ISSN: 2321-7782 (Online)

Volume 2, Issue 9, September 2014

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Prototyping and Detecting Disguise Worm

Anil Kumar Nampalli[1]
M.Tech(CNIS), Dept. of IT
Sreenidhi Institute of Science and Technology
Yamnampet, Ghatkesar, Rangareddy
Telangana – India

Vamsee Krishna.K[2]
Associate Professor, Dept. of IT
Sreenidhi Institute of Science and Technology
Yamnampet, Ghatkesar, Rangareddy
Telangana – India

Samson.Ch[3]
Associate professor & Associate Head, Dept. of IT
Sreenidhi Institute of Science and Technology
Yamnampet, Ghatkesar, Rangareddy
Telangana – India

Abstract: In this paper we propose an approach to Prototype and Detection of Disguise Worm. The Disguise Worm (D-Worm) is different from traditional worms because of its ability to intelligently control its scan traffic volume over time. We design analyze characteristics of the D-Worm and conduct a comprehensive examination on its traffic. We observe that it will scan all its neighbour's IP addresses first and then selects alive addresses. It will randomly select an IP address among alive addresses and ping the IP address. If ping is successful then D-Worm can execute on remote machine. When worm is executing on remote machine, it will create a worm file in all existing folders and can send an acknowledgement to the master so that master can have an idea regarding particular host.

Keywords: Prototyping; Grabbing; Disguise; Worm; Camouflaging.

I. Introduction

Worm is a program that makes copies of itself in different locations on a computer system. Dynamic worms propagate in an automated fashion and continuously compromise computers on the cyberspace. The extension of the worm is based on tapping exposures of hosts. This worm similar to Code-Red[1], Slammer[2]. Here dynamic worms are used to taint a multitude hosts and enroll as bots or zombies and act as botnets [3] perform (a) Launch massive Distributed Denial-of-Service (*DDoS*) attacks (b) Access secret matter that can be abused through large scale traffic sniffing, key logging, identity theft etc, (c) Destroy data that has a high monetary value [4]. There is evidence showing that infected computers are being rented out as "Botnets" for creating an entire black-market industry for renting and an aging "owned" computers, leading to economic incentives for attackers. Earlier worm grabbing system assumes that each infected host keeps scanning the IP address and propagates itself at the highest possible speed. In particular, 'stealth' is one attack strategy used by a recently-discovered dynamic worm called the "self-stopping" worm [5] circumvent detection by hibernating with a pre-determined period.

In the present paper, we propose the prototype of smart worm called Disguise Worm. The D-Worm has a self-propagating nature and can disguise any noticeable trends in the number of infected computers over time. Here scan traffic handling involve steady increase, followed by a decrease in the scan traffic volume, such that the changes do not manifest as any trends in the time domain such that the scan traffic volume does not cross thresholds that could reveal the D-Worm extension. Based on the above observation, we adopt frequency domain analysis techniques and develop a detection scheme against wide-spreading of the D-Worm.

This paper is organized as follows: In Section II, we present the related work. In Section III, we describe the proposed method. The Performance Evaluation and Results are provided in Section IV. Finally draw conclusions in Section V.

II. Related Work

The basic form of dynamic worms can be categorized as having the Pure Random Scan (PRS) nature. In the PRS form, a worm-infected computer continuously scans a set of random Internet IP addresses to find new vulnerable computers. Worms use hit list to infect previously identified vulnerable computers at the earlier stage of extension to increase its propagation effectively and using network topology and routing information it can identify dynamic computers instead of randomly scanning IP addresses. The Disguise Worm (D-Worm) studied in this paper aims to elude the detection by the worm defense system during worm extension. Polymorphic worms are able to change their binary representation or signature as part of their extension process. This can be achieved with self-encryption mechanisms or semantics preserving code manipulation techniques. The D-Worm also shares some similarity with stealthy port-scan attacks. Such attacks try to find out available services in a particular system. Generally to avoid D-worm detection decreasing the port scan rate, hiding the origin of attackers, etc. Due to the nature of self-extension, the D-Worm must use more complex mechanisms to manipulate the scan traffic volume over time in order to avoid detection. Generally Worm detection classified into two categories: “host-based” detection and “network-based” detection. In Host-based detection system detect worms by monitoring, collecting, and analyzing worm behaviors on end-hosts. Since worms are malicious programs that execute on these computers, analyzing the behavior of worm executables plays an important role in host based detection systems. In network-based detection systems detect worms primarily by monitoring, collecting, and analyzing the scan traffic (messages to identify vulnerable computers) generated by worm attacks. The widely adopted worm detection framework consists of multiple distributed monitors and a worm detection center that controls the former. Each monitor passively records irregular port-scan traffic, such as connection attempts to a range of void IP addresses (IP addresses not being used) and restricted service ports. Periodically, the monitors send traffic logs to the detection center. The detection center analyzes the traffic logs and determines whether or not there are suspicious scans to restricted ports or to invalid IP addresses. Network-based detection schemes commonly analyze the collected scanning traffic data by applying certain decision rules for detecting the worm propagation. Venkataraman *et al.* and Wu *et al.* in [6], [7] says that analyze the statistics of scan traffic volume. Zou *et al.* presented a trend-based detection scheme to examine the exponential increase pattern of scan traffic [8]. Payload-based worm signature detection [9],[10]. Regardless of, the different methods described above, we believe that detecting widely scanning anomaly behavior continues to be a useful weapon against worms, and that in practice multifaceted defense has advantages.

III. Propose Method

The D-Worm disguises its extension by controlling scan traffic volume during its extension. The simplest way to manipulate scan traffic volume is to randomly change the number of worm instances performing port-scans. D-worm uses a closed loop control for regulating the propagation speed based on the feedback propagation status. In order to effectively evade detection, the overall scan traffic for the D-Worm should be comparatively slow and variant enough to not show any notable increasing trends over time. On the other hand, a very slow propagation of the D-Worm is also not desirable, since it delays rapid infection damage to the LAN. Hence, the D-Worm needs to adjust its propagation so that it is neither too fast to be easily detected, nor too slow to delay rapid damage on the LAN. To regulate the D-Worm scan traffic volume, we introduce a control parameter called *attack chance* P(t) for each worm-infected computer. P(t) is the probability that a D-Worm instance participates in the worm propagation (i.e., scans and infects other computers) at time t. Our D-Worm model with the control parameter P(t) is generic. P(t) = 1 represents the cases for traditional worms, where all worm instances dynamically participate in the propagation. For the D-Worm, P(t) needs not be a constant value and can be set as a time varying function. In order to achieve its Disguise behavior, the D-Worm needs to obtain an appropriate P(t) to manipulate its scan traffic. Specifically, the D-Worm will regulate its overall scan traffic volume such that it is similar to non-worm scan traffic in terms of the scan traffic volume over time and does not exhibit any notable trends, such as an exponentially increasing pattern or any mono-increasing pattern even when the number of infected hosts increases (exponentially) over time. The average value of the overall scan traffic volume is sufficient to make the D-Worm propagate fast enough to cause rapid damage on the LAN.

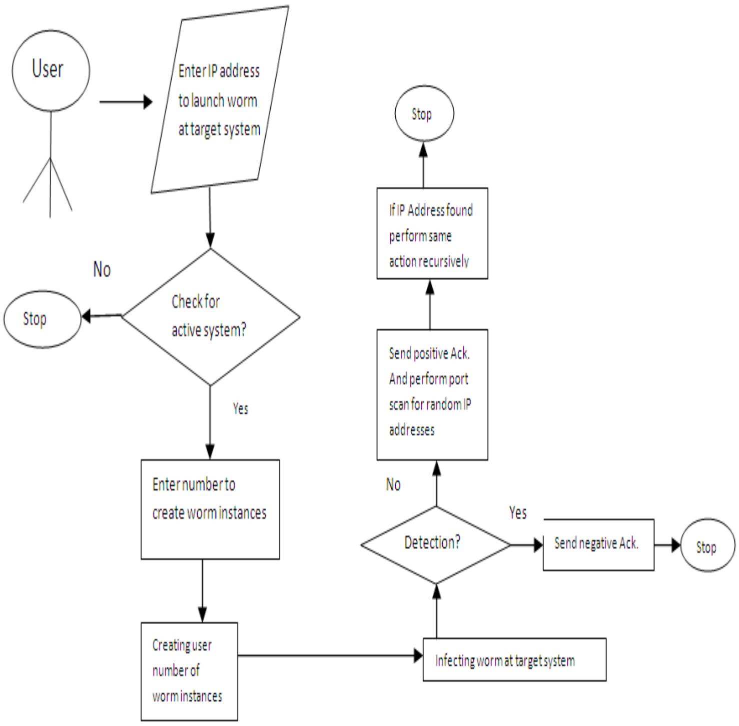
User
Enter IP address to launch worm at target system
Stop
If IP Address found perform same action recursively
No
Stop
Check for active system?
Send positive Ack. And perform port scan for random IP addresses
Yes
Enter number to create worm instances
No
Yes
Detection?
Send negative Ack.
Stop
Creating user number of worm instances
Infecting worm at target system

Fig. 1 Schematic Diagram

IV. PERFORMANCE EVALUATION AND RESULTS

The Epidemic model for a finite population of D-Worms can be expressed as given bellow

$$\frac{dR(t)}{dt} = \beta \cdot R(t) \cdot [B - R(t)],\text{----}(1)$$

R (t)	the number of infected Folders at time t;
B (= I· $C_1 \cdot C_2$)	The number of vulnerable Folders on remote machine;
I	total number of IP addresses on the Intranet;
C1	the ratio of the total number of hosts on the Intranet over T
C2	the ratio of total number of *vulnerable* hosts on the LAN over the total number of hosts on the intranet;
β = F/V	pair wise infection rate
F	Scan rate defined as the number of scans that an infected host can launch in a given time interval.
We assume that at t = 0, there are R (0) computers being initially infected and B−R(0) computers being susceptible to further worm infection.	

Table 1: Description of epidemic model

The D-Worm has a different propagation model compared to traditional PRS worms because of its P (t) parameter. Consequently, Formula (1) needs to be rewritten as,

$$\frac{dR(T)}{dt} == \beta \cdot R(t) \cdot P(t) \cdot [N - R(t)]$$

P(t) - The attack probability that a worm-infected computer participates in worm propagation at time t.

Initially worm scans the all its neighbor's IP addresses and among them it will select alive addresses then among them it will randomly select an IP address then it will ping the IP address. If ping is success then D-Worm can execute on remote machine. When worm is executing on remote machine then it will create a worm file in all existing folders then after it will send an acknowledgement to the master so that master can have idea regarding particular host.

In order to detect we have followed MD5 algorithm. Initially detection system converts existing file into their binary form and perform bitwise and operation with a byte which consists of only one's then after it will add 256 i.e., 2^8, then it will consider only first 16 digits numbers and then after it will ignore first MSB(Most Significant Bit) and then it will convert the value into their respective UNICODE value.

Now detection system has enrolled the pattern of this worm into their database then the detection system will scan existing file sequentially and generate their respective MD5 hashing pattern, if the generated hashing pattern matches the enrolled hashed pattern then the detection system deletes the worm.

From implementation we noticed following things:

Initially D-Worm scans all the IP-addresses of its neighboring hosts if any then create the worm instances and perform their actions at remote system recursively. To detect the D-worm we have analyze their worm signatures using this we can grabbing the worm.

The D-Worm can adjust its propagation so that it is neither too fast to be easily detected, nor too slow to delay rapid damage on the LAN.

This worm can also help to launch action remotely on particular situations given by master. This worm also helps to reveal secret information of remote system in which it has been being executing.

V. Conclusion

In present analysis, we have modeled a worm in stand-alone system then implemented it in Local Area Network successfully among three systems. Here we have noticed couple of things as follows: a) the D-Worm successfully disguises its propagation in the time domain, b) camouflaging nature inevitably manifests as a distinct pattern in the frequency domain. For detection of worm we have focused on the Message Digest 5 (MD5) Hashing technique to catch the worm. Using this worm we can have remote file system information in which worm has been being propagated.

This paper lays the foundation for ongoing studies of "smart" worms that intelligently adapt their propagation patterns to reduce the effectiveness of countermeasures.

We can model a smart worm so that it will gather remote file system's information and transfer it to the master and also it will transfer files with them. We hope to develop a full prototype of such smart worm as our future work.

References

1. D. Moore, C. Shannon, and J. Brown, "Code-red: a case study on the spread and victims of an internet worm," in Proceedings of the 2-th Internet Measurement Workshop (IMW), Marseille, France, November 2002.
2. D. Moore, V. Paxson, and S. Savage, "Inside the slammer worm," in IEEE Magazine of Security and Privacy, July 2003.
3. P. R. Roberts, Zotob Arrest Breaks Credit Card FraudRing,http://www.eweek.com/article2/0,1895,1854162,00.asp.
4. Worm.ExploreZip,http://www.symantec.com/avcenter/venc/data/worm.explore.zip.html.
5. J. Ma, G. M. Voelker, and S. Savage, "Self-stopping worms," in Proceedingsof the ACM Workshop on Rapid Malcode (WORM), WashingtonD.C, November 2005.
6. S. Venkataraman, D. Song, P. Gibbons, and A. Blum, "New streaming algorithms for superspreader detection," in Proceedings of the 12-thIEEE Network and Distributed Systems Security Symposium (NDSS), San Diego, CA, Febrary 2005.
7. J. Wu, S. Vangala, and L. X. Gao, "An effective architecture and algorithm for detecting worms with various scan techniques," inProceedings of the 11-th IEEE Network and Distributed System Security Symposium (NDSS), San Diego, CA, Febrary 2004.
8. C. C. Zou, W. Gong, and D. Towsley, "Code-red worm propagation modeling and analysis," in Proceedings of the 9-th ACM Conference on Computer and Communication Security (CCS), Washington DC, November 2002.
9. R. Perdisci, O. Kolesnikov, P. Fogla, M. Sharif, and W. Lee, "Polymorphic blending attacks," in Proceedings of the 15-th USENIX Security Symposium (SECURITY), Vancouver, B.C., August 2006.
10. H. Kim and B. Karp, "Autograph: Toward automated, distributed worm signature detection," in Proceedings of the 13-th USENIX Security Symposium (SECURITY), San Diego, CA, August 2004.

AUTHOR(S) PROFILE

Anil Kumar Nampalli, pursuing the M.Tech degree in Computer Networks and Information Security at Sreenidhi Institute of Science and Technology and B.Tech degree in Information Technology from Jawaharlal Nehru Technological University-Hyderabad in 2012.

K.Vamsee Krishna is an Associate Professor in the Department of Information Technology in SNIST, Hyderabad. He received his BE in CSE from Madras University in 2002 and his ME in CSE from Anna University in 2006. Since 2006 he has taught courses in Computer Networks, Client/Server Computing, Network Security, Network Management Systems, etc. His research areas of interests include Security & Privacy, Network Security.

Mr. Ch.Samson obtained his Diploma from Govt Polytechnic, Hyderabad in 1994, B. E. from Osmania University in 1998 and M. E from SRTM University in 2000. He submitted Ph.D. thesis to JNTUH in May 2014. He published 18 research papers in international journals and two papers in national conferences. He is currently working as Associate Professor & Associate Head in the Dept. of Information Technology (IT), SNIST. His research interests are Image Processing, Image Cryptography and Network Security.