Botnet Identification & Challenges

Manikannan D.[1] Shyam Shankar U.[2] Siddharth V.[3]
[1]Assistant Professor [2,3]UG Scholars
[1,2,3]Department of Computer Science & Engineering
[1,2,3]SRM Institute of Science and Technology, Chennai, Tamil Nadu, India

Abstract— Serious problems are becoming more transparent in this digital world giving rise to various threats. Botnets are the imminent among the various threats in cyber security. They are group of compromised nodes, in other words bots which are remotely commanded by its controller (Bot Master). This scheme is a typical Command-and-Control (C&C) infrastructure. Various Internet attacks, including spam, distributed denial-of-service (DDoS), phishing, malware dissemination and identity theft are facilitated through Botnets. This paper provides an overview of Botnets and latest advances in Botnet detection research. It classifies Botnet detection techniques into two approaches. One approach is based on setting up Honey nets and another approach is based on Intrusion Detection System (IDS). Peer to Peer Botnet consists of only two nodes and they are harder to detect than the normal Botnets. This paper also presents various remedies to the Botnet threats and also paves way for future directions for Botnet detection research.

Key words: Botnet, C&C Channels, Intrusion Detection, Cyber Security

I. INTRODUCTION

The current evolution of information and communications technology, particularly the internet, subscribes to the improvement of the quality of everyday life, and is therefore considered as a vital infrastructure. But, it is also true that this wide development also provides negative impact on society. The continuous increase in threats and incidents related to cyber security has revealed to be very serious issue in this digital world. Botnets, stand out to be the most emerging threats against cyber security. It is a type of malware which is regarded as new and is installed into a compromised computer that can be remotely controlled by Bot Master for the execution of orders through the received commands [1]. A Botnet is a number of internet-connected devices (bots), which are controlled by attackers remotely (bot Masters) to setup various network attacks such as malware dissemination, distributed denial of service (DDOS), phishing, and click fraud [2]. The concept of Botnets is quite different from other forms of malware as they use command and control (C&C) channels for their communication. The key purpose of these channels is to disseminate the botmasters' commands to their bot armies. These channels can operate over various network topologies and use different communication mechanisms, from established Internet protocols to more recent P2P protocols. However in today's world, majority of Botnets make use of the Internet Relay Chat (IRC) protocol which was originally designed to form large social chat rooms [3].

II. FEATURES & CLASSIFICATION OF BOTNETS

Bots are self-propagating applications that infects its host like viruses and worms do. Even so, compared to other malware classes, the typical characteristic of Botnet is the use of Command & Control channels (C&C) through which they can be updated and instructed. Features like high transmission rate, easy availability, low user awareness and monitoring, and distant location are generally used by bot masters to identify their victims [4]. A typical function that bots give for their masters includes the automated extraction of a victim's credentials, the organized distribution of spam, ability to participate in denial of service attacks, or the extension of the Botnet by recruiting new bots [5].

According to N.S. Raghava [8], the lifecycle of Botnet can be divided into three phases:

1) Searching

Searching for vulnerable and unprotected computers.

2) Distributing

The bot code is distributed to the computers (targets), so that the targets become bots.

3) Sign-on

The bots connect to botmaster and become ready to receive command and control traffic.

B. Components of Botnet

A similarity can still be found in the structures of the components of Botnets, although it has been found that there are different structures associated with Botnets. The Command & Control infrastructure is the most vital component of a Botnet. It comprises of bots and an entity that is responsible for controlling these bots. More than one communication protocols are used by the bot masters in order to command the target systems and to control and co-ordinate their actions. It is observed that the only way to control bots is by C&C infrastructure within the Botnet. Within this infrastructure, the bots are expected to maintain a stable connection in order to operate efficiently. Thus, from the architecture of the C&C infrastructure, stability, robustness and reaction time are determined. Botnet structures can be centralized or decentralized [5].

1) Centralized Botnet

In this type of structure, all the bots act as clients and connect to the centralized servers. The bots receive commands from the server. The Botmaster possesses the power to send commands to all the bots. This is primarily due to the fact that all the bots are connected to the servers. This proves to be a better way of communication since it maintains low message latency. A command and control architecture of centralized Botnets is shown in Figure-1.

The origin of Botnets is mainly attributed to the Internet Relay Chat (IRC), a text-based chat-system that enables channel communications. The IRC protocol continues to serve as a pioneer technology for the control and provides a centralized communication model. The significant feature of this protocol is due to the fact that there is no limitation for the number of potential participants within one channel. This further permits the collection of many bots in

one channel and also provides the ability to command them in parallel. In addition, it is possible to have private conversations on a one-to-one basis. Single bots can be directly manipulated using this. It is easy to modify and implement as the IRC protocol is text-based.

The communication with bots by C&C servers is established through a prominent protocol called Hypertext Transfer Protocol (HTTP). This HTTP protocol is the most commonly used for the data transmission over the Internet. This includes websites and images, and also machine-readable binary data transported in downloads and uploads. Due to these main features, HTTP is found readily in almost every network connected to the Internet. This is particularly interesting and useful for cyber criminals who operate Botnets because the command and control mechanism of the Botnets is achieved through this protocol. HTTP bots are determined to issue requests to the target C&C server in a periodic fashion. Usually these requests contain information about the status of the Botnet, the underlying basis using which the server decides which commands are transferred to this particular bot [7].

As it is obvious that C&C server helps in performing every connection and action, it therefore proves to be the vulnerable and critical point. If somehow someone is able to control and shut down the C&C server, then the whole Botnet would become incapable.

2) Decentralized Botnet

Unlike the centralised Botnet model, this model allows the bots to act autonomously. Bots have the ability to connect to several infected nodes in a Botnet rather than to a C&C center. These links are useful for communicating with other bots within Botnets. A decentralised Botnet model with a command and control architecture is depicted in Figure-2. This type of Botnet is also referred as Peer-to-Peer Botnet (P2P Botnet), as this is the name of the corresponding network model. It is much harder to recover from P2P Botner. Also P2P Botnet is not easily manageable, because the command transfer rate is slower than that of a centralised Botnet's. Since a single point of failure in P2P Botnet does not cause noteworthy disruptions, this class of Botnets are very difficult for ethical experts and defenders to track.

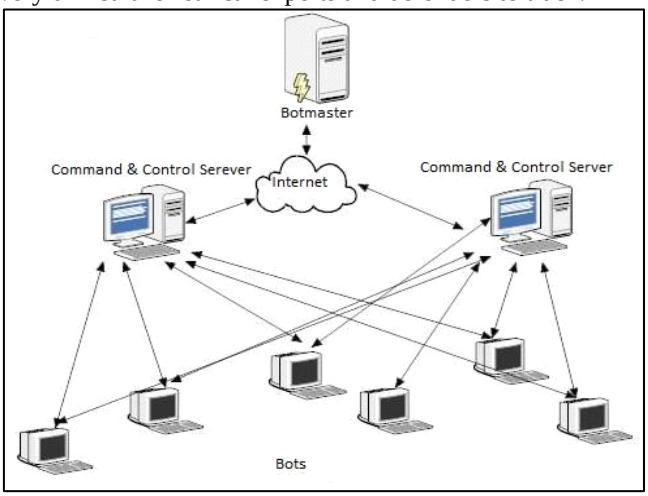

Fig. 1: C&C Architecture of a Centralized Botnet Model [31].

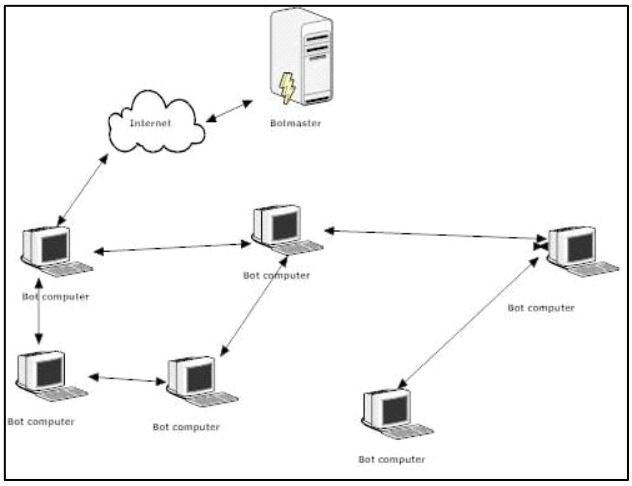

Fig. 2: C&C Architecture of a Decentralized Botnet Model [31].

III. BOTNET IDENTIFICATION

Botnet identification simply means the detection of bots (infected machines) in a network. The characteristics of bots and their properties and features are to be studied effectively for a Botnet detection to be successful. According to [14], common characteristics of a bot are associated to usual activities in any network as some basic communication with the C&C servers is required by the bots. To detect the presence of Botnets one could observe some common activities such as making anti-virus software useless by disabling it, establishing a number of unwanted network connections, downloading and executing certain applications, creating new processes that consume memory to slow down the processing power of CPU, and so on. Botnet detection is viable using two approaches. The first approach is based on employing Honeynets in the network. And the second approach is Intrusion Detection System (IDS).

A honeynet is nothing but a mere network or environment that contains purposefully introduced vulnerabilities to monitor intrusions and attacks. Honeynets serve useful to get binaries of bots and penetrate those Botnets [13]. The Honeynet project [15] stations an infrastructure that comprises a honeypot and the honeywall network. J.S. Bhatia et al. [19] proposed mechanisms to deploy a Virtual Honeynet for gathering of packets to detect HTTP and IRC based Botnet Command signatures. But, honeynets are limited to monitoring and capturing activities that contact them directly. Apparently, attacks against other systems are not captured by them. IDS Botnet detection techniques can be either an anomaly-based or signature-based technique. A signature-based Botnet detection technique utilizes the signatures of present Botnets for its purpose of detection. This method has several advantages, such as very low false alarm rate, easy implementation, faster rate of detection, and finer insight about the detected attack's type. S. Behall et al. [10] Proposed a signature-based N-EDPS which weighs only outgoing packet traffic to detect Botnet based infectious packet traffic using several open source and free software. They expressed that their proposed framework is superior to Network-based Intrusion detection & Prevention system (N-IDPS) because a smaller signature database is sufficient for N-EDPS as opposed to N-IDPS and thus

provides improved results. Unfortunately, the proposed N-EDPS is not capable of detecting C&C channels that possess the ability to encrypt data, if they exist. The nicknames within one Botnet channel may be a combination of random letters and numbers, but ultimately they must contain the same structure. So this similarity is used to detect IRC-based Botnets [11], [16]. Signature based detection method is successful only in the case of well-known Botnets. Thus, it is not possible to detect new and unfamiliar Botnets using this method. However, Anomaly-based detection techniques are capable of solving this problem.

The basis behind anomaly-based detection technique is to detect Botnets by taking into account many different network traffic anomalies (oddities), including high traffic volume, traffic on unusual ports, high network latency and abnormal system behaviour that could signal the presence of malicious bots in the network. Anomaly-based detection techniques are further classified into two techniques namely host-based detection and network-based detection.

A host-based technique is a detection technique which monitors and examines the computer system instead of network traffic. A previous study [12] presents a cross-breed model for bot detection, which bonds an operating system event log analyzer and a host intrusion detection system. The proposed system, Model of Multi-Agent Bots Detection System (MABDS) consists of the administrative agent, user agent, a central knowledge base, and groups of agents (network analysis, system analysis, and honey nets).

A network-based technique is a detection technique which attempts to detect Botnets by examining traffic at network level. Network-based techniques can further be categorized: Active monitoring and passive monitoring.

The active monitoring is based on the potential to inject test packets into the network databases and servers for determining the possible responses of the network. Thus, it has the probability to assemble extra data traffic on the network. Gu et al. [21] put forward BotProbe which perkily joins mainly to dynamically inject data packets that probe the internal client to find out whether a human or bot is managing that session. A cause-effect correlation seems suitable to compare its working because for a big portion of Botnet C&C channels, a typical command-and-control communication has a deterministic command response sequence. This approach unveils effectiveness on IRC-based Botnet detection in real world. Passive monitoring techniques analyze traffic in the network in which they are employed and search for sceptical interactions by bots or C&C servers. They do not expand the traffic on the network for observation.

Binkley and Singh [20] proposed an anomaly-based Botnet detection algorithm for detecting IRC-based Botnets. The algorithm aggregates two main factors for its working which are, an IRC mesh detection component and a TCP scan detection heuristic called the TCP work weight.

An IRC-based three layer architecture has been developed by Strayer et al. [25] that first eliminates traffic that is not part of a Botnet (Filter), classifies the remaining traffic into a group that is likely to be part of a Botnet (Classifier), then correlates or corresponds the likely traffic to determine common communication sequences that would evince the presence of a Botnet (Correlator).

In order to interact with the C&C server, bots initiate a number of DNS queries to identify the particular server that a DDNS (Dynamic DNS) provider hosts. Thus, it is quite possible to create a detection mechanism that observes DNS traffic and searches for some anomalies in the DNS traffic. Botnets routinely use DNS to reunite infected hosts, begin attacks and modify their codes accordingly. Choi et al. [23] have proposed BotGAD (Botnet Group Activity Detector) based on group activity model and metric. They have defined an inborn property of Botnets, known as group activity. They have also developed metric model to evaluate the characteristic and detection mechanism that has the capability to detect Botnets of large scale networks in real-time.

M.M. Masud et al. [24] have proposed flow-based Botnet traffic detection by mining multiple log files. The proposed work utilizes log correlation for C&C traffic detection. They classify an entire flow to identify C&C Botnet traffic. This proposal can detect P2P based Botnets [24].

A tool for tracking, monitoring and analyzing spamming Botnets is BOTMAGNIFIER [9]. This tool is capable of magnifying an initial seed pool of spamming IP addresses by grasping the characteristics of known spamming bots and checking for a match of the obtained patterns against a log containing the email transactions in the Internet.

A. Karasaridis et al. [28] have propounded an algorithm for the detection and characterization of Botnets using flow data based passive analysis. The authors have stated that they have been able to identify several compromised nodes in networks over a period of few months with very low false positive rate. Also, their system is capable of detecting Botnets that use encrypted communications.

Zeidanloo et al. [1] have proposed a non-specific Botnet detection structure. The proposed detection structure examines the collection of nodes that exhibit similar communication pattern in one step and doing malicious activities in another step and try to locate common nodes in them. A better version of this structure has been proposed using Artificial Immune System (AIS) for detection of malicious activities among the bots [18].

A passive bot detection system which employs IDS dialog correspondence to map IDS events with bot infected models is BotHunter [27]. As BotHunter aims to locate bot-related behaviour at the network level, sneaky bots could evade detection by simply dodging event timing correlation or performing normal attacks with no correspondence to any network activities. In BotHunter, the infection cycle of Botnet is represented by viewing the following activities: binary download and execution, target lookup, C&C channel establishment, and outbound mail scanning. The procedure then detects Botnets using IDS-driven dialog interdependence as stated by the bot infection life cycle model.

Guofei Gu et al. [17] have proposed a Botnet detection framework, BotSniffer which has been designed specifically for detecting IRC and HTTP Botnets in a local area network. Botsniffer observes that the bots within the same Botnet possess strong resemblances in their activities and responses, that is, scanning and transmitting email spam, thus sharing common communication contents. It uses a detection technique called spatial-temporal correlation and

believes that all Botnets opt a highly synchronized communication fashion.

Botminer [26] is the latest technique which uses data mining approaches for C&C traffic detection of Botnet. It is an improvement of Botsniffer [17]. Botminer gathers malicious traffic and similar communication. After that, it performs cross-cluster correlation to determine hosts that share both malicious activity patterns and similar communication.

IV. REMEDIES

Protection against Botnets is possible by enactment of some mentioned strategies [22]. It is the responsibility of all Internet users for defence, starting from home computer users, system administrators, developers, up to ISPs. The issue of defence must be regarded as a permanent and complete process in which all the activities must aim to control the situation rather than just responding to it. This is the best way to achieve good results and to safeguard computers. There are two methods to deal with Botnets. The first method is associated with technical approaches while the second method follows social and regulatory approaches. The second method is not dealt in this paper as it is not in the scope of this study.

The majority of Botnet remedies rely on the C&C infrastructure of Botnets by separating Botnet-based traffic, sinking domains with the help of DNS registrars or shutting down malicious servers in data centers. The technical approaches include Port Blocking, Blacklisting, Reverse engineering and Packet Filtering.

Blacklists may provide a list of IP addresses of suspicious hosts or entire subnets exhibiting malicious activities. A single blacklist is sufficient to prevent all traffic from included addresses and also to sift websites that possess proven malicious contents [29]. The Spamhaus Project [30] offers several real-time lists that help in identifying malicious activities and finally blocking them. Domain Block List (DBL) and the Spamhaus Block List (SBL) possess a group of domain names and IP addresses respectively that provide information about which emails to be not accepted.

Packet filtering can be applied at a host, network and ISP level. A desktop firewall is a typical component that performs packet filtering at host level. The motive of a desktop firewall is to analyze the network activities of all active processes. As it is usually the case that the amount of traffic at host level is manageable, application of deep-packet inspection is quite possible. Frequently, user or administrator interaction is required to permit or restrict network access for some applications, if they are not bound by specific rules [7]. The process of retrieving the functionality of a program without the source code is known as reverse engineering. It helps in extorting the details of the installation and increasing of malware. The process involves static analysis and dynamic analysis. The binary is not executed in static analysis. This phase deals with the restoration of certain aspects of the functionality. The dynamic analysis focuses on the execution of the sample.

A preventive measure to reduce the amount of spam emails traversing a wide network like the network of an ISP is known as Port Blocking. It is the process of disabling certain ports of a machine, making it unable to interact with other machines connected to the network using those ports. The use of insincere services via port 25, like direct email exchange or access of open relay mail servers suggest that port 25 is worthy of being blocked to enhance safety of a network. It has been recommended as best practice to block port 25 at ISP level [7].

V. CONCLUSION

It is not surprising to realize the fact that Botnets prove to be an useful catalyst in the process of launching several Internet-related attacks. The diversity of Botnet protocols and frameworks renders Botnet detection a challenging task. A majority of the present Botnet detection techniques work only on specific C&C communication protocols and structures. Accordingly, since Botnets change their C&C communication architecture, these detection methods will become unproductive and validate the difficulty in Botnet detection. A main challenge in Botnet detection is bots have become steadily more advanced, so evasion techniques have become much more powerful and capable of avoiding detection mechanisms. Another challenge for researchers is the problem of testing their schemes in a real scenario or validating their results using real data.

REFERENCES

[1] H. R. Zeidanloo and Azizah Bt Abdul Manaf, "Botnet Detection by Monitoring Similar Communication Patterns" ,In International Journal of Computer Science and Information Security, Vol. 7, No. 3, pp. 36 - 45, 2010.
[2] Lei Zhang, Shui Yu, Di Wu, and Paul Watters, " A Survey on Latest Botnet Attack and Defense", In proceeding of International Joint Conference of IEEE TrustCom'11, pp. 53 - 60 , 2011.
[3] Moheeb Abu Rajab, Jay Zarfoss, Fabian Monrose, and Andreas Terzis, "A Multifaceted Approach to Understanding the Botnet Phenomenon", In IMC '06 Proceedings of the 6th ACM SIGCOMM conference on Internet measurement, pp. 41 - 52, 2006.
[4] Saha and A, Gairola, "Botnet: An overview", CERT-International Journal of Network Security & Its Applications (IJNSA), Vol.3, No.5, pp. 177 -189, 2011.
[5] A.Upadhyaya, D.Jayaswal, and S. Yadav, "Botnet: A New Network Terminology", IEEE International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), pp. 424 –428, 2011.
[6] Arshad Hussain, "Botnet Tracking and Intrusion Detection", Eastern Michigan University, 2011.
[7] D. Plohmann, E. Gerhards-Padilla, and F. Leder, "Botnets: Detection, Measurement, Disinfection & Defence", European Network and Information Security Agency (ENISA), 2011.
[8] N.S. Raghava, D. Sahgal, and Seema Chandna, "Classification of Botnet Detection Based on Botnet Architecture", IEEE International Conference on Communication Systems and Network Technologies (CSNT), pp. 569 - 572, 2012.

[9] G. Stringhini, T, Holz, B. Stone-Gross, C Kruegel, and G. Vigna, "BOTMAGNIFIER: Locating Spambots on the Internet", Proceedings of the 20th USENIX conference on Security '11, 2011.
[10] S. Behal1, A. S. Brar, and K. Kumar, "Signature-based Botnet Detection and Prevention", ISCET, pp.122-127, 2010.
[11] W, Wang,B.fang, Z. Zhang, and C. Li, "A Novel Approach to Detect IRC-based Botnets", In IEEE International Conference on Networks Security, Wireless Communications and Trusted Computing, pp. 408 - 411 , 2009.
[12] Miroslaw Szymczyk, "Detecting Botnets in Computer Networks Using Multi-Agent Technology", In IEEE proceeding on 2009 Fourth International Conference on Dependability of Computer Systems, pp. 192 –201, 2009.
[13] F. Pouget, M. Dacier, "Honeypot-based Forensics", Asia Pacific Information technology Security Conference, 2004.
[14] Naveen Davis, "Botnet detection using correlated anomalies", Thesis at Technical University of Denmark, 2012.
[15] http://www.honeynet.org/papers/bots/
[16] Jan Goebel, and Thorsten Holz, "Rishi: Identify Bot Contaminated Hosts by IRC Nickname Evaluation", In Proceedings of USENIX Workshop on Hot Topics in Understanding Botnets (HotBots), 2007.
[17] Guofei Gu, Junjie Zhang, and Wenke Lee, "BotSniffer: Detecting Botnet Command and Control Channels in Network Traffic", Georgia Institute of Technology, 2008.
[18] H.R. Zeidanloo, F. Hosseinpour, and P. N. Borazjani, "Botnet Detection Based on Common Network Behaviors by Utilizing Artificial Immune System(AIS)", IEEE 2nd International conference on Software Technology and Engineering (ICSTE), volume 1, pp. 21-25, 2010.
[19] J.S.Bhatia, R.K.Sehgal, and Sanjeev Kumar, "Botnet: Steps to Reducing Unwanted Traffic on the Internet", 2nd conference on Software Technology and Engineering (SRUTI'06), San Jose, CA, July 2006.
[20] J.R. Binkley and S. Singh, "An Algorithm for Anomaly-based Botnet Detection", In Proceedings of
[21] G. Gu, V. Yegneswaran, P. Porras, J.Stoll, and W.Lee, "Active Botnet Probing to Identify ObscureCommand and Control Channels", ACSAC '09 Proceedings of the 2009 Annual Computer SecurityApplications Conference, pp. 241 - 253 , 2009.
[22] S. Stanković, and Dejan Simić, "Defense Strategies against Modern Botnets", in proceedings of International Journal of Computer Science and Information Security, Vol. 2, No. 1, 2009.
[23] H. Choi, H. Lee, and H Kim, "BotGAD: Detecting Botnets by Capturing Group Activities in Network Traffic", ACM Proceedings of the Fourth International ICST Conference on Communication System software and middleware, 2009.
[24] M.M. Masud, T.Al-khateeb, L. Khan, B.Thuraisingham, and K. W. Hamlen, "Flow-based Identification of Botnet Traffic by Mining Multiple Log Files", proceedings of the first International conference on Distributed Framework and Applications- DFmA 2008, pp. 200 - 206 , 2008.
[25] W.T. Strayer, R. Walsh, C. Livadas, and D. Lapsley, "Detecting Botnets with Tight Command and Control", Proceedings of 31st IEEE Conference on Local Computer Networks, pp. 195 - 202, 2006.
[26] G. Gu, R. Perdisci, J. Zhang, and W. Lee, "BotMiner: Clustering Analysis of Network Traffic for Protocol- and Structure-Independent Botnet Detection", In SS'08 Proceedings of the 17th conference on Security symposium, 2008.
[27] G. Gu, P. Porras, V. Yegneswaran, M.Fong, W.Lee. "BotHunter: Detecting Malware Infection through IDS-Driven Dialog Correlation", USENIX Security '07, 2007.
[28] A .Karasaridis, B. Rexroad, D. Hoeflin. "Wide-scale Botnet Detection and Characterization", In Proceedings of the first conference on First Workshop on Hot Topics in Understanding Botnets, 2007.
[29] Jian Zhang, Phillip Porras and Johannes Ullrich, "Highly Predictive Blacklisting", SS'08 Proceedings of the 17th conference on Security symposium, pp. 107-122, 2008.
[30] http://www.spamhaus.org/
[31] http://www.hotforsecurity.com

