Criminology of BotNets and their Detection and Defense Methods

Jivesh Govil, *Student Member*, IEEE
Dept. of Electrical Engineering & Computer Science
University of Michigan,
Ann Arbor, Michigan, USA
jivesh@umich.edu

Jivika Govil
Dept. of Information Tech. and Computer Science
Apeejay College of Engineering, MD University
Gurgaon, Haryana, India
jivikag@email.com

Abstract—**Internet has been recently witnessing dramatic increase in malwares. Maliciously compromised machines termed Bots have been figured to be the major reason for Internet malware epidemic. Further, the BotNet—the network of Bots—detection is difficult and possible only after they have spread widely. Though the existence of these BotNets has been acknowledged, their attributes have not yet been fully construed owing to their distributed nature. A recent issue has been to develop efficient detection technologies to combat BotNets. This paper essentially presents our efforts to disseminate an understanding of BotNets presenting an outline on the types of BotNets and their despiteful characteristics. This paper highlights various detection mechanisms with an aim to seek insight into their efficiency and subsequent issues arising from variety of perspectives. Moreover, recommendations for defense against BotNets have also been mentioned. The goal of the paper is to present research community an expatiation to develop a unique efficient solution for BotNet detection and control.**

***Index Terms*—Bot, BotNet, IRC, Malware**

I. Introduction

BotNet refers to a collection of Bot—a type of malware which allows an attacker to gain complete control over the affected computer—running on a Internet Relay Chat (IRC) network that has been created with a Trojan. They are a member of Internet computers that have been setup to forward transmission to other computers on the internet. They are basically a collection of software robots, or Bots centrally controlled using the IRC protocol.

BotNets very greatly in size, with small networks being only tens of strategically placed hosts and the largest one being hundreds of thousands of hosts [1]. Attackers with automatic techniques install IRC Bot (also referred as zombie and drone) on it. The majority of BotNets uses IRC Protocol [2]. IRC is a form of real time communication between one to one, and one to many. The Bot joins a specific IRC channels on an IRC server and waits for further commands as attackers remotely control the Bot and use them according to their own requirement. Furthermore, attackers bring different Bots together and manage them from IRC channel collectively forming BotNet (freely available IRC servers are used by attackers). Applications running on the BotNet platform include click fraud, identity theft, denial of services, key cracking etc [3]-[4]. Spam is another security problem posed by BotNet platforms. The numbers of new Bots have been observed to increase rapidly everyday. BotNets with more than 100,000 members have been discovered [5]-[6]. This is an indication of serious problem. Majority of victims are home computers and small businesses running on less secure operating systems like Windows, even without firewalls, with an access to high speed Internet. Malicious programmers use forbidding threats like worms, Trojans and spy wares etc to attack these computers, generally not having specialized and sophisticated anti-virus packages. Malicious HTML files are also prepared to exploit the vulnerabilities of browsers and spread via peer-to-peer network.

In this paper we outline the problems arising from BotNets. The paper has been structures as follows. Firstly, an overview about different BotNets along with an insight to their characteristics has been presented. This is followed by exemplification of malicious nature of BotNets. Then, BotNet detection techniques and their evasion by BotNets have been discussed. Finally, some recommendations for defense against malicious BotNets have been made. This study has been performed to provide a strong foundation for scientific community and motivate then to develop unified approach to combat BotNet maleficence.

II. BotNet

Bots are software Robots that are responsible for performing tasks automatically. Bots gather information for search engines (called search Bots) and shopping sites (called shop Bots) as well as thousands other process. As aforementioned, BotNets are the networks of computer systems using IRC or related capabilities for communication, command and control.

A. Basic IRC Operation

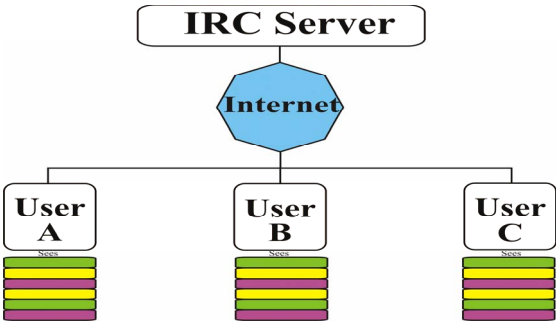

Fig. 1. Basic IRC Operation user submits the message.

Fig. 1 provides a simplified view of IRC operation. It can be observed that each user is being connected to same IRC server and each user sees the same message whenever any particular user submits the message.

In other words IRC becomes multiple 'channels' or discussion area, so a user wishing to participate in a particular discussion must (a) have an IRC client (b) know the IP address of a server hosting the desire clients (c) know the channel name. The default ports of IRC service are 6665-6669/TCP and port 6667/TCP is the most common. IRC popularity has led to the establishment of IRC networks, which are collections of connected IRC servers, providing higher capacity and redundancy.

B. BotNet Lifecycle

BotNets are often run by malicious programmers with specialized skills while advance attackers operate the control channel. These people use the BotNet for commercial uses and 'sell' the services. They also install special softwares to steal information. Some attackers are highly skilled and belong to well organized crime structures. Armed with power of thousand Bots, it is viable for them to take down almost any website or network instantly. In other words BotNets are power weapons. Fig. 2 illustrates the various stages in a typical BotNet life-cycle.

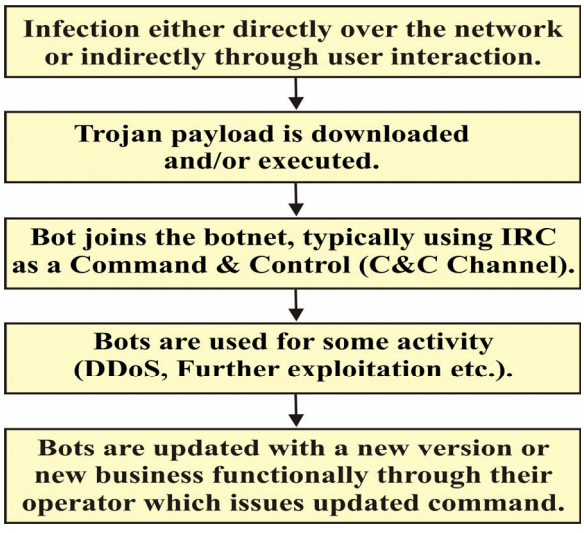

Fig. 2. Life Cycle Phases of BotNet

C. BotNet Structures

The 'Bot' itself is actually a computer code which runs on a client system, not all Bots can participate in a BotNet, but many that do are widely available. There are two main types of BotNet structures, Hub-Leaf BotNet and Channel BotNet.

1) In Hub-Leaf BotNet, two Bots are connected by installing the Bots on the target client systems, and then one Bot becomes a Hub and other a Leaf. The building of Leaf Bots continues and is connected to the same Hub. Also independent BotNet can be merged to join their Hubs. This Hub-Leaf BotNets does not use IRC for communication but communicate via Bot specific listeners and hence each Hub-Leaf pair of ports could be unique.

2) *In Channel BotNet*, Bot identifies or establishes an IRC channel (generally a password, or key) for BotNet communication. Each Bot joins the channel and the controller issues command by posting messages to the IRC channel, which the Bots reads and interprets.

D. BotNet Initiation and Spread

It is to be noted that installation of a Bot on a client system has the same requirements as installing any application. The client system operator must perform the install and must be tricked into performing the install, or vulnerability must be exploited to perform the install.

BotNets propagate like worms, hide like viruses and launch coordinated attacks. Bot masters gather Bot infected PCs into the BotNet evolving own new Domain Name Servers (DNS) analogous to an ISP that uses a dynamic DNS to assign an internet domain name to a computer with a varying IP address. Bots include hard coded domain names assigned by dynamic DNS providers. Some newer BotNets run their own distributed DNS service that run at high port numbers to evade detection by security devices.

E. Communication by Bots

1) *IRC based command and control:* This is the most common technique used by Bot masters to communicate with their Bots. BotNets in most cases uses existing communication protocols rather than creating a new network protocols. IRC protocols are used by Bot masters to command their whole BotNet army as well as to command a few Bots selectively. Firewalls can be configured to block IRC traffic; however it is more difficult to detect IRC channels tunneled in HTTP. HTTP protocol is a recently been used as a new communication method by BotNets blending themselves into the internet traffic and making themselves difficult to detect. BotNets using the HTTP protocol usually bypass security policies.

2) *Centralized:* A central node forwards message between the various clients. This system can be easily detected as all communication passes through a single point in the network. Once the central location is discovered, the whole BotNet can be easily compromised. Hence, it is a weak method of communication for BotNet.

3) *P2P: Advance BotNets* uses IM protocols and peer-to-peer (P2P) protocols, the emerging technology. P2P offers

many incentives for their uses in BotNets. P2P communication is not disrupted even if one or few of the Bots are compromised. Also some existing anonymous techniques make the P2P communication anonymous. Presently, they are relatively small but in future they may get wider and will possess more challenges for BotNet detection [7-8].

F. Analysis of BotNets

As explained above, a BotNet is a network of compromised machines that can remotely be controlled by an attacker [4]. Bots joins a specific IRC channels and waits for further commands by the remote controller (attacker). Attackers also bring different Bots together and are called BotNet. IRC is not the best solution since the communication between Bots and their controllers is rather blotted; a simple communication protocol would suffice. Free availability and easy setup are the advantages of IRC. A relatively small BotNet with only 1000 Bots can cause huge damage. These 1000 Bots could have a combined bandwidth (1000 home PCs with an average upstream of 128KBit/s can offer more than 100MBit/s), which is empirically higher than the Internet connection of most corporate systems. In addition, the IP distribution of the Bots renders maintenance and deployment difficult. Further, incident response is hampered by the large number of separate organizations involved. Thus, the use of BotNets in stealing sensitive information or identity theft and searching some thousands home PCs for password.txt, or sniffing their traffic etc., can be effective. The spreading mechanism used by Bots is a leading cause for "background noise" on the Internet, esp. on TCP ports 445 and 135. These malware scan large network ranges for new vulnerable computers and infect them. The traffic on four ports (445/TCP, 139/TCP, 137/UDP, 135/TCP) account for more than 80% of the whole traffic captured.

III. BotNet Malicious Activities

Bots facilitate IRC channels administration and monitoring. BotNet enhances the Bots' capabilities, and also add capabilities (to establish 'private' networks). Bots are capable to create and operate private network. These networks are then widely use for file transfer and distributed file storage. There is no control mechanism to detect what type of files are stored or transferred. These files may include illegitimate material.

BotNets can be abused for coordinated network attacks under single control. They are often misused for denial of service attacks, either against channel operators, in order to obtain channel control, or against distinct targets like web servers, in order to render them unavailable for long period. BotNets are also misused for distributed scanning, vulnerability exploitation, distributed computation (breaking codes by brute force), email spamming/bombing, malware distribution, and any illegitimate activity which can be partitioned among multiple systems.

Various malicious activities that are performed by BotNets are enumerated below.

1) *Attacking IRC Networks:* BotNets are used for attacking IRC networks. The victim is flooded by service request from thousands of Bots and thus victim IRC network is brought down.

2) *Distributed Denial of Services (DDoS):* DDoS is a attack on a computer system or network that causes a loss of services/network to users by consuming the bandwidth of the victim network. The resource path is exhausted if the DDoS-attack causes many packets per second (PPS). The DDoS attacks are not limited to Web servers, virtually any service available on the internet can be target of such an attack. Higher level protocols can be misused to increase the load even more effectively by using very specific attacks such as such as running exhausting search queries on bulletin boards or recursive HTTP-floods on the victim's website called spidering.

3) *Key Logging:* With the help of a key logger it is very easy for an attacker to retrieve sensitive information. There exists filtering mechanism that aid in stealing secret data.

4) *Sniffing Traffic:* Bots can also use a packet sniffer to watch for clear text data passing by compromised machine. The sniffers are used to retrieve sensitive information such as usernames and passwords.

5) *Spamming:* Some bots can open a SOCKS v4/v5 proxy—a generic proxy protocol for TCP/IP-based networking applications—on a compromised machine. After having enabled the SOCKS proxy, this machine can then be used for nefarious tasks such as spamming. With the aid of BotNet, an attacker can then send massive amounts of bulk e-mail (spam). Some Bots also harvest e-mail addresses (by opening a SOCKS v4/v5 proxy).

6) *Advertisement Installation:* BotNets setup a fake web site with some advertisements. The operator of this website negotiates a deal with some hosting companies that pay for clicks on ads. With the help of BotNet, these clicks can be 'automated' so that instantly a few thousands Bots clicks on the pop-ups, hijacks the start page of a compromise machine so that the 'clicks' are executed each time the victim uses the browser.

7) *Spreading New Malware:* This is easy since all Bots implement mechanisms to download and execute a file via HTTP or FTP. Thus, spreading virus via e-mail is very easy using a BotNet.

8) *Manipulating Online Polls or Games:* These are very easy to manipulate due to high attention. Since every Bot has a distinct IP address and do the manipulation. Every vote will have the same credibility as a vote cast by a real person. Online games can be manipulated in a similar way.

9) *Mass Identity Theft:* By combining above different functions, they are used for large scale identity theft which is one of the fastest growing crimes on the internet. Bogus e-mails that pretend to be legitimate (such as banking e-mail) ask their internet victims to go on line and submit their private information. The fake e-mail are generated and sent by Bots via their spamming mechanism. These Bots can also host multiple fake web sites and as and when one of these fake sites is shut down, another one can pop up. In addition, key-logging and sniffing of traffic can also be used for identity theft.

IV. BotNet Detection and Characterization

IRC (Internet Relay Chat) is an internet protocol which allows multiple connected users to engage in a real-time textual dialogue, so that a user can monitor a 'conversation' between multiple entities, and can participate in the conversation also [9].

A. Detection of BotNet

Detecting BotNet is a difficult task since BotNets can be detected only when they have spread largely, such as DDoS attack. Information is needed from variety of data sources before a BotNet malicious activity can be detected. There are active and passive methods for discovering active BotNets [4].

1) *Active detection* involves capturing live instances of running BotNets which are trying to propagate either directly by exploiting a vulnerable network service or indirectly through user interaction, i.e. through malicious spam email or web page.

2) *Passive Detection* involves capturing of suspicious traffic from the network.

B. Characteristics of BotNet Detection

BotNet detection can be summarized in terms of parameters:

- Scope indicates the capability of BotNet detection mechanism to track BotNet infections from individual hosts to multi-organisation networks.
- Detection time indicates the time that a detection mechanism takes to identify new BotNets, which can be early (before the BotNet has propagated) or late (BotNet has already caused a massive infection).
- User indicates the kind of user is the detection mechanism is used by.
- Type indicates whether the tracking by detection mechanism is direct, where mechanism measures BotNet activity directly, or indirect, where only side-effects of BotNet activity are measured.

C. Indicators of BotNet Activity

The following enumerates various activities that can be suspected to have been caused by malicious BotNets. These vary with particular network, system or traffic involved.

1) Unauthorized traffic on port 6667/TCP, the default location for IRC servers or IRC traffic on other ports.

2) IRC traffic which includes non-human messages. Normal IRC traffic looks like human conversation as person to person; traffic which contains apparent codes, system IDs, or structured messages.

3) Traffic between systems which have no legitimate reason to communicate. Floods of UDP or ICMP traffic may indicate a BotNet attack in progress.

4) Attempts to compromise systems indicate an attempt to set up a BotNet.

6) High volume of legitimate looking traffic is an indicator of compromise system which may or may not be BotNet attack

7) Unusual system behavior to be investigated for unexpected high volume of traffic.

V. Evading Detection by BotNets

With the growth of computer science knowledge and programming abilities, BotNets are becoming more and more sophisticated everyday, and thus better at evading detection. Not only are state-of-the-art Bots better able to evade Antivirus (AV) engine and signature-based intrusion detection systems, they are also becoming more evasive to anomaly-based detection systems. BotNets evade AV and signature based IDS systems via methods such as executable packers, rootkits, and protocol evasion techniques, which also improve the survivability of BotNets and the success rate of compromising new hosts. BotNets continue to add new mechanisms that hide traces of their communication. Now-a-days, BotNets are moving away from IRC, to either modified IRC protocols or HTTP or VoIP protocols. Sometimes, Bots use encryption schemes to prevent their content from being revealed. State-of-the-art BotNets now use TCP, ICMP tunneling, and even IPv6 tunneling. Hence, as more modifications are introduced by attackers, there becomes a pressing need to evolve better detection methods.

VI. Detection Methods for BotNets

As mentioned in previous section, the range of data sources for detecting an active BotNet varies in scope, latency and interactivity. Different Data sources are used for detecting an active BotNet; some of them are described below and summarized in Table I.

A. Victims

The first detector of BotNet infection is often a end-user, whose computer has been compromised. Investigation can be carried out about the control mechanism used from the victim computer but it has a limitation that usually yields little about the criminal activity it has been used for.

B. Honeypots and Spampots

Honeypots are placed on the network and closely monitored for infection. Honeypots gather information about the operation of malware, including obtaining the Trojan payload and monitoring the BotNet traffic.

TABLE I
DIFFERENT BOTNET DETECTION METHODS

Data Source	Scope	Detection Time	User	Type
Victim	Individual machine	After infection	Unhappy end-user	Direct, Indirect
Honeypot or spampot	Varies	Early	Security researcher	Direct
Antivirus software	Individual machine	Infection attempt	End-user, network operator	Direct
IDS with signature	Network	Infection attempt	Network Operator	Direct
IDS without signature	Network	After Infection	Network Operator	Indirect

DNS-based IDS	Network	After Infection	Network Operator	Indirect
Flow Data	Several networks	Early to postmortem	Network operator	Direct, indirect.

Nepenthes is a tool that interacts with the worm trying to spread to the honeypot. Spampots collect malware by harvesting e-mail spam to find new and unknown instances [10]. The drawbacks of honeypot are (a) they require being infected first and when the infection occurs through a worm, which attempts to attack as many host on the internet as possible, attention needs to be paid to the egress filtering of the traffic generated by the infected host (b) operating a honeypot poses a legal problem in terms of data privacy and liability issue; (c) bots may contain antiforensic capabilities, such as honeypot detection mechanisms. However, properly operated honeypots are capable to provide valuable intelligence on Bots software and the C&C channels used.

C. Anti-Virus Software

Anti-Virus (AV) software is generally used to stop malware by matching the signature of malicious activity as changes to the operating system and/or its network connectivity. Once the signature is matched, the normal procedure seems to be an attempt to quarantine the malicious code and to notify the computer owner or a central AV management console. In a corporate setting, this can yield in a heads-up for the systems administrators, but unfortunately AV engines can detect only malicious code which has been identified before. Evaluating the usefulness of AV software as an information source for BotNet investigations depends largely on the particular deployment of software.

D. Intrusion Detection Systems

Data collected from the network do not interact with the mechanisms installed. Attacks are signature based and detect malicious activity only when system is tuned to detect. BotNets used IRC as control mechanism, but other protocols like P2P are in the process of being adopted. BotNets are using ephemeral port numbers for their IRC servers and some BotNets are encrypting the C&C traffic; hence detecting BotNets traffic from network is difficult. Passive detection mechanism is required to identify the secondary features of Bot infection like propagation or attack behavior detection [4].

E. DNS-based Intrusion Detection Systems (IDS)

A new type of IDS for BotNets uses analysis of DNS queries to find misbehaving hosts. BotNets typically uses DNS to find the IP address of the controller, and the controller quickly moves to new host as previous ones are disconnected [11]. A DNS based IDS looks for anomalous DNS queries and logs them. DNS based IDS are, however, very susceptible for false-positives and are not conclusive for marking bad hosts with 100% percent certainty. Therefore, additional information such as NetFlow data should be used to find compromised hosts and controllers. While implementing passive DNS, only the queries and their responses are logged, not the individual host generating the queries [12]. DNS logging infrastructure will not be beneficial unless a detection mechanism is incorporated into this approach.

TABLE II
TYPES OF DATA AVAILABLE IN TYPICALNETFLOW RECORD

Start Time	End Time	Source Interface	Source Address	Source Port	Destination Interface
Destina -tion address	Destina -tion port	Protocol	TCP Flag	Packets	Bytes

F. Flow Data

One of the sources of information about BotNet infection that is available to organizations is NetFlow data gathered on the traffic crossing the network border. NetFlow contains data for each flow of traffic traversing a network router. Several different formats are used to encapsulate NetFlow data. The most recent version of NetFlow is extensible format, which currently defines 89 field types (.e.g. MPLS labels, IPv6 addresses and AS numbers associated with the data) [13]. Table II shows kinds of fields typically are available in a NetFlow record.

Existence of BotNet could be identified but effective analysis require correlation of flows between organizations, which often is not possible due to technical and legal reasons. Also, storing and sharing data may be a problem for large organizations because additional backups of gigabytes of data are required for the fact that volume of traffic is very high. Isolating the BotNet traffic from regular traffic (and possibly anonymizing it) makes sharing of the data possible. Address anonymization can be done by Cryptography-based Prefix-preserving Anonymization algorithm (Crypto-Pan) [14].

The above method for anonymizing has many benefits. It is prefix-preserving, that is unanonymized addresses with a k-bit prefix will share a k-bit prefix when anonymized. The anonymization is also cryptography-based and is consistent across traces (if same cryptographic key is used). CANINE is a tool for applying this algorithm to NetFlow data [15]. Finally, while analyzing the data address, anonymization causes difficulty as the same C&C hosts appears in several traces. Thus, there is a need for a mechanism for querying whether two anonymized addresses are the same (without revealing the actual identity).

VII. DEFENSE AGAINST BOTNET

BotNet defenses can be grouped by phase (prevention, detection, and response) and by role (agent or target). Defenses for each of the resulting six situations (three phases and two roles for each) are presented in Table III.

If the system has BotNet.exe file it has to be fixed immediately otherwise it may cause number of problems such as slow performance, loss of data and leaking information to web sites. Recommendations for defense against BotNet for home and corporate user have been summarized in Table IV.

TABLE III
BOTNET DEFENSES

1(a) Prevention for Potential Agent	- Patch and maintain systems - Implement and monitor perimeter defenses. - Enforce safe e-mail handling policy. - Protect remote users systems. - Educate users.
1(b) Prevention for Potential Targets	- All above - Establishing DDoS defenses as appropriate (multiple internet connection, IP switching ability, traffic filtering/throttling etc.
2 (a) Detection of Agent Activity	- Monitor network traffic flow for BotNet activity. - Monitor file transfers suspect e-mail etc. - Monitor IDs systems for known BotNet activity. - Monitor systems for BotNets behavior. - Monitor traffic profiles for anomalous activity.
2 (b) Detection of Target Activity	- Follow General Detection techniques. - Distinguish through distributed source.
3 (a) Response for Agents	- Remove the systems from network. - Isolate the system if a trap and trace strategy is employed. - For preserving evidence consider hard shut down. - Preserve the data and relevant system locks (firewalls, mail servers, IDs, DHCP Server, Web proxy locks etc.) - Report to Law Enforcement Agency.
3 (b) Response for Targets	• Try filtering if small set of source IPs are detected. • Change the target's IP address and black whole the attack traffic upstream. • In sophisticated, flexible and coordinate attacks employ the following defenses. - Recognize (fingerprint) the attack traffic and block it upstream. - Add bandwidth (depends on the size of the attack) - Provide only essential services, or service only a subset of the population. - Employ rate-limiting (prior to or during an attack). Employ attack-specific defenses (i.e. SYN flood defense using host stack tuning perimeter defenses. etc.)

TABLE IV
DEFENSE AGAINST BOTNET FOR HOME AND CORPORATE USERS

Defense against BotNet for Home Users	• Keep operating system updated • Keep antivirus/anti spyware programs updated • Avoid installing anything i.e. no unnecessary and undesired installation • Avoid clicking on links in e-mail messages, instant messages and social networking websites • Stay away from file sharing networks
Defense against BotNet for Corporate Users	• Look for some key information e.g. IP address of the server or nick name of the Bot. • Observe BotNets • Collect binaries of Bots and extract the sensitive information • Monitor the typical commands issued by attackers and try to capture their communication • Learn about the motive of attackers and their tactics • Develop an automated method to catch information about BotNets and a mechanism to effectively track BotNets

VIII. CONCLUSION

The paper rightfully serves the purpose to disseminate knowledge about BotNets, their behavior pattern and growth. We have presented the study on the types of BotNets, vicious techniques used by Bot masters and the misuses of BotNets. Examples have been provided to demonstrate the extent of ravage that can be caused by BotNets. However, detection techniques are still emerging but contemporary BotNets have even evolved methods to evade them. This calls for more intensified research into the effectiveness of detection mechanisms. It is expected with the proliferation of computer science prowess that new state-of-the-art Bots will use non-IRC protocols, sophisticated Control & Command (C&C) and will utilize de-centralized P2P communication. It has been concluded that there is a need for developing more advanced powerful Honeypots that can combat growth of BotNets in different emerging protocols and prevent this cyber threat from spreading as less as possible. Thus, more research is needed to delve deeply into these issues as threats continue to rise and attackers adopt new methodologies.

IX. ACKNOWLEDGEMENT

Authors are thankful to the National Digital Certification Agency, Tunisia, Honeynet Research Alliance, Jim Jones, FedCIRC and Perkka Pietikainen & Lari Huttunen of NBI Finalnd IT Crime Unit for the valuable information received from their research work.

REFERENCES

[1] N. Ianelli, A. Hackworth, "BotNets as a Vehicle for Online Crime", CERT Coordination Center, Tech. Rep., December, 2005.
[2] C. Kalt, "Internet Relay Chat: Client Protocol," RFC 2812, April 2000.
[3] S.E. Schechter and M.D. Smith, "Access for Sale," Proc. 2003 ACM Workshop Rapid Malcode (WORM 03), ACM SIGSAC, 2003, pp.19-23
[4] E. Cooke, F. Jahanian, and D. McPherson, "The Zombie Roundup: Understanding, Detecting, and Disrupting Botnets," Usenix Workshop on Steps to Reducing Unwanted Traffic on the Internet (SRUTI) July 2005.
[5] K. Legelis, Symantec Corporation, "Combating online fraud: An update," http://information-integrity.com/article.cfm?articleid=100, 2005
[6] A. Householder and R. Danyliw, CERT Advisory CA-2003-08 Increased Activity Targeting Windows Shares, 2003 .
[7] I. Clarke, O. Sandberg, B. Wiley, and T.W. Hong, "Freenet: A Distributed Anonymous Information Storage and Retrieval System," Proc. Designing Privacy Enhancing Technologies, Int'l Workshop on Design Issues in Anonymity and Unobservability, 2001.
[8] V. Scarlata, B.N. Levine, and C. Shields, "Responder Anonymity and Anonymous Peer-to-Peer File Sharing," Proc. Ninth Int'l Conf. Network Protocols, pp. 272-280, Nov. 2001.
[9] http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1459.html
[10] P. Baecher, M. Koetter, T. Holz, M. Dornseif, F. Freilings, "The Nepenthes Platform: An Efficient Approach to Collect Malware," 9th Int'l Symp. On Recent Advances In Intrusion Detection, Germany, September 2006.
[11] A. Schonewille, D-J van Helmond, "The Domain Name Service as an IDS: How DNS can be used for detecting and monitoring badware in a net- work," Research report, University of Amsterdam, February 5, 2006
[12] F. Weimer, "Passive DNS Replication," Annual FIRST Conference Singapore, April 2005.
[13] Cisco IOS NetFlow Version 9 Flow-Record Format. Available at: http://www.cisco.com/warp/public/cc/pd/iosw/prodlit/tflow_wp.htm#wp1002063.
[14] J. Fan, J. Xu, M. H. Ammar, S. B. Moon, "Prefix-Preserving IP Address Anonymization", Computer Networks, Elsevier, Vol. 46, Issue 2, October 2004, pp. 253-272.
[15] Y. Li, A. Slagell, K. Luo, and W. Yurcik, "CANINE: A Combined Converter and Anonymizer Tool for NetFlows for Security," Int'l Conf. on Telecommunication Systems – Modeling and Analysis (ICTSM), Texas, November 2005.