A Wide Scale Survey on Botnet

Amit Kumar Tyagi
Department of Computer Science,
School of Engineering and Technology,
Pondicherry University
Puducherry-605014, INDIA

G.Aghila
Department of Computer Science,
School of Engineering and Technology,
Pondicherry University
Puducherry-605014, INDIA

ABSTRACT
Among the diverse forms of malware, Botnet is the serious threat which occurs commonly in today's cyber attacks and cyber crimes. Botnet are designed to perform predefined functions in an automated fashion, where these malicious activities ranges from online searching of data, accessing lists, moving files sharing channel information to DDoS attacks against critical targets, phishing, click fraud etc. Existence of command and control(C&C) infrastructure makes the functioning of Botnet unique; in turn throws challenges in the mitigation of Botnet attacks.

Hence Botnet detection has been an interesting research topic related to cyber-threat and cyber-crime prevention. Various types of techniques and approaches have been proposed for detection, mitigation and preventation to Botnet attack. This paper, discusses in detail about Botnet and related research including Botnet evolution, life-cycle, command and control models, communication protocols, Botnet detection, and Botnet mitigation mechanism etc. Also an overview of research on Botnets which describe the possible attacks performed by various types of Botnet communication technologies in future.

Keywords: Bot; Botnet; C&C mechanism; communication protocols; honeynet; passive traffic; attacks; defense; preventaation; mitigation .

1. INTRODUCTION
The term 'Bot' is nothing but a derived term from "ro-Bot" [40] which is a generic term used to describe a script or sets of scripts designed to perform predefined function in automated fashion. Botnet is the collections of bots or collection of compromised computers that are remotely controlled by its BotHerder [17]. The terms BotHerder and Botmaster are used interchangeably in the literature. Even though Botnets shows the trace of existence for several years ago, Botnet have only recently sparked the interest of the research community.

Generally *Botnet* is used to define networks of infected end-hosts, called *bots* that are under the control of a human operator commonly known as a *Botmaster*. Botnets recruit vulnerable machines using methods utilized by other classes of malware (*e.g.*, remotely exploiting software vulnerabilities, social engineering, etc.) [5], these machines create a C&C infrastructure between them to perform malicious activity. Hence the following services are provided by bots to its Botmaster[38]:

- Robust network connectivity
- Individual encryption and control traffic dispersion
- Limited Botnet exposure by each bot
- Easy monitoring and recovery by its Botmaster

Hence in the mechanism of C&C, it will disseminate the Botmasters's commands to their bot armies, where these channels can operate over a variety of (logical) network topologies and use different communication mechanisms, from established Internet protocols to more recent P2P protocols [5].

Now in general the main difference between Botnet and other kind of malwares is the existence of C&C infrastructure. Hence in the mechanism of detection of Botnet, if we identify the location of C&C then Botnet can be detected, removed and prevented from various types of cyber-crimes. But this depends on the weakness and strengths in communication protocol which is adopted by Botnet to perform malicious attacks. Now on the other side, bots are used by search engines to spider online website content and by online games to provide virtual opponents e.g. the games sometimes we play against computer while online, bot act as our artificial opponents [DALNET] for e.g. Google bot, Google search engine use Google bot to search any information from its database. More specifically on Internet relay chat (IRC) network bot's function in channels include managing access lists, move files, share users, share channel information, anything else if right scripts are added. IRC bots are automated and controlled by events which could be commands given in a channel by other IRC bot or client with necessary privileges.

In this paper, an overview of current Botnets technology research has been provided. The remainder of the paper is organized as follows: Section 2 discusses background of Botnets. Section 3 describes related work about bots and Botnet terminology. Section 4 describes Botnet phenomenon. In this section, Botnet characteristics, and Botnet life-cycle are explained to provide better understanding of Botnet technology. Classification of bots is explained in Section 5. Section 6, describe about the communication protocols used by Botnet to communicate. Section 7, explain Botnet attacks which are till now traced and measured. Section 8 classifies Botnet detection approach which is explained in two classes: Honeynet based and passive traffic monitoring. Furthermore, it summarizes Botnet detection techniques in each class and provides a brief comparison of these techniques. In section 9, Botnet mitigation strategies to reduce the effect of Botnet

have been discussed. In section 10, we explain about the further Botnet attacks possible or Botnet developments in future. Section 11, concludes the total work done in this paper.

2. BACKGROUND OF BOTNET

In order to discuss more about Botnet it is required to know about bot, its origin, and effect of Botnet in real world situation. It is also required to know that how an attacker changed the trend to use Botnet in searching or in providing better service to users and used bot as malicious activity more. First bot is created in 1988 but it was not as a malicious activity. First malicious bot is created in 1998[5]. And after 1998, much more type of Botnet is traced as a dangerous activity to online user. This section discuss about the key terms to understand better about the Botnet, and a timeline of bot and Botnet evolution.

2.1 Definitions

Some definitions related to Botnet and its malicious activity is described here:

1. *Bot*: It is typically an executable file, capable of performing a set of functions, each of which could be triggered by a specific command [40]. A bot when installed on victim machine copies itself into configurable install directory and changes system configuration each time system boots.
2. *Victim machine*: It is the compromised internet host on which the malicious bot is installed after the attacker has exploited an application or operating system vulnerability or has duped the user into executing a malicious program [40]. Once infected the target host are also referred to as Zombies.
3. *Attacker*: It is the one that configures the bot; it comprises a machine to install a malicious bot, controls and directs the bots once it joins the designated IRC channel.
4. *Control channel*: It is a private IRC channel created by the attacker as rendezvous point for all the bots to join once they are installed on infected machine and are online, it comprises of a channel name and a password 'key' to authenticate.
5. *IRC server: It* is a server providing IRC services, this could be a legitimate public service provider like DALNET etc. or another attacker's compromised machine to perform attack [40].

2.2 Botnet evolution

To discuss about Botnet history, the first bot IRC was invented in August of 1988 by Jarkko Oikarinen of the University of Oulu, Finland [44]. In 1989, Greg Lindahl, an IRC server operator, created the benevolent bot called GM which would play a game of Hunt the Wumpus with IRC users. Till now they are emerging with varieties of Botnets among them Eggdrop (non malicious bot) created by Jeff Fisher for assisting IRC channel management in 1993 has significance [28]. Actually these bots were not malicious or malicious activity has not been measured about these Botnets at that time. But after this, more Botnet emerged that is first malicious bot, GT-BOT, found in April 1998[5]. And now there, at least a hundred variant of GT-BOT are available which include IRC client, .mirc .exe as part of bot [40]. Pretty party worm [36] was the first worm which emerged to make use of IRC as a means of remote control in June 1999. In April 2002, Agobot's source code was published on many websites [20] and slapper [43] was the first worm with P2P communications. So many more different types of Botnet measured after 2002, on various communication techniques. And now recently Zeus Botnet [2009], spy eye [2010], Mariposa [2009], Asprox(a P2P Botnet) [2009][34] Botnet is measured in which mariposa Botnet infected 10.3 million computers all over the world [39] , and Zeus Botnet has the report of infecting over 3.6 million computers in the united states [15]. Hence as time passing, Botnet is also using the stronger techniques and performing attacks on a large scale. Fig-1, shows the evolution of Botnet.

Appendix-A, lists the various of bots with the following features: the time, who invented, Infected hosts, Architectural features, kinds of threats, preventation mechanism, detection method, how they perform and which operating system, they support. The table gives detail about the existing bots but the following bots (Bobax, Torping, Trojan, Donbot, Mega-D, Grum, Maazben, Onewordsub, Cheg, Wopla, Xarvester, Spamthru, Rambot, Internbot, Akbot, Gumblar, Social bot and Decbot) have not been mentioned. Since they belong to the same category of Agobot.

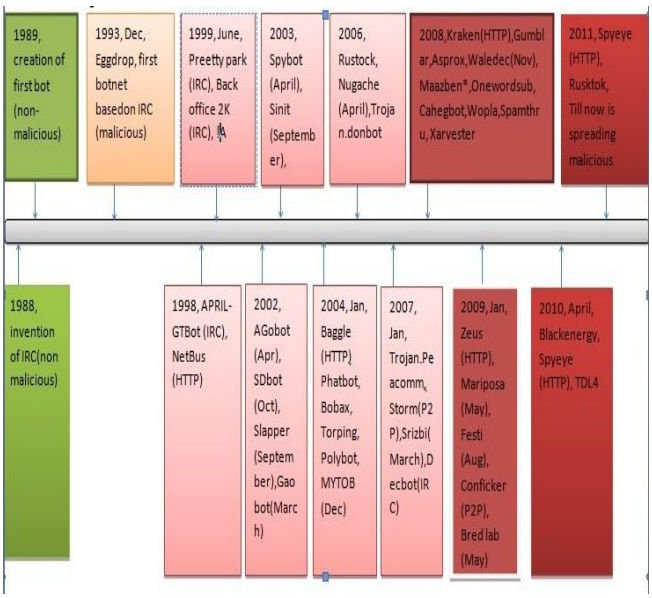

Figure 1-Evolution of Botnet technology

Now following schemes used to discover prevent from all Botnet which are explained in Appendix.A:

1. Patch, Patch, Patch
2. Teach user's safe computing habits that is user awareness,
3. Use up to date Anti-virus Signatures
4. Host-based Anti-virus
5. Network Intrusion Detection
6. Prevention Signatures Sets

3. RELATED WORK

As discussed, Bots and Botnets are hot topics for last few years due to measuring the large number of attacks through

cyber crime to steal valuable data of users. Basically a bot when perform an attack then for this, it distribute themselves across the Internet by looking for vulnerable and unprotected computers to infect. When they find an unprotected computer, they infect it and then send a report to the BotMaster. The bot stay hidden until they are informed by their BotMaster to perform an attack or task. Other ways in which attackers use to infect a computer in the Internet with bot includes sending email and using malicious websites. Based on our understanding, we could say that the activities associated with Botnet can be classified into three parts [18, 19]:

1. *Searching* – searching for vulnerable and unprotected Computers.
2. *Distribution* – the Bot code is distributed to the computers (targets), so the targets become Bots.
3. *Sign-on* – the Bots connect to Botmaster and become ready to receive C &C traffic.

The various Botnet infection measuring techniques and understanding the Botnet life-cycle, prevent, mitigate and defense from Botnet attack, various types of views, approaches, and tasks have been explained in this paper. So at last, to identify the C&C channel from a network, we will explain various types of techniques to detect Botnet.

4. BOTNET PHENOMENON

As discussed, Botnets are emerging as the most significant threat which is used to perform cyber –crime attack to steal the valuable data of users that is we can say Botnet perform attack after facing online ecosystems and computing assets [28]. Malicious Botnets are distributed computing platforms predominantly used for illegal activities such as launching Distributed Denial of Service (DDoS) attacks[21], sending spam[23], Trojan and phishing emails[21, 23], illegally distributing pirated media and software, force distribution, stealing information and computing resource, E-business extortion, performing click fraud, and identity theft [28] to financial gain. In the phenomena of Botnet, the high light value of Botnets is the ability to provide anonymity through the use of a multi-tier C&C architecture. Note that the individual bots are not physically owned by the Botmaster, and may be located in different locations spanning to itself as globe. And differences in time zones, languages, and laws make it difficult to track malicious Botnet activities across international boundaries [28]. So this characteristic makes the Botnet an attractive tool for cybercriminals, and in fact poses a great threat against cyber security. So for better understanding of Botnet phenomenon, its characteristics and its life-cycle has been explained here.

4.1 Botnet characteristics

Like the previous generations of viruses and worms, a bot is a self-propagating application that infects vulnerable hosts through exploit activities to expand their reach. Bot infection methods are similar to other classes of malware that recruit vulnerable systems by exploiting software vulnerabilities, Trojan insertion, as well as social engineering techniques leading to download malicious bot code[2, 11, 21]. According to measurement studies in [33] modern bots are equipped with several exploit vectors to improve opportunities for exploitation. However, among the other classes of malware, the defining characteristic of Botnet is the use of C&C channels through which they can be updated and directed. The multi-tier C&C architecture of Botnets provides anonymity for the Botmaster. C&C channels can operate over a wide range of logical network topologies and use different communication protocols. Generally Botnets are usually classified according to their command and control architecture [11, 14, 33, and 46].

According to their command and control architecture, Botnets can be classified as IRC-based, HTTP-based, DNS based or Peer to Peer (P2P) Botnets [46]. P2P Botnets use the recent P2P protocol to avoid single point of failure. P2P Botnets are harder to locate, shutdown, monitor, and hijack [22, 38]. According to the analysis in [33] the most prevalent Botnets are based on IRC protocol [6] with a centralized C&C mechanism due to the inherent flexibility and scalability of this protocol. Furthermore, there are several open-source implementations that enable Botmasters to extend them according to their demands [1, 33].

4.2 Botnet life cycle

A typical Botnet can be created and maintained in five phases. This is depicted in Fig. 2.

1. In first phase, firstly Botmaster infect victim host with Bot through the social engineering, mail attachments, automatic scan, exploit and compromise etc mechanisms.
2. In second phase, Bot connected to command and control channel
3. In third phase, Botmaster send command through IRC/HTTP/P2P C&C Channel to bots
4. In fourth phase, repeat, soon the Botmaster has a large number of army bots to control from a single point.
5. And in last phase, bots are updated with a new version or new business functionally through their operator which issue payload command.

Hence the above discussion elaborates all five steps about how a bot is infected to other hosts. In addition it also gives insight into how the Bot increase their quantity means its capacity on a network to perform malicious activity and harm the users.

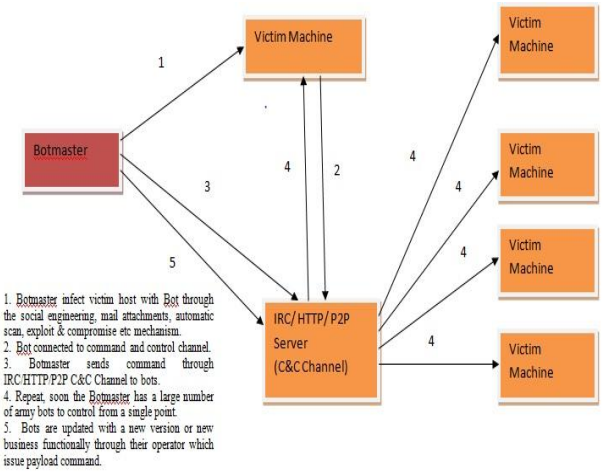

Figure2. A Typical Botnet Life-cycle

5 CLASSIFICATIONS OF BOTNET

5.1 Based on network protocols

For a Botmaster to send commands to a bot, it is essential that a network connection must be established between the zombie machine and the computer transmitting commands to control it. Here all network connections are based on protocols that define rules for the interaction between computers on the network. Botnets can be classified according to network protocols follow as:

5.1.1 IRC-oriented: This is one of the very first types of Botnet in which bots are controlled via IRC channels. Each infected computer connected to the IRC server (master) indicated in the body of the Bot program, and waited for commands [48] from its master on a certain channel (eg-IRC Botnet).

5.1.2 IM-oriented: This type of Botnet is not particularly common. It differs from IRC-oriented Botnets only in that it uses communication channels provided by IM (instant messaging) services such as AOL, MSN, and ICQ etc and due to the difficulty of creating individual IM accounts for each bot. The Biggest problem in this, Bots should be connected to the network and must remain online all the time [48] and each bot needs its own IM account to perform malicious activity. As result, owners of IM-oriented Botnets only have a limited number of registered IM accounts at their disposal, which limits the number of bots that can be online at any one time. Of course, they can arrange for different bots to share the same account, come online at predefined times, send data to the owner's number and wait for a reply for a limited period of time, but this is inefficient because it takes such networks too long to respond to their masters' commands to perform an activity.

5.1.3 Web-oriented: This is a relatively new and rapidly evolving type of Botnet designed to controlling zombie networks over the World Wide Web. A bot connects to a predefined web server (master), receives commands from it and transfers data to it in response. And wait to get a signal from its master to perform some activity for eg-HTTP Botnet.

5.1.4 Other: In this, there are other types of Botnets that communicate via only their own protocol that is only based on the TCP/IP stack, i.e., they only use transport-layer protocols such as TCP, ICMP and UDP.

5.2 Based on communication topologies

In this section we will describe about "how bot communicate" between each other. So according to the C&C channel, we categorized Botnet topologies into three different models, the Centralized model and the Decentralized model and Unstructured C&C Model [5].

5.2.1. Centralized model: Hossein et al [20] explain the model where, one central point (C&C server) has been used for exchanging commands and data between the Botmaster and Bots. Actually C&C server runs certain network services such as IRC or HTTP. So advantage of this model is small message latency which cause Botmaster easily arranges Botnet and launch attacks. Here, all connections and action performs through the C&C server; therefore, the C&C is a critical (weak) point in this model. If somebody manages to discover and eliminates the C&C server, the entire Botnet will be useless and ineffective. Fig -3 shows that, IRC and HTTP are two common protocols that C&C server uses for communication.

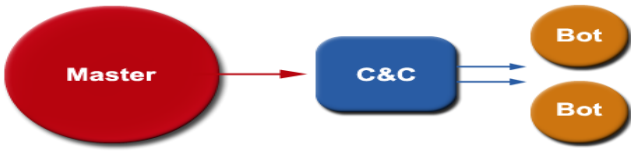

Figure 3. Command and control architecture of a Centralized model

5.2.2. Decentralized model: In this model the communication system does not completely depend on some selected servers, for discovering and destroying a number of Bots. As a result, attackers exploit the idea of Peer-to-Peer (P2P) communication as a Command-and Control pattern which is more resilient to failure in the network. Figure 4 shows that, depicts the decentralized (P2P) model where there is no Centralized point for communication. In this, each bot keeps some connections to the other Bots of the Botnet where Bots act as both Clients and servers. A new bot must know some addresses of the Botnet to connect there. Here if Bots are offline, the Botnet can still continue to operate under the control of Botmaster. Since P2P Botnets usually allow commands to be injected at any node in the network, the authentication of commands become essential to prevent other nodes from injecting incorrect commands [20] for eg: DNS, P2P protocol based botnet.

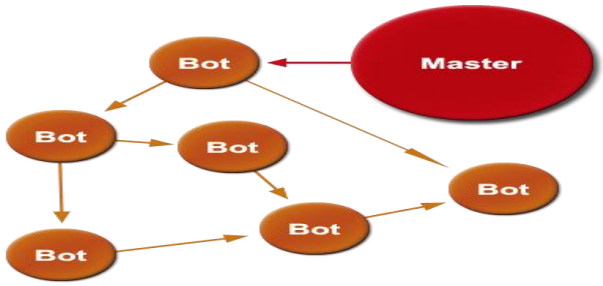

Figure 4. Example of Peer-to-peer Botnet Architecture

5.2.3. Unstructured C&C model: A bot will not actively contact other bots or the Botmaster, and would listen to incoming connections from its Botmaster. The Botmaster randomly scan the Internet and pass along the encrypt message when it detected another bot [5].

6 COMMUNICATION PROTOCOLS

A communications protocol is a system of digital message formats and rules for exchanging those messages in or between computing systems and in telecommunications [49]. Today Botnet usually use well defined communication

protocols to perform attack. So studying about communication protocols can help us determine the origins of a Botnet attack and decode conversations between the bots and the Botmasters [5].

Communication protocol can be classified in three different categories:

6.1 IRC protocol: A most common protocol used by Botmasters to communicate with their Bots. IRC protocol mainly designed for one to many conversations but can also handle one to one, which is very useful for Botmasters control their Botnet. However, security devices can be easily configured to block IRC traffic [5].

Weaknesses of IRC bots:

- Usually unencrypted
- Easy to get into, take over or shut down
- Due to the dependability more on C&C Server, Single point of failure is there [18].

6.2 HTTP protocol: Generally HTTP protocol is a popular Botnet due to its communication method by sending message as HTTP response and HTTP GET response to perform attack which is difficult to be detected. So Using the HTTP protocol, Botnet usually bypass security devices.

Weaknesses of HTTP based bots:

- Due to the dependability more on C&C Server, single point of failure is there [18].
- Bypass attack possible

6.3 P2P protocol: Recently, more advanced Botnet used decentralized model for their communications [5, 12]. For eg; Phatbot[18] , Storm, Nugache [18], Peacomm [18], Conficker and Slapper[33] used P2P communication protocols to perform malicious activity.

Note: Which protocols, used to communicate between two networks, through which a Botnet attack is possible, e.g. FTP (file transfer protocol), ICMP, IGMP, DHCP, TCP, UDP, DNS etc.

Weaknesses of P2P based bots:

- Strict Dependent ability on previous or others nodes
- These will not generate a sound Botnet
- Not mature
- If these have poor connectivity then easily traced
- Compared to HTTP Botnet, these have no hardly encryption /authentication code
- For large number of nodes, creates a complex structure and generates a large amount of traffic
- WASTE P2P protocol [12] is not scalable across a large network.

In last several years, Botnets such as Slapper [33], Sinit [18], Phatbot [18], and Nugache [18] have implemented different kinds of P2Pcontrol architectures. They have shown several advanced designs. For example, Sinit [18] uses public key cryptography for update authentication [33] but has poor connectivity for the constructed Botnet and easily detect due to extensive probing traffic. Nugache [18] attempts to thwart detection by implementing an encrypted/obfuscated control channel and so on [38]. But its weakness lies in reliance on a seed list of 22 IP addresses during its bootstrap process [18].

7 BOTNET ATTACKS

A Botnet is a tool for malicious users (attackers). There are as many different motives for using Botnets as there are people with malicious intent [23]. But generally most used of Botnet used for financial gain or for destructive purposes and to misuse or steals the valuable data of users. Hence here some of Botnets attacks are enumerated which is traced till now:

7.1 DDoS (Distributed denial of service) attacks
7.2 Spamming
7.3 Click fraud/Harvesting of information
7.4 Spreading new malware
7.5 Manipulating online polls
7.6 Google AdSense abuse
7.7 Attacking IRC networks
7.8 Fast Flux
7.9 Sniffing traffic
7.10 Key logging
7.11 Adware
7.12 Adware
7.13 Mass identity theft

Hence now description of each attack is explained as:

a. **Distributed denial of service (DDoS) attacks**: It is an attack on a network that causes a loss of service to users, typically the loss of network connectivity and services, by consuming the bandwidth of the victim's network or high bandwidth or overloading the computational resources of the victim's system(s).
b. **Spamming**: It is an attack which is performs by sending spamming mails (malicious links) to user through over internet.
c. **Click fraud**: It is a type of Internet crime that occurs in pay per click online advertising when a person, automated script or computer program imitates a legitimate user of a web browser clicking on an ad, for the purpose of generating a charge per click without having actual interest in the target of the ad's link (fraud link).
d. **Spreading new malware**: Botnets are used to spread new bots and malware. This is easy since all bots implement mechanisms to download and execute a file via HTTP or FTP.
e. **Manipulating online polls/games**: Online polls/games are getting more attention for online users. It is a technique, easy to manipulate with Botnets. Every bot with distinct IP address, every vote will have the same credibility as a vote cast by a real person. Online games can be manipulated in a similar way to perform malicious activity.
f. **Google AdSense abuse**: In this attack, attacker offers companies the possibility to display Google advertisements on their own website and earn money this way. The company earns money due to clicks on these ads. For example; per 10,000 clicks in one month, attacker can abuse online users through click on these advertisements in an automated incremented fashion.
g. **Attacking IRC Chat networks**: This attack similar to a DDoS attack. In this attack, victim network is flooded by service requests from thousands of bots or by thousands of channel-joins by bots. Through which the victim IRC network is brought down.

h. **Fast-flux network:** This is a service network in which networks of hijacked computer (that are part of a Botnet) systems with public DNS records that are constantly changing, with short time span [38] to perform illegal content from the Botnet end point to a central server . So the main aim of this technique is to provide high availability of the malicious contents by hiding location of the mothership.

i. **Sniffing Traffic:** Through using the sensitive information like usernames and passwords, this attack is done. In which Bots can also use a packet sniffer to watch for interesting clear-text data passing by a compromised machine.

j. **Keystroke logging:** Keylogging is the action of tracking (or logging) the keys struck on a keyboard, typically in a covert manner. The person using the keyboard is unaware that their actions are being monitored and traced by an attacker. Through this attack, very easy for an attacker to retrieve sensitive information. A key logger can run also on thousands of compromised machines in parallel.

k. **Adware:** It also called advertising-supported software. Any software package which automatically plays, displays, or downloads advertisements, to a computer. During the installation process, they may also be in the user interface of the software or on a screen presented to the user. Today Adware is so much harmless to generate revenue for its author or its master with integrated spyware such as key loggers and other privacy invasive software to perform malicious activity.

l. **SQL injection attack:** SQLi is a code injection technique that exploits security vulnerability in some computer software. An injection occurs at the database level of an application (like queries). The vulnerability is present when user input is either incorrectly filtered for string literal escape characters embedded in SQL statements or user input is not strongly typed and unexpectedly executed.

m. **Mass identity theft:** This is the one of the fastest growing crimes on the Internet to identity theft. Bogus emails ("phishing mails") that pretend to be legitimate (such as fake PayPal or banking emails) ask their intended victims to go online and submit their private information. These fake emails are generated and sent by bots via their spamming mechanism to perform an illegal activity [21]. Just as quickly as one of these fake sites is shut down, another one can pop up.

This section described the various about Botnet attacks. The following section will describe the methodologies to Botnet detection.

8 BOTNET DETECTION

Generally Botnet detection and tracking has been a major research topic in recent years due to increase in the malicious activity. Due to the presence of malicious activity (Botnet) from a long time, till now only few formal studies have examined the Botnet problem. So different techniques have been proposed to detect bot which are described as follows:

8.1. Honeynet-based methods:

Generally Honeynet based method consists of Honeypot and Honeywall [44]. Honeypot denotes an end host which is very vulnerable to malicious attacks and is often successfully compromised in a very short time span. And Honeywall denotes software which is used to monitor, collect, control, and modify the traffic through the Honeypot, eg. Snort [14]. Honeynet work used only unpatched versions of all versions of Windows as Honeypot, and Snort inline used as Honeywall device to track Botnets on a daily basis report (i.e., the Honeynet would have been rebuilt in every 24 hours). So based on using both results we detect the location and behavior of bots in a network.

Now beside of this functionality, this project has also listed a set of suggestions from which we can elaborate "how to write a useful Botnet tracking IRC clients" [44]. First, this client shall have SOCKSv4 and multi-server support to tracking bots. Second, some useful packages, such as lbadns, libcurl, and Perl Compatible Regular Expression (PCRE) shall be included in this client. And at last, the modularity and certain functionalities, such as no threading, shall be inconsideration throughout the design of this client.

A similar Honeynet has been constructed [17], in which Honeywall element shall be able to capture and inspect all the traffic payloads to retrieve Botnet information such as the DNS/IP address of the C&C server with the corresponding port number and the authentically data to join the C&C channel and capable of isolating the Honeypots from other machines in the local network by blocking outgoing connections containing suspicious keywords linked to possible malicious activities. These projects only offer a single vantage point of view on Botnet activities, thus missing a substantial portion of Botnet spreading behaviors.

So in order to capture the comprehensive actions of the Botnets, Rajab et al. [32] have constructed a multifaceted and distributed measurement infrastructure by combining a modified version of the nepenthes platform with the Honeynets i.e. we can say confidentially here, Honeynet is a powerful tool for understanding Botnet technology and characteristics, and tracking Botnet behaviors.

8.2. Passive traffic monitoring:

Beside of a successfully project Honeynet, to collect Botnet data or attacker location is a difficult task much more today. So another approach is setting up here "passively monitor the real Internet traffic" which is used to detect or extract the Botnet related packets [44]. Till now, presence of various types of different data such as Internet traffic data, DNS data, BGP route views, Netflow data, and proprietary enterprise data, and on the complexity and response time requirements, many Intrusion Detection System (IDS) designs have been already proposed to detect the Botnets and their location but no solution is perfect to give a better result in compare to this technique. This technique classified as behavior-based, DNS-based, and data-mining based respectively as described and summarized in the following sections:

8.2.1. Behavior-based detection: Based on the presence information and data, this method can be further categorized into two ways as signature based and anomaly based which can be explained as:

8.2.1.1 Signature-based detection: In this technique based on the available knowledge and signature of existing bot is sufficient to capture bots. In this to detect the botnets, collected a library of specific Botnet commands and function names which could be summarized and included in the proposed IDS. Once the IDS found matching keywords while inspecting the payload content, it can trigger the alert and take further actions against the Botnet. But this technique is limited to detect only the known Botnets. For example, Snort [30] is an open source IDS that monitors network traffic to find signs of intrusion by searching matches based on the predefined set of rules and signature.

8.2.1.2 Anomaly-based detection: This system detects threat, malicious threat by searching abnormal behavior of the network. Here"abnormal" behavior means detects the bots as deviation from "normal" behavior predefined by the appropriate templates. Binkley and Sigh [24] proposed an effective TCP based anomaly detection technique with IRC tokenization and IRC message statistics to detect Botnet clients and reveal Botnet servers. First, this anomaly based system implements an IRC parsing component to collect information on TCP packets and to determine an IRC channel. Next, these IRC channel traffics are correlated over a large set of sampled data in search of scanning activities [8]. And at last, the IRC channels with high scanning count would be stamped as the possible Botnet channels.

Akiyama et al. [31] proposed a three-metrics based measurement to detect abnormal Botnets behaviors under the assumptions that bots from the same Botnet will have regularities in relationship, response and synchronization.

Gu et al. [13] have proposed Botnet detection system (BotHunter) that recognizes the bot infection phase by running a correlation algorithm with the help of the user defined bot infection life cycle model. But weakness of BotHunter is, it cannot detect IRC bot based communication (detect the known signature Botnet only). From the same authors, BotSniffer [12] has been developed as an anomaly based algorithm designed to detect Botnet C&C channels in a local area network using the observation that bots within the same Botnet would demonstrate strong synchronization in their response and activities (e.g., sending spam, scanning and binary downloading). So advantage of this algorithm is, it does not require prior knowledge of Botnet and has low false positive and false negative rates. In future remember that Behavior based detection algorithms are not useful to identify C&C traffic.

8.2.2 DNS-based detection: This technique is a hybrid of behavior based and data-mining based techniques which is performed on DNS traffic. Generally for a Botmaster, to maintain and hide its bots is so much easy so consider these points DNS queries have been implemented in multiple Botnet stages, such as the rallying process after infection, malicious attack initiation, and C&C server update which is an advantage of this technique. And now these two factors distinguish Botnet DNS queries from legitimate DNS queries [4]. In which, first weakness is that queries to C&C servers, often in the form of DDNS, come only from Botnet members not from other areas or links. So for these weaknesses, in 2005, Dagon [9] has proposed a mechanism to identify the domain names of the C&C servers with abnormally high or temporally concentrated DDNS query rates. However, this technique could be easily evaded by using faked DNS queries. But this technique generates many false positives due to misclassification of legitimate and popular domain names that use DNS with short TTL. So another improved approach has been proposed in 2006 [3] with the additional utilization of NXDOMAIN reply rates.

So this technique than existing previous technique, provide so much more effective in revealing suspicious domain names and generates less false positives because, NXDOMAIN replies are more likely to refer to DDNS than to other names. And recently, in 2007, Choi et al. [16] proposed a Botnet detection mechanism that monitors group activities which are often consist of DNS queries simultaneously sent by a large number of distributed bots with more robustness and to detect Botnets with encrypted channels.

8.2.3 Data-mining based detection: This technique is mostly used to detect the bot in an abnormal traffic and in a high volume of traffic data. Abnormal DNS traffic has been successfully distinguished from the legitimate one but till now Botnet C&C communication pattern recognition or detection remains one of the most challenging tasks in IDS designs [44]. Because Botnets utilize some regular protocols for C&C communications and some modern technique to perform attack and show the traffic is similar to regular traffic. So along with the continuous evolution of various Botnets detection scheme data mining techniques also classified which including data classification, clustering, statistical data analysis, pattern recognition and artificial neural networks, support vector machines, etc.

Further Geobl and Holz [25] introduced Rishi, a mining based system in 2007 to distinguish Botnet C&C traffic. Rishi [44] constructs its data set by collecting IRC server nicknames, port numbers and implement an-gram analysis and a scoring system to detect bots that use uncommon communication channels which have evaded detections from other conventional IDS. But disadvantage of Rishi, it can easily be misguided by the disguised nicknames and cannot detect encrypted communication as well as non-IRC Botnets. Hence as conclusion here, our goal with the integration of data mining methods is to provide a mechanism for the detection of new classes and variants of bots.

9 BOT / BOTNET MITIGATION

Generally mitigation means reduce the effect of an activity which is harmful to others. So for identification of a bot/malicious activity on online system, the system will generate an alert signal to identifying malicious data. When once a Bot or Botnet or malicious activity has been detected, the classification information is sent to the user and to the

mitigation subsystem to perform action against that violation activity. Then an appropriate mitigation strategy will be recommended by the system listing all actions necessary to mitigate a bot attack, for which no automated mitigation approach will be implemented; only user intervention will be required [33].

- So there commended operations [32] include but are not limited to the following: Physically disconnecting infected computers from the network
- Considering an option of immediate blocking all outbound traffic to external networks
- Implementing filters on internal routers, firewalls and other networking equipment as appropriate to isolate infected segments and to monitor network traffic to ensure internal containment or identify how this infection is spreading and which hosts are infected
- Monitoring all network traffic in order to address possible multifaceted attacks
- Reviewing appropriate log files to attempt to identify the first system infected and what the attack vector was?
- Notification of users and external cyber support groups per policy
- Reinstall OS of infected systems (from Ghost image)
- Fully follow all bot packet streams, for analysis and additional detection
- Contact ISP or Network provider (company, organization, etc.) of Botnet offenders
- Perform additional forensics on affected systems (possible additional exploitation)

10 FURTHER BOTNET DEVELOPMENTS

In this section, we will discuss about possible future architectural Botnet threats that can be challenging to the Internet defense community [34]. In future further improvement with bot can be use of strong cryptography means asymmetric cryptography and Tor architecture features, onion routing for setting up the Botnets in order to be anonymous on the Internet, to harden the Botnet traffic detection process and to perform malicious activity.

In general today Botnet used so much advanced technologies for example; mariposa Botnet [39] is one of the dangerous Botnet which used its own protocol [Iredo communication protocol] to perform malicious activity. Mostly other protocols have been experimented for a remote control mechanism in various malicious activities to hidden its signature and its existence e.g, FTP has been designed as the C&C channel for Botnet such as Dumador and Haxdoor to perpetrate key logging to steal sensitive information [44]. In general, HTTP based Botnets use more encrypted C&C channels (which used Base64 obscured) to perform attacks. So use of encrypted HTTP has increased the difficulties in detection and de-obfuscation to itself. Moreover must be remembering one thing, IRC Botnet also can perform malicious activity using the same functionality as HTTP and P2P Botnet in future.

10.1. P2P bot

In P2P architecture, bots communicate client server rather centralized system. So P2P communication system is much harder to disrupt. In this whenever the Botmaster attempts to launch an attack, it publishes one of the predefined commands on the P2P system, and all the bots which subscribed to the set will be able to execute this command.

For eg. P2P Botnets such as Slapper [33], Sinit [18], Nugache [18], and Conficker [33]. However, P2P systems are more complicated and there are typically no guarantees on message delivery or latency [10, 38].

So in near future, the combination of HTTP and P2P protocols used Botnet means Hybrid P2P Botnet may be arise with a strong asymmetric cryptography, strong encryption and private key usage for communication between bots. This hybrid P2P architecture provide robust network connectivity, individualized encryption and control traffic dispersion, limited Botnet exposure by each captured Bot, and easy monitoring and recovery by its Botmaster [7] which is so much hardly traceable compared to other existing Botnet in current and in future.

10.2. Fast - flux service network

A new technology implementing the DNS protocol within C&C communications, referred as the FFSN, has emerged in recent years [34]. FFSN refers to rapidly changing the mapping between IP address and domain name. Fast-flux service networks are networks of hijacked computer (a part of Botnet) systems with public DNS records that are constantly changing, with short time span [38].

Now conceptually, fast-flux networks are two types: single-flux network and double-flux service networks [38]. Single-flux network puts the IP address of the domain name in flux and double –flux refers to dynamically and repeat changing the IP address of both and bots and their authoritative DNS. Today mostly Cyber-criminals or attackers engaged in illegal activities (e.g. Phishing, Spamming, etc) also use fast flux technique, for eg: first time in 2007, Storm Botnet[34] creators used such service networks. Later Asprox BotHerders[2009] also utilized the double-flux service networks to increase its strengthen and provide the best availability of the malicious content the Botnet architecture[34] and serve the content or commands to the bots globally. Double–flux refers has an additional layer of protection by changing the IP address for the authoritative NamerServers. Hence Botnets gradually utilize more protocols and FFSN network for specific malicious attacks and adapts more decentralized C&C structures.

Now at last, in future, BotHerders can configure the infected machine to allow IPv6 traffic and use this [46] novel approach to construct the covert channel that can be used for the malicious purpose. Though system administrators are aware of the IPv6 auto configuration feature, most firewall and IDS are not configured to filter the IPv6 traffic [34].

11 CONCLUSION AND FUTURE WORK

As well discussed above since 1988, Botnet have evolved from the beginning assistant tool to the predominant threat in modern internet and as discussed in this paper, in 1988 Botnet was not a malicious activity but later in 1998 , attacker use the bot to perform malicious activity via cyber crime. That is Botnets pose a significant and growing threat against cyber-security as they provide a key platform for many cybercrimes such as DDoS attacks against critical targets, malware dissemination, phishing, and click fraud etc. Although the number of bots to each Botnet seems to be decreasing, the monetary damaging power of the Botnets is continuously increasing given the development of internet bandwidth due to change in technology.

So now in current days, instead of using a centralized, IRC based C&C channel to perform multiple nefarious attacks, the Botnets have been gradually developed into more complicated, stealthy, and modular based package which perform particular malicious activity with diverse C&C protocols and structures [44] e.g. Hybrid P2P Botnet. Hence existing the long presence of malicious Botnets, only a small amount of studies have examined on Botnet problem and Botnet research is still in its infancy. This paper described about Botnet and various Botnet detection techniques (Honeynet based and Passive traffic monitoring) to detect these malicious activity and explained also some mitigation techniques to mitigate Botnet.

Finally it is necessary to discuss about further Botnet developments which may arise in future.

Hence the following points summarize the future trends to be carried out in Botnet research.

1. Build a list of all known Botnets and a data repository for associated traffic data samples that could be used to develop and test detection and mitigation algorithms in future to detect Botnet attack.
2. Develop an algorithm using the characteristics which can be identified as common among all Botnets as a first order detector.
3. Determine if it is practical for network providers to use network flow data to detect and mitigate Botnets [23].
4. Investigate various ideas developed and its possible extension be extension to build anti-bot applications that could be applied the way anti-virus or anti-spyware are used today [23].
5. Generation of a robust Botnet capable of maintaining control of its remaining bots even after a substantial portion of the Botnet population has been removed by defenders [38].
6. How to prevent significant exposure of the network topology when some bots are captured by defenders [38].
7. Monitoring and obtaining the complete information of Botnet by its Botmaster [38].

Apart from these points listed, an in-depth analysis of Botnets at various levels will really bring the Botnet to an entity, which will perform only for constructive work rather than destruction work.

12. REFERENCES

[1] A. Karasaridis, B. Rexroad, and D. Hoeflin, "Wide-scale Botnet detection and characterization," in Proc. 1st Workshop on Hot Topics in Understanding Botnets, 2007.

[2] A. Ramachandran and N. Feamster, "Understanding the network-level behavior of spammers," in Proc. ACM SIGCOMM, 2006.

[3] A. Schonewille and D. van Helmond, "The domain name service as an IDS", Research Project for the Master System-and Network Engineering at the University of Amsterdam, 2006.

[4] Brett Stone-Gross, Marco Cova, Lorenzo Cavallaro, Bob Gilbert, Martin Szydlowski, Richard Kemmerer, Christopher Kruegel, and Giovanni Vigna, "Your Botnet is My Botnet: Analysis of a Botnet Takeover", 2009.

[5] Chao Li, Wei Jiang, Xin Zou,"Botnet: Survey and Case Study", 4th International Conference on Innovative Computing, Information and Control, 2009.

[6] C. Kalt, "Internet Relay Chat: Client Protocol," Request for Comments (RFC) 2812 (Informational), April 2000.

[7] C. Mazzariello, "IRC traffic analysis for Botnet detection", In Information Assurance and Security, 2008. ISIAS'08. Fourth International Conference on, pages 318–323, 2008.

[8] D. Dagon, G. Gu , C.P. Lee, W. Lee, "A Taxonomy of Botnet Structures," in Proc. 23rd Annual Computer Security Applications Conference (ACSAC 2007), 2007, pp. 325-339.

[9] D. Dagon, "Botnet detection and response", In OARC Workshop, 2005.

[10] Evan Cooke, Farnam Jahanian, Danny Mcpherson, "The Zombie Roundup, Understanding, Detecting, and Disrupting Botnets", IEEE, 2005.

[11] E. Cooke, F. Jahanian, and D. McPherson, "The zombie roundup: Understanding, detecting, and disrupting botnets," in Proc. Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI'05), 2005, pp. 39-44.

[12] G. Gu, J. Zhang, and W. Lee, "Botsniffer: Detecting Botnet command and control channels in network traffic," in Proc. 15th Annual Network and distributed System Security Symposium (NDSS'08), 2008.

[13] G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee, "BotHunter: Detecting malware infection through ids-driven dialog correlation" In Proceedings of the 16th USENIX Security Symposium, pages 167–182, 2007.

[14] G. Schaffer, "Worms and Viruses and Botnets, Oh My: Rational Responses to Emerging Internet Threats", IEEE Security & Privacy, 2006.

[15] H. Binsalleeh , T. Ormerod , A. Boukhtouta , P. Sinha , A. Youssef , M. Debbabi , and L. Wang, "On the Analysis of the Zeus Botnet Crimeware Toolkit" Eighth Annual International Conference on Privacy, Security and Trust , 2010.

[16] H. Choi, H. Lee, H. Lee, and H. Kim, "Botnet Detection by Monitoring Group Activities in DNS Traffic," in Proc. 7th IEEE International Conference on Computer and Information Technology (CIT 2007), 2007, pp.715-720.

[17] Hossein Rouhani Zeidanloo, Azizah Bt Manaf, Payam Vahdani, Farzaneh Tabatabaei, Mazdak Zamani, "Botnet Detection Based on Traffic Monitoring" IEEE transaction,2010.

[18] Hossein Rouhani Zeidanloo, Azizah Abdul Manaf,"Botnet Command and Control Mechanisms", IEEE transactions, 2009.

[19] Hossein Rouhani Zeidanloo, Mohammad Jorjor Zadeh, M. Safari, Mazdak Zamani,"A Taxonomy of Botnet Detection Techniques", 3rd IEEE Conference paper , 2010.

[20] Hossein Rouhani Zeidanloo, Farhoud Hosseinpour , Farhood Farid Etemad, "New Approach for Detection of IRC and P2P Botnets" , International Journal of Computer and Electrical Engineering, Vol.2, No.6, December, 2010, 1793-8163.

[21] The Honeynet Project. (November 7, 2007), "Know your enemy: Behind the Scenes of Malicious Web Servers" Retrieved October 31, 2009, http://honeynet.org/papers/wek/

[22] J. B. Grizzard, V. Sharma, C. Nunnery, B. B. Kang, and D. Dagon, "Peer-to-peer botnets: Overview and case study," in Proc. 1st Workshop on Hot Topics in understanding Botnets, 2007.

[23] Joseph Massi, Sudhir Panda, Girisha Rajappa, Senthil Selvaraj, and Swapana Revankar, "Botnet Detection and Mitigation", presented on Proceedings of Student-Faculty Research Day, CSIS, Pace University , May 7th, 2010.

[24] J. Binkley and S. Singh, "An algorithm for anomaly-based botnet detection" In Proceedings of USENIX Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI), pages 43–48, 2006.

[25] J. Goebel and T. Holz, "Rishi: Identify bot contaminated hosts by irc nickname evaluation", In USENIX Workshop on Hot Topics in Understanding Botnetsc(HotBots 07), 2007.

[26] K. K. R. Choo, "Zombies and Botnets," Trends and issues in crime and criminal justice, no. 333, Australian Institute of Criminology, Canberra, March 2007.

[27] Kapil Singh, Abhinav Srivastava, Jonathon Giffin , Wenke Lee, "Evaluating Email's Feasibility for Botnet Command and Control" International Conference on Dependable Systems &Networks: Anchorage, Alaska, June 24-27, 2008.

[28] Maryam Feily, Alireza Shahrestani, "A Survey of Botnet and Botnet Detection", 3rd International Conference on Emerging Security Information, Systems and Technologies, 2009.

[29] M. Kola, "Botnets: Overview and Case Study", PhD thesis, IBM Research, 2008.

[30] M. Roesch, "Snort-lightweight intrusion detection for networks", In Proceedings of the 13th USENIX conference on System administration, pages 229–238. Seattle, Washington, 1999.

[31] M. Akiyama, T. Kawamoto, M. Shimamura, T. Yokoyama, Y. Kadobayashi, and S. Yamaguchi, "A proposal of metrics for botnet detection based on its cooperative behavior", In Applications and the Internet Workshops, 2007. SAINT Workshops 2007. International Symposium on, pages 82–82, 2007.

[32] Rajab, M., Zarfoss, J., Monrose, F.,&Terzis, A. (October 2006). "A multifaceted approach to understanding the botnet phenomenon" Retrieved October 31, 2009 from http://www.imconf.net/imc- 2006/papers/p4-rajab.pdf

[33] Robert F. Erbacher, Adele Cutler, Pranab Banerjee, Jim Marshall,"A Multi-Layered Approach to Botnet Detection", IEEE conference, 2010.

[34] Ravishankar Borgaonkar,"An Analysis of the Asprox Botnet", 4th International Conference on Emerging Security Information, Systems and Technologies, 2010.

[35] R.Villamarin-Salomon and J.C. Brustoloni, "Identifying Botnets Using Anomaly Detection Techniques Applied to DNS Traffic," in Proc. 5thIEEE Consumer Communications and Networking Conference (CCNC 2008), 2008, pp. 476-481.

[36] Ping Wang, Lei Wu, Ryan Cunningham, Cliff C. Zou," Honeypot Detection in Advanced Botnet Attacks" Int. J. Information and Computer Security, Vol. x, No. x, xxxx.

[37] P. Vixie, S. Thomson, Y. Rekhter, and J. Bound, "Dynamic updates in the domain name system (DNS Update)," 1997, http://www.faqs.org/rfcs/rfc2136.html/.

[38] P. Wang, S. Sparks, and C. C. Zou, "An advanced hybrid peer-to-peer botnet," in Proc. In Workshop on Hot Topics in understanding Botnets, 2010

[39] Prosenjit Sinha, Amine Boukhtouta, Victor Heber Belarde, Mourad Debbabi, "Insights from the Analysis of the Mariposa Botnet"

[40] SANS Institute InfoSec Reading Room provided a description on "Bot & Botnet: An overview" research on topics in information security, 2003.

[41] Shun-Zheng Yu, Wei-Zhou Lu, "An HTTP Flooding Detection Method Based on Browser Behavior" IEEE transaction, 2006.

[42] Taxonomy of Botnet Threats. Trend Micro Inc. White Paper, November, 2006.

[43] Xuhua Ding, Wei Yu, Ying Pan,"A Dynamic Trust Management Scheme to Mitigate Malware Proliferation in P2P Networks" IEEE conference on communications, 2008.

[44] Xiaonan Zang, Athichart Tangpong, George Kesidis and David J. Miller, CSE Dept Technical Report on "Botnet Detection through Fine Flow Classification" Report No. CSE11-001, Jan. 31, 2011.

[45] Yang-Seo Choi, Jin-Tae Oh, Jong-Soo Jang, Jae-Cheol Ryou, "Integrated DDoS Attack Defense Infrastructure for Effective Attack Prevention", and 2nd International Conference on Issue Date: 11-13 Aug. *2010, 2010.*

[46] Z. Zhu, G. Lu, Y. Chen, Z. J. Fu, P. Roberts, K. Han, "Botnet Research Survey," in Proc. 32nd Annual IEEE

International Conference on Computer Software and Applications (COMPSAC '08), 2008, pp.967- 972.

[47] The botnet business, available at http://www.viruslist.com/en/analysis?pubid=204792003

[48] The botnet business, available at http://www.securelist.com/en/analysis/204792003/The_botnet_business?print_mode=1

[49] Communications protocol, Available at http://en.wikipedia.org/wiki/Communications_protocol

13. AUTHORS PROFILE

Amit Kumar Tyagi received his Bachelor of Technology degree in Computer Science and Engineering from Uttar Pradesh Technical University, Lucknow, India, in 2009. He is currently pursuing his Master's degree in Computer Science and Engineering in Department of Computer Science from the School of Engineering and Technology, Pondicherry University, India. His research interests include Network Security, Denial-of Service resilient protocol design, VoIP, Mobile Computing, Cloud Computing, Smart and Secure Computing.

Prof. Dr. G. Aghila is currently working as Professor in Pondicherry University. She completed her PhD in Computer Science and Engineering from Anna University, Chennai in 2004. She completed her Master of Technology in Computer Science and Engineering from Anna University, Chennai in 1991 and completed her Bachelor of Technology in Computer Science and Engineering from Thiagarajar College of Engineering, Coimbatore. Her research interest includes Knowledge Representation and Reasoning, Semantic Web Engineering, Ontology Engineering, Cheminformatics, Context aware Computing, Smart and Secure Computing.

Appendix-A: Table 1- Botnet History

S.No	Bots	Invented by (Author)	Evolution Year	Month	Architectural Features	Infected Hosts	Kind of Threat	Detection Method or Tool	How it Works	OS
1	Eggdrop	Robey pointer	1993	December	IRC	-	-	-	-	Unix /Linux
2	GTBot	-	1998	April	mIRC	-	DDoS Files installation and deletion	Bot signature analysis (looking for mIRC scripts), Data Mining methods: Neural Networks, SVM, Expert Systems, etc.	Uses mIRC as a core. UsesIRC-channel. Spreads using email attachment or downloads via the hacker's site.	Windows
3	Netbus	Carl-Fredrik Neikter	1998	-	HTTP	-	-	-	-	Windows
4	!A	-	1999	-	-	1 billion	-	-	-	Windows
5	Backorifice2k	-	1999	-	IRC	-	-	-	-	Windows
6	Sdbot/Rbbot/ Urbot/Urxbot	-	2002	October	IRC	-	Unauthorized remote access to computer (Executing programs, Opening files Downloading files, Redirecting information sent to a local port to a remote port,	Data Mining methods: Neural Networks, SVM, Expert Systems, etc.	Uses IRC-port to receive commands Spreads exploiting vulnerabilities in Windows operating systems	Windows

							Sending system information from the local host, such as operating system, processor speed, free ram, etc.) File deletion		and via network shared drives.	
7	GaoBot	-	2002	March	HTTP	-	-	-	-	Windows
8	Slapper	-	2002	April	P2P	-	-	-	-	Windows
9	Agobot	Ago alias wonk	2002	April	IRC	-	Releases confidential info (steals the CD keys of several popular computer games, steals Windows product ID) Unauthorized remote access to computer Kills processes, belonging to antivirus and firewall software	Data Mining methods: Neural Networks, SVM, Expert Systems, etc	Uses IRC-port For messaging. Spreads using numerous vulnerabilities in OS, applications, via P2P applications such as Kazaa, Grokster, and Bear Share, and via network shared drives.	Windows
10	Spybot	-	2003	April	IRC	-	-	-	-	Windows
11	Sinit	-	2003	September	P2P	-	-	-	-	Windows
12	Rbot	-	2003	-	-	-	-	-	-	Windows
13	Bagle	-	2004	January	HTTP	2,30,000	-	-	-	Windows
14	Phatbot	-	2004	March	P2P	-	-	-	-	Windows
15	Polybot	-	2004	-	IRC	-	-	-	-	Windows
16	Mytob	-	2005	-	IRC	-	-	-	-	Windows
17	Rustok	-	2006	April	HTTP	1,50,000	-	-	-	Windows
18	Nugache	-	2006	April	P2P	-	-	-	-	Windows
19	Trojan.peacomm	-	2007	January	P2P	-	-	-	-	Windows
20	Srizbi	-	2007	March	-	4,50,000	-	-	-	Windows
21	Storm	No identification till now	2007	September	P2P, Fast Flux Network	1.7 billion	-	-	-	Windows
22	Kraken	-	2008		HTTP	4,95,00	-	-	-	Windows
23	Asprox	-	2008	January	HTTP, Advanced Fast Flux Network	50,000	-	-	-	Windows
24	Waledec	-	2008	November	HTTP	80,000	-	-	-	Windows
25	Zeus bot(in US only)	-	2009	January	HTTP	3.6 million	Mobile Banking Threat	Zeus tool kit	-	Windows
26	Conficker	-	2009	-	P2P, Fast Flux	27,08,259	-	-	-	Windows

					Network					
27	Mariposa	-	2009	May	HTTP	13.5 million	-	-	-	Windows
28	Festi	-	2009	August	-	2.25 billion	-	-	-	Windows
29	Bredolab	-	2009	May	-	3.6 billion	-	-	-	Windows
30	Blackenergy	-	2010	April	-	-	DDos attack, phishing attack	-	-	Windows
31	Spyeye	-	2010	-	HTTP	-	Mobile Banking Threat, DDos attack, phishing attack	-	-	Windows
32	Tdl 4	-	2010	-	HTTP	4,50,0000	-	-	-	Windows
33	Social bot	-	2011	August	-	-	-	-	-	Windows
34	Anti spyware	-	2011	June	-	-	-	-	-	Windows
35	Q8 bot	-	-	-	IRC	-	DDoS (SYN-flood and UDP-flood). Execution of arbitrary commands	Bot signature analysis, Data Mining methods: Neural Networks, SVM, Expert Systems, etc.	Uses IRC-Channel	Linux/ Unix
36	Clickbot(hitbot)	-	-	-	IRC	-	Click Frauds , DDoS attacks	User Intension Analysis	Uses IRC-port to communicate with hacker Spreads using e -mail Attachment.	Windows
37	Push do	-	-	-	HTTP	-	-	-	-	Windows
38	Web based C&C	-	-	-	HTTP	-	-	-	-	Windows
39	Cyber bot	-	-	-	HTTP	-	-	-	-	Windows
40	Cutwail	-	-	-	HTTP	1,50,0000	-	-	-	Windows
41	Netsky.Q	-	-	-	HTTP	1,50,0000	-	-	-	Windows
42	Bluecode.worm	-	-	-	HTTP	-	-	-	-	Windows
43	Kaiten	-	-	-	IRC	-	DDoS attacks Download files from a Web site of the hacker's choice Run commands or files of the hacker's choice	Bot signature analysis, Data Minig Methods: Neural Networks, SVM	Uses IRC-Channel	Windows /Linux/ Unix
44	Lisp IRC	-	-	-	IRC	-	DDoS attacks	Signature analysis, Data Mining	Lisp commands to process operations	Windows (Unix /Linux Rarely

								methods: SVM, Neural Network	Uses IRC-port for C&C communications	Only)
45	Perl based	-	-	-	IRC	-	DDoS attacks	Data mining methods: SVM, Neural Network	uses IRC-Channel for C&C communications Has limited basic set of commands	Unix /Linux