Detection of Compromised Machines by Monitoring Outgoing Messages

[1]S.Yuvaraj
M.E. Computer and Communication Engineering
SNS College of Technology
Coimbatore, India
Connect2yuvanice@gmail.com

[2]M.Suguna
Associate Professor, Department of IT
S.N.S College of Technology,
Coimbatore, India
suguna.marappan@gmail.com

[3] Dr.D.Sharmila, PhD.
Professor & Head EIE Department,
Bannari Amman Institute of Technology,
Sathyamangalam.
sharmiramesh@rediffmail.com

Abstract— **In the internet nowadays compromised machines are the key security threats which are often used to spread various security attacks. These security attacks include spamming, spreading malware, DDoS and identity theft in which spamming provides large number of compromised machines. The process involves the detection of compromised machines involved in spamming activities named as spam zombies. The existing system uses the effective spam zombie detection algorithm named SPOT which detect by supervising outgoing messages of a network which only detects spam content present in the message. The proposed system is designed in a novel method by using semantic aware statistical algorithm (SAS) which improve the performance of SPOT by detecting virus/worm attachment in a message. The SAS will use a data flow analysis technique which processes when packets in suspicious flow pool to ignore the non-critical bytes. Then state-transition-graph based signatures are generated by refining the data's using Hidden Markov Model (HMM). By using this SAS algorithm the worm signatures are automatically generated and also this is first work of generating worm signatures by the combination of semantic analysis with statistical analysis. The evaluation shows that proposed system SAS is efficient and effective in detecting compromised machines in a network when compared to existing SPOT system.**

Keywords: spot, sas, spam zombies

Introduction

I. Introduction

E-mail is an efficient form of communication that has become widely adopted by both individuals and organizations. Today, more and more people are relying on e-mail to connect them with their friends, family, colleagues, customers and business partners. Unfortunately, as e-mail usage has evolved, so too has its threats. For example, according to Message Labs, spam accounted for 67% of all e-mail traffic in October 2006, up from 57% the same time a year before. In particular, it is a now a non-trivial task to find legitimate e-mails in an e-mail inbox cluttered with spam. There are also published reports, which suggest that spam has resulted in lost opportunity costs of several billions of dollars because of organizations that have lost faith in the security industry's ability to fight this problem.

An effective spam zombie detection system is developed which is 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. SPOT has bounded false positive and false negative error rates. In addition to the spam detection to detect the virus attachments proposed a SAS algorithm which detect more compromised machines in a network.

The objective of the project is

- To introduce a new approach to spam filtering by first investigating the SMTP transactional behavior differences between legitimate mail sources and illegitimate mail sources such as spam zombies.
- Next, the objective is to introduce a new machine learning filtering technique that can continuously learn efficiently in real-time and also identify differences in mail sending behavior between a legitimate and illegitimate-mail source.
- The existing detection based techniques is used to learn the mechanism to produce a solution designed to identify illegitimate mail servers, such as spam zombies, and filter mail from said sources.
- The final objective is to conduct experimentation to prove that the new filtering technique provides an advantage over existing detection based filtering.

II. RELATED WORK

Spam zombie detection method is a new real-time machine learning based spam filtering technique that uses the Semantic Aware Signature(SAS) to learn Simple Mail Transfer Protocol(SMTP) transactional behaviour of spam zombies. This technique was implemented as a single layer perceptron plug-in that learns the behavior of spam zombies and makes decisions as to whether an incoming source is likely to send spam or not. This also creates and integrates a reverse Domain Name Service(DNS) module into this design to prevent spammers from forging legitimate domains and making it difficult for them to overcome the proposed technique.

This technique is deployed on a large corporate network, in which the technique can able to demonstrate the SAS list. In particular, the SAS technique had successfully detected more number of compromised machines in a network.

III. SYSTEM IMPLEMENTATION

The virus/worms can be locally or remotely injected using the HTTP protocol. To generate high quality signatures of such worms, proposed SAS, a novel Semantics Aware Statistical algorithm that generates semantic aware signatures automatically. When SAS processes packets in the suspicious flow pool, it uses data flow analysis techniques to remove non-critical bytes irrelevant to the semantics of the worm code. Then apply a HMM to the refined data to generate STG based signatures. Since modern polymorphic engines can completely randomize both the encrypted shell code and the decryptor , which uses a probability STG signature to defeat the absence of syntactic invariants. STG, as a probability signature, can adaptively learn token changes in different packets, correlate token distributions with states, and clearly express the dependence among tokens in packet payloads. The experiments show that the proposed technique exhibits good performance with low false positives and false negatives, especially when attackers can indistinguishably inject noisy bytes to mislead the signature extractor. SAS places itself between the pattern-based SAS for Polymorphic Worm Detection and the semantic-derived detection methods, by balancing between security and the signature matching speed.
As a semantic-based technique, SAS is more robust than most pattern-based signatures, sacrificing a little speed in signature matching.

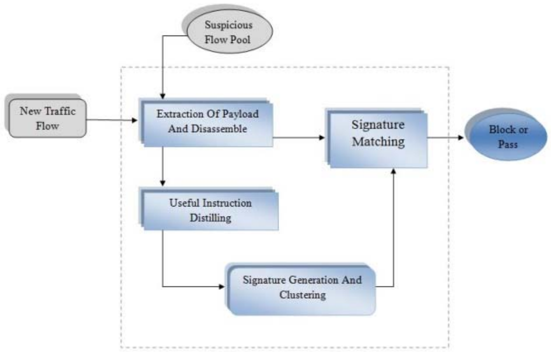

Suspicious Flow Pool
New Traffic Flow
Extraction Of Payload And Disassemble
Signature Matching
Block or Pass
Useful Instruction Distilling
Signature Generation And Clustering

Fig 3.1: Structure Of SAS

System Overview

To describe the framework of the approach. it consists of two phases, semantic-aware signature extraction phase and semantic-aware signature matching phase. The signature extraction phase consists of the phases such as payload extraction, payload disassembly, useful instruction distilling, clustering, and signature generation. The five phase is comprised into two modules namely payload extraction and signature matching module in which the Payload extraction module extracts the payload which contains malicious intent, from a flow. For example, in a HTTP request message, a malicious payload will exist either in Request-URI or in the Request-Body of the whole flow. These two parts are extracted from the HTTP flows for further analysis. Then the disassembly module disassembles an input byte sequence and it finds consecutive instructions in the input sequence which will be used to generate disassembled instruction sequence as output. If an instruction is valid means it should have at least one execution path from the entry point. Next the useful instruction distilling module extracts useful instructions from the disassemble input sequences. Useless instructions are identified and pruned by control flow and data flow analysis techniques. Payload clustering module will be used to cluster the payloads containing similar set of useful instructions to generate the signature. Signature generation module will generate STG based signatures from the payload clusters which will be used for checking the incoming packets. The Signature matching module starts detecting worm packets by matching STG signatures against input packets.

IV. MODULES

Compose Mail Process

The form is used to compose mail to receiver address. Here the form stores the details about the content, the attachment, the receiver’s mail address, the subject and the body. This will helps the spam detection server to collect the details of the sender’s details and are maintained in a separate list for further clarification. These list may stored in the spam detecting system which will uses for collecting the details about the senders details such as IP address and the details about the amount of data send etc.

Filter spam and detect spam

Once the mail has been composed the spam detecting system will involved in the process separating the doubted machines. These separations may takes place by keeping the frequency of the sending content not more than that frequency to be crossed. Once the frequency has been crossed the spam detecting system will indicate that the machine may contains any spam content and the doubted machines are separated and allowed to store it in a list view control.

IP Capture

With the help of the compose mail process form, the multiple recipient sender IP address are captured to separate the compromised machines and the normal machines.

Extraction of Payloads And Payload Disassemble

The extraction of payloads from the suspicious flow is the process of splitting /extracting the areas where the intruders will inject the threats. For example, in the http request the noisy payloads will occur only in the Request URI and in the Request body. Those places must be taken for further consideration. The disassemble phase is that the process of disassemble the input byte sequences.

It will first take the consecutive instruction and it will generate the output as disassemble instructions. If the instruction should be a disassemble instruction it should have an entry point and an output path. This disassemble instruction phase will check it and prune those codes which are not satisfy the condition and let the process to continue for next phase.

Useful Instruction Distilling and Clustering

The useful instruction distilling phase will take the disassembled instructions and find the useful instructions and useless instructions. These useful instructions are pruned with the help of three data flow anomalies named define-define, define-undefine and undefine reference. The define-define anomaly is that the process of identifying the threats by checking whether the variable has assigned a value twice. The define-undefine anomaly will check that the defined variables is later set by an undefined variables. In the undefined-reference anomaly the variable will be referenced before it is ever assigned by a value. From the above dataflow anomalies the useful instruction distilling phase will split the useful instructions separately from the flow. These useful instructions and useless instructions are separated and clustered separately.

Signature Generation

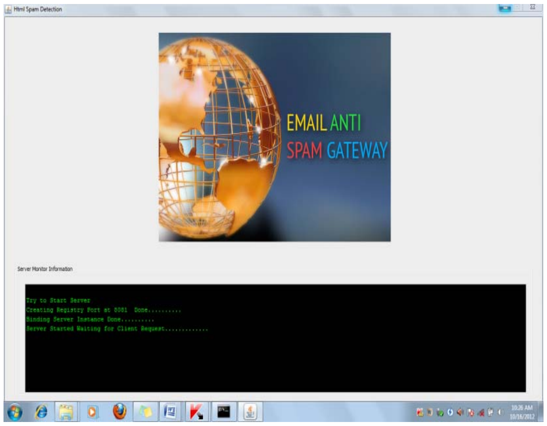
The signature generation module is used to generate the STG based signatures with the useful instructions distilled from useful instruction distilling phase. Then these useful instructions are clustered and STG based signatures are generated according to the signatures of similar types.

Signature Matching

These STG based signatures are used to check the live network packets. To check it the new packets that are send through mail whose payload length may be of n-byte. Then the detecting detector will checks the n-byte payload either it will match any of the existing signatures or not. If the n-byte payload matches any of the signature means it will be considered as the worm packet. If it does not match the existing signature means it will be allowed to check in the detecting system.

V. RESULTS

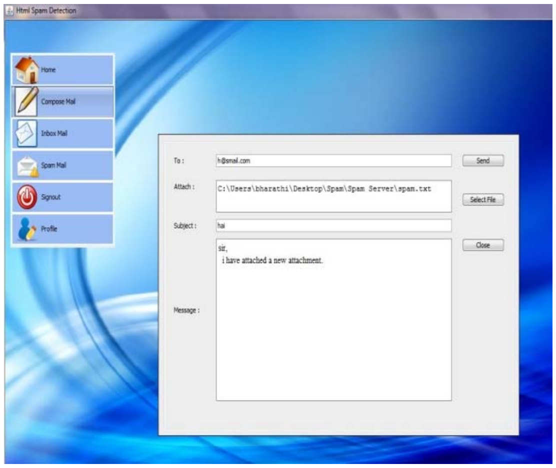
An effective spam zombie detection system is developed which is named SPOT by monitoring outgoing messages in a network. SPOT was designed based on a simple and powerful statistical tool named Sequential Probability Ratio Test to detect the compromised machines that are involved in the spamming activities. SPOT has bounded false positive and false negative error rates. It also minimizes the number of required observations to detect a spam zombie which has already done in existing system. In addition to it the virus/worm attachment is detected by Semantics Aware Statistical algorithm which automatically detects the virus/worm attached packets in a network. By using the SAS algorithm the virus/worm attached packets can be easily detected using the dataflow anamolies. The SAS algorithm will helps to detect more number of compromised machines in a network when comparing to the SPOT algorithm. To show the existing system will detect more number of compromised machines, live review results are made by taking the send mails from any browsing centre. The comparison were made to show that the SAS algorithm have detected more number of compromised machines compared to SPOT.



```
Try to Start Server
Creating Registry Port at 8081  Done..........
Binding Server Instance Done..........
Server Started Waiting for Client Request............
```

Figure 4.1 Server

The above Figure 4.1 represents the server in which all the operation performed will be displayed such as user account creation, compose mail, received mail etc. Even the operation such as attachment details will also be denoted in the server during the compose mail process takes place. These list may stored in the spam detecting system which will uses it for collecting the details about the senders details such as IP address and the details about the amount of data send.



To : h@smail.com

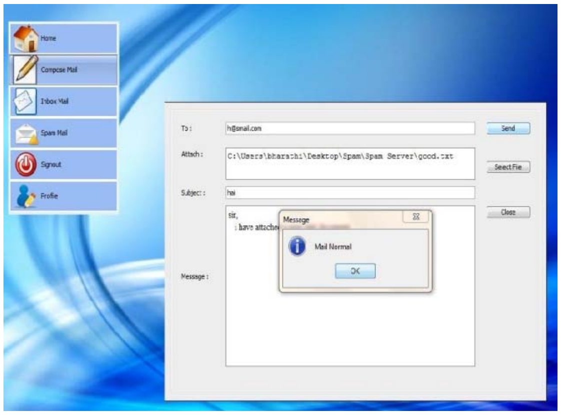
Attach : C:\Users\bharathi\Desktop\Spam\Spam Server\spam.txt

Subject : hai

Message : sir,
 i have attached a new attachment.

Figure 4.2 Compose Mail Process

The above Figure 4.2 may contains login form in which it is used to login using their username and password which will verify for the account details and will login into their account. The new users can also able to have an account by signup process in which the users can have an new account. As the login happen the entry into the home page appears. The process such as compose mail takes place in which the mail has been composed. Once the mail composed the spam detecting system will indicate that it is spam mail or normal mail.



To : h@smail.com

Attach : C:\Users\bharathi\Desktop\Spam\Spam Server\good.txt

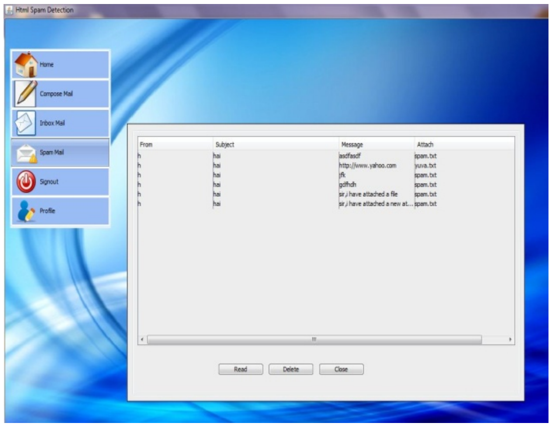
Subject : hai

Message : sir,
i have attache

Mail Normal

Figure 4.3 Normal Mail

The front page may contains login form in which we used to login using their username and password which will verify for the account details and will login into their account. The new users can also able to have an account by signup process in which the users can have an new account. As the login happen the entry into the home page appears. The process such as compose mail takes place in which the mail should be composed. Once the mail composed the spam detecting system will indicate that it is spam mail or normal mail. In the above process it shows that the composed mail is normal and donot contain any spam attachment.



From	Subject	Message	Attach
h	hai	asdfasdf	spam.txt
h	hai	http://www.yahoo.com	yuva.txt
h	hai	jfk	spam.txt
h	hai	gdfhdh	spam.txt
h	hai	sir,i have attached a file	spam.txt
h	hai	sir,i have attached a new at...	spam.txt

Figure 4.4 Receivers Spam Mail Box

The detected affected mails are separated and send for the further scheme which shows the details of the affected mails list. These mails are collected separately in form shows the details about the composed spam mails which are stored in the receivers spam mail box which collects all the spam mails and their IP addresses for further process to be performed.

ACKNOWLEDGMENT *(HEADING 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression, “One of us (R. B. G.) thanks . . .” Instead, try “R. B. G. thanks”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

REFERENCES

[1] Duncan, C., Jacky, H., Kevin, M., & Joel, S: Catching Spam Before It Arrives: Domain Specific Dynamic Blacklists, Proceedings of the 2006 Australasian Workshops on Grid Computing and E-research, Hobart, Tasmania, Australia,. pp. 193-202. (2006)

[2] The Spam Problem and the Brightmail Filtering Engine Technical White Paper, Brightmail Anti-Spam Enterprise Edition Version 5.5.

[3] Liang, L.: A comparison of email filtering techniques, Master Thesis, Dalhousie University. (2005)

[4] Rejeb, J., Le, T. T., &Anand, N.:High Speed and Reliable Anti-Spam Filter , Proceedings of IEEE International Conference on Software Engineering Advances (ICSEA2006), Tahiti, French Polynesia, October 29 - November 3, 2006, (ISBN 0-7695-2703-5) pp. 66-66. (2006)

[5] Delany, S. J., & Derek, B.: Catching the Drift: Using Feature Free Case-based Reasoning for Spam Filtering, In: R Weber & M. Richter (eds.) Case-Based Reasoning Research and Development, Procs of the 7th International Conference on Case-based Reasoning (ICCBR 2007), pp. 314-328. (2007)

[6] Yong Tang and Shigang Chen: An Automated Signature-Based Approach against Polymorphic Internet Worms, ieee transactions on parallel and distributed systems, vol. 18, no. 7, july .(2007)

[7] Kumar Simkhada, TarikTaleb, Yuji Waizumi, Abbas Jamalipour, Nei Kato, and Yoshiaki Nemoto: An Efficient Signature-Based Approach for Automatic Detection of Internet Worms over Large-Scale Networks.

[8] Zhichun Li MananSanghi Yan Chen Ming-Yang Kao Brian Chavez:Hamsa_: Fast Signature Generation for Zero-day Polymorphic Worms with Provable Attack Resilience, Northwestern University Evanston, IL 60208, USA flizc,manan,ychen,kao,cowboyg@cs.northwestern.edu.

[9] James Newsome, Brad Karp, Dawn Song: Polygraph: Automatically Generating Signatures for Polymorphic Worms

[10] Deguang Kong, Yoon-Chan Jhi, Qihe Pan, Sencun Zhu, Peng Liu, and Hongsheng Xi: SAS: Semantics Aware Signature Generation for PolymorphicWorm Detection

[11] J.P. John, A. Moshchuk, S.D. Gribble, and A. Krishnamurthy, “Studying Spamming Botnets Using Botlab,” Proc. Sixth Symp. Networked Systems Design and Implementation (NSDI ’09), Apr. 2009.

[12] J. Jung, V. Paxson, A. Berger, and H. Balakrishnan, “Fast Portscan Detection Using Sequential Hypothesis Testing,” Proc. IEEE Symp.Security and Privacy, May 2004.

[13] J. Klensin, “Simple Mail Transfer Protocol,” IETF RFC 2821, Apr. 2001.

[14] J. Markoff, “Russian Gang Hijacking PCs in Vast Scheme,” The New York Times,

http://www.nytimes.com/2008/08/06/technology/06hack.html, Aug. 2008.

[15] P. Wood et al., "MessageLabs Intelligence: 2010 Annual Security Report," 2010.

AUTHORS PROFILE

Mr. S.Yuvaraj doing final year M.E. Computer and Communication Engineering in SNS College of Technology, Coimbatore from Anna University-Chennai.