ISSN : 2231-0711

Detecting Spam Zombies

Vivek Fernando, Siddharth Nayak, Suraj Naik

Computer Department
Xavier Institute Of Engineering
Mumbai,India

Abstract: **One of the key security threats on the Internet are compromised machines which are often used to launch various security attacks such as spamming and spreading malware, DDoS, and identity theft. Spamming provides a key economic incentive for attackers to recruit a large number of compromised machines hence we focus on the detection of the compromised machines in a network that are involved in the spamming activities. These are commonly known as spam zombies. We have developed an effective detection system named SPOT which detects spam zombties by monitoring outgoing messages of a network. SPOT is designed based on a powerful statistical tool called Sequential Probability Ratio Test, which bounds false positive and false negative error rates.**

Keywords—Compromised machines, spam zombies, compromised machine detection algorithms.

I. Introduction

A major security challenge on the Internet is the existence of a large number of compromised machines. Such machines are being increasingly used to launch various security attacks such as spamming and spreading malware, DDoS, and identity theft [1][3][6]. Two natures of the compromised machines on the Internet—sheer volume and being widespread—render many existing security countermeasures less effective and hence makes defending attacks involving compromised machines extremely hard. On the other hand, identifying and cleaning compromised machines in a network remains a significant challenge for system administrators of networks of all sizes. In this paper, we 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.

In this paper, we will be developing 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) [1]. SPRT is a powerful statistical method. It can be used to test between two hypotheses i.e. whether a machine is compromised or the machine is not compromised, as the events, in this 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 without greater error rates. This means that the SPOT detection system can identify a compromised machine quickly.

II. Problem Formulation And Assumptions

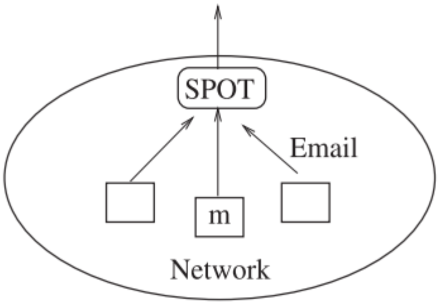

Fig. 1. Network model.

In this section, we will be formulating the spam zombie detection problem in a network [1][4]. Particularly, we will focus on the network model and assumptions we will be making in the detection problem. Fig. 1 illustrates the logical view of the network model. We assume that messages originated from machines inside the network will pass the deployed spam zombie detection system. This assumption can be achieved in a few different scenarios. For eg., the outgoing e-mail traffic can be replicated and redirected to the spam zombie detection system. A machine in the network is assumed to be either compromised or normal i.e. not compromised. In this paper, we only focus on the compromised machines that are used for spamming. Hence we use the term compromised machine to denote a spam zombie, and these two terms can be used interchangeably.

Let X_i for i = 1, 2, ... denote the successive observations of a random variable X which corresponds to the sequence of messages originating from machine m inside the network. Let $X_i = 1$ if message i from the machine is spam, and $X_i = 0$ otherwise. The detection system assumes that the behavior of a compromised machine is different than that of a normal machine in terms of the messages they send. Specifically, a compromised machine has a higher probability of generating a spam message than a normal machine. Formally let H_1 denote that machine m is compromised and H_0 that the machine is not compromised. The spam zombie detection problem can be formally stated as follows: When X_i arrives sequentially at the detection system, the system determines with a high probability if machine m has been compromised [1]. Once a decision is reached, the detection system reports the result, and further actions can be taken, e.g. the machine can be cleaned. We assume that a content-based spam filter is deployed at the detection system so that an outgoing message can be classified as either spam or non-spam.

Additionally, an assumption is made that IP address corresponds only to a unique machine and we ignore what impact dynamic IP addresses may have on the detection algorithms.

III. Spot Algorithm

SPOT [2] is designed based on the statistical tool SPRT which we discussed in the last section. In the context of detecting spam zombies in SPOT, we consider H_1 as a detection parameter and H_0 as a normality parameter i.e. H_1 is true if the concerned machine is compromised, and H_0 is true if it is not compromised. In addition, we assume $X_i = 1$ if the ith message from the concerned machine in the network is spam, and $X_i = 0$ otherwise. SPRT requires four configurable parameters from users, namely, the desired false positive probability α, the desired false negative probability β , the probability that a message is spam when H_1 is true (1), and the probability that a message is spam when H_0 is true (0).

Algorithm 1 outlines the steps of the algorithm. When an outgoing message arrives at the SPOT detection system, the sending machine's IP address is recorded, and the message is classified as either spam or non-spam by the spam filter. For each observed IP address, SPOT maintains the logarithm value of the corresponding probability ratio n, whose value is updated according to (3) as message n arrives from the IP address (lines 6 to 12 in Algorithm 1). Based on the relation between n and A and B, the algorithm determines if the corresponding machine is compromised, normal, or that a decision cannot be reached and additional observations may be needed (lines 13 to 21).

```
1: An outgoing message arrives at SPOT
2: Get IP address of sending machine m
3: // all following parameters specific to machine m
4: Let n be the message index
5: Let $X_n = 1$ if message is spam, $X_n = 0$ otherwise
6: if ($X_n$ == 1) then
7: // spam, Eq. 3
8: $\Lambda_n$ += $\ln \frac{\theta_1}{\theta_0}$
9: else
10: // nonspam
11: $\Lambda n$ += $\ln \frac{1-\theta_1}{1-\theta_0}$
12: end if
13: if ($\Lambda_n \geq B$) then
14: Machine m is compromised. Test terminates for m.
15: else if ($\Lambda_n \leq A$) then
16: Machine m is normal. Test is reset for m.
17: $\Lambda_n = 0$
18: Test continues with new observations
19: else
20: Test continues with an additional observation
21: end if
```

In the context of spam zombie detection, from the viewpoint of network monitoring, it is more important to identify the machines that are compromised than the machines that are not. After a machine is identified as being compromised (lines 13 and 14), it is added into a list of potentially compromised machines that system administrators can clean later. The message sending behavior of the machine is also recorded should further analysis be required. Before the machine is cleaned and removed from the list, the SPOT detection system does not need to monitor the message sending behavior of the machine any further.

IV. Parameters Used In Spot Algorithm

SPOT requires four user-defined parameters: α , β , 1, and 0. Here, we discuss how a user of the SPOT algorithm can configure these parameters, and how these parameters may affect the performance of SPOT. As discussed in the previous section, α and β are the desired false positive and false negative rates. These are normally small values in the range of 0.01 to 0.05, which users of SPOT can easily specify independent of the behavior of the compromised and normal machines in the network. The values of α and β will affect the cost of the SPOT algorithm i.e. the number of observations needed for the algorithm to reach a conclusion.

In general, a smaller value of α and β will require a larger number of observations for SPOT to reach a decision. Ideally, 1 and 0 should indicate the true probability of a message being spam from a compromised machine and a normal machine, respectively, but these are hard to obtain. A practical way to assign values to θ_1 and θ_0 is to use the detection rate and the false positive rate of the spam filter deployed along with the spam zombie detection system. Given that all the widely used spam filters have a high detection rate and low false positive rate [1][5], values of 1 and 0 assigned in this way should be very close to the true probabilities.

To get some intuitive understanding of the average number of required observations for SPRT to reach a decision. we assume the probability of a message being spam when H_1 is true to be 0.9 ($\theta_1 = 0.9$). When $\theta_0 = 0.2$, SPRT requires about three observations to detect that the machine is compromised given that the desired false positive rate is 0.01. As the behavior of a normal machine gets closer to that of compromised machine, i.e., θ_0 increases, a slightly higher number of observations are required for SPRT to reach a conclusion. As the behavior of a compromised machine gets closer to that of a normal machine, i.e., 1 decreases, a higher number of observations are required for SPRT to reach a detection. Also, as the desired false positive rate decreases, SPRT needs a higher number of observations to reach a conclusion. The same observation applies to the desired false negative rate. These observations illustrate the trade-offs between the desired performance of SPRT and the cost of the algorithm.

In the above discussion, we only show the average number of required observations when H_1 is true+

V. Conclusion

The proposed system detects the spam mails by monitoring the outgoing mails. The proposed system uses the Sequential Probability Ratio Test algorithm in order to detect the spam zombies. Depending upon the threshold limit given by the user this system minimizes the number of the required observation for detecting the spam zombies.

The proposed system also provides the blocking mechanism in which if the system is identified as the spam zombie then the system gets blocked so that it cannot send the spam messages further. Also the proposed system helps to recover the blocked system in case if the system was hacked by an attacker and was used as a spam zombie.

References

[1] Z.Duan, P.Chen, F.Sanchez, Y.Dong, M.Stephenson, J.M.Barker, ”Detecting Spam Zombies by Monitoring Outgoing messages”, IEEE transaction on dependable and secure computing,Vol 9,No.2,March/April 2012

[2] Ar.Arunachalam, V.Vevek, V.Yogeswaran, “Detecting Spam Zombies Using Spot Tool By Monitoring Outgoing Messages”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 4, April 2013

[3] T.Shinde, D.Kolpe, P.Kutal, A.Khairkar, “Spam zombie detection”, Proceedings of 4th IRF International Conference, Pune, 16th March-2014

[4] R.Ansari, Dr.V.N.R. Varman, “Spot protocol detecting outgoing spam messages”, IJCSMC, Vol. 2, Issue. 4, April 2013

[5] A.A.Chaudhari, R.S.Apare, “Spam Detection and Blocking Mechanism”, Proceedings of 10th IRAJ International Conferences, 25th December 2013

[6] http://en.wikipedia.org/wiki/Zombie_(computer_science) accessed on 1 st October