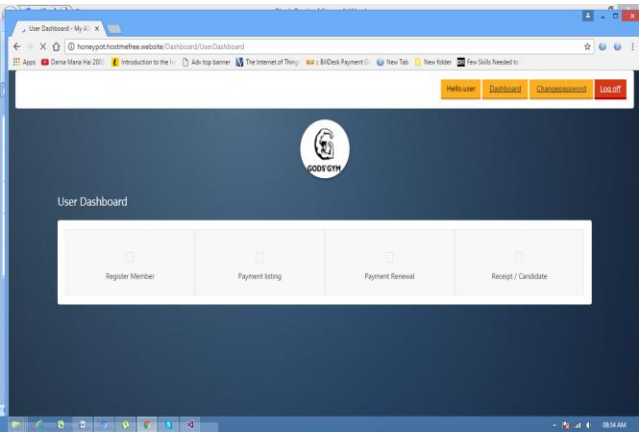


Fig.4.1.2. User Dashboard

If the user is a genuine user and logs into the website he is given access to the user dash board. But is the attacker tries to attack the website the record is immediately sent to the admin.

Now, the admin dashboard allows the admin to control the entire websites. The various options of the provided to the admin are as follows.

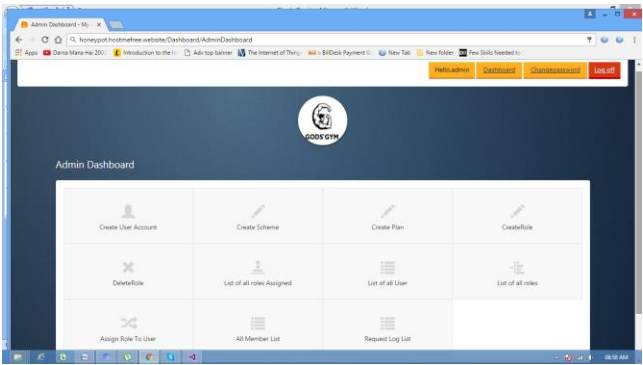


Fig.4.1.3. Admin Dash Board

Since the website for security purpose makes use of the honeypot facility to identify the pattern and the behavior of the attack an additional option is provided to the user where the user will be able to check the records in the logs to receive complete information of the attack.

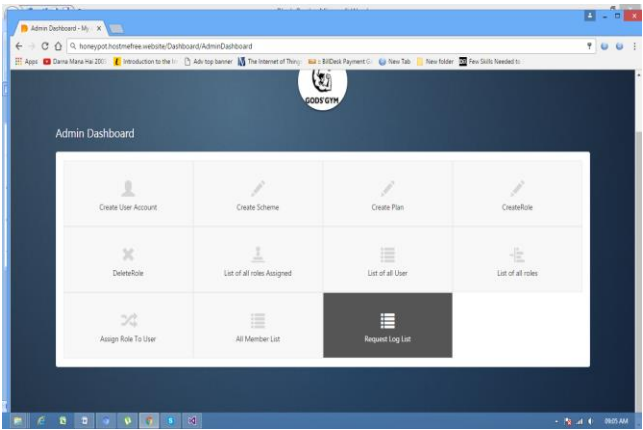


Fig.4.1.4. Request Log List

The honeypot which is designed for the gym one website focuses upon two attacks denial of service attack and distributed denial of service attack.

Once the attack is done the following information of the attacker is visible in the logs table

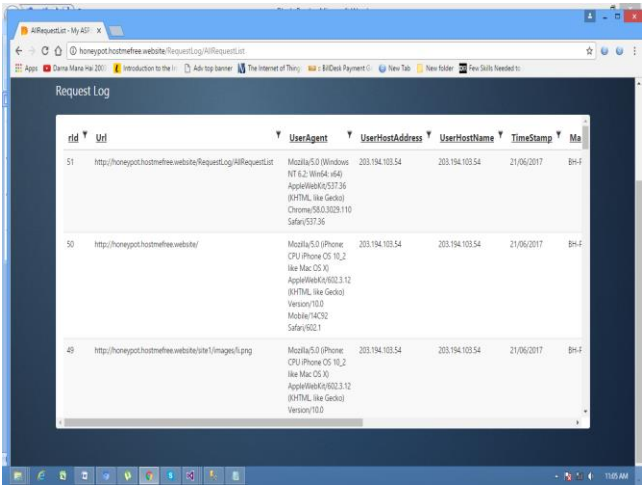
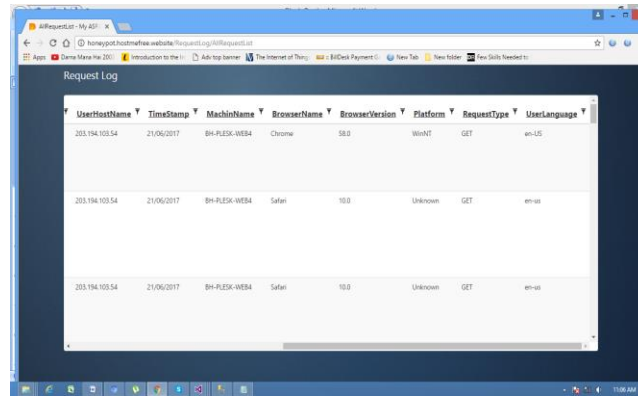


Fig.4.1.5. Logs



UserHostName	TimeStamp	MachineName	BrowserName	BrowserVersion	Platform	RequestType	UserLanguage
203.194.103.54	21/06/2017	BH-FLESK-WEB4	Chrome	58.0	WinNT	GET	en-US
203.194.103.54	21/06/2017	BH-FLESK-WEB4	Safari	10.0	Unknown	GET	en-us
203.194.103.54	21/06/2017	BH-FLESK-WEB4	Safari	10.0	Unknown	GET	en-us

Fig.4.1.6. Logs

CONCLUSION AND FUTURE SCOPE

In this thesis, we have discussed the various types of denial of service attacks and distributed denial of service attacks. The two attacks the have been implemented into the project are reloaded attack and ping attack for denial of service and distributed denial of service attack. We have analyzed the various functions features and uses of a honeypot. We have here implemented a honeypot for a gym website which allows the admin of the website to have a complete record of the DoS and DDoS attacks done on the website. The logs contain various information of the website such as the User Agent, User Host Agent, User Host Name, Time Stamp, Machine Name, Browser Name, Browser Version, Platform, Request Type, and User Language.

This system can be further improved by identifying the various other types of denial of service and distributed denial of service attacks. The honeypot above designed only tracks the identity of the attacker through the network and contains no personal data of the attacker. In the future, the design can be improved by even identifying the personal information of the attacker by using the technique of like jacking.

REFERENCES

- [1] OWASP, "OWASP Top 10 – 2013- The Ten Most Critical Web Application Security Risks", The Open Web Application Security Project, 2013.
- [2] L. Spitzner, "Honeypots: Tracking Hackers", Boston: Addison-Wesley Professional, 2002.
- [3] R. Chandramouli, "Battery power-aware encryption," ACM Transactions on Information and System Security (TISSEC), vol. 9, no. 2, pp. 162-180, May 2006.
- [4] L. Rist, S. Vetsch, M. Kobin, and M. Mauer, "Glastopf: A dynamic, low-interaction web application honeypot", The HoneyNet Project, 2010. [4] M. Muter, F. Freiling, T. Holz, Andl Matthews, "A generic toolkit for converting web applications into high-interaction honeypots", Clarkson University, New York, 2007.
- [5] A. Sintsov, "Hon Eypoth at can bite: Reverse penetration", Black Hat Europe Conference, 2013.
- [6] R. McGeehan, B. Engert, and M. Mueter, "Using Honeypots to learn about HTTP-based attacks", HoneyNet Project, 2008.
- [7] R. Barnett, "W ASC Distributed Open Proxy Honeypot Project", OW ASP, and W ASC AppSec Conference, 2007.
- [8] Aggressive Web Application Honeypot for exposing Attackers Identity 2014 1st International Conference on Information Technology, Computer, and Electrical Engineering.
- [9] Implementing High Interaction Honeypot to study SSH Attacks 2015 1st International Conference on Information Technology, Computer, and Electrical Engineering.
- [10] Web-based honeypot for detecting and tracking attackers www.ijariie.com