www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 6 June 2015, Page No. 12687-12691

Social Network Based Security Schema for Botnet Detection and prevention

Ms. Geerthidevi K G, Dr. T. Senthil Prakash M.Phil., Ph.D, Ms. S. Tharani

PG Scholar
Shree Venkateshwara Hi-Tech Engg College
Gobi,India
geerthivya@gmail.com
Professor and HoD,
Department of CSE, College, Shree Venkateshwara Hi-Tech Engg College,
Gobi,Tamilnadu, India,
PG Scholar
Shree Venkateshwara Hi-TechEngg College
Gobi,India
tharanimahalakshmi@gmail.com

Abstract: Generally, the botnet is one of the most dangerous threats in the network. It has number attackers in the network. The attacker consists of DDOS attack, remote attack, etc., Bots perform perform repetitive tasks automatically or on a schedule over the internet, tasks that would be too mundane or time-consuming for an actual person. But the botnets have stealthy behavior as they are very difficult to identify. These botnets have to be identified and the internet have to be protected. Also the the activity of botnets must be prevented to provide the users, a reliable service. The past of botnet detection has a transaction process which is not secure. A efficient stastical data classifier is required to train the botent preventions system. To provide the above features clustering based analysis is done. our approach can detect and profile various P2P applications rather than identifying a specific P2P application. Anomaly based detection technique is used to obtain this goal.

Keywords: Botnet, anomaly base detection, hash function, DDOS

1. INTRODUCTION

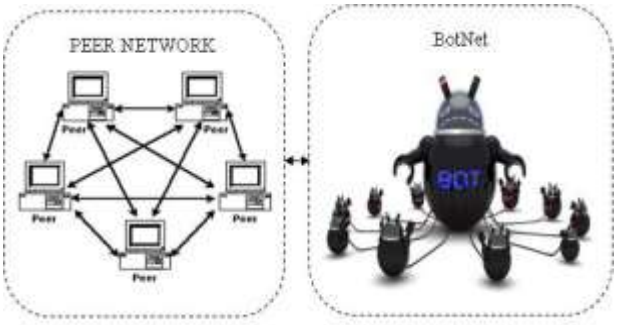
A botnet is a collection of Internet-connected programs communicating with other similar programs in order to perform tasks. Botnets sometimes compromise computers whose security defenses have been breached and control conceded to a third party. [1]It is remotely controlled by an attacker through a command and control (C&C) channel. Botnets serve as the infrastructures responsible for a variety of cyber-crimes, such as spamming, distributed denial of-service (DDoS) attacks, identity theft, click fraud, etc. The C&C channel is an essential component of a botnet because botmasters rely on the C&C channel to issue commands to their bots and receive information from the compromised machines.

Bots perform perform repetitive tasks automatically or on a schedule over the internet, tasks that would be too mundane or time-consuming for an actual person. Search engines use them to surf the web and methodically catalogue information from websites, trading sites make them look for the best bargains in seconds, and some websites and services employ them to deliver important information like weather conditions, news and sports, currency exchange rates.

Unfortunately, not all bots roaming the internet are useful and harmless. Cyber crooks have also noticed their potential and have come up with malicious bots – programs designed to secretly install themselves on unprotected or vulnerable computers and carry out whatever actions they demand. And that could be anything from sending spam to participating in a distributed denial of service attack (DDoS) that brings down entire websites

Once infected, your computer becomes part of a botnet – a network of infected or zombie-computers controlled from the distance by a cybercriminal who rented it to carry out his illegal plans. So not only is your computer infected and your internet security compromised, but your system resources and your bandwidth are rented out to the highest bidder to help them attack other unsuspecting users or even legitimate businesses. This huge potential for cybercrime makes these botnets what some security experts believe to be the most dangerous threat on the internet today.

Such networks comprising hundreds or thousands of infected devices have the resources needed to perform high-scale malicious actions such as: (1) Mass-spam delivery that floods millions of inboxes in a matter of seconds (2) DoS and DDoS attacks that crash entire websites and can put legitimate businesses in serious trouble (3) Brute-force hacking attacks by cracking passwords and other internet security measures (4) Identity theft and internet fraud by collecting private information from infected users



Bots can sneak up on you in many ways. They can use the vulnerabilities and outdated software in your system to infect it while you're casually surfing the web. They can be delivered by Trojans or questionable software you get tricked into downloading (like rogue antivirus programs). Or they can be sent directly to your inbox as an email attachment by spammers.

Botnets perform many malicious activity in internet like sending spams to emails, increasing network traffic and even takes control of the system by running Trojans. But the botnets have stealthy behavior as they are very difficult to identify. These botnets have to be identified and the internet have to be protected. The information shared in social media are sensitive and personal. Hence the activity of botnets must be prevented to provide the users, a reliable service.

To provide the above features clustering based analysis is done. our approach can detect and profile various P2P applications rather than identifying a specific P2P application. Anomaly based detection technique is used to obtain this goal.

2. RELATED WORKS

Many approaches have been proposed to detect botnets have been proposed. For example, BotMiner [7] identifies a group of hosts as bots belonging to the same botnet if they share similar communication patterns and meanwhile perform similar malicious activities, such as scanning, spamming, exploiting, etc.[4] Unfortunately, the malicious activities may be stealthy and non-observable A efficient stastical data classifier is required to train the botent preventions system. Acquiring such information is a challenging task, thereby drastically limiting the practical use of these methods. Some of the older approach involves content signature, encryptions, profiling, fixed source port. our approach does not need any content signature. our analysis approach can estimate the active time of a P2P application, which is critical for botnet detection

3. SYSTEM DESIGN

A Botmaster has to be designed with P2P protocol. Therefore P2P bots exhibit some network traffic patterns that are common to other P2P client applications either legitimate or malicious. Hence our system is divided into two phases. In the first phase, we aim at detecting all hosts within the monitored network that engage in P2P communications. We analyze raw traffic collected at the edge of the monitored network and apply a pre-filtering step to discard network flows that are unlikely to be generated by P2P[1]. We then analyze the remaining traffic and extract a number of statistical features to identify flows generated by P2P clients. In the second phase, our system analyzes the traffic generated by the P2P clients and classifies them into either *legitimate* P2P clients or P2P *bots*. Specifically, we investigate the active time of a P2P client and identify it as a *candidate* P2P bot if it is persistently active on the underlying host. We further analyze the overlap of peers contacted by two *candidate* P2P bots to finalize detection. After analyzing with the use of anomaly based detection algorithm the network has to be revoked from malwares.

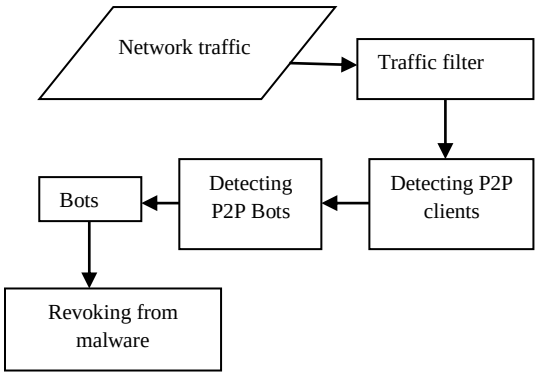


Fig 1: System architecture

3.1 Detecting P2P client

Traffic filter is used to sort out th traffic which is unlikely to P2P networks. In this first phase, fine grained detection of P2P botenets is implemented. This component is responsible for detecting P2P clients by analyzing the remaining network flows after the Traffic Filter component. For each host h within the monitored network we identify two flow sets, denoted as Stcp(h) and Sudp(h), which contain the flows related to successful outgoing TCP and UDP connection, respectively.

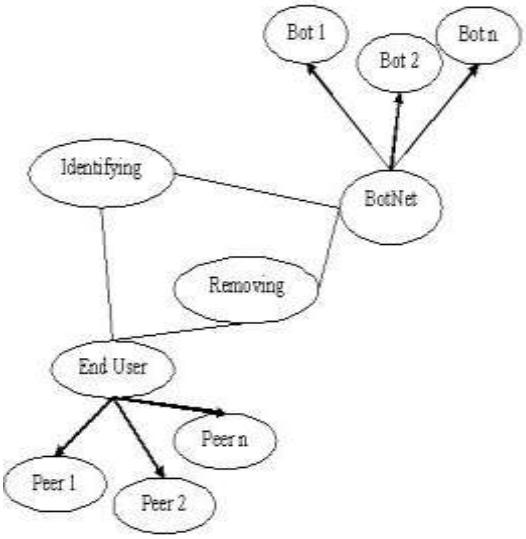
To identify flows corresponding to P2P control messages,we first apply a flow clustering process intended to group together similar flows for each candidate P2P node *h*. Given sets of flows *Stcp(h)* and *Sudp(h)*, we characterize each flow using a vector of statistical features *v(h)* = [*Pkts* , *Pktr* , *Bytes* , *Byter*], in which *Pkts* and *Pktr* represent the number of packets sent and received, and *Bytes* and *Byter* represent the number of bytes sent and received, respectively.

The distance between two flows is subsequently defined as the *euclidean distance* of their two corresponding vectors. We then apply a clustering algorithm to partition the set of flows into a number of clusters. Each of the obtained clusters of flows, *Cj (h)*, represents a group of flows with similar size.

Flows corresponding to ping/pong and peer-discovery share similar sizes, and hence they are grouped into two clusters (FC1 and FC2), respectively. Since the number of destination BGP prefixes involved in each cluster is larger, we take FC1 and FC2 as its fingerprint clusters. A fingerprint cluster summary, (Pkts , Pktr , Bytes , Byter , proto), represents the protocol and the average number of sent/received packets/bytes for all the flows in this fingerprint cluster. We implemented the flow analysis component and identified fingerprint cluster for the sample P2P traces including two traces.

3.2 Detecting P2P bots

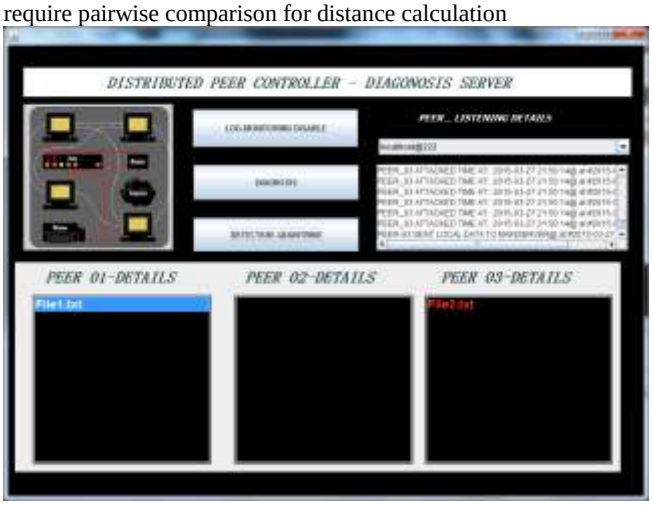
To detect the bots coarse grained detection method is used. Since bots are malicious programs used to perform profitable malicious activities, they represent valuable assets for the botmaster, who will intuitively try to maximize utilization of bots. This is particularly true for P2P bots because in order to have a functional overlay network (the botnet), a sufficient number of peers needs to be always online. In other words, the active time of a bot should be comparable with the active time of the underlying compromised system.



The distance between each pair of hosts is computed. We apply hierarchical clustering, and group together hosts according to the distance defined above. In practice the hierarchical clustering algorithm will produce a dendrogram (a tree-like data structure). The dendrogram expresses the "relationship" between hosts. The closer two hosts are, the lower they are connected at in the dendrogram. Two P2P bots in the same botnet should have small distance and thus are connected at lower level. In contrast, legitimate P2P applications tend to have large distances and consequently are connected at the upper level. We then classify hosts in dense clusters as P2P bots, and discard all other clusters and the related hosts, which we classify as legitimate P2P clients.

4. SYSTEM IMPLEMENTATION

Out of four components in our system, "Traffic Filter" and "Coarse-Grained Detection of P2P Bots" have linear complexity since they need to scan flows only once to identify flows with destination addresses resolved from DNS queries or calculate the active time. Other two components, "Fine-Grained Detection of P2P Clients" and "Fine-Grained P2P Detection of P2P Bots", require pairwise comparison for distance calculation



We use a two-step clustering approach to reduce the time complexity of "Fine-Grained P2P Client Detection". For the first-step clustering, we use an efficient clustering algorithm to aggregate network flows into K sub-clusters, and each subcluster contains flows that are very similar to each other. For the second-step clustering, we investigate the global distribution of sub-clusters and further group similar sub-clusters into clusters.

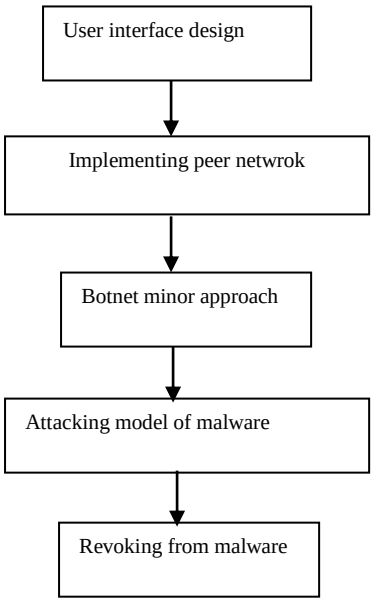
The distance of two flows is defined as the Euclidean distance of their corresponding vectors, where each vector [Pkts , Pktr , Bytes , Byter] represents the number of packets/ bytes that are sent/received in a flow.

For the second-step clustering, we use hierarchical clustering with DaviesBouldin validation [24] to group sub-clusters into clusters. Each sub-cluster is represented using a vector ([Pkts , Pktr , Bytes , Byter]), which is essentially the average for all flow vectors in this sub-cluster.

Hierarchical clustering is used to build a dendrogram. Finally, DaviesBouldin validation is employed to assess the global distribution of inter- and intra-cluster distances of clusters based on various clustering decisions and yield the best cut for the dendrogram. The two-step clustering algorithm has the time complexity of O(nK I + K2).

4.1 Modules

The goal of guarding the large scale scrabble in social network is implemented by the following modules,



4.1.1 User Interface Design

The user interaction is effective operation and control of the machine on the user‘s. The user interface module has login and registration phases. The registration phase gets details from user and stores it in database. It also checks the details are valid or not.

4.1.2 Implementing peer network

The peer network contain decentralized networks. All the nodes contains separate IP address and separate port number. The peer one node have stored separate list of files which in the global respository.

4.1.3 Botnet minor approach

The global respository contains the decentralized network details. The botnet minor store and reteive the information about port and IP details from the database. Identification scenario always visible botnet minor. If any dispute in the identification scenario overall network may be crashed.

4.1.4 Attacking model of Malware

Botnet minor contain all the details about the peer network. The botnet minor handles all the request processed by the decentralized network. The botnet major attack decentralized scenario spread the warm data to the peer network. The node connected with the attacked node that specific node also get the warm data.

4.1.5 Revoking the network from Malware

Data matching have the law data and the original data. The proposed technical approach can identify the warm data it is spreaded by the botnet. Revoke the original data instead of warm data it can identify the problem and revoke the botnet minor from the attacking model.

5. EXPERIMENTAL RESULTS

We prepared a data set (D) for evaluation. Specifically, we randomly selected half (8) of the P2P bots from NETbots .Then for each of the 5 P2P applications we ran, we randomly
selected one out of its two traces from NETP2P and overlaid its traffic to the traffic of a randomly selected host We applied our detection system on data set D. The traffic filter drastically reduced the workload for the whole system. As indicated in Figure 4, it reduced the number of hosts subject to analysis by 67% (from 953 to 316) but retained all P2P clients.

Among 26 P2P clients identified in the previous step, 25 out of them exhibit persistent P2P behaviors. We further evaluate the similarity of fingerprint clusters and peer IPs for each pair of persistent P2P clients and
derive a dendrogram.

If botmasters get to know about our detection algorithm, they could attempt to modify other bots’ network behavior to evade detection. This situation is similar to evasion attacks against other intrusion detection systems

6. CONCLUSION

To summarize, although our system greatly enhances and complements the capabilities of existing P2P botnet detection systems, it is not perfect. We should definitely strive to develop more robust defense techniques, where the aforementioned discussion outlines the potential improvements of our system.

In this paper, we presented a novel botnet detection system that is able to identify *stealthy* P2P botnets, whose malicious activities may not be observable. To accomplish this task, we derive *statistical fingerprint*s of the P2P communications to first detect P2P clients and further distinguish between those that are part of legitimate P2P networks (e.g., filesharing networks) and P2P bots. We also identify the performance bottleneck of our system and optimize its scalability. The evaluation results demonstrated that the proposed system accomplishes high accuracy on detecting stealthy P2P bots and great scalability.

7. REFERENCES

[1] S. Stover, D. Dittrich, J. Hernandez, and S. Dietrich, “Analysis of the storm and nugache trojans: P2P is here,” in Proc. USENIX, vol. 32. 2007, pp. 18–27.

[2] Junjie Zhang, Roberto Perdisci, Wenke Lee, Xiapu Luo, and Unum Sarfraz, “Building a Scalable System for Stealthy P2P-Botnet Detection”, IEEE transactions on information forensics and security, vol. 9, no. 1, january 2014

[3] Pratik Narang, Subhajit Ray, Chittaranjan Hota, Venkat Venkatakrishnan, "PeerShark: Detecting Peer-to-Peer Botnets by Tracking Conversations",2014 IEEE Security and Privacy Workshops

[4] P. Porras, H. Saidi, and V. Yegneswaran, "A multi-perspective analysis of the storm (peacomm) worm," Comput. Sci. Lab., SRI Int., Menlo Park, CA, USA, Tech. Rep., 2007.

Authors:

[5] P. Porras, H. Saidi, and V. Yegneswaran. (2009). Conficker C Analysis [Online]. Available: http://mtc.sri.com/Conficker/addendumC/index.html

[6] G. Sinclair, C. Nunnery, and B. B. Kang, "The waledac protocol: The how and why," in Proc. 4th Int. Conf. Malicious Unwanted Softw., Oct. 2009, pp. 69–77.

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

[5] R. Lemos. (2006). Bot Software Looks to Improve Peerage[Online]Available:http://www.securityfocus.com/news/11390

[6] Y. Zhao, Y. Xie, F. Yu, Q. Ke, and Y. Yu, "Botgraph: Large scale spamming botnet detection," in Proc. 6th USENIX NSDI, 2009,
pp. 1–14.

[8] T.-F. Yen and M. K. Reiter, "Are your hosts trading or plotting? Telling P2P file-sharing and bots apart," in Proc. ICDCS, Jun. 2010, pp. 241–252.

Ms. Geerthidevi K G, PG Scholar Currently persuing her M.E CSE degree in Shree Venkateshwara Hi-Tech Engg College, Gobi, Tamilnadu , India. Her research interests include Networking, Network Security etc.,

Mr.S.Prakadeswaran, received the Bachelor of Engineering in Anna University,Chennai,Tamilnadu in 2008.He received the Master of Engineering in Anna University, Chennai, Tamilnadu in 2013. He has the experience in Teaching of 6+Years. He is currently working as Assistant professor in Shree Venkateshwara Hi Tech Engineering College, Gobichettipalayam, Tamilnadu. His research interest includes Wireless Networks and Pervasive computing. He has published several papers in 4 International Journals

S. Tharani, received the B.Tech(IT) degree from Bannari Amman Institute of Technology, Sathyamangalam, India in 2007 -2010 and worked as lecturer in Shree Venkateshwara Hi-Tech Polytechnic College, Erode, India in 2011 - 2013. Currently Pursuing ME(CSE) degree in SVHEC, Erode, India in 2013 - 2015. Her research interests include Database, Network Security and Data Mining