International Refereed Journal of Engineering and Science (IRJES)
ISSN (Online) 2319-183X, (Print) 2319-1821
Volume 5, Issue 5 (May 2016), PP.71-75

Detecting Spam Zombies By Monitoring Outgoing Messages

Birru Devender[1],Korra Srinivas[2],Ch.Tulasi Ratna Mani[3]
[1]*Working as Associate Professor, Holy Mary Institute of Technology and Science, Keesara , Affiliated to JNTU-Hyderabad, T.S, India.*
[2,3]*Working as Assistant Professor, Holy Mary Institute of Technology and Science, Keesara, Affiliated to JNTU-Hyderabad, T.S, India*

Abstract: Research have studied numerous means of One of the key security threats on the Internet occurs because of Compromised machines, such systems often used to launch various security attacks such as spamming and spreading malware, DDoS, and identity theft. The spamming provides a key economic incentive for attackers to recruit the large number of compromised machines, so keeping focus on the detection of the compromised machines in a network that are involved in the spamming activities, commonly known as spam zombies. So here to develop effective spam zombie detection system named SPOT by monitoring outgoing messages of a network. SPOT is designed based on a statistical tool called Sequential Probability Ratio Test, this is a very powerful technique which gives the result by considering very few number of observations made at less time consumption and improves the speed to execute the process. A methodology called Paul-Graham Implementation is used to detect the spams by applying the SPOT filter using SPRT which is tracked when a message is passed from the network which is called outgoing messages. This implementation is based on Bayesian calculation which determines the rating of spam and based on this rating, the technique identifies the mail is either spam or not and also verifies weather the system is compromised system or not.
Keywords – DDoS, and identity theft. The spamming provides a key economic incentive for attackers to recruit the large number of compromised machines.

I. INTRODUCTION

A major Security challenge on the Internet is the existence of the large number of compromised machines. Such machines have been increasingly used to launch various security attacks including spamming and spreading malware, DDOS, and identity theft. Two natures of the compromised machines on the Internet sheer volume and wide spread render many existing security countermeasures less effective and defending attacks involving compromised machines extremely hard. On the other hand, identifying and cleaning compromised machines in a network remain a significant challenge for system administrators of networks of all sizes.

In this paper, I focus on the detection of the compromised machines in a network that are used for sending spam messages, which are commonly referred to as spam zombies. Given that spamming provides a critical economic incentive for the controllers of the compromised machines to recruit these machines, it has been widely observed that many compromised machines are involved in spamming. A number of recent research efforts have studied the aggregate global characteristics of spamming botnets (networks of compromised machines involved in spamming) such as the size of botnets and the spamming patterns of botnets, based on the sampled spam messages received at a large email service provider.

Rather than the aggregate global characteristics of spamming botnets, I aim to develop a tool for system administrators to automatically detect the compromised machines in their networks in an online manner. I consider ourselves situated in a network and ask the following question: How can I automatically indentify the compromised machines in the network as outgoing messages pass the monitoring point sequentially? The approaches developed in the previous work cannot be applied here. The locally generated outgoing messages in a network normally cannot provide the aggregate large-scale spam view required by these approaches. Moreover, these approaches cannot support the online detection requirement in the environment I consider.

The nature of sequentially observing outgoing messages gives rise to the sequential detection problem. In this paper, I will develop a spam zombie detection system, named SPOT, by monitoring outgoing messages. SPOT is designed based on statistical method called Sequential Probability Ratio Test (SPRT), developed by Wald in his seminar work. SPRT is a powerful statistical method that can be used to test between two hypotheses (in our case, a machine is compromised vs. the machines is not compromised), as the events (in our case, outgoing messages) occur sequentially. As a simple and powerful statistical method, SPRT has a number of desirable features. It minimizes the expected number of observations required to reach a decision among all the sequential and non-sequential statistical tests with no greater error rates. This means that the SPOT detection system can

identify a compromised machine quickly. Moreover, both the false positive and false negative probabilities of SPRT can be bounded by user defined thresholds. Consequently, users of the SPOT system can select the desired thresholds to control the false positive and false negative rates of the system.

In this paper, I develop the SPOT detection system to assist system administrators in automatically identifying the compromised machines in their networks.

II. METHODOLOGY

Problem Formulation

In this paper, I formulate the spam zombie detection problem in a network. In particular, I discuss the network model and assumptions I make in the detection problem.

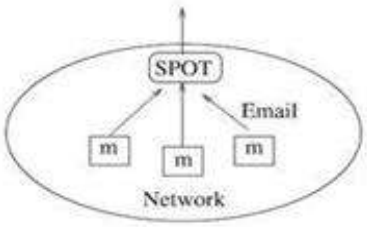

Fig. 1. Network Model

. 1.1.1 illustrates the logical view of the network model. I assume that messages m originated from machines inside the network will pass the deployed spam zombie detection system. This assumption can be achieved in a few different scenarios. First, in order to alleviate the ever-increasing spam volume on the internet, many ISPs and networks have adopted the policy that all the outgoing messages originated from the network must be relayed by a few designated mail servers in the network. Outgoing email traffic from all other machines in the network is blocked by edge routers of the network. In this situation, the detection system can be co-located with the designated mail detection system can be co-located with the designated mail servers in order to examine the outgoing messages. Second, in a network where the aforementioned blocking policy in not adopted, the outgoing email traffic can be replicated and redirected to the spam zombie detection system. I note that the detection system does not need to be on the regular email traffic forwarding path, the system only needs a replicated stream of the outgoing email traffic. Moreover, as I will show in paper, the proposed SPOT system works well even if it cannot observe all outgoing messages, SPOT only requires a reasonably sufficient view of the outgoing messages originated from the network in which it is deployed.

Architectural Block Diagram

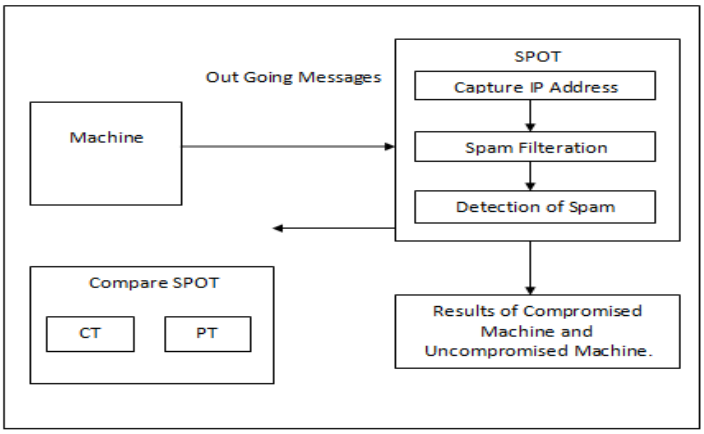

Fig. 1.2.1 Architecture of Spam Detection Process

The Fig. 2 depicts the architecture following the three major components. Machine, SPOT and Comparision of SPOT. Here the Machine is a local system that assists the system administrators supporting the network behavior environment through which various IP Addresses are generated randomly from the same machine. An account authentication process is followed in order to send mail to the recipient address. Now these messages are the outgoing messages after the mail is sent and now the system automatically detects the IP

Address randomly generated through some logic implemented by code and list of mails and its message files are displayed to the administrator so that he can view the messages and later these are applied for filtration process which implements SPRT.

As I discussed SPOT implements the testing procedure to find the problem of detection using Sequential Probability Ratio Test. This includes the filtration process and executes the output with the ratio. This includes the Paul-Graham procedural process which executes based on two hypothesis considering two files. One file contains the records of non-spam that is good message file and another file is a spam which contains records of data usually it is called as a bad message file. These both files may contain more number of records which can be greater than thirty thousand lines of text. This text may contain both readable and non-readable data. Whenever the mails are sent than the attachments containing the files are tested between these two trained files, I call this trained files as the two hypothesis and finally rating is measured and based on this rating SPRT decides whether the mail file is spam or genuine.

There are more two algorithms bounded to SPOT. One I call as Count Threshold and other is the Percentage Threshold. A particular choice is made when SPRT is finished so as to choose one technique among two and later the control is transferred to that technique. These two thresholds are called as user-defined threshold algorithms because here user will give the constraint limited values to detect the mail spams which are independent for system administrators.

The Count Threshold (CT) detection used to count the number of times the mail messages arrives at each IP Address location. Here user constraints the limited value usually integer. So if the count of the mail message files arrived at particular IP Address is less than this limited value specified and if and only if the records in the file contains greater than twenty lines than it displays as it is the spam mail.

The Percentage Threshold (PT) detection is executed between two limited constraints. One I call it as minimum limit and another is the maximum limit, both are integer values. The minimum limit is used to count the total number of files sent from various address locations and if it exceeds its limit than the mail containing the file is a spam file. Another limit the maximum value is used to check whether the number of mails sent are within its value specified than is less than the limit. But both will be displayed as spam if and only if its records containing lines are greater than twenty lines.

In the above discussion of the spam zombie detection algorithms I have for simplicity ignored the potential impact of dynamic IP addresses and assumed that an observed IP corresponds to a unique machine. In the following I informally discuss how well these algorithms fair with dynamic IP addresses.

SPOT can work extremely well in the environment of dynamic IP addresses. To understand the reason I note that SPOT can reach a decision with a small number of observations and shows the average number of observations required for SPRT to terminate with a conclusion. In practice, I have noted that 3 or 4 observations are sufficient for SPRT to teach a decision for the vast majority of cases. If a machine is compromised, it is likely that more than 3 or 4 spam messages will be sent before the (unwitting) user shutdowns the machine and the corresponding IP address gets re-assigned to a different machine. Therefore, dynamic IP addresses will not have any significant impact on SPOT.

So, only CT and PT detection are the only two techniques that can support the dynamic behavior. Both the algorithms have surpassed the system SPRT technique in automatically detecting the spam files which are independent of dynamic support.

There are no error rates like false positive or false negative in case of SPRT implemented by SPOT. So, higher efficiency is maintained by successful execution with less number of observations. It is a very simple process for administrators in detecting the comprised machines because of faster mode of execution.

Machine

The machine is any local system. In this paper it is either a connected network or it can be unconnected network. It is not mandatory to have an internet connection in order to send the mails. An IP Address is generated for each login of a user and it is different from the previous IP Address. In this way though without having the internet connection the system can act as if it is connected to a network supporting the dynamic behavior. So whatever the mails are sent from this machine is tracked in the next phase and these mails are now called the outgoing messages.

Spot

In the above Fig. 1.2.1, this technique now follows three sequential steps. One Capture IP Address, Spam Filteration and Detection of Spam. This phase considers outgoing messages as input and displays the corresponding results as output.

Capture IP Address

As explained above the IP Address is generated logically through some implementation using java

program. The logic will randomly generate a different IP at each login when accessing the user account. This IP Address can be viewed by the administrators when they can click the button **Capture IP** developed in a module. This phase also created to serve the administrators to view the mail list of a user to whom he sent the mails generated from the IP at login. Next the control is transferred to the Spam Filter phase.

Spam Filteration

Filteration is the execution of program by which it implements some methodology to detect the spams generated from the mails of that particular IP Address and it uses the sequential observations made to test the process. This Technique uses only fewer observations to complete the task. Next the output result is displayed.

Detection of Spam

Now comes the detection of spam messages with respect to the IP Address of a machine. This phase detects the mails are either spam content or genuine and regarding this it decides whether the system is a compromised system or an uncompromised system.

III. RESULTS

All the results of the mail sent by the user are here stored in a database. It can be further used in the upcoming phases so as to detect the spams with other two more techniques which support dynamic behavior. Here all the mails sent by user are stored from various users accessing the accounts. So the database stores each and every details of user mails sent from various IP Address generated. This is how the other two techniques as CT and PT supports the dynamic behavior.

Compare Spot

As explained above SPOT comparison uses two techniques providing a choice to select one among them so as to transfer the control to another technique. This phase has the two more sub phases nothing but called as CT Detection and PT Detection techniques.

Ct

CT stands for Count Threshold and the features of this phase are explained above. This is the technique which executes the program based on database connectivity in order to detect the spams and uses a user defined threshold limit value.

Pt

PT stands for Percentage Threshold and the features of this phase are explained above. This is also the technique which executes the program based on database connectivity in order to detect the spams and uses two user defined threshold limit values.

IV. AN OVERVIEW OF PROPOSED SYSTEM

In this paper, I focus on the detection of the compromised machines in a network that are used for sending spam messages, which are commonly referred to as spam zombies. The nature of sequentially observing outgoing messages gives rise to the sequential detection problem. In this paper, I will develop a spam zombie detection system, named SPOT, by monitoring outgoing messages. SPOT is designed based on a statistical method called Sequential Probability Ratio Test (SPRT), as a simple and powerful statistical method, SPRT has a number of desirable features. It minimizes the expected number of observations required to reach a decision among all the sequential and non-sequential statistical tests with no greater error rates. This means that the SPOT detection system can identify a compromised machine quickly.

In proposed system to develop an effective spam zombie detection system named SPOT is used in monitoring outgoing messages of a network. SPOT is designed based on a statistical method called sequential probability ratio test (SPRT). SPOT can be used to test between two hypotheses whether the machine is compromised or not.

Advantages Of Proposed System

SPOT is an effective and efficient system in automatically detecting compromised machines in a network. For example, among the 440 internal IP addresses observed in the e-mail trace, SPOT identifies 132 of them as being associated with compromised machines. Out of the 132 IP addresses identified by SPOT, 126 can be either independently confirmed (110) or are highly likely (16) to be compromised. SPOT has surpassed the false positive and false negative error rates. It also minimizes the number of required observations to detect a spam zombie. In this application SPOT was designed based on a simple and powerful statistical tool named Sequential Probability Ratio Test to detect the compromised machines that are involved in the spamming

activities. SPOT has surpassed the false positive and false negative error rates. It also minimizes the number of required observations to detect a spam zombie.

V. CONCLUSION

In this paper, I developed an effective spam zombie detection system named SPOT by monitoring outgoing messages in a network. SPOT was designed based on a simple and powerful statistical tool named Sequential Probability Ratio Test to detect the compromised machines that are involved in the spamming activities. SPOT has surpassed both the false positive and false negative error rates. It also minimizes the number of required observations to detect a spam zombie. In addition, I also showed that SPOT outperforms two other detection algorithms based on the number and percentage of spam messages sent by an internal machine, respectively. The main usage of the application is sender can identify the sending mails as either spam or not and weather his system is compromised system or an uncompromised one and the user defined thresholds algorithms which are CT and PT can support the dynamic behavior to detect the spam mails associated with different address locations.

Future Scope

This paper only deals with detecting the spam zombies across the network by monitoring the outgoing messages but not provided a resultant solution to stop or interrupt the hacker or third party from transmission of messages. The whole paper is about detecting the spams by using different techniques as SPOT filter, CT Detection and PT Detection but not provided any implementation procedure to resolve the issue of spams that leads to attacks. So our future enhancements are going to be prepared to discover new algorithms to stop these hackers from the transmission of spam mails and providing an efficient utilization of environment in resolving the spam mails also to provide the network environment for SPRT in detecting the spams through various ip addresses dynamically.

REFERENCES

[1]. P. Bacher, T. Holz, M. Kotter, and G. Wicherski, "Know Your Enemy: Tracking Botnets," http://www.honeynet.org/papers/ bots, 2011.
[2]. Z. Chen, C. Chen, and C. Ji, "Understanding Localized-Scanning Worms," Proc. IEEE Int'l Performance, Computing, and Comm. Conf. (IPCCC '07), 2007.
[3]. R. Droms, "Dynamic Host Configuration Protocol," IETF RFC 2131, Mar. 1997.
[4]. Z. Duan, Y. Dong, and K. Gopalan, "DMTP: Controlling Spam through Message Delivery Differentiation," Computer Networks, vol. 51, pp. 2616-2630, July 2007.
[5]. Z. Duan, K. Gopalan, and X. Yuan, "Behavioral Characteristics of Spammers and Their Network Reachability Properties," Technical Report TR-060602, Dept. of Computer Science, Florida State Univ., June 2006.
[6]. Z. Duan, K. Gopalan, and X. Yuan, "Behavioral Characteristics of Spammers and Their Network Reachability Properties," Proc. IEEE Int'l Conf. Comm. (ICC '07), June 2007.
[7]. G. Gu, R. Perdisci, J. Zhang, and W. Lee, "BotMiner: Clustering Analysis of Network Traffic for Protocol- and Structure-Independent Botnet Detection," Proc. 17th USENIX Security Symp., July 2008.
[8]. G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee, "BotHunter: Detecting Malware Infection through Ids-Driven Dialog Correlation," Proc. 16th USENIX Security Symp., Aug. 2007.
[9]. G. Gu, J. Zhang, and W. Lee, "BotSniffer: Detecting Botnet Command and Control Channels in Network Traffic," Proc. 15th Ann. Network and Distributed System Security Symp. (NDSS '08), Feb. 2008.
[10]. N. Ianelli and A. Hackworth, "Botnets as a Vehicle for Online Crime," Proc. First Int'l Conf. Forensic Computer Science, 2006.