Proc. of the Second Intl. Conf. on Advances in Electronics, Electrical and Computer Engineering -- EEC 2013
Copyright © Institute of Research Engineers and Doctors. All rights reserved.
ISBN: 978-981-07-6935-2

Botnet Detection, Measurement and Analysis: Research Challenges

Meenakshi Thapliyal, Neha Garg, Divya Bhatt, Shivani Pant

***Abstract*— At a present time lots of compromised computers also known as botnets, represents a very serious threat to Internet security. Botmaster have developed the ability of controlling large network of infected hosts, characterized by complex executable command set. This paper we have brief review of the recently approach related botnet detection, measurement and analysis. Now we will propose common process model for botnet detection and also concluded by presenting the identified research challenges in the botnet defense.**

***Keywords*—Botnet, Detection, Measurement, Analysis.**

I. Introduction

In the past, Internet Relay Chat that is a text-based chat-system, responsible for botnets origin which communicated by botmaster simultaneously. Eggdrop (1993) was the first known IRC bot. After some time, denial of service and then distributed denial of service (DDoS) attacks were implemented in these bots. Bots makes lethal infection like worms, invisible as viruses and a major threat to the Internet. A few examples of these bots are AgoBot and SDBot. The current generation of bots can spread through file sharing peer-to-peer (P2P) networks, email attachments and infected websites, or may be previously installed in backdoors. IRC, HTTP, P2P and hybrid are the main communicated protocols between bots [1].

The traditional common control infrastructures were mostly based on IRC which designed for huge area communication through specific channel. IRC-based botnets are essentially flexible and scalable. HTTP server for commands distribution. Both IRC-based and HTTP-based C&C performed as a centralized mechanism under the control of the botmaster[2]. These may be used for analyzing direct feedback, botnet status and significant properties like number of active bots. P2P Botnets are advanced botnet which utilize peer-to-peer C&C structure. P2P Botnets possess a lack of central point, consequently more discoverable and robust than centralized botnet, and bots are connected to each other acting as both C&C server and client[3].

Further, a botnet can be understood by its life-cycle [4], how a botnet is created and infects systems across infrastructure. There are ***five*** main phases of botnet life cycle. In the **initial infection phase**, the attacker, scans a target subnet for a known vulnerability, and infects victim machines through different exploitation methods. After initial infection, the infected hosts' written script, known as shell-code, is executed in **secondary phase**. In **connection phase**, the bot program establishes a command and control (C&C) channel and connects the zombie to the C&C server. In the next **command & control phase** the actual botnet C&C activities are started. Zombie is a part of botnet during the setup of control channel. The botmaster uses the C&C channel to give out commands to his bot army. Botmaster sends bot programs for receiving and executing commands. The C&C channel is used for sending the command and bots are commanded to download the updated binary.

Attack phase …

Last phase is **maintenance & update**, bot controller may update the bot binary to hide the detection of bot or may be for some changes in the bot army. Moreover, sometimes the updated binary moves bots to a different C&C server. This process is called server migration and it is very useful for botmaster to keep their botnet alive.

Botnet detection is very complicated and time consuming process. Botnet size is peculiar characteristics which help to evaluate the botnet threat. In gene*ral*, botnet measurement which deals counting of bots and botnet analysis are defined a botnet behaviour.

The rest of paper is organized into the sections. Section 2 research related studies, common process model of botnet detection which is required by mitigation of botnet is detail in Section 3, botnet measurement techniques, Section 4, brief summary of botnet analysis techniques, some research challenges in Section 5, at last, we discuss conclusion and future directions in Section 6.

II. Botnet Detection

Botnets have recently gained high interest by the scientific area, media and industry. In recent year there have been many approaches to detect botnet.

Al-hamaadi et al. [5] proposed a correlation algorithm to detect bots on the system by correlating their behavioural activities. Correlation of different activities can enhance the detection mechanisms and reduce the false alarms. Correlation algorithm is used because P2P bots are difficult to detect as there is no central point of communications. In addition, analyzing network traffic looking for signatures can be tedious task because bots signatures can be dynamic and encrypted.

Meenakshi Thapliyal, Neha Garg
Graphic Era University, Dehradun, India

Divya Bhatt, Shivani Pant
Graphic Era University, Dehradun
India

Different peer-to-peer bots such as Sinit and Nugache to examine the behaviour of these bots. In their analysis, some peer-to-peer bots communicate on a fixed port. They argue that by monitoring traffic on that port, one could detect these bots. They also discover that some of these bots generate a large number of destination unreachable error messages (DU) and connection reset error messages while trying to connect to other peers. In addition, some bot's communications are encrypted which make the traffic analysis a difficult task and resulting in high false alarms.

A. *Host based Detection*

Yuanyuan Zeng et al. [6] proposed a C&C independent framework that combines with both host-and network-level information. Network flow analyzer searches for trigger action traffic patterns among different hosts without accessing the packets' payloads, and clusters. Host analyzer then obtains suspicion-level information along with a few network statistics on a host-by-host basis for verification. Finally, Correlation engine detects each and every host by taking into account both suspicion level and clustering results. The network analyzer can be effective in forming suspicious clusters of aggressive bots but may fail to separate benign hosts from bot infected hosts if the latter are stealthy at the network level. When the stealthy bots are present, it is the host analyzer that provides correct detection results by generating distinguishing suspicion levels. By using combined host-and network-level information, it is able to detect different kinds of botnets with slow false-positive and negative rates.

Ping wang et al. [7] proposed the design of an advanced hybrid peer-to-peer botnet. Compared with current botnets, the proposed botnet is tough to be shut down, monitored, and captured. It provides robust network connectivity, individualized encryption and control traffic dispersion, limited botnet exposure by each bot, and easy monitoring and recovery by its botmaster. Therefore, invest more research into determining how to deploy honeypots efficiently and avoid their exposure to botnets and botmasters. Honeypot is an effective way to trap and spy on malware and malicious activities. Because compromised machines in a botnet need to cooperate and work together, it is particularly effective to use honeypot techniques in botnet spying.

B. *Network based Detection*

Saad et al. [8] proposed the characterization of network traffic behaviour to detect P2P botnet command and control phase under the radar through malicious email, websites, file sharing networks, ad hoc wireless networks. They discussed the main requirements of an online botnet detection framework and investigated the power of different machine learning (ML) techniques that are commonly used in the literature in addressing these requirements. Although the performance of these techniques was promising, none of these techniques can satisfy all the requirements of an online botnet detection framework. This underscores the need to investigate new ML technique or an hybrid of existing ones that can satisfy all the requirements of online botnet detection.

Wen-hwao et al. [9] proposed a P2P botnet detection method relying on monitoring traffic at the gateway and using data mining technology to analyze network behaviour. The following objects were achieved:

1. Using Anomaly detection: regardless of the packets were encrypted or not;
2. Achieved higher distribution in practice without installing any software in computer or changing any network or routing architecture;
3. Made pre-warning instead of post review possible only for the network behaviour in P2P botnet without any attack (syn flood and port scan) as exploration parameters
4. P2P botnet flow was found among various mixed flows in the same computer without being disturbed.

On the other hand, the research was conducted only for LAN environment with more distributed structure from the internet. Therefore, it was necessary to be distributed in ISPs if we want to stop P2P botnet massively and effectively, indicating that the NAT technology would be generally used in internet causing all the flows integrating as the same IP, which eventually made the judgment of P2P botnet flow more difficult since the characters were extremely diluted.

C. *Common Process Botnet Detection model*

We suggest a common process model for Botnet detection based on general steps: Netflow Capture, Flow Correlation which comprised the basic steps like Filtering traffic, Packet sample, Packet Assembler than Classifier and Clustering.

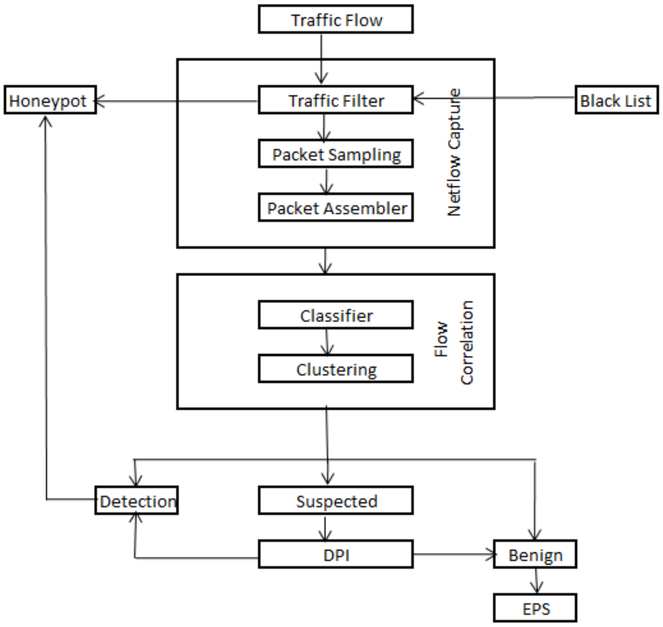

Fig1. Common Process model of Botnet Detection

Based on classification result the traffic is detected malicious of benign. Now Fig1.Shows the steps of common detection model:

1) Net*flow Capture*

- The incoming traffic based on either the source or scanning like behaviour. The traffic black listed DNS (Domain) source is directed to honeypots. The traffic is resulting from the port scanning attacks also directed to honeyport.
- After that, the adaptive packet sampling algorithm can be used with dynamic sampling interval to reduce the capture traffic rate.
- In next step packet assembler will assemble the samples the packet.

2) *Flow Correlation*

- In the flow correlation phase the classifier can be used to segregate the traffic ports, protocols and applications.
- Further classified traffic is clustered which is further analyzed to detect the benign, suspected and malicious traffic.
- Malicious traffic is directed to honeyports and suspected traffic is directed to DPI (Deep Packet inspection), Phase of the detection model which identified the traffic is malicious and benign.
- Malicious traffic is again directed to honeyport and benign traffic forward the EPS (Enterprises Production Server).

III. Botnet Measurement

There are two main techniques of measurement the botnet.

A. *Active Measurement*

This method can directly get the topology of the network and precise result. This technique is used to provide in-depth analysis of several facets of botnets, including inferring their membership by directly counting the bots observed on individual C&C channels [10-12].

- **Infiltration and Redirection**

Botnet infiltration provides information related identities of all active bots. Similarly, the botnet's live population is measured by counting the number of bots simultaneously present on the channel at a particular time[11]. The tracker cleverly mimics the behaviour of actual bots and joins a number of botnets, all the while recording any information observed on the command and control channel [13, 14].

- **DNS Redirection**

An alternative technique for counting infected bots by manipulating the DNS entry associated with a botnet's IRC server and redirecting connections to a local sinkhole. First, this technique can only measure the botnet's footprint. Second, as the sinkhole does not host an actual IRC server, there is no way of knowing if the bots are connecting to the same C&C channel. Finally, it is conceivable that botmasters can detect DNS redirection and subsequently redirect their bots to another IRC server thus distorting the estimate provided by this technique.

- **DNS cache snooping**

This method based on the caching property implemented. A DNS server is asked for a domain for storing the recorded data. It estimates a botnet's DNS footprint by probing the caches of a large collection of DNS servers and recording all cache hits. A cache hit implies that at least one bot has queried its name server within the time to live (TTL) interval of the DNS entry corresponding to the botnet server. The total number of cache hits provides an indication of the botnet's DNS footprint [14].

B. *Passive Measurement*

In passive measurement, the monitors are fixed on the edge of the backbone, core routers, and bound of the ISP. The monitors can get node numbers, flow characters to measure botnet. This method does not require any prior knowledge [15, 16].

- **Packet Inspection**

A popular concept for increasing a network's security is to inspect the network data packets. The basic idea is to match various protocol fields, or the payload of a packet, against pre-defined patterns of abnormal or suspicious content.

- **Analysis of Flow Records**

It is a technique for tracing network traffic at an abstract level. Instead of inspecting individual packets, communication streams are considered in an aggregated form.

- **Analysis of Spam Records**

One common purpose of botnets is the distribution of unsolicited email, known as spam. A fairly indirect approach to measuring botnets and their corresponding activities is the analysis of spam records. In this context, indirect means that, instead of observing communication such as command and control messages, information is derived from the investigation of the spam messages sent by a botnet.

IV. Botnet Analysis

Botnet analysis techniques allow the analyst to quickly and in detail understand the risk and purpose of a bot. Analyst to understand the behaviour of a bot[17] and the opposing intention of botnet. As analysis tools and techniques become more involved, attackers come up with evasion techniques to prevent their botnet from being analyzed. Such techniques cover self modifying or dynamically generated code, as well as approaches that detect the presence of an instrumented analysis environment allowing a malicious behaviour. Firstly the process of analyzing a given program during execution is called dynamic analysis, while static analysis refers to all techniques that analyze a program by inspecting it.

Shangdong et al. [18] investigates the nature of botnet size, upon which four issues are introduced.

- The measurement of botnet live population, which is a problem of botnet detection in nature.
- The measurement of botnet footprints.
- Dynamic tracing of botnet size.
- Area issue of botnet size.

Live population can be obtained by network anomaly detections, but footprint contains offline bots which cannot be detected by network anomaly detections. They describe that local size is used to estimate first then this size is used to estimation of global size.

A. *Static Analysis Botnet Techniques*

Static analysis can be applied on different software program, without executing it. If the source code is available, static analysis can help finding memory corruption flaws, botnet activity and precise models.

Christodorescu et al. [19] Static analysis tools can also be used on the binary program. When compiling the source code of a program into a binary executable, the size of data structures or variables gets lost. Static analysis tools can be used to extract useful information of a program. Call graphs give an overview about functions code of program.

Sasaki et al. [20] Static analysis is able to calculate the possible values of parameters then this knowledge can be used for advanced defense mechanisms.

Problems : Sommer et al. [21] Generally, the source code of botnet samples is not readily available which makes problems to apply static analysis techniques for botnet analysis to those that recover the information from the binary Therefore, it is necessary to develop analysis techniques that are resilient to such modifications, and are able to reliably analyze malicious software.

B. *Dynamic Analysis Botnet Techniques*

Manuel *et al.* [22] Dynamic analysis applied on executed program with the different approaches.

- **Function Call Monitoring**

A function consists of code that performs a specific task, such as, calculating the factorial value of a number. Functions can be easily define, maintain, and commonly used to abstract from implementation details to a semantically richer representation.

- **Application Programming Interface**

Application programming interface (API) is a coherent set of function which available on different layers of abstraction such as manipulating files or communicating over the network. On Windows API refers to a set of APIs that provide access to different functional categories such as networking, security, system services, and management.

- **System Calls**

Software executing computer systems such as word processors or image manipulation programs are called user-mode. The operating system is executed in kernel-mode. Only code that is executed in kernel-mode has direct access to the system state. This separation prevents user-mode processes from interacting with the system and their environment directly. Using system calls, a user-mode application can request the operating system to perform a limited set of tasks on its behalf. Thus, to create a file, a user-mode application needs to invoke the specific system call indicating the file's path, name, and access method. Once the system call is invoked, the system is switched into kernel-mode (privileged operating system code is executed). Upon verification that the calling application has sufficient access rights for the desired action, the operating system carries out the task on behalf of the user-mode applications.

V. Research Challenges

There are many challenges which regularly faced by botnet research. This section discusses the current botnet detection, measurement and analysis challenges that need to be addressed.

1) *Botnet Detection*

Botnets have several characteristics (e.g. developed by skillful developers, dynamic nature, and high flexibility) that make them difficult to detect.

- **Small – Scale and Single Bot Detection:** Current botnet detection methods are designed based on analyzing the cooperative behaviour posed by bots from the same botnet. These techniques are more effective in large-scale botnet detection where there are high numbers of bots in a botnet. Hence the detection of small-scale botnets and single bots can still considered as a challenge.

- **Risk: False positives & negatives :** Botmaster use HTTP protocol to hide their activities among the normal web flows and easily avoid current detection and analysis methods like firewalls. Because of the wide range of HTTP services used, unlike the IRC and P2P, it is not easy to block this service .Moreover, this service is commonly used by normal applications and services in the Internet, thus, detection of the HTTP botnets with low rate of false alarms (e.g. false negative and false positive) has become a challenge.

2) *Botnet Measurement*

The botnet measurement is very challenging research area, especially in the peer to peer network.

- **Botnet Size:** The size of a botnet is an important metric to evaluate the threat posed by it. In general, it is hard to measure the real size of a botnet. Since attackers can complicate this information by modifying the C&C server. To obscure the size of the botnet, botherder can modify the C&C server to discard the responses of the users and commands. In addition, botherder can set the user mode of the bots to be invisible: the status messages of the bots are then not visible for all members of a channel. In such situations, the size of these botnets is hard to estimate.

- **Active Botnet Infiltration:** Goal-Rewrite C&C messages on either dialog side. Understand both sides of C&C protocol – message and Field structure. Access to one side dialogs only and handles encryption /obfuscation.

3) Botnet Analysis

- **Reverse Engineering:** Reverse engineering methods can only recover the format of plain text. One gap in recovering the format of encrypted message is how to recover the plain - text message from the cipher-text message.

- **Complexity:** The major challenges of binary analysis are binary code is complex. Binary analysis needs to the model this complexity accurately in order for the analysis to be accurate.

VI. Conclusion and Future work

This paper analyzes the research of botnet detection, measurement and analysis. The complexity of botnets continues to increase and suffocated. These networks can be harmful for constructive purpose, however currently they are primarily being used to penetrate computer related crime such as Spam, DDOS, TCP SYN Flood, UDP Flood. Previous and current botnet strategies are any indication, the problem is likely to get worse before it gets better. New methods for intrusion, infection, and avoidance are making it increasingly difficult to detect bots and track a attacker. So this paper intended to techniques on botnet detection, measurement and analysis based on bot behaviour. Common process botnet detection model helps to detect the bot and malicious activities.

As part of future work of the analysis presented in this paper, we may consider detecting botnet activity on a host, find the list of peers it connects to before it joins the P2P network and automate an attack. For this we would simulate a network on the OverSim simulator. Using the simulator , it is possible to define malicious behaviour and the probability of malicious nodes on the network

References

[1] Silva, S.S., Silva, R. M., Pinto,R.C., and R. M. Salles, "Botnets: A survey," *Computer Networks*, 2012.

[2] C. Mazzariello, "IRC traffic analysis for botnet detection," in *Fourth International Conference on Information Assurance and Security, ISIAS'08.* , 2008, pp. 318-323.

[3] J. Dae-il, K. Minsoo, J. Hyun-chul, and N. Bong-Nam, "Analysis of HTTP2P botnet: case study waledac," in *9th Malaysia IEEE, International Conference on Communications (MICC), 2009*, pp. 409-412.

[4] M. Feily, A. Shahrestani, and S. Ramadass, "A Survey of Botnet and Botnet Detection," in *Third International Conference on Emerging Security Information, Systems and Technologies, 2009. SECURWARE '09.*, 2009, pp. 268-273.

[5] Y. Al-Hammadi and U. Aickelin, "Behavioural Correlation for Detecting P2P Bots," in *Second International Conference on Future Networks, ICFN '10.* , 2010, pp. 323-327.

[6] Y. Zeng, X. Hu, and K. G. Shin, "Detection of botnets using combined host-and network-level information," in *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2010* pp. 291-300.

[7] W. Ping, S. Sparks, and C. C. Zou, "An Advanced Hybrid Peer-to-Peer Botnet," *IEEE Transactions on Dependable and Secure,* vol. 7, pp. 113-127, 2010.

[8] S. Saad, I. Traore, A. Ghorbani, B. Sayed, D. Zhao, L. Wei, J. Felix, and P. Hakimian, "Detecting P2P botnets through network behaviour analysis and machine learning," in *Ninth Annual International Conference on Privacy, Security and Trust (PST), 2011* pp. 174-180.

[9] L. Wen-Hwa and C. Chia-Ching, "Peer to Peer Botnet Detection Using Data Mining Scheme," in *International Conference onInternet Technology and Applications,* , 2010, pp. 1-4.

[10] W. Binbin, L.Zhitang, T. Hao, H. Zhengbing, and H. Jun, "Actively Measuring Bots in Peer-to-Peer Networks,"in *International Conference on Networks Security, Wireless Communications and Trusted Computing, 2009. NSWCTC '09.* 2009, pp. 603-607.

[11] M. S. T. Holz, F. Dahl, E. Biersack, and F. Freiling, ""Measurements and mitigation of peer-to-peer-based botnets: a case study on storm worm,," in Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats, Ed., 2008, pp. 1-9.

[12] K. Brent Byung Hoon, C.-T. Eric, P. L. Christopher, T. James, K. Hun Jeong, N. Chris, W. Zachariah, S. Greg, H. Nicholas, D. David, and K. Yongdae, "Towards complete node enumeration in a peer-to-peer botnet," in *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security* Sydney, Australia: ACM, 2009.

[13] M. A. R. J. Z. Fabian and M. A. Terzis, "My botnet is bigger than yours (maybe, better than yours): why size estimates remain challenging," in *Proceedings of the 1st USENIX Workshop on Hot Topics in Understanding Botnets, Cambridge, USA*, 2007.

[14] B. B. H. Kang, E. Chan-Tin, C. P. Lee, J. Tyra, H. J. Kang, C. Nunnery, Z. Wadler, G. Sinclair, N. Hopper, and D. Dagon, "Towards complete node enumeration in a peer-to-peer botnet," in *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*, 2009, pp. 23-34.

[15] I. Castle and E. Buckley, "The Automatic Discovery, Identification and Measurement of Botnets," in *Second International Conference on Emerging Security Information, Systems and Technologies, 2008. SECURWARE '08.* 2008, pp. 127-132.

[16] M. A. R. J. Z. Fabian and M. A. Terzis, "A multifaceted approach to understanding the botnet phenomenon," in *Proceedings of the ACM SIGCOMM Internet Measurement Conference (IMC)*, 2006.

[17] H. Binsalleeh, T. Ormerod, A. Boukhtouta, P. Sinha, A. Youssef, M. Debbabi, and L. Wang, "On the analysis of the Zeus botnet crimeware toolkit," in *Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on*, 2010, pp. 31-38.

[18] L. Shangdong, G. Jian, Y. Wang, and A. Jakalan, "A Survey of Botnet Size Measurement," in *Second International Conference on Networking and Distributed Computing (ICNDC)*, 2011, pp. 36-40.

[19] M. Christodorescu and S. Jha,"Static analysis of executables to detect malicious patterns," DTIC Document 2006.

[20] R. Sasaki, S. Qing, E. Okamoto, H. Yoshiura, P. Akritidis, E. P. Markatos, M. Polychronakis, and K. Anagnostakis, "STRIDE: Polymorphic Sled Detection Through Instruction Sequence Analysis," in *Security and Privacy in the Age of Ubiquitous Computing*. vol. 181: Springer US, 2005, pp. 375-391.

[21] R. Sommer, D. Balzarotti, G. Maier, M. Lindorfer, C. Kolbitsch, and P. Milani Comparetti, "Detecting Environment-Sensitive Malware," in *Recent Advances in Intrusion Detection*. vol. 6961: Springer Berlin Heidelberg, 2011, pp. 338-357.

[22] E. Manuel, S. Theodoor, K. Engin, and K. Christopher, "A survey on automated dynamic malware-analysis techniques and tools," *ACM Comput. Surv.*, vol. 44, pp. 1-42, 2008.