Botnet Detection through DNS based approach

Amit Dange¹, Prashant Gosavi²

¹²M. Tech Computer Engineering Student, VJTI, Mumbai.

ABSTRACT
Botnets is group of compromised computers controlled remotely by attackers. Botnets create widespread security and data safety issues and are effective tools for propagating cyber-crime. It is imperative for the IT community to develop effective means of detecting and mitigating the malicious behavior of botnets. In this paper, we will understanding about the botnet and botnet detection by (a) creating a simple botnet and studying the flow of information among the bot and C&C server through activity diagram; and (b) study the detection of botnet using the DNS query patterns generated by the botnet.

Keywords: Botnet, Botnet Detection, DNS based approach.

1. INTRODUCTION
Today, computer networks are present everywhere from commercial organizations to governmental organizations. A significant threat to computer networks is botnet. A botnet is network of compromised computer machines which can be present anywhere from offices, home, school, business organization to government offices around the world. They work under the control of a single or group of user, commonly known as a Bot-master. Botnets have come to flood the Internet, largely unnoticed by the public.

Incident like DDoS attack on numerous Estonian websites[1], which is basically flooding attacks by botnets, and the detection of Stuxnet worm in July 2010 [1], which attempts to take control of critical physical infrastructure and connects to command and control server for updates, has shown the capabilities of the botnet to harm any potential targets.

The financial systems operating 24/7 linking intermediaries globally, power plants and electrical grids, gas and oil distribution pipelines, water treatment systems, oil and chemical refineries, transportation systems, and even essential military communications which rely heavily on an interdependent computer network, they become the potential targets for botnet. Therefore, the study of botnet and botnet detection has become the need of the time. In this paper we will define the working of the simple botnet and the detection technique. The paper is organized as follows: Section II will give details about the botnet and their detection. Section III will describe the creation of the simple bot and their communication topology and detection of the botnet. Section IV will conclude the paper.

2. LITERATURE REVIEW
2.1. Botnet Review
Botnet is defined as a collection of software “robots” that run on host computers autonomously and automatically, controlled remotely by an attacker or attackers [2]. The typical botnet consists of a bot server and one or more bot clients. The most important part of a botnet is the so-called command-and-control infrastructure (C&C). This infrastructure consists of the bots and a control entity that can be either centralized or distributed. The control entity or Bot-Master communicates to bots through C&C channel, which sends commands to bots and stolen information to the Bot-Master. The C&C infrastructure typically serves as the only way to control bots within this infrastructure in order to operate efficiently. Therefore, the architecture of the C&C infrastructure determines robustness, stability and reaction time.

A typical botnet can be created and maintained in five phases including: initial infection, secondary injection, connection, malicious command and control, update and maintenance [3]. During the initial phase, the attacker, scans a target machine for known vulnerabilities, and infect victim machines through different exploitation methods. After initial infection, in secondary injection phase, the infected hosts execute a script known as shell-code. In connection phase, the bot program establishes the connection with C&C server. Upon establishment of C&C channel, the zombie becomes part of attacker’s botnet army. Now actual botnet activity is started i.e. malicious command and control phase. Bot programs receive and execute commands sent by Bot-Master. In Update and Maintenance phase, bots are commanded to be lively and updated.
Botnets are generally characterized with respect to the C&C mechanism used for communication [4, 5 and 6]. C&C mechanism specifies how the bots retrieve the commands from the botmasters. They can be characterized as Centralized and De-Centralized C&C architecture. In a centralized C&C architecture, all bots establish their communication channel with one, or a few single connection points. These connection points are the central servers to issue command to selected set of bot. The server can be IRC or web server, then the bots use IRC or HTTP protocol to communicate with C&C for communication respectively. In decentralized C&C architectures, loosely coupled links between the bots enable communication within the botnet and provide the basis for its organization. A common term for this class of botnets is peer-to-peer botnets, as this is the name of the corresponding network model. Each bot is a peer acting as a server and a client at the same time. Botconnect to each other when they join the botnet. In order to achieve this goal, bots has a list of known peers. Bots try to connect using the list and when a connection established bots exchange their lists to have an improved connectivity.

2.2. Botnet Detection Technique Overview

The botnet detection techniques can be categorized into Passive techniques and Active Techniques. Passive Techniques are those where data is collected through observation without tampering with the environment. Therefore, these techniques are transparent and unknown to the botmasters. The passive techniques are Packet Inspection, DNS monitoring and Honeypots.

In packet inspection the basic idea is to match various protocol fields, or the payload of a packet, against pre-defined patterns of abnormal or suspicious content. These patterns are also called detection signatures.

DNS-based approaches is based on the property that, in order to access the C&C server bots carry out DNS queries to locate the particular C&C server that is typically hosted by a DDNS (Dynamic DNS) provider. So DNS monitoring will be a good approach to detect Botnet DNS traffic and detect DNS traffic anomalies. This is most famous and easy technique to botnet detection.

A honeypot is defined as an “environment where vulnerabilities have been deliberately introduced to observe attacks and intrusions”. It is a computer system that is used as a trap to draw the attention to attack this computer system. All Honeypots have a unique concept. They are computer systems that don’t have any production value [7]. They have a strong ability to detect security threats, to collect malware signatures and to understand the motivation and technique behind the threat used by perpetrator. While using Honeynet for Botnet detection, we have to wait until one bot in the network infect our system then we can track or analyze the machine.

Active measurement techniques contain approaches that involve interaction with the information sources being monitored. Although this enables the performance of deeper measurements, their application may leave traces that influence results, or include activities that can be observed by the botherder. The active measurement techniques are Sink holing, Infiltration.

Sink holing is a technical countermeasure to cutoff the control source from the botnet. One of the ways to do this is to change the malicious domain name with the trusted domain name controlled by the investigator.

The ‘infiltration’ of botnets can be divided into software-and hardware-based techniques. Software-based infiltration extends the ideas of the enumeration approaches. Instead of emulating or modifying the bot software on a controlled host with the intention of joining the botnet and measuring it internally, infiltration goes a step further and aims to take control of the botnet. The other approach, hardware-based infiltration, may be applied if an IP address belonging to a command-and-control server has been identified and a relationship to a data processing centre or hosting company can be established. By obtaining a connection to a mirror port on the suspected servers, the communication can be wiretapped and analyzed.

3. PROJECT COMPONENTS

There are two major components of this work: 1) designing of the botnet 2) detection of the botnet.

3.1 Designing the botnet

While designing the botnet we have to answer the some question such as: What will be the C&C architecture? Which protocol will be used for communication by the bots? How the bots will be uniquely identified in the network? And what the bots can do?

Let’s look at this question one by one. What will be the C&C architecture? Here, we will be using the Centralized C&C architecture, all the bots will be reporting to the central point which will be the apache web server. Next comes, the...
protocol used for communication, as the C&C server is web server the protocol used will be the HTTP. To uniquely identify the bot in the network, each bot is tied to the hardware of the host machine. The bot look for serial number and compute the MD5 checksum and it will be used as the bot ID, and last question what the bots can do here, we are designing the botnet which will be able to sniffing traffic, sending the spam mail as instructed by the C&C, scan the network for vulnerabilities and capturing the screen shot of the infected machine. The block diagram of the botnet is given below.

![Bot Mode and Transition](image)

**Figure 4 Bot Mode and Transition**

The working of the botnet in different mode is as follows:  

**INIT Mode:** In init mode the bot will connect to the C&C for the first time. The bot will provide the password for establishing the connection with C&C if the password matches the C&C will establish the connection with the bot otherwise, it will terminate the connection with the bot.

**START Mode:** Once the connection is established, the bot will send the host detail such as bot-ID, hostname, OS, bot creation time and bot last updated time and register itself on the C&C.

**SNIFFING Mode:** Once the bot is registered the bot will start capturing the traffic. After some random time interval it will send this detail to C&C and wait for command from the C&C. The C&C will reply with either of the sleep, screenshot or spam command. After receiving the command the bot will perform the particular operation.

**SCREENSHOT Mode:** In the screenshot mode, the bot will capture the image of the current working window of the host computer and send it to the C&C.

**SPAM Mode:** In this mode, the bot will send the spam messages. The bot will receive the XML file from the C&C which will contain the information such as victim list, subject, body, return address etc required to the mail.

**SLEEP Mode:** If the C&C replies the sleep command then the bot will go into the Sniffing Mode and then again contact to C&C after some random time interval.

The communication between bot and the C&C starts with bot connecting to the C&C server with URL and password encoded in the bot. At this stage bot is in INIT mode. If the password matches then the C&C server will reply with either START or SLEEP command. If the command is SLEEP than the bot will again contact to C&C server after some random interval, but if the command is START then the bot will send the bot ID, hostname, OS Details, IP address and bot creation time to the C&C server. The C&C server will store this information into the database. Now, the bot will change state into SNIFFING mode and capture traffic and send it to C&C server, the C&C server then reply either with SLEEP, SPAM, SCAN or SCREENSHOT mode. If the command is SPAM mode then C&C will send the XML file containing the list of victim address, mail subject, content information and then bot will send the spam messages and return to SNIFFING mode. If the command is SCAN then the bot will scan the network to find the vulnerabilities and send it to C&C server and goes to the SNIFFING mode. If the command is SCREENSHOT then the bot will capture the image of the victim machine and send it to the C&C server and goes to the SNIFFING mode and if the command is SLEEP then it will directly go to SNIFFING mode. The more about botnet creation can be found here [8].
The communication between the Bot and C&C server is explained with the activity diagram in figure 2:

![Activity Diagram for bot and C&C Communication](image)

**Figure 5: Activity Diagram for bot and C&C Communication**

### 3.2 Detecting the Botnet

For detecting the botnet we will use the DNS based approach. The detection method is based on the property of botnet DNS query. The DNS queries can be differentiate from the legitimate DNS queries. The difference between the DNS queries and the legitimate DNS queries are: 1) only botnet members query about the domain name of the C&C server. Therefore, the number of IP address queried C&C domain is fixed. (2) The botnet DNS query occurs simultaneously whereas, the legitimate DNS query occurs continuously not simultaneously.

The detection algorithm makes use of this property. The algorithm compares the list of IP address accessing the particular domain in different time period. If there is similarity between this IP lists then the domain is declared as the botnet domain and blacklisted.

The algorithm work as follows:

1) Capture the DNS packet from the network for analysis.
2) Store the triplet <Domain Name, Source IP Address, Timestamp> into the database.
3) When the new packet arrives it checks whether the Domain Name and IP address is present in the database if not insert into the database.
4) Form the cluster of IP address for each domain name.
5) If the number of IP address in the cluster for a domain name is less than the threshold then delete their entries from the database. This is required to reduce the amount of data to be processed in later stage.
6) For each cluster of IP address of domain name, divide the IP list into the two lists A and B in respect to the time stamp $T_1$ and $T_2$. 
7) Calculate the similarity between the two lists as defined above. 
8) If similarity is closer to 0, then remove the IP lists for this domain and white-list the domain name. 
9) If similarity is closer to 1, then black list this domain as the botnet domain name. 

For comparing the similarity between the two lists Kulczynski similarity measure is used which is defined as 
\[ S = \frac{1}{2} \left( \frac{2C}{A+B} \right) \]
Where, 
A is the size of IP list in time t1. 
B is the size of IP list in time t2. 
C is the number of IP common in both lists i.e. \((A \cap B)\). 
If A = 0 and B = 0 then we define similarity as -1. If S is closer to 0, white list the domain and delete the IP address. And if the similarity is closer to 1 then black list the domain name as the botnet domain name. 

4. CONCLUSION

Since the computer networks are present everywhere their security has become the major concern. Here, we have studied the botnet which have become a one of the biggest threat of network security and major contributor to unwanted network traffic. We have studied the working of botnet by creating a simple botnet and studied how the DNS based approach can be used to detect the botnet domain and mitigate the botnet threat. 

References