LAYERED BOTNET DETECTION FRAMEWORK BASED ON SIGNAL PROCESSING AND DISCRETE TIME ANALYSIS

By

LOAI KAYED HASSAN BANI MELHIM

UNIVERSITI SINSE MALAYSIA

September 2012

Layered Botnet Detection Framework Based on Signal Processing and Discrete Time Analysis

By

LOAI KAYED HASSAN BANI MELHIM

Thesis submitted in fulfillment of the Requirements for the degree of Doctor of philosophy

September 2012

ACKNOWLEDGMENTS

بسم الله الرحمن الرحيم

﴿وَعَدَ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَعَمِلُوا الصَّالِحَاتِ لَيَسْتَخْلِفَنَّهُمْ فِي الأَرْضِ كَمَا اسْتَخْلَفَ الَّذِينَ مِنْ قَبْلِهِمْ وَلَيُمَكِّنَنَّ لَهُمْ دِينَهُمُ الَّذِي ارْتَضَى لَهُمْ وَلَيُبَدِّلَنَّهُمْ مِنْ بَعْدِ خَوْفِهِمْ أَمْنًا يَعْبُدُونَنِي لا يُشْرِكُونَ بِي شَيْئًا وَمَنْ كَفَرَ بَعْدَ ذَلِكَ فَأُولَئِكَ هُمُ الْفَاسِقُونَ﴾ (55)النور

All Praise go to almighty Allah, we thank him, seek his guidance and forgiveness for giving me the patience and health to finish this research.

There have been many hands helping me in my work, and some people stand out in my mind. The first idea related to this thesis came from my friend and my previous supervisor, Dr Ahmad Manasreh. Who helped, encouraged and support me. If the meaning of pursuing a Ph.D. is more than a degree, all of the lessons I learned from this journey has permeated my whole life. I feel profoundly enriched and fortunate to have met all of those whom I interacted with.

I would like to express my deepest gratitude to my main supervisor, Assoc. Professor Wan TatChee for his valuable insights, comments, patient guidance and continual encouragement and to my co-supervisor Professor Sureswaran Ramadass Director of NAV6 Center of Excellence for his invaluable guidance through the course of my research. Without their knowledge and assistance, this thesis would not have been successful.

Special thanks to National Advanced IPv6 Center of Excellence (NAv6), and to all NAv6 staff and to the whole NSST team especially to Hairul Nizar, Othman, Kunalan dava Rajoo. Without your time, patience and resources I would never have been able to accomplish this research.

Thanks to my friends Ammar al-momani, Khaleel Shaqfa, Eisa, Sami, mohammad, and all the staff in Al-Zelfi colledge and to all my colleagues for making my research days happy and memorable. I also would like to express my gratitude to Eng.Muyad bani melhim for his constant support in Matlab programming as I have pursued my research.

I would like to express my deep thanks to those who are close to my heart; my brothers, my sisters, and all of my family members, thank you so much for your love and support.

Above all, I would like to express my love and gratitude to my wife, Rasha, and my dearest son Ahmad, for their understanding and endless love through the duration of my studies.

I dedicate this thesis sincerely to my father, and my mother who guide me and helped me to be what I am. Without them, I would not have reached what I have reached today. For their continuous support and guidance I owe much to their patience, and their honest advices. It was their many encouragements that kept me going when I had doubts about myself.

تمت بحمد الله

Loai Kayed Bani Melhim

Penang, Malaysia. March 2013.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACF	Auto Correlation Function
ACK	Acknowledgment
ASCII	American Standard Code for Information Interchange
bpp	Byte Per Packet
bps	Byte Per Second
C&C	Command and Control
DNS	Domain Name Service
DDNS	Dynamic Domain Name System
f	Frequency
FFT	Fast Fourier Transform
FIN	Finish
FN	False Negative
FP	False Positive
fph	Flow Per Hour
FTP	File Transfer Protocol
HTTP	Hyper Text Transfer Protocol
ICMP	Internet Control Message Protocol
ICQ	Instant Messaging Computer
IDS	Intrusion Detection System
IM	Instant Messaging
Ipsrc	The IP Source
IPv4	IP Address Version 4
IRC	Internet Relay Chat

ISP	Internet Service Provider
LBDS	Layered Approach Detection System
LAN	Local Area Network
MAN	Metropolitan Area Networks
MSN	Microsoft Networks
NXDOMAIN	Non Existent Domain
P2P	Peer to Peer
PCF	Partial Creation Filter
PSD	Power Spectral Density
RST	Reset
SMS	Short Message Service
SSL	Secure Socket Layer
SYN	Synchronize
TCP/IP	Transmission Control Protocol\Internet Protocol Stack
TN	True Negative
TP	True Positive
VoIP	Voice over Internet Protocol
WAN	Wide Area Network
Whs	Weight of the High Severity Ports
Wls	Weight of the Low Severity Ports
WWW	World Wide Web

LIST OF APPENDICES

RANGKA KERJA PENGESANAN BERLAPIS BOTNET BERDASARKAN PEMPROSESAN ISYARAT DAN ANALISIS MASA DISKRET

ABSTRAK

Transaksi kewangan dalam talian dan maklumat sensitif yang banyak saling bertukar di Internet. Ini mengalih tumpuan penyerang siber daripada perasaan ingin tahu kepada mendapatkan keuntungan kewangan. Penyerang menggunakan perisian hasad yang berbeza untuk mencapai matlamat meraka. Botnet dianggap antara perisian hasad yang berbahaya kerana kuasanya yang mampu mengawal pelbagai mesin dan memberi ancaman kepada pengguna Internet.

Tesis ini membentangkan suatu pendekatan baru dalam bidang pengesanan botnet. Ia memperkenalkan rangka kerja baru yang dinamakan Rangka Kerja Pengesanan Botnet Berlapis (LDBF) yang dapat mengesan rakan botnet dengan berkesan. Rangka kerja ini berfungsi dalam domain kekerapan dan bukannya dalam masa domain. LBDF dilengkapi dengan alogritma pengesan 'pengimbasan-hasad'. Alogritma LBDF menggunakan peraturan SYN, ACK (SNAK) untuk mengurangkan jumlah kesesakan rangkaian dan menukarkan trafik yang telah dikurangkan menjadi sampel data diskret. Kemudian LBDF mengaplikasikan kedua-dua periodogram dan fungsi autokorelasi membulat bagi mengesan sebarang keberkalaan tersembunyi di dalam jujukan sampel. Jika pelakuan berkala dikesan, kekerapan jujukan dan alamat IP komputer akan direkodkan. Oleh itu, alamat komputer peribadi dengan pelakuan berkala akan disimpan ke dalam pangkalan data dan dilabelkan sebagai mencurigakan. Jika mana-mana mesin yang mencurigakan menunjukkan pelakuan

pengimbasan-perosak, ia akan diisytiharkan sebagai bot. Bot yang mempunyai ciri yang sama dikumpulkan sebagai jenis botnet yang sama. Walaupun LBDF tidak mengesan bot yang tidak berkala atau tidak aktif contohnya, tidak berkomunikasi dengan master bot atau melakukan sebarang tindakan, ia akan mengesan mereka apabila mereka menunjukkan pelakuan yang mencurigakan. Pendekatan ini adalah berbeza dengan pendekatan lain kerana ia tidak terhad kepada protokol yang spesifik kepada protocol C&C (contohnya; HTTP, IRC) atau struktur botnet yang spesifik (contohnya; P2P, Berpusat) atau pelakuan serangan (iaitu; SPAM, DDOS) yang tidak memerlukan sebarang pengetahuan terdahulu bot yang dikesan.

Penilaian LBDF menunjukkan bahawa algoritma pengesanan adalah tepat, pantas dan berskala jika dibandingkan dengan rangka kerja pengesanan yang ada. LBDF mampu mengesan P2P, HTTP, IRC, bot berpusat atau yang tidak berstruktur. Justeru, prestasi LBDF F-measure adalah 26% lebih baik berbanding rangka kerja pengesanan botnet yang lain. Hasil daripada pengaplikasian algoritma pengurangan trafik rangkaian yang diadaptasikan oleh LBDF, kadar pengurangan dalam trafik yang dikaji adalah dalam julat 20%-90%, pengurangan ini meningkatkan prestasi LBDF dan meningkatkan daya pemprosesan tanpa menjejaskan matlamat utama LBDF.

LAYERED BOTNET DETECTION FRAMEWORK BASED ON SIGNAL PROCESSING AND DISCRETE TIME ANALYSIS

ABSTRACT

A massive volume of online financial transactions and sensitive information is exchanged over the Internet. This has shifted the focus of cyber attackers from curiosity to financial gain. Attackers use different malware to achieve their goals. Among the various forms of malware; the botnet is considered as the worst, because of its vast computing power, ability to control many machines and its significant threat to the Internet users.

This thesis presents a new approach in the area of botnet detection. It introduces a framework called Layered Botnet Detection Framework (LBDF) that can detect botnet members efficiently. This framework works in the frequency domain rather than in the time domain. LBDF is equipped with a 'malicious-scanning' detection algorithm. The LBDF algorithm uses SYN, ACK (SNAK) rules to reduce the volume of network captured traffic and to convert the reduced traffic into discrete time sequences. Then LBDF applies both a periodogram and circular autocorrelation function to these sequences to detect any hidden periodicities. If periodic behavior were detected, the frequency of the sequence and the IP address of the monitored computer will be recorded. Thus the IP address of PCs with periodic behavior will be saved in a database and labeled as suspicious. If any of the suspicious machines performs a malicious-scanning action, it will be declared as a bot. Bots that have similar features are grouped together as members of the same botnet.

Although LBDF does not detect bots that are non-periodic or inactive i.e. not communicating with their bot master or performing any action, it will detect them as soon as they exhibit suspected bot behavior. The proposed approach is different than other approaches, since it is not limited to specific C&C protocols (e.g., HTTP, IRC) or to specific botnet structures (e.g., P2P, Centralized) or attack behaviors (i.e. SPAM, DDOS), neither does it require any prior knowledge of the detected bots.

The evaluation of LBDF shows that the detection algorithm is accurate, fast and scalable compared to existing bot detection frameworks. LBDF is capable of detecting P2P, HTTP, IRC, centralized and even unstructured bots. In this respect, the LBDF F-measure is better by 26% compared with other botnet detection frameworks. As a result of applying the network traffic reduction algorithm adopted by LBDF, the reduction rate in the analyzed traffic was in a range of 20% - 90%, this reduction improves the performance of LBDF and increases its throughput without affecting the main goal of LBDF.

CHAPTER ONE

INTRODUCTION

A few years ago, protecting a computer system or networks was mainly required to prevent the threat from viruses and worms. Nowadays, the situation has changed drastically; the biggest threat faced by network hosts is malware, which is written by cyber programmers with the intent of malicious activities. Malware may take the form of viruses, worms, Trojans, botnet or other malicious programs.

Among the various forms of malware, botnet is considered as the most serious means for conducting online crimes (FBI, 2011b). This threat is triggered due to its large scale and geographical diversity of the network hosts enlisted in a Botnet. The large number of enlisted bots; gave the Botnet its vast computing power (Guofei Gu, Roberto Perdisci, Junjie Zhang, & Wenke Lee, 2008). This vast computing power coupled with the easy controlling of botnet from anywhere in the world; makes the botnet a powerful cyber weapon and an effective tool for performing malicious activities. Botnets become sophisticated more and more every day by employing variety of techniques (e.g., sophisticated executable packers, rootkits, protocol evasion techniques, such as moving away from IRC and taking control of, HTTP, VoIP, IPV6, ICMP, Skype protocols, etc). Bots are more evasive to signature based detection systems, anomaly-based detection systems as well as DNS and data mining based intrusion detection systems. These evasion techniques; improve the survivability of botnets and the success rate of compromising new hosts. Additionally, botnets have also added (and continue to add) new mechanisms to hide traces of their communications.

1.1 The scale of Botnet problem

A bot is defined as a computer that is compromised by malicious software that enables a remote computer to control it. Bots are part of a network of infected machines, known as a "botnet" that spread globally as shown in Figure 1.1. The process of estimating botnet size and calculating the botnets population is a tedious task; the size and growth of botnets differ widely. For example, the Mariposa botnet (FBI, 2011a), it contains 12.7 million infected computers, while Zeus (Binsalleeh, et al., 2010) has more than 1,400 command and control servers with undetermined numbers of infected hosts. The figures are scary. The total number of zombies is near to 60 million. Table 1.1 shows some evaluations of the number of active bots at the end of 2010 according to Message Labs Intelligence 2010 Annual Security Report (Symantec, 2011).

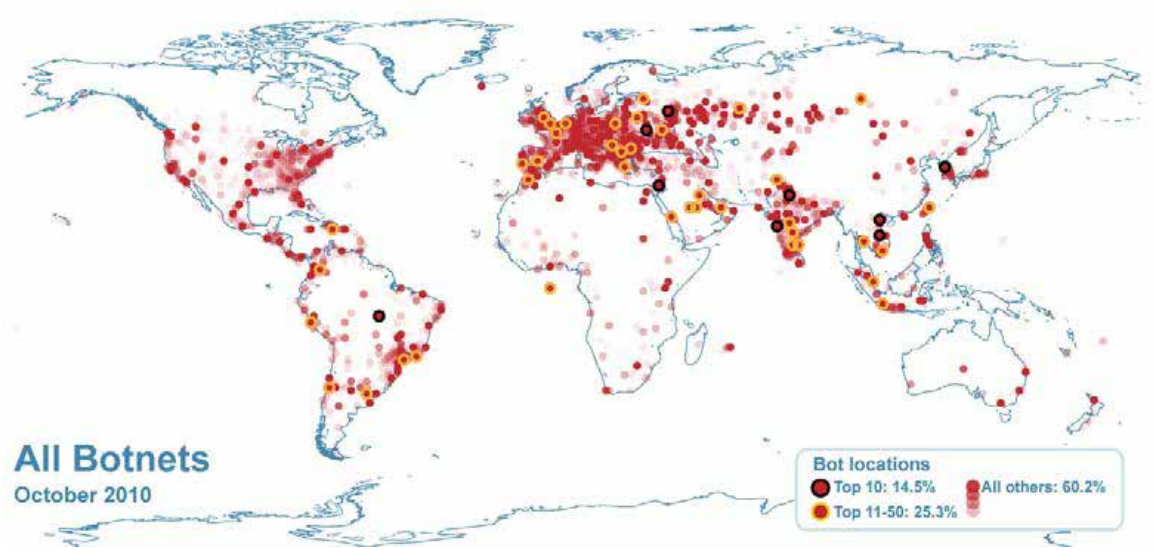

Figure 1.1 Global Bot Infection, Bots are a global problem. The map shows the geographic locations of active bots at October 2010.(Symantec, 2011)

Table 1.1 State of Botnet at the end Of 2010 (CISCO, 2011)

Botnet	Est.Botnet size	Country of infection
Rustock	1100k to 1700k	USA (17%), Brazil (7%), India (7%)
Grum	310k to 470 k	Russia (12%), India (8%), Vietnam (8%)
Cutwail	560k to 840k	India (17%), Russia (16%), Ukraine (8%)
Maazben	510k to 770k	Russia (11%), India (10%), Brazil (7%)
Mega-D	8ok to 120k	Russia (15%), Ukraine (14%), Brazil (7%)
Cimbot	32k to 48k	Italy (27%), Spain (25%), France (14%)
Bobax	250k to 370k	India (32%), Russia (25%), Ukraine (9%)
Xarvester	17k to 25k	Italy (15%), UK (10%), Poland (8%)
Festi	8k to 12 k	Vietnam(24%), Indonesia(21%), India
Gheg	8k to 12 k	Spain (12%), Indonesia (21%), India (10%)
Unnamed	490k to 740k	
other	220k to 340k	
Total	3500k to 5400k	India (9%), Russia (9%), USA (7%)

1.2 Research Motivation

Regardless of how malware reaches a computer, the challenge is to identify the infected machine and heals it as soon as possible before any harm is caused. The past recent years are witnessed of different approaches that have been proposed to detect botnets and to combat their threat against cyber-security, but these approaches were based on a specific part of botnet lifecycle like, scan, spam, etc. or a specific abnormal behavior of a network traffic or, a specific communication protocols like, IRC, P2P and HTTP that are used by botnet Command and Control servers (C&C) or a certain topology e.g, centralized. All of these properties are specific properties and it is not necessary that all types of botnets contain it. Therefore, previous methods are suitable only for specific botnet type or structure. Diversity of botnet protocols and different structures; make botnet detection a very challenging task.

1.3 Problem Statement

One of the most critical issues in Cyber Security is the botnet detection problem. Bots are stealthy in nature and usually do not aggressively consume CPU/memory/bandwidth resources, or perform noticeable damage to computers, such as disabling existing antivirus. Thus, a host-based solution method that is very specific to a certain botnet's structure or a certain communication protocol is not desirable because:

- Bots are flexible in their nature.
- Continuously evolving with flexible design.
- Different protocols and structures are used to organize and control the botnet.
- Bot life cycle consists of several different stages and aspects that developed and changed continuously.

That is why many existing Bot detection techniques become ineffective, as bots change their structure or C&C techniques. Despite the concerted efforts given in the literature, diversity of botnets protocols and structures makes botnet detection a challenging task and unsolved problem for the online community (ENISA, 2011; FBI, 2011a; IBM, 2010). Botnet detection problem can be solved through the detection of the command and C&C communication channels and the host's malicious-activities that is proposed in this thesis.

1.4 Research Objective

The objective of this thesis is to construct a new approach to detect botnet members in the monitored network. This approach is independent of botnet C&C protocols and structures. In addition, it does not require any priori knowledge (signature) of bots. It is assumed that the detection of the periodic C&C communication channels traffic together with the detection of the malicious-scan activities makes it possible to detect botnet members in the monitored network.

Therefore, the main objectives of this thesis are:

- To create a traffic representative that functions in frequency domain.
- To detect bots, independent of the bots structure and the communication protocols used.
- To evaluate the performance and accuracy of the proposed framework compared to other existing frameworks.

1.5 Thesis Contribution

Computer users, applications and bots utilize network in the same manner, but with different intentions. The naïve nature of users and applications activities differs than the malicious activities performed by bots. The proposed framework should be able to distinguish the normal traffic caused by a legitimate user or applications from the malicious traffic caused by bot activity.

The main expected contribution of this thesis is to propose and to design a new bot detection model. However, this thesis contribution summarizes as follow:

- **Traffic reduction**, to introduce a new technique that will be able to create a true representative of the monitored network traffic with a discrete time sequences.
- **An algorithm that computes the PSD of the resultant Discrete Time Sequences**, an algorithm that can be used to understand, and to model the normality of the network traffic in the frequency domain rather than the time domain.
- **A bot detection model with low false positives**, a new detection model that is capable of identifying the existence of all known and unknown types of bots, independent of both, the botnet structure and the used communication protocols.
- Enhanced model in terms of **performance and accuracy**, compared to the existing models.

1.6 Thesis Scope

The scope of this work as shown in Figure 1.2 is limited to Inbound/Outbound, IPv4 and TCP traffic, captured from the observed hosts in the monitored local area network (LAN). In the captured traffic, only details within the packet headers are of interest.

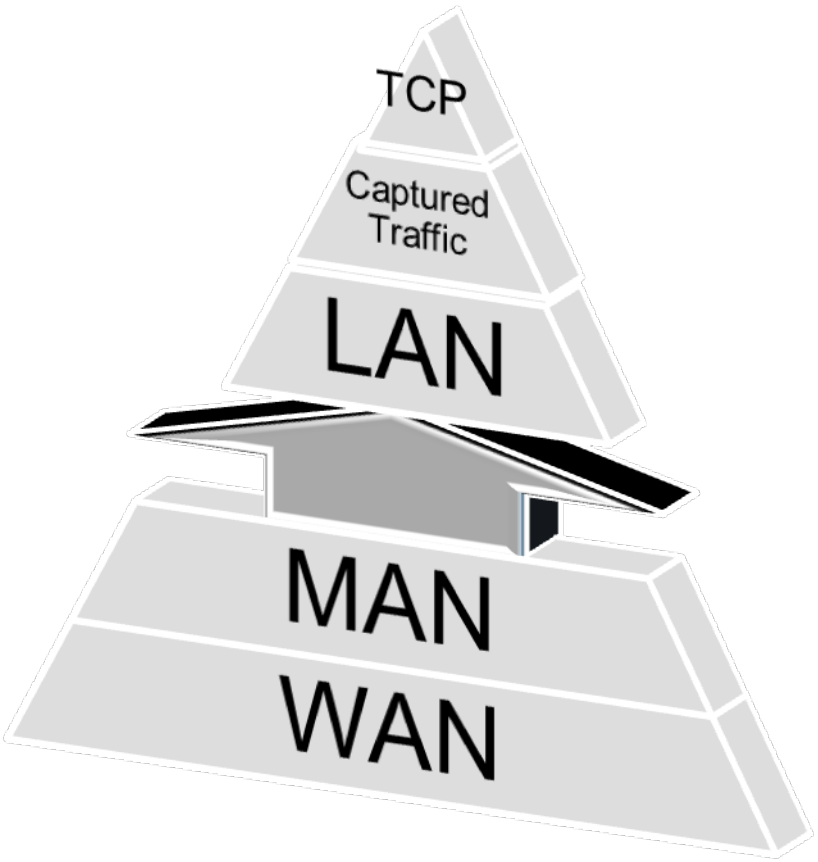

Figure 1.2 Thesis Scope

1.7 Research Framework

Figure 1.3 describes the complete research framework of this thesis.

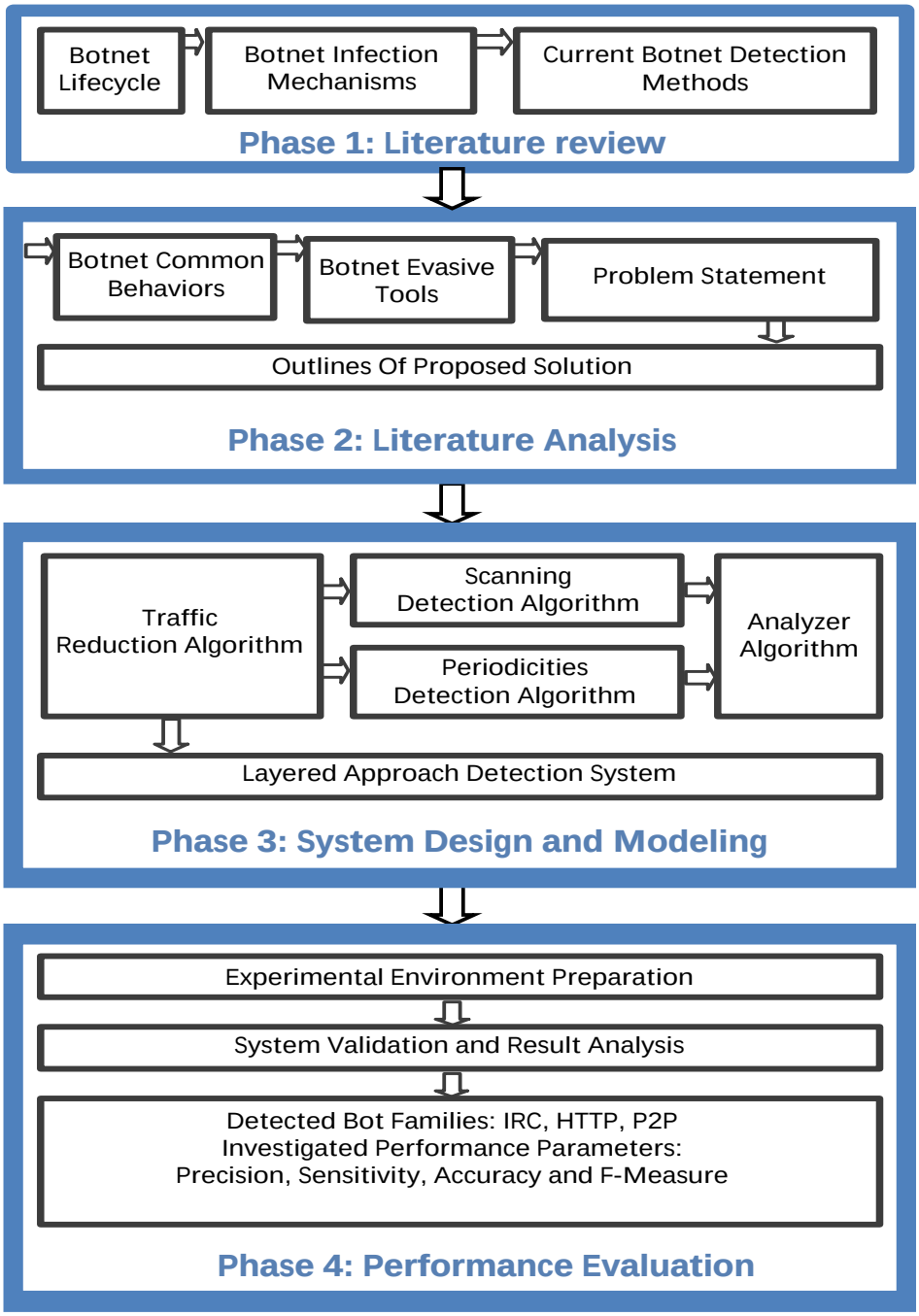

Figure 1.3 Research Framework

1.8 Thesis Outline

This thesis is organized into seven chapters. This chapter **(Chapter 1)** presents the objectives of this thesis. It starts by presenting a background discussion for the Bot problem along with our research objectives and contributions.

In **Chapter 2**, literature review will be presented along with some fundamental concepts related to this work and issues surrounding it. Other botnet detection models will be discussed, as well as the most current and related works related to botnet detection. This chapter also provided motivation for our work by describing some candidate architectures and the limitations of those proposed solutions.

Chapter 3 covers the methodology discussion on how the proposed solution was designed. The algorithm to shift captured traffic from time domain to frequency domain, and to accurately -detect and specify- the frequency (time) of the monitored signal (traffic) will be introduced in this chapter.

The implementation details and issues regarding the illustration of the detection model implementation were presented in **Chapter 4**. While the explanation of the performed experiments and the used datasets are presented in **Chapter 5**.

The results obtained by the experiments in Chapter 5 are the primary content of **Chapter 6**. Finally, **Chapter 7** presents three main headings conclusion, recommendation and the possible future work for this study.

CHAPTER TWO

LITERATURE REVIEW

Botnet threats are a magnified extension of previous computer threats, combined with C&C systems and capable of infecting hundreds of thousands of computer systems. Despite botnet recency, this area witnessed a significant number of researches and proposed solutions. In this chapter, an overview about botnets is introduced. Also, this chapter presents the botnet and the related researches including infection mechanisms, botnet communication protocols, C&C models, malicious behavior, previous and current, bot detection methods and botnet defense. Moreover, some related topics like periodic signals, periodograms and circular autocorrelation function, also will be discussed.

2.1 Introduction

The related work that will be discussed in this chapter concerns previous approaches of botnet detection methods; their pros and cons. In particular, Signature Based, Anomaly Based, DNS Based and Data Mining Based Techniques will be discussed. Primarily focus will be concentrated on the efforts that have been made to detect botnets, based on known, anomalous, preprogrammed, repetitive and correlated behavior of botnet members. These effort's advantages and disadvantage will be discussed individually. Moreover, a comparison between those models will be performed and some examples of each of them will be provided.

2.2 Background of Botnet

For better understanding of botnet, some key terms are introduced that are related to the botnet community. The most related topics to botnet detection are shown in the general outline presented in Figure 2.1.

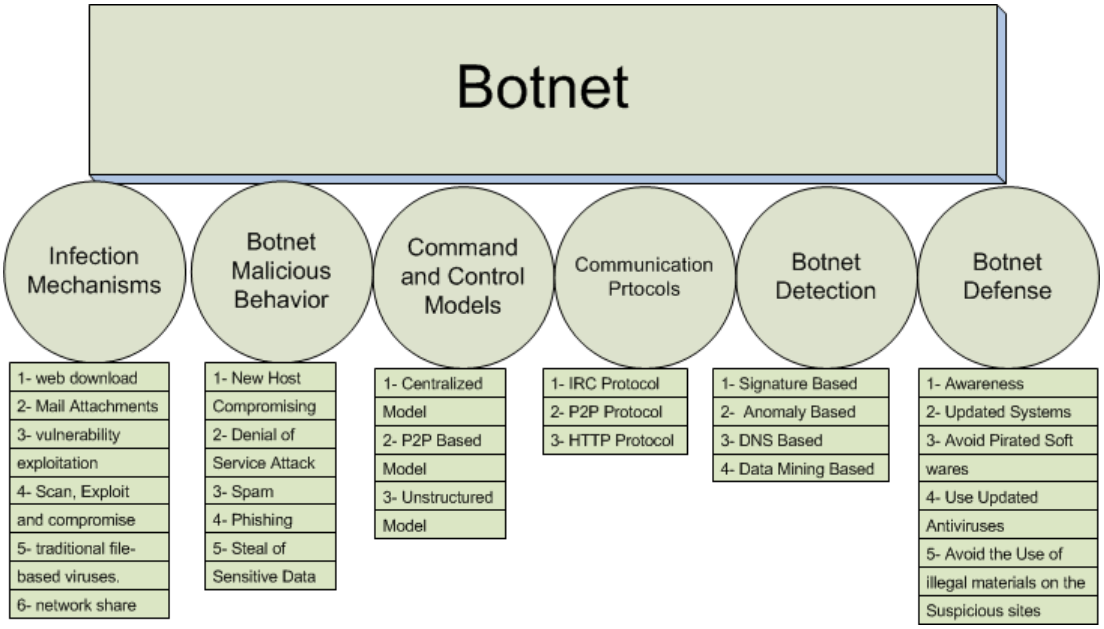

Figure 2.1 Botnet Overview

2.2.1 Definitions Related to Botnet

- ***Bot*** - the term bot comes from the word robot, which means "worker". In the world of computers, bot is a generic term, used to describe an automated process (Geer, 2005; Ianelli & Hackworth, 2005; Saha & Gairola, 2005). A bot is usually referred to as automated software, which is capable of performing certain predefined tasks repeatedly.
- ***Botnet*** - are a group of compromised computers (or zombies) that are under the control of a single entity called botmaster (Barford & Yegneswaran, 2007; Gu, Porras, Yegneswaran, Fong, & Lee, 2007; Saha & Gairola, 2005).

Ø ***Command and Control (C&C)*** is the commander channel that receives commands from the botmaster and conveys these commands to their bots in order to carry out various distributed and coordinated attacks remotely (B. AsSadhan, Moura, Lapsley, Jones, & Strayer, 2009; Bailey, Cooke, Jahanian, Xu, & Karir, 2009; Gu, Zhang, & Lee, 2008).

Bot infected computers can be controlled as:

- Directly, by initiating a connection with the infected computer known as the channel. Then, controlling it by commands hardcoded into the bot program, e.g. IRC and HTTP botnets.
- Indirectly, the bot initiates the connection with the control center/peer, sends a request and then performs the returned command e.g. P2P botnets.

2.2.2 Current and Expected Future Structure of Botnet

Botnet can be classified according to:

- Topology
- Communication protocols

∨ Botnet Classification According to Topologies

Botnets can be classified based on their C&C architectures as follows: (Chao, Wei, & Xin, 2009).

a- ***Centralized architecture***: in a centralized architecture, all bots are connected to a certain centralized C&C server, such as IRC and HTTP based botnets as shown in Figure 2.2. This architecture is considered as the easiest to construct

and implement; that is why this structure is still in use in the cyber world till now. However, this architecture suffers from the one single point of failure architectures, where it is easy to identify the C&C server and thus, bring down the whole botnet.

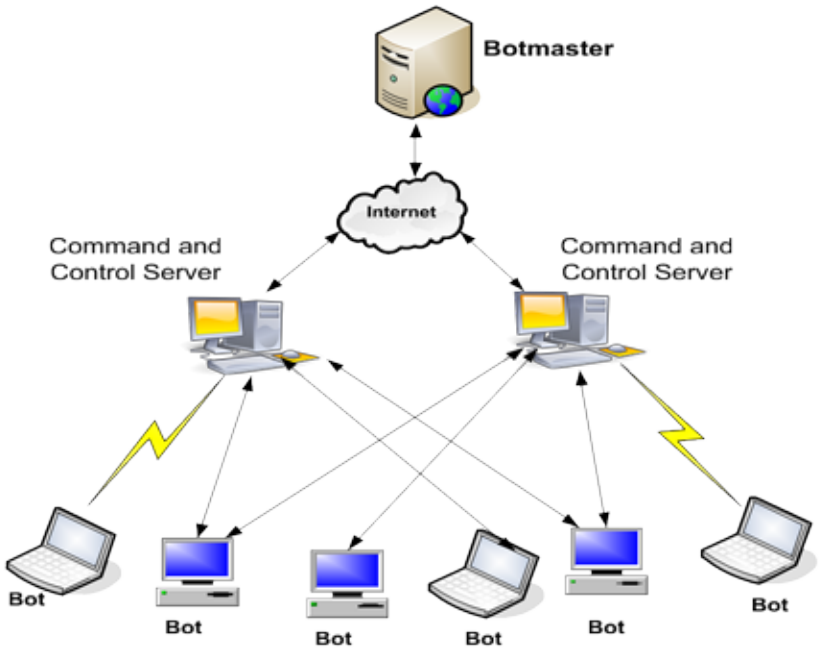

Figure 2.2 A typical Centralized Botnet structure.

b- ***P2P or Decentralized architecture***: shown in Figure 2.3, in this architecture there is no centralized point for C&C. So that any node in the network can act as a client and as a server, P2P architecture employs the P2P protocols to present a various distributed C&C servers. This architecture is considered difficult to discover and destroy, due to the anonymity and the distributed nature of the P2P architecture (Grizzard, Sharma, Nunnery, Kang, & Dagon, 2007).

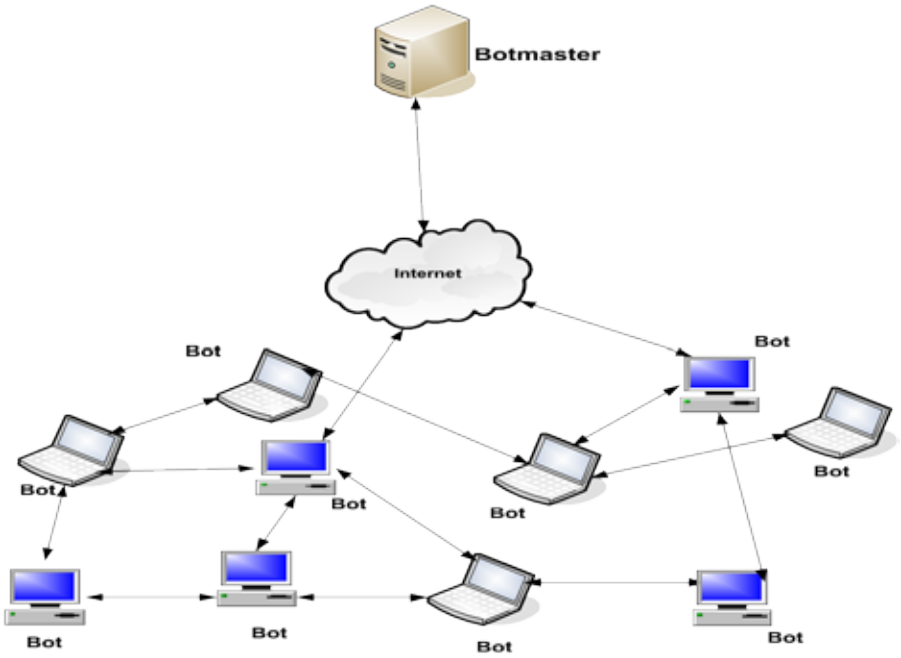

Figure 2.3 A typical Peer-to-Peer Botnet Structure

c- ***Unstructured C&C architecture***: shown in Figure 2.4 and considered to be the extreme case of P2P botnets; where each bot is connected to one peer and doesn't know anything about other peers in the botnet, and more importantly, the bots in this structure are randomly organized (Clarke, Sandberg, Wiley, & Hong, 2001; Gnutella, March 2001). In this model there is no direct connection between the bot and the bot master; the bot master has to search the Internet and posts the required tasks to the bot when it finds one. Such a system is simple to design and to implement. The single bot detection would never compromise the whole botnet. However, this structure will not be effective as other structures; as it doesn't have a guarantee of delivery and suffer from high message latencies.

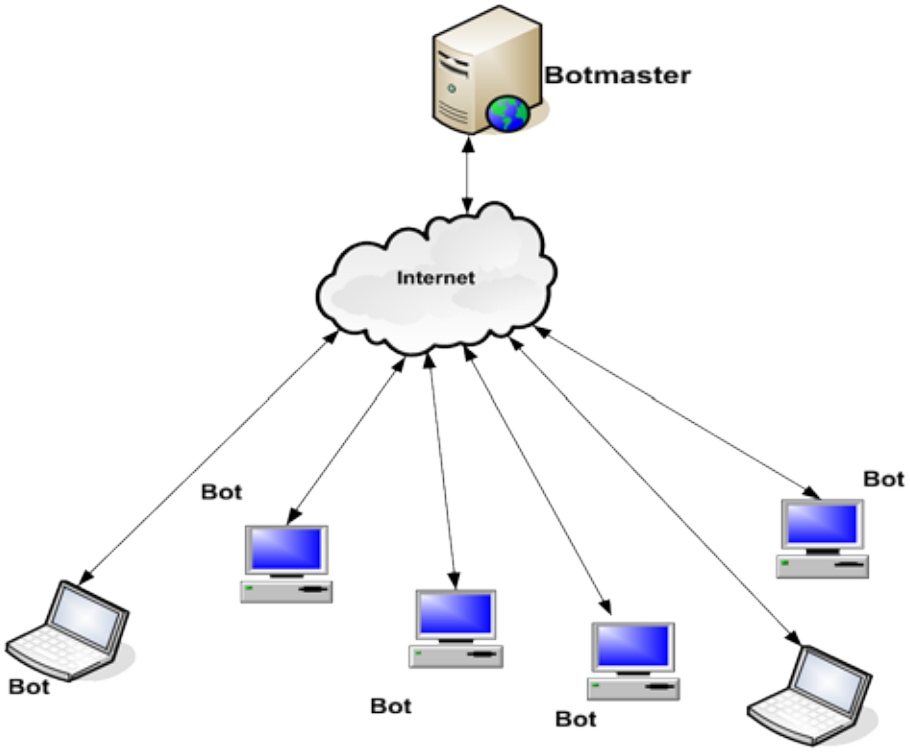

Figure 2.4 A typical Unstructured Botnet structure

The general properties of the different structures are summarized in Table 2.1

Table 2.1 C&C topologies and basic properties (Bailey, Cooke, Jahanian, Xu, et al., 2009)

Topology	Design Complexity	Detectability	Message Latency	Survivability
Centralized	Low	Medium	Low	Low
Peer-to-Peer	Medium	Low	Medium	Medium
Unstructured	Low	High	High	High

∨ Botnet Classification According to Communication Protocols

It is essential to have a communication channel between the bots and their owner, so that; the botnet owner can control his bots and send them the required commands. Establishing these channels (connections) and maintaining them are based on network communication protocols.

Therefore, based on the used network protocols; botnets can be classified as (Tyagi & Aghila, 2011):

a- ***IRC-based***: in this botnet, bots are controlled via Internet Relay Chat (IRC) channels. IRC was the first and the common protocol used by botmasters, to send commands to the infected machines. IRC can be easily detected and network security devices can be configured to block IRC traffic.

b- **IM-based**: which uses communication channels provided by instant messaging (IM) services such as AOL, MSN, and ICQ etc. IM is Low popular botnet communication channel; because it is difficult to create an individual IM accounts for each bot. Bots should be online all the time and keep connected to the network. IM services do not permit the same account, to log on to the system, from more than one host at the same time; each bot needs its own IM account; because automatic account registration is prevented in most of IM services. This will limit the number of registered IM accounts i.e. limits the number of concurrent online bots.

c- ***Web-based***: This technique is based on the popular communication HTTP protocol, which is difficult to be detected and can easily bypass network security devices. The botmaster controls his zombies from anywhere in the world through the World Wide Web by using the HTTP. The bot master defines a web server, the bot connects to the defined web server, receives commands and responses back to the same web server.

d- ***Other***: Botnets that use their own protocols to communicate, protocols that are based on the TCP/IP stack, i.e., that use transport-layer protocols such as TCP, ICMP and UDP.

2.2.3 How Bots Work

Bots spread across the Internet by exploiting vulnerabilities on unprotected computers to infect them and to report back to their master. Then the bots stay hidden until they are instructed to carry out another task. Based on 2010's results, the top 20 most common vulnerabilities were found in software developed by four companies: Microsoft, Adobe, Oracle and ACDSee, as shown in Figure 2.5.

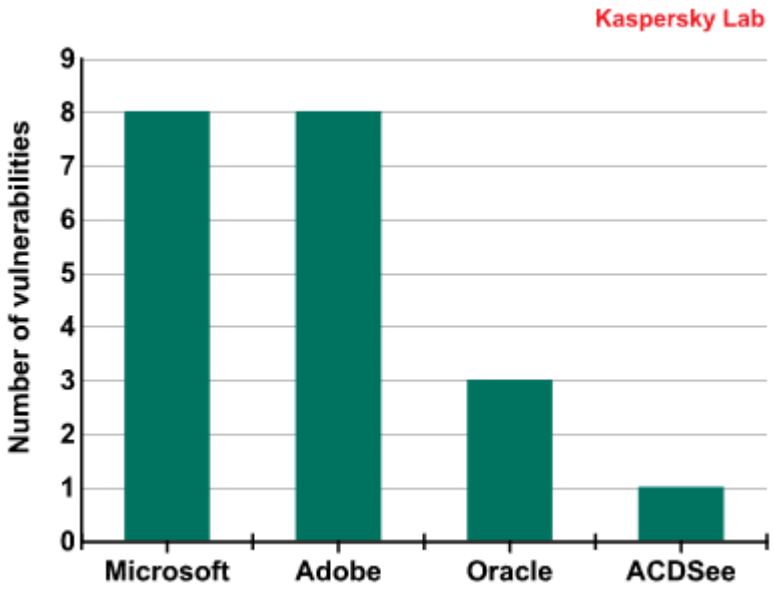

Figure 2.5 list of vulnerabilities among some of the well known softwares.

The compromised computer can be used to carry out a variety of automated tasks. Such as, Sending (Spam, Viruses, and Spyware), stealing personal information (credit card numbers, bank credentials, email address lists and Other sensitive personal information), launching denial of service (DoS) attacks, clicking on internet ads to boost up web advertising billings and extortion in which attackers ask to be paid, or they will attack the online services or the website of a certain company.

2.2.4 Botnet evolution

Like many services on the Internet, bots started as a useful tool without any malicious intent. Bots originally, were developed to sit on an IRC channel and perform several tasks, for its owner. Bots evolved from playing games with IRC users (GM: IRC bot, 1989, by Greg Lindahl), to a password stealer and backdoor (PrettyPark, 1999 'Trojan.PSW.CHV'), then it has the ability to remote control IRC clients by utilizing IRC vulnerabilities (SubSeven Trojan/Bot:By the late 1990s) (Tyagi & Aghila, 2011).

In 2000, Global Threat (GTBot) appeared, this bot can execute commands in response to events on the IRC server, and it supports raw TCP and UDP socket connections. GTbot had the capabilities of port scanning, flooding and cloning etc. (Jing Liu, Yang Xiao, Kaveh Ghaboosi, Hongmei Deng, & Jingyuan Zhang, 2009). In 2002, SDBot appeared which represents a new era in the evolutionary chain for bots with available source code, which made it accessible to many hackers. Moreover these types were easy to modify and to maintain (Jing Liu, et al., 2009). In 2002 bots with modular design appeared (Agobot, aka Gaobot, 2002), the modular design; allows the botmaster to update modules as new techniques or sites are available (Kola, 2008).

The available bot evolution techniques, lead to the creation of botnet that depends on unique characteristics, rather than depending on the original code, like Spybot and MyTob, for example (Polybot, March of 2004) has the capability to appear in many different forms. The use of hybrid, social engineering and spoofed e-mail addresses appeared with (Mytob, 2005). Botnets moved away from the original IRC Command

& Control channel, and began to communicate over HTTP, ICMP and SSL ports, Fast Flux network based on DNS servers and of course the P2P protocols. (Sinit, 2003) is an example of the early malicious Peer-to-Peer bot. In January 2007, the Trojan.Peacomm bot appeared; it was the most recently known peer-to-peer bot (Chao, et al., 2009).

After exploiting all available protocols, botnet developers turned their attention to network architecture, moving their botnets structure from the legacy classic architecture (i.e. a centralized structure with one or more C&C), into the dynamic P2P structure, which has no C&C, large botnet based on P2P architecture appeared in 2007. P2P botnet have attracted Bonet developer as well as botnet researcher.

2009 was characterized by the increased sophistication and the complex malicious programs that have rootkit functionality, for example the year 2009 witnessed (global epidemics, web attacks, web botnets, SMS fraud, the use of new platforms such as Mac OS and mobile operating systems and attacks on social networks) (Szor & Kaspersky).

2010, The Year of the Vulnerability (Bail, 2011), Web malware grew by 139 percent in 2010 compared to 2009 as shown in Figure 2.6. Numbers of used botnet-technologies have progressed dramatically. Botnet is constantly growing more and more complex, (e.g. Mariposa, ZeuS, Bredolab, TDSS, Koobface, Sinowal and Black Energy 2.0 botnets); which are considered among the most sophisticated malware ever created. 2010 also witnessed compound efforts of law enforcement agencies, antivirus vendors and telecom providers in trapping of cybercriminal and illegal

services or business on the Internet. The P2P share was present to declare an increase in peer-to-peer (P2P) activity and again to focus more on the rule of P2P in the future direction of botnet.

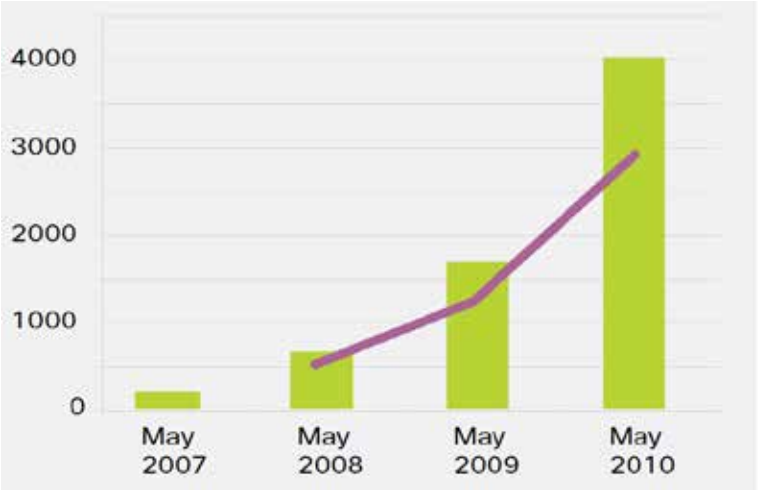

Figure 2.6 The growth of Web Malware for the years (2006-2010).

Also it was noticed that in the year 2010, attackers have shifted from internet and user's pc toward mobiles. A significant increase in mobile malwares gives us a black picture for the future of mobile botnets as shown in Figure 2.7.

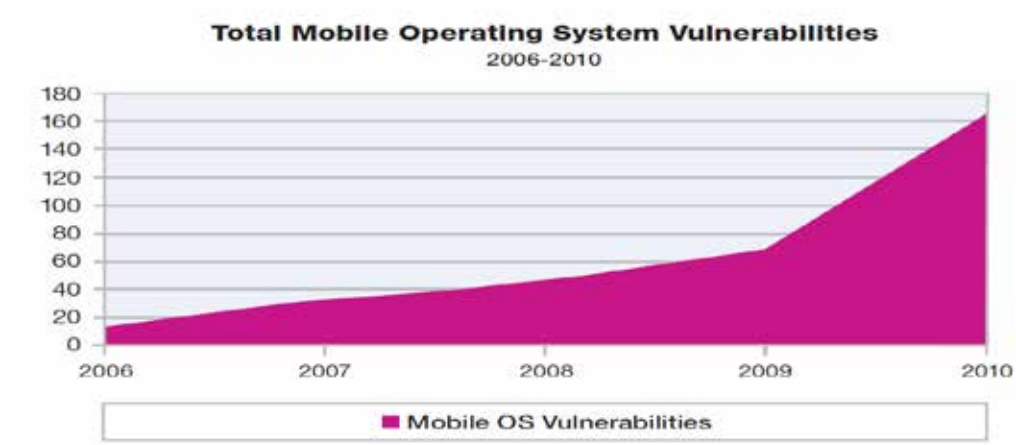

Figure 2.7 Total Mobile Operating System Vulnerabilities for the years 2006 to 2010, IBM

2.2.5 Botnet Potential Benefits

Botnets constructing have one or more of the following information processing, information harvesting and information dispersion. Information processing is used to process data, such as cracking passwords. While the use of information harvesting may includes obtaining financial data, password data, identity data, relationship data (i.e., email addresses, list of friends) or any available data on the host. Information dispersion includes providing false information from illegally controlled sources, creating denial of service attacks and sending out spam.

2.2.6 Botnet Infection Mechanisms

Botnets utilize many different infection mechanisms, such as employing malware (i.e. worms, trojan insertion), web driven-by download, mobile media, vulnerability exploitation, mail attachments, automatically scan-exploit-and-compromise, traditional file-based viruses, network share, as well as social engineering techniques and P2P file sharing networks, etc (Chao, et al., 2009).

2.3 Botnet Life Cycle & Detection Systems

Creating Botnet begins by sending a malware to vulnerable machines. Once vulnerability is found, the machine will be compromised; leading to the malicious bot binaries to be downloaded into the compromised hosts, turning it into a zombie (bot). This new bot in return will be redirected to a dynamic/static server address that is known for both the bot and his master. This server is known as a C&C server, where the botmaster (attacker) can login and issue commands to his bots to start an attack, scanning, infection…etc, (Chao, Wei et al. 2009). The most general phases in

Botnet lifecycle are: spread, infection, C&C, and attack, as shown in Figure 2.8. Botnet life cycle events include:

1- Victim browses a website or clicks a link on email (e.g. phishing, drive-by download), then the browser is redirected to a malicious dropper site.
2- Victim is directed into downloading the dropper - or dropper is automatically downloaded through an exploit.
3- Dropper unpacks on the infected machine and runs.
4- Dropper informing its botmaster that it joined to the botnet.
5- The C&C secure the new client, sends encrypted malware with new instructions.
6- Dropper decrypted the malware and installs it. The dropper has to vanish by hiding, or delete itself so that users believe that no infection has occurred. Infected machine is turned into zombie (bot)
7- Malware contacts C&C, sends passwords/data/etc. as encrypted payload.
8- C&C updates the bot status and sends new instructions.
9- Bot responds by executing the commands and performing the required actions.
10- The bot contacts C&C sending its report.
11- In certain situations the bot is recommended to erase all commands and vanishes to remove any evidence on the botnet existence.

Steps 7, 8, 9 and 10 repeat indefinitely with the malware ‘evidence’ and C&C connection instructions changing constantly. The malware can be told to lay silent for a specified period of time.

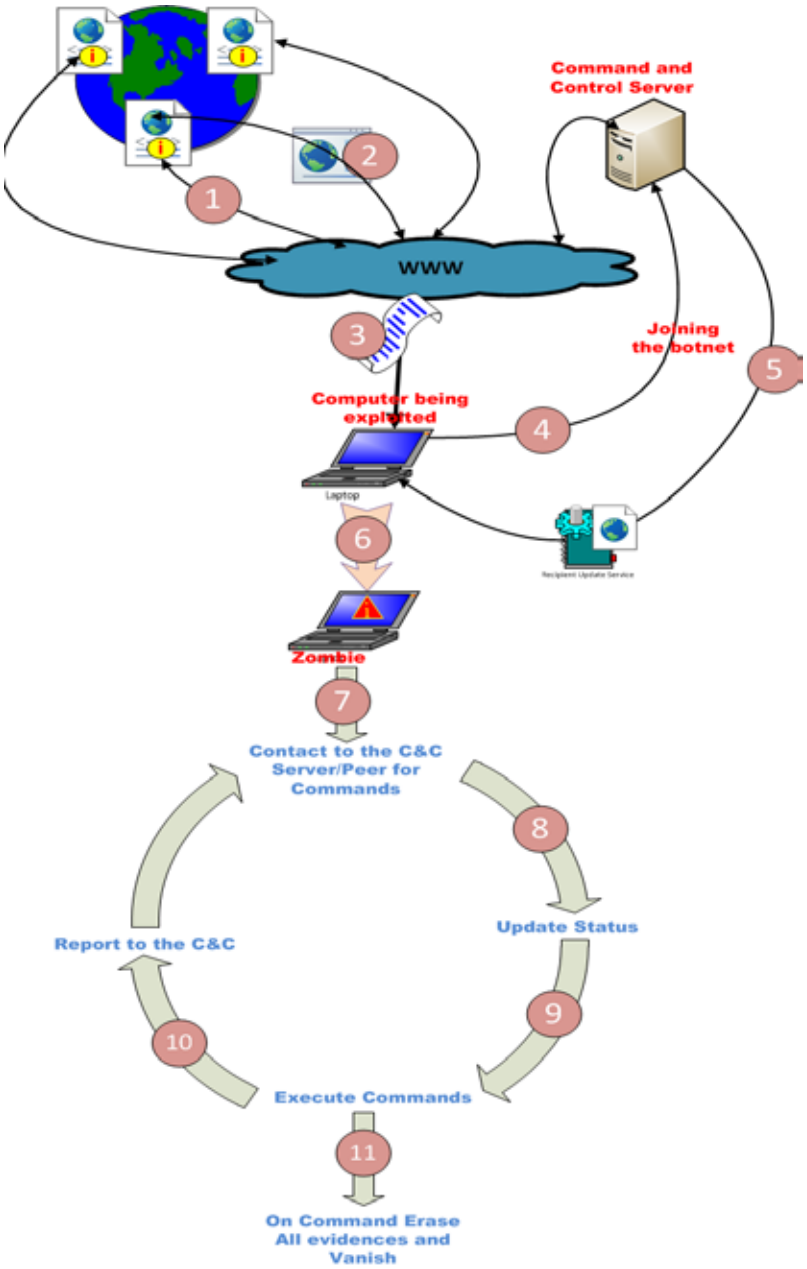

Figure 2.8 botnet life cycle.

Analyzing the malware after it enters the organization to obtain C&C details, can help removing it. Unfortunately, analyzing alone will not be enough as the infection lifecycle changes so quickly in a way that the analyzed malware no longer exists on the victim's machine.

2.4 Botnet Detection Methods

How the host is infected is not important as how to heal it, once the bot is created within the network hosts, the first priority is to identify the infected host and to heal it. Therefore, many efforts were initialized to detect botnets. The past recent years witnessed different approaches that have been proposed to detect botnets and to combat their threat against cyber-security. These approaches can be grouped into Signature Based, Anomaly Based, DNS based and Data Mining Techniques.

2.4.1 Signature-based Detection

Signature-based Detection explained in Figure 2.9 examines the network traffic for known patterns of a malicious activity; new types of attacks are not detected. Signature-based detection involves searching among network traffic, for a series of bytes or packet sequences or a set of attributes and matches them against a set of predetermined attribute lists. In case some particular network traffic has a match, the system has to alert administrators or to take a pre-defined action.

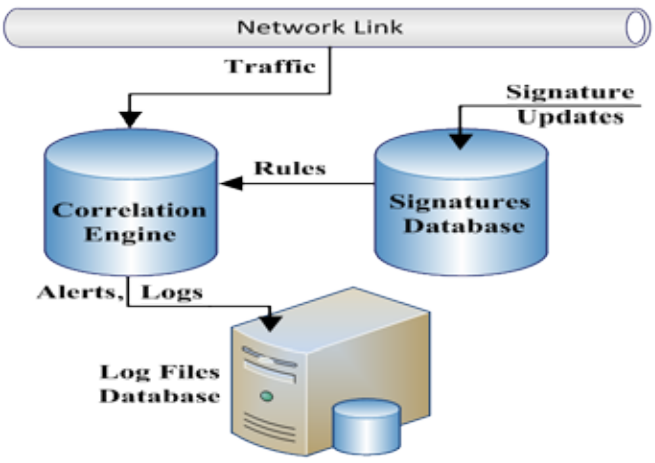

Figure 2.9: a simplified figure representing a signature-based detection model