Review on Botnet Threat Detection in P2P

Mohini N. Umale#, Prof. A. B. Deshmukh*, Prof. M. D. Tambakhe*

#Computer Science and Engineering Department, Sipna COET SGBAU, Amravati, India, *mohiniumale30@gmail.com*
*Associate Professor Computer Science and Engineering Department, Sipna COET SGBAU, Amravati, India, *abd_07@rediffmail.com*
*Associate Professor Computer Science and Engineering Department, Sipna COET SGBAU, Amravati, India, *manoj20t@gmail.com*

Abstract— Botnets are nothing but the malicious codes such as viruses which are used for attacking the computers. These are act as threats and are very harmful. Due to distributed nature of botnets, it is hard to detect them in peer-to-peer networks. So we require the smarter technique to detect such threats. The automatic detection of botnet traffic is of high importance for service providers and large campus network monitoring. This paper gives the review on the various techniques used to detect such botnets.

Keywords— *Botnets, detection, peer-to-peer.*

*****

I. Introduction

Malware is an especially serious threat to fashionable networks. In recent years, a brand new type of general malware called bots has arisen. Bots area unit distinctive in this they conjointly maintain communication structures across nodes to resiliently distribute commands from a command and management (C&C) node. One in every of the foremost important threats to the net these days is that the threat of botnets, that are a unit networks of compromised machines below the management of associate offender. It's troublesome to live the extent of harm caused on the net by botnets, however it's wide accepted that the harm done is important. Further, the potential for orders of magnitude additional harm exists within the future.

The beginning of botnets will be copied back to basic types of benign bots. The EggDrop bot is one in every of the earliest standard bots used for automating basic tasks on net relay chat (IRC). Today, there are units several botnets that use IRC as a type of centralized command and management (C&C). The fundamental scripting tasks that a benign bot like EggDrop offers may be wont to coordinate bots. It's variety of unintended ways exist to sight and stop botnets, and these ways still mature. As techniques for botnet detection and mitigation advance, the hardiness and resiliency of botnets also will advance. Today, the foremost simply detected botnets use IRC as a type of communication for command and management (C&C). IRC has several properties that create it enticing for associate offender like its redundancy, measurability, and flexibility [11]. Further, there's an outsized base of information and ASCII text file for developing IRC-based bots. Several botnet authors employ existing code so as to form their own botnet.

One key property of IRC-based botnets is that the use of IRC as a type of central C&C. This property provides the attackers with terribly economical communication. However, the property conjointly is a significant disadvantage to the offender. The threat of the botnet will be slaked and probably eliminated if the central C&C is incapacitated. It's seemingly that new architectures can emerge because the ability to prevent IRC-based botnets matures. The botnet is commandeered by a "botmaster" and used as "resource" or "platform" for attacks. the power to coordinate and transfer new commands to bots offers the botnet owner immense power once performing arts criminal activities, as well as the power to orchestrate police investigation attacks, perform DDoS extortion, causation spam for pay, and phishing. Botnets are unit collections of compromised computers running code below a typical command-and-control infrastructure, typically used for malicious functions. The automated detection of botnet traffic is of high importance for service suppliers and huge field network observation [1]. The development of P2P networks is on the highest of IP layer, generally with a localized protocol permitting 'peers' to share resources. As P2P networks are a unit inherently sculptured with none centralized server, they lack one point-of-failure. This resilience offered by P2P networks has conjointly attracted the eye of adversaries within the type of bot-masters. A 'bot' could be a worm that permits the operator to remotely management the infected system wherever it's put in [2],[10].
A network of such compromised end-hosts below the remote command of a master (i.e., the bot-master) is named a 'Botnet'. Peer-to-Peer overlay networks area unit distributed systems consisting of interconnected nodes that self-organize into network topologies. They're engineered with specific functions of sharing resources like content, CPU cycles, storage and information measure, and have the power to accommodate a transient population of nodes whereas

maintaining acceptable property and performance, while not requiring the intercession or support of a worldwide centralized server or authority [3]. Ancient botnets were well-known to use IRC (Internet Relay Chat), that inexplicit a centralized design for his or her 'Command & Control' (C & C) operations. Sleuthing the centralized C & C server meant delivery down the complete botnet[12]. Botmasters have used the resilience offered by P2P networks to make botnets whereby bots communicate, pass away commands and update alternative bots in an exceedingly P2P fashion. Even as a P2P network is resilient to break-down if a couple of peers leave the network, P2P botnets have tested to be extremely resilient not withstanding a particular variety of bots are a unit known and taken-down. The power to sight botnets could be a crucial part of a network's security system.

Botnet architecture: There are two elementary approaches in botnet architecture: (a) the centralized approach, that uses Command and management (C&C) channels like net Relay Chat (IRC) to receive directions from one supply, (e.g. R-Bot, Spybot, or Gaobot), and (b) the localized approach, that utilizes a peer-to-peer protocol to coordinate its operation (e.g. Storm and Nugache). The localized (or P2P) approach offers higher resiliency and therefore the botnets implementing it area unit tougher to each sight and take down.

A P2P bot's life cycle consists of the subsequent stages:

• Infection stage, throughout that the larva spreads (this may happen through drive-by downloads, a malicious code being put in by the end-user, infected USB sticks, etc.).
• Rally stage, wherever the larva connects with a peer list so as to hitch the P2P network.
• Waiting stage, wherever the larva waits for the bot-master's command (and doesn't exhibit a lot of activity otherwise).
• Execution stage, during which it truly carries out a command, like a Denial of Service (DoS) attack, generates spam emails, etc.

It is troublesome to sight hosts infected with such malware, since by default they are doing very little to arouse suspicion: e.g., usually their communications neither consume important information measure nor involve an outsized variety of targets. Analysis will be additional sophisticated by infected hosts encrypting their network traffic, or act over peer-to-peer protocols to "blend-in" with peer-to-peer file-sharing traffic. Whereas bots noncommissioned in aggressive scanning for alternative vulnerable hosts will be detected exploitation well-known techniques, it'd be higher to sight the infected hosts before them participating in malicious activities.
For these reasons, bot detection has remained a difficult problem.

II. Related Work

Most previous work has either centered on P2P traffic classification from the attitude of a additional general downside of net traffic classification or has given special attention to detection of botnets (centralized or distributed) in net traffic. though many approaches are planned to discover P2P botnets through the analysis of their network behavior, most of them propose a binary classification of P2P hosts [2].

Most of the authors make a case for the botnet detection in peer to look communication, one in all them is Alexander V. Barsamian, he projected botnet detection exploitation applied mathematics and behavioural methods[1]. Their approach to find botnet is to think about the categories of behaviors common to any or all bot package and to think about ways in which will detect those behaviors at the network level. In short, they appear for the network-based ways in which bot-infected peers reveal themselves and every different. a number of these behaviors are going to be easier to find than others, and a few are going to be a lot of (or less) reliable indicators of botnet infection than others. Such proof will be thought-about on its own or, a lot of seemingly, consolidated with different proof gathered from systems. They used the tools NetSAW and NetFEE for the detection. however this analysis has immeasurable questions on network and host behavior, the more work that may be exhausted this uses information bases, automatic flow correlation, behavioural procedure.

There is another system that is employed for the identification of unwanted peer to look traffic called Peerrush developed by authors Babak Rahbarinia, Roberto Perdisci, Andrea Lanzi and Kang Li. PeerRush could be a novel system for the identification of unwanted P2P traffic[4]. It goes on the far side P2P traffic detection, and may accurately categorise the detected P2P traffic and attribute it to specific P2P applications, together with malicious applications like P2P botnets. PeerRush achieves these results while not the necessity of deep packet review, and may accurately determine applications that use encrypted P2P traffic Peerrush notice the traffic concerning 99.5% true.

Shishir Nagaraja, Prateek Mittal, Chi-Yao Hong, Matthew Caesar, Nikita Borisov these authors justify BotGrep system for locating P2P Bots with Structured Graph Analysis[5]. BotGrep is associate abstract thought rule that identifies botnet hosts and links inside network traffic traces. BotGrep works by finding out structured topologies, and separating them from the background communication graph. that they had done the prefiltering and cluster of P2P nodes. that they had used privacy protective graph algorithms for establishing a typical symbol house, stochastic process and performance. however

the performance of application can not be improved to it a lot of extent.

Guofei Gu, Roberto Perdisci, Junjie Zhang, and Wenke Lee developed a system named as BotMiner: agglomeration Analysis of Network Traffic for Protocol- and Structure-Independent Botnet Detection, freelance of botnet C&C protocol and structure for act, additionally freelance of content of C & C communication, needs no a priori information of botnets [6]. BotMiner notice teams of compromised machines at intervals a monitored network that square measure a part of a botnet. this can be done by passively analyzing network traffic within the monitored network. agglomeration and correlation is employed. BotMiner shows wonderful detection accuracy on numerous forms of botnets with a awfully low false positive rate on traditional traffic. however this work needs some a lot of strong techniques for cluster communication and activity patterns of botnets than these.

Ting-Fang Yen justify Detection of sneak Malware exploitation behavioural options in Network Traffic exploitation network analysis[7]. The infected hosts exhibit similar characteristics in their network activities that area unit distinct from those of benign hosts. This approach therefore identifies bots by aggregating “similar” network traffic, that area unit collected within the sort of flow records that contain coarse summaries of every association. below this framework, authors gift techniques to spot each infected hosts collaborating in centralized botnets and people that communicate over peer-to-peer networks. They additional develop a passive browser procedure methodology to sight malware that aren\'t confined to hosts of one software platform. They conjointly study peer-to-peer botnets analytically exploitation models from network theory, and investigate however a structural characteristic of networks affects the effectiveness of botnet takedown methods. however it\'s disadvantage of requiring potential strategies to boost sneak malware detection systems because it doesn\'t correlate to host profile changes.

The authors Pratik Narang, Subhajit Ray, Chittaranjan Hota, and Venkat Venkatakrishnan have developed a system known as PeerShark: sleuthing Peer-to-Peer Botnets by chase Conversations [2]. PeerShark, a system that detects P2P botnet traffic and differentiates it from benign P2P traffic in a very network. rather than the standard 5-tuple ‘flow-based’ detection approach, a 2-tuple ‘conversation-based’ approach is employed that is port-oblivious, protocol-oblivious and doesn\'t need Deep Packet scrutiny. Its system style supported chase conversations and categorization of P2P application. They use the algorithmic programs like packet filtering algorithm, oral communication creation algorithmic program and oral communication aggregation algorithmic program. however this method is merely applicable to 2 connected nodes.

Huy Hang, Xuetao Wei, Michalis Faloutsos, Tina Eliassi-Rad these authors has developed a system referred to as Entelecheia: detection P2P Botnets in their Waiting Stage [8]. we have a tendency to had studied the waiting stage of botnets in introduction section; we all know that these botnets square measure non-active in their waiting stage. These authors have used graph primarily based approach for developing this technique. They trace the traffic and use superflow graph rule and developed superflow graph that is then experience filtering and cluster module to list the suspected bots. Limitations of this technique embrace that, it (Entelecheia) will determine associate initial set of bots, this approach works with centralized botnets as a result of their traffic can merely kind easy-to-detect clusters with high period and low-volume flows.

BotTrack: following Botnets victimization NetFlow and PageRank is another system developed by authors Jerome Francois, Shaonan Wang, Radu State, and Thomas Engel[9]. It analyze communication activity patterns and to infer potential botnet activities. This approach relies on process NetFlow-related data to make a bunch dependency model that captures data concerning that host is reproval that alternative host. Linkage analysis on this host dependency model is including a agglomeration algorithmic program so as to make clusters of equally behaving nodes.

III. Conclusion

Due to distributed nature of botnets, it is hard to detect them in peer-to-peer networks. So we require the smarter technique to detect such threats. In this study, we get aware to botnets and its some of detection techniques. But this is not sufficient to know thoroughly the botnets. The techniques given above have somewhat disadvantages so; we still require some more influenced techniques for this. The work is going on the detection of such botnets by the scientists and authors.

References

[1] “Network characterization for botnet detection using statistical-behavioral methods” by Alexander V. Barsamian in June 2009.

[2] “PeerShark: Detecting Peer-to-Peer Botnets by Tracking Conversations” by Pratik Narang, Subhajit Ray, Chittaranjan Hota, Venkat Venkatakrishnan in 2014 IEEE Security and Privacy Workshops.

[3] S. Androutsellis-Theotokis and D. Spinellis, “A survey of peer-to-peer content distribution technologies,” ACM Computing Surveys (CSUR), vol. 36, no. 4, pp. 335–371, 2004.

[4] B. Rahbarinia, R. Perdisci, A. Lanzi, and K. Li, "Peerrush: Mining for unwanted p2p traffic," in Detection of Intrusions and Malware, and Vulnerability Assessment, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, vol. 7967, pp. 62–82.

[5] S. Nagaraja, P. Mittal, C.-Y. Hong, M. Caesar, and N. Borisov, "Botgrep: Finding p2p bots with structured graph analysis." in USENIX Security Symposium, 2010, pp. 95–110.

[6] G. Gu, R. Perdisci, J. Zhang, W. Lee et al., "Botminer: Clustering analysis of network traffic for protocol-and structure-independent botnet detection." in USENIX Security Symposium, 2008, pp. 139–154.

[7] T.-F. Yen, technical report given on the topic "Detecting Stealthy Malware Using Behavioral Features in Network Traffic" in August 2011 Department of Electrical and Computer Engineering Carnegie Mellon University Pittsburgh, PA 15213.

[8] H. Hang, X. Wei, M. Faloutsos, and T. Eliassi-Rad, "Entelecheia: Detecting p2p botnets in their waiting stage," in IFIP Networking Conference, 2013, 2013, pp. 1–9.

[9] J. Franc¸ois, S. Wang, R. State, and T. Engel, "Bottrack: Tracking botnets using netflow and pagerank," in Proceedings of the 10th International IFIP TC 6 Conference on Networking -Volume Part I, 2011, pp. 1–14.

[10] P. Narang, J. M. Reddy, and C. Hota, "Feature selection for detection of peer-to-peer botnet traffic," in Proceedings of the 6th ACM India Computing Convention, 2013, pp. 16:1–16:9.

[11] D. Dittrich and S. Dietrich, "P2p as botnet command and control: a deeper insight," in Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on. IEEE, 2008, pp. 41–48.

[12] C. Rossow, D. Andriesse, T. Werner, B. Stone-Gross, D. Plohmann, C. J. Dietrich, and H. Bos, "Sok: P2pwned-modeling and evaluating the resilience of peer-to-peer botnets," in Security and Privacy (SP), 2013 IEEE Symposium on. IEEE, 2013, pp. 97–111.