DDoS Attack using Botnets

Dr. T. Arumuga Maria Devi
Associate professor
Center for Information Technology and Engineering
Manonmaniam Sundaranar University

M. Arunima
PG Scholar
Center for Information Technology and Engineering
Manonmaniam Sundaranar University

K. Rooban Prakash
PG Scholar
Center for Information Technology and Engineering
Manonmaniam Sundaranar University

***Abstract*:- Botnets are becoming the single biggest danger to Internet security due to their constantly evolving harmful capabilities and capacity to infect the vast majority of machines connected to the Internet. This project's goal is to do a thorough examination of botnets and the flaws they took use of to expand themselves and carry out numerous nasty actions like DDoS assaults. Without a question, DDoS attacks are the most effective type of attacks committed by bot networks. To comprehend this expanding phenomenon better and To create efficient defenses, it's essential to be able to mimic DDoS attacks in a supervised setting. a DDoS attack simulation with customizable simulation settings.**

The assault attributes will provide us with information about how attacks sneak up on us and avoid detection. A thorough examination of suggested and existing DDoS defense methods paired with the learnings from simulation should allow us to develop creative and workable defenses against and countermeasures to DDoS attacks that use Botnets.

Index Terms: DDoS Attacks, Botnets, BYOB, Shell Access.

1.INTRODUCTION

A collection of security-compromised computers that have been infected with bots, or malicious computer programs, is known as a botnet. Numerous harmful actions, including distributed denial of service (DDoS) assaults, click fraud, spam, theft of banking information, identity theft, and theft of other sensitive data from infected machines, have been carried out via botnets (Collins, 2007). Large internal networks pose a particular hazard to enterprises because only one compromised machine in the internal network puts the entire network at risk. A botnet has the potential to be more profitable for attackers than viruses and worms, which has caused malware developers to concentrate increasingly on botnets. It is vital to be able to comprehend this expanding problem to create effective countermeasures to recreate them in a supervised setting.

This project's goal is to do thorough research on botnets and the vulnerabilities that they use to proliferate and carry out various destructive actions, such as DDoS attacks. Without a doubt, DDoS attacks are the most effective type of attacks committed by botnets. We can learn how attacks become stealthy and escape detection by simulating a DDoS attack with control over numerous simulation and attack settings. We should be able to develop creative and workable methods to prevent and mitigate DDoS attacks launched using botnets after conducting a thorough examination of the DDoS defensive tactics and ideas now in use and combining them with the knowledge gained from simulation.

DDoS ATTACK

A distributed denial-of-service (DDoS) attack is a malicious attempt to obstruct a server, service, or network's regular traffic by saturating the target or its surrounding infrastructure with an excessive amount of Internet traffic.

By using numerous compromised computer systems as sources of attack traffic, DDoS attacks are made effective. Computers and other networked resources, like IoT devices, can be exploited by machines.

When viewed from a distance, a DDoS assault resembles unexpected traffic congestion that blocks the roadway and keeps ordinary traffic from reaching its destination.

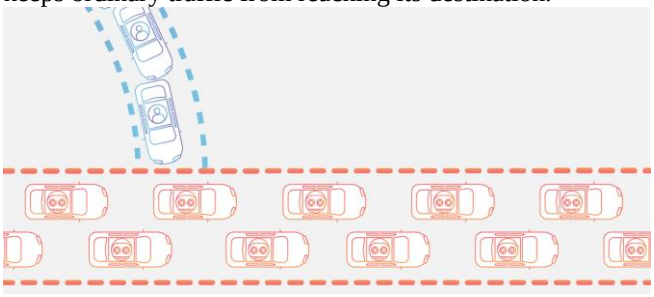

Fig:1 DDoS Attack

BOTNETS

A botnet is a collection of vulnerable computers that have been infected with bots, a type of malicious software. An attacker can take full control of the compromised machine once it has been infected with a bot and set up to its specifications. The terms "botmaster" and "botnet herder" both refer to the person who attacks or spreads a botnet.

Architecture for Botnet Control and Communication

Using the control and communication infrastructure, the botmaster orchestrates the distribution of directives to the whole botnet. Both botnet builders and security researchers looking to identify and stop botnets are very interested in the control and communication architecture of botnets. An efficient detection system for an entire genre of botnets and their variants can be created by having a thorough grasp of the architectural strategy used by a certain botnet.

Traditional Botnet architecture vs. centralized architecture

The bulk of current and conventional botnets have a centralized command and control architecture for their operations. All of the bots in this type of network are directly linked to a small number of specialized computers known as command and control servers. These few command and control servers are the only means by which the botmaster communicates with the whole botnet. In terms of scalability and the simplicity with which the botmaster may give commands to his whole network of infected computers, the classic design of centralised command and control servers is quite effective. However, it has a significant drawback. The botmaster may lose control of all connected computers if some of these command and control servers are discovered or fail via means of those servers. 2007's Usenix The identities (IP addresses) of these constrained command and control servers become known after a bot is seized. Therefore, in this architecture, the command and control servers serve as a single point of failure. Newer strains of botnet use a P2P architectural approach to try and minimize some of the issues with centralized command and control architecture.

BYOB

For people interested in learning about offensive security, BYOB is meant to be a tool that is user-friendly for beginners. Details about the project's architectural and design decisions are provided here. This web application makes use of a modified version of the console-based program me. BYOB that was made to work with a web-based front-end.

STEP1: Install byob tool, BYOB is an open-source post-exploitation framework for students, researchers and developers. It includes features such as:

- Command & control server with intuitive user-interface
- Custom payload generator for multiple platforms
- 12 post-exploitation modules

It is designed to allow students and developers to easily implement their own code and add cool new features *without* having to write a C2 server or Remote Administration Tool from scratch.

Step 2: Generate a Payload

Once you are logged in, the first thing you need to do is generate a payload. To get started, click the Payloads button at the top of the screen to navigate to the payloads page.

The payloads page has 2 main parts:

- Payload Generator
- Payloads Table

Payload Generator

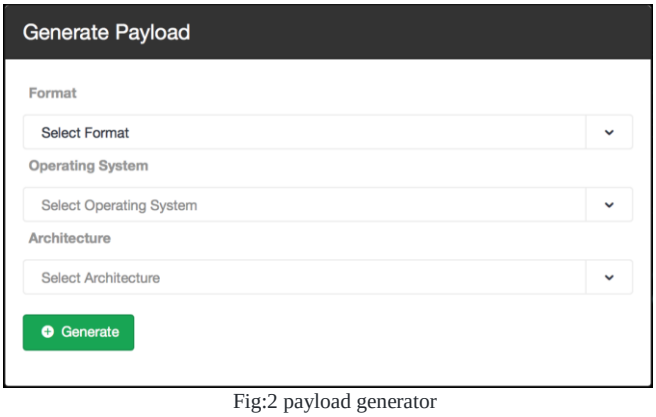

Fig:2 payload generator

Python

Select "Python" format as the format to generate an obfuscated Python script. Python is platform independent, so you will not need to select a target operating system and architecture. The file size is small, however, Python must be installed on the target machine to execute it.

Executable

Select "Executable" format to compile a binary executable for a target operating system and architecture. You must select the operating system and architecture of the target machine(s) in order to compile an executable. This will run on systems which do not have Python installed, however, the file size is substantially larger.

Payloads Table

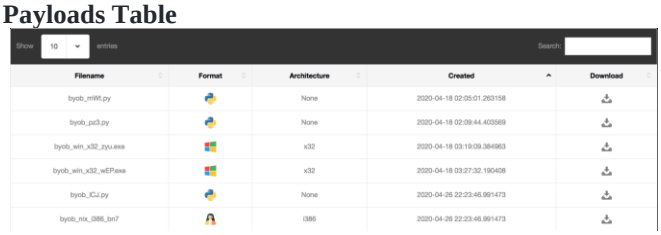

Fig:3 payloads table

Step 3: Create Bots

After downloading a payload, you can create bots by executing the payload on target machines. This platform is strictly for authorized testing and education purposes, so this is done by simply downloading the payload onto your virtual machine or testing environment. In a real world scenario an attacker would most likely use a social-engineering trick to get the target to execute the payload, such as sending it as an email attachment disguised as a software update.

Step 4: Command & Control

Once you have a payload running on a target machine, you are ready to command and control your bots! To get started, click the Control Panel button at the top of the screen to navigate to your command and control dashboard. The control panel dashboard is designed to provide an intuitive command & control interface. It has 3 main parts:

- Post-Exploitation Modules
- Bots Table
- Shell Access

Post-Exploitation Modules

Select a post-exploitation module using the panel on the left. Each module contains a description and a list of supported platforms. Next, select the bots to execute it on by either

clicking "select all bots" or selecting bots from the Bots Table below. Now click execute and watch the results stream in.

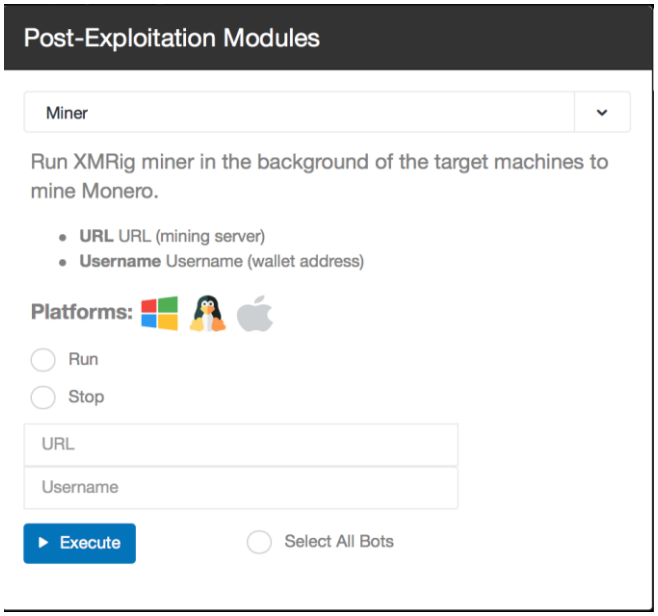

Fig:4 post-exploitation modules

Bots Table

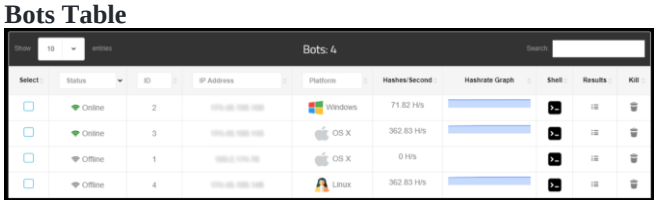

Fig:5 bots table

The bots table lets you keep track of your bots' status and have direct conversations with them because it contains their unique identifiers. For your convenience, it is completely searchable and columnar sortable. The "Hashes/Second" column and "Hash rate Graph" will update every second if your bots are mining Montero, allowing you to keep track of their progress in real time. By selecting the "Results" button on the right, you can see a bot's command history and results. You can also delete a bot by selecting the Trash icon. And sure, the bot has direct shell access when you click the terminal icon.

Shell Access

Click the terminal icon for any bot to connect directly to the bot via reverse TCP shell. A fully-featured terminal emulator runs in the browser which behaves exactly the same as the terminal on the machine. This provides you with direct access full control over the machine so you can run standard red team operations which require terminal access.

Fig:6 shell access

CONCLUSION

Prevention is the best medicine, and this couldn't be more true for DDoS attacks. Equip your network, applications, and infrastructure with multi-level protection strategies. This may include prevention management systems that combine firewalls, VPN, anti-spam, content filtering and other security layers to monitor activities and identity traffic inconsistencies that may be symptoms of DDoS attacks.

REFERENCE

[1] https://www.dsm.net/it-solutions-blog
[2] https://www.techtarget.com
[3] https://www.cloudflare.com

AUTHOR'S PROFILE

Dr. T. Arumuga Maria Devi Received B.E. degree in Electronics & Communication Engineering from Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India, in 2003, M.Tech degree in Computer & Information Technology from Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India, in 2005, also received Ph.D degree in Information Technology—Computer Science and Engineering, from Manonmaniam Sundaranar University, Tirunelveli, Tamil Nadu, India, in 2012 and also the Associate Professor of Centre for Information Technology and Engineering of Manonmaniam Sundaranar University since November 2005 onwards. Her research includes Signal Processing, Remote Communication, Multime- dia and Mobile Computing .

M.Arunima, M.sc.Cyber Security II year, Centre for information Technology and Engineering, Manonmaniam Sundaranar University, Abishekapatti, Tirunelveli - 627012, Tamilnadu, India. She received her Bachelor of Information Technology in Manonmaniam Sundaranar University.Her research interests include DDoS Attacks,Metasploit, Web app vulnerability, Digital forensics, Ethical hacking.

K.Rooban Prakash, M.sc.Cyber Security II year, Centre for Information Technology & Engineering, Manonmaniam Sundaranar University, Abishekapatti, Tirunelveli - 627012, Tamilnadu, India. He received his Bachelor of Networking in Madurai Kamaraj University. His research interests include USB rubber ducky, raspberry pico pi, and Metasploit, Web app vulnerability, Digital forensics, Ethical hacking.