Efficiently Detecting the Active Worm

[1]Yugandhar Tirumani, [2]Gumpula Raju
[1,2]Dept. of CSE, Akula Gopayya college of Engineering and Technology, Tadepalligudem, AP, India

Abstract
Active worm's major security threats to the Internet. This is due to the ability of active worms to propagate in an automated fashion as they continuously compromise computers on the Internet. Active worms evolve during their propagation, and thus, pose great challenges to defend against them. In this paper, we investigate a new class of active worms, referred to as Camouflaging Worm (C-Worm in short). The C-Worm is different from traditional worms because of its ability to intelligently manipulate its scan traffic volume over time.Thereby, the C-Worm camouflages its propagation from existing worm detection systems based on analyzing the propagation traffic generated by worms. We analyze characteristics of the C-Worm and conduct a comprehensive comparison between its traffic and nonworm traffic (background traffic). We observe that these two types of traffic are barely distinguishable in the time domain. However, their distinction is clear in the frequency domain, due to the recurring manipulative nature of the C-Worm. Motivated by our observations, the design a novel spectrum-based scheme to detect the C-Worm. Our scheme uses the Power Spectral Density (PSD) distribution of the scan traffic volume and its corresponding Spectral Flatness Measure (SFM) to distinguish the C-Worm traffic from background traffic. Using a comprehensive set of detection metrics and real-world traces as background traffic, we conduct extensive performance evaluations on our proposed spectrum-based detection scheme. The performance data clearly demonstrates that our scheme can effectively detect the C-Worm propagation. Furthermore, we show the generality of our spectrum-based scheme in effectively detecting not only the C-Worm, but traditional worms as well. In the existing system, traditional worms are more threats to the internet and also would produce lot of overall network traffic. It is very easy to identify the worm using traditional worm detection as the overall network traffic is increased. In the proposed model, camouflage worm is modeled and detection using spectrum based approach. Worm targets only vulnerable node so that overall traffic level is not increased. Spectrum based approach which is used to kill the C-worm. Modifications are mode in designing a worm which is used to increase the CPU load in the system, and also compared with traffic level of a application initiation and the C-worm. This process makes very clear process of execution.

Keywords
Worm, Camouflage, Anomaly Detection

I. Introduction

In recent years, Internet worms have proliferated because of hardware and soft- ware mono-cultures, which make it possible to exploit a single vulnerability to compromise a large number of hosts.

Most Internet worm follow a scan/compromise/ -replicate pattern of behavior, where a worm instance first identifies possible victims, then exploits one or more vulnerabilities to compromise a host, and finally replicates there. These actions are performed through network connections and, therefore, Network Intrusion Detection Systems (NIDSs) have been proposed by the security community as mechanisms for detecting and responding to worm activity

However, as worms became more sophisticated and efficient in spreading across networks, it became clear that countermeasures based on human reaction time were not feasible. In response, the research community focused on devising a number of techniques to automatically detect and contain worm outbreaks.

In particular, the need for the timely generation of worm detection signatures motivated the development of systems that analyze the contents of network streams to automatically derive worm signatures. These systems, such as Earlybird [1] and Autograph [6], implement a content sifting approach, which is based on two observations. The first observation is that some portion of the binary representation of a worm is invariant; the second one is that the spreading dynamics of a worm is different from the behavior of a benign Internet application. That is, these worm detection systems rely on the fact that it is rare to observe the same byte string recurring within network streams exchanged between many sources and many destinations. The experimental evaluation of these systems showed that these assumptions hold for existing Internet worms.

A limitation of the systems based on content sifting is the fact that strings of a significant length that belong to different network streams are required to match (for example, byte strings with a length of 40 bytes are used in [9]). Unfortunately, the next generation of Internet worms is likely to be polymorphic. Polymorphic worms are able to change their binary representation as part of the spreading process. This can be achieved by using self-encryption mechanisms or semantics-preserving code manipulation techniques. As a consequence, copies of a polymorphic worm might no longer share a common invariant substring of sufficient length and the existing systems will not recognize the network streams containing the worm copies as the manifestation of a worm outbreak.

Active worms have been a persistent security threat on the Internet since the Morris worm arose in 1988. The Code Red and Nimda worms infected hundreds of thousands of systems, and cost both the public and private sectors millions of dollars. Active worms propagate by infecting computer systems and by using infected computers to spread the worms in an automated fashion. Staniford et al. show that active worms can potentially spread across the Internet within seconds [5]. It is therefore of great importance to characterize and monitor the spread of active worms, and be able to derive methods to effectively defend our systems against them.

About ten years ago, Kephart and White presented the Epidemiological model to understand and control the prevalence of viruses [6]. This model is based on biological epidemiology and uses nonlinear differential equations to provide a qualitative understanding of virus spreading. White pointed out, however, that the "mystery" of the Epidemiological model is that it fails to predict that virtually most viruses will be slow in global prevalence.

A. Spread in Active Warm

When an active worm is fired into the Internet, it simultaneously scans many machines in an attempt to find a vulnerable machine to infect. When it finally finds its prey, it sends out a probe to infect the target. If successful, a copy of this worm is transferred to this new host. This new host then begins running the worm and tries to infect other machines. When an invulnerable machine or an unused IP address is reached, the worm poses no threat.

During the worm's spreading process, some machines might stop functioning properly, forcing the users to reboot these computers or at least kill some of the processes that may have been exploited by the worm. Then these infected machines become vulnerable machines again, and are still inclined to further infection. When the worm is detected, people will try to slow it down or stop it. A patch, which repairs the security hole of the machines, is used to defend against worms. When an infected or vulnerable machine is patched, it becomes an invulnerable machine.

To speed up the spread of active worms, Weaver presented the "hitlist" idea [10]. Long before an attacker releases the worm, he/she gathers a list of potentially vulnerable machines with good network connections. After the worm has been fired onto an initial machine on this list, it begins scanning down the list. Hence, the worm will first start infecting the machines on this list. Once this list has been exhausted, the worm will then start infecting other vulnerable machines. The machines on this list are referred to as the "hitlist". After the worm infects the hitlist rapidly, it uses these infected machines as "stepping stones" to search for other vulnerable machines.

II. Related Works

Worms are a common phenomenon in today's Internet, and despite significant research effort over the last years, no general and effective countermeasures have been devised so far. One reason is the tremendous spreading speed of worms, which leaves a very short reaction time to the defender [2]. Another reason is the distributed nature of the problem, which mandates that defense mechanisms are deployed almost without gap on an Internet-wide scale.

Research on countermeasures against worms has focused on both the detection and the containment of worms. A number of approaches have been proposed that aim to detect worms based on network traffic anomalies. One key observation was that scanning worms, which attempt to locate potential victims by sending probing packets to random targets, exhibit a behavior that is quite different from most legitimate applications. Most prominently, this behavior manifests itself as a large number of (often failed) connection attempts.

Other detection techniques based on traffic anomalies check for a large number of connections without previous DNS requests [7] or a large number of received "ICMP unreachable messages" [3]. In addition, there are techniques to identify worms by monitoring traffic sent to dark spaces, which are unused IP address ranges [2], or honeypots [4].

Active worms use various scan mechanisms to propagate themselves efficiently. The basic form of active 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. Other worms propagate themselves more effectively than PRS worms using various methods, e.g., network port scanning, email, file sharing, Peer-to-Peer (P2P) networks, and instant messages. In addition, worms use different scan strategies during different stages of propagation. In order to increase propagation efficiency, they use a local network or hitlist to infect previously identified vulnerable computers at the initial stage of propagation [12],

III. Existing System

1. Sharif et al. [6] presented an obfuscation-based technique that automatically conceals specific condition dependent malicious behavior from virus detectors that have no prior knowledge of program inputs.
2. Popov et al. [7] investigated a technique that allows the worm programs to be obfuscated by changing many control transfers into signals (traps) and inserting dummy control transfers and "junk" instructions after the signals. The resulting code can significantly reduce the chance to be detected.

IV. Proposed System

1. Worm detection has been intensively studied in the past and can be generally classified into two categories: "host-based" detection and "network-based" detection.
2. Host-based detection systems detect worms by monitoring, collecting, and analyzing worm behaviors on end-hosts.
3. Network-based detection systems detect worms primarily by monitoring, collecting, and analyzing the scan traffic (messages to identify vulnerable computers) generated by worm attacks
4. The C-Worm camouflages its propagation by controlling scan traffic volume during its propagation
5. A worm attacker may use an open-loop control (non-feedback) mechanism by choosing a randomized and time related pattern for the scanning and infection in order to avoid being detected.
6. The open-loop control approach raises some issues of the invisibility of the attack.
7. A very slow propagation of the CWorm is also not desirable, since it delays rapid infection damage to the Internet. Hence, the C-Worm needs to adjust its propagation so that it is neither too fast to be easily detected. where all worm instances actively participate in the propagation. nor too slow to delay rapid damage on the Internet.
8. To regulate the C-Worm scan traffic volume, we introduce a control parameter called attack probability P(t) for each worm-infected computer. P(t) is the probability that a CWorm instance participates in the worm propagation (i.e.,scans and infects other computers) at time t. Our C-Worm model with the control parameter P(t) is generic. P(t) = 1 represents the cases for traditional worms.
9. In order to achieve its camouflaging behavior, the C-Worm needs to obtain an appropriate P(t) to manipulate its scan traffic.
10. Specifically, the C-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.
11. The average value of the overall scan traffic volume is sufficient to make the C-Worm propagate fast enough to cause rapid damage on the Internet..
12. In this paper, we focus on a new class of worms, referred to as the camouflaging worm (C-Worm). The C-Worm adapts their propagation traffic patterns in order to reduce the probability of detection, and to eventually infect more computers.
13. The C-Worm is different from polymorphic worms that deliberately change their payload signatures during propagation.
14. Recent studies also showed that existing commercial anti-worm detection systems fail to detect brand new worms and can also be easily circumvented by worms that use simple mutation techniques to manipulate their payload.

A. Spectrum Based Analysis

1. Power Spectral Density
2. Spectral Flatness Measure

B. Power Spectral Density

Transfer data from time domain Scan traffic determined the discrete fourier transform. Compare c-worm traffic and normal worm traffic Transform data from time domain into frequency domain Random process x(t),t€[0,n], X(t)-source count in time period PSD SCAN traffic data is determined using Discrete fourier transform Φ(Rx[L],k)=Є(Rx[L].e^-j2Πkn/N Compare non worm traffic and c-worm traffic.

C. Spectral Flatness Measure

Distinguish the scan traffic of c-worm and normal non worm traffic Spectral flatness measure concentration of data at narrow frequency range higher concentration at small spectrum comparatively.

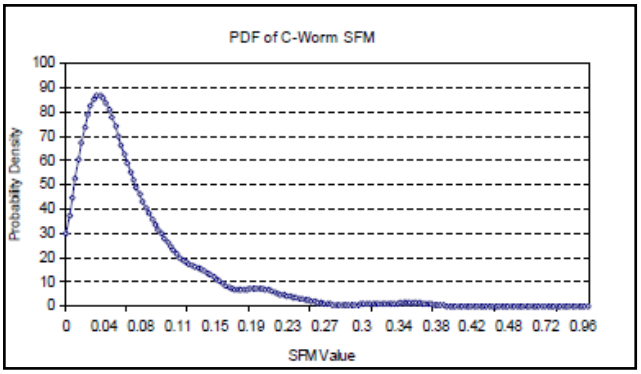

Fig. 1: PDF of SFM on C-Worm Traffic

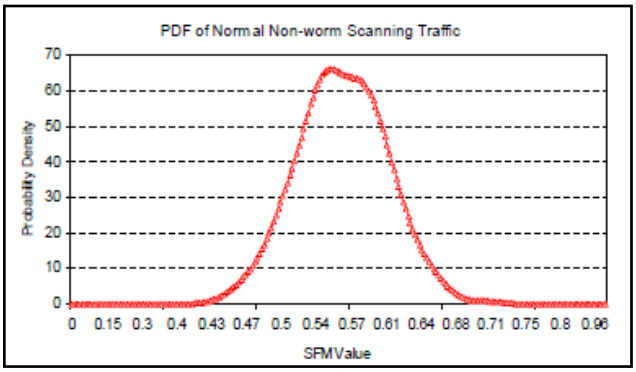

Fig. 2: PDF of SFM on Normal Non-Worm Traffic

We are modeling the Camouflaging Worm (C-Worm), in which the Behavior is hidden and its action is implicitly kept secret. So this Process of Detecting the C-Worm is not possible using the usual Traditional Worm Detection Techniques as well as IP Trace Back Systems. The Major Advantage of the C- Worm is it scans all the IP Present in the Network first then identifies the number of protected systems, number of Worm Affected Systems, number of Vulnerable Systems. C-Worm rather focusing all the IP, instead it focuses only the Vulnerable Systems, because these systems are the Target of C-Worm.

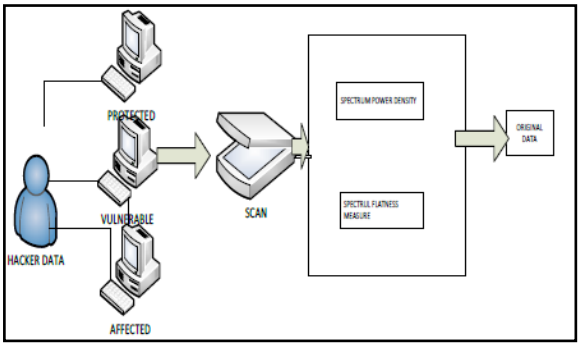

Fig. 3:

The Main aim of C-Worm is the overall scan traffic for the C-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 C-Worm is also not desirable, since it delays rapid infection damage to the Internet. Hence, the C-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 Internet. The Detection method is Spectrum method based Process to continuously monitor the Traffic Process. Even though smaller amount of Traffic is only going to generate the Spectrum based process will identify the C-Worm based on the Behavior.

System Overall Architecture.

There are thre types of system

- Affected system
- Protected system
- Vulnerable system

Power Spectral Density (PSD) distribution of the scan traffic volume and its corresponding Spectral Flatness Measure (SFM). It detect and delete the cworm.

V. Conclusion

C-Worm could use based on the limited network and computing resources available during its propagation. Incorporating the Peer-to-Peer techniques to disseminate information through secured channels actually a worm could take advantage of the knowledge that an infection attempt was a new hit reaching a previously uninfected vulnerable computer and duplicate hit reaching a previously infected vulnerable computer. The approach used by the “self-stopping” worms that do not require a global overlay control network for Realizing their behavior in practice. We call our approach to estimate the Distributed Co-ordination method. In this method, there is no centralized coordination between the C-Worm instances to obtain feedback information about the value. The distributed co-ordination requires each C-Worm infected computer to be marked with a watermark indicating that the C-Worm infection code has already been installed on the scanned host as with “Code-Red” worms.Thus, when an already infected computer .A scans another infected computer then computer will detect the water mark and know that computer B has already been infected. By scanning vulnerable computers and obtaining the water-marks information during the scanning.

References

[1] D.Moore, V. Paxson, S. Savage,“Inside the slammer worm”, in IEEE Magazine of Security and Privacy, July 2003.

[2] W32.Sircam.Worm@mm, [Online] Available: http://www.symantec.com/ avcenter/venc/data/w32. sircam. worm@mm.html.

[3] Z. S. Chen, L.X. Gao, K. Kwiat,“Modeling the spread of active worms”, in Proceedings of the IEEE Conference on Computer Communications (INFOCOM), San Francisco, CA, March 2003.

[4] M. Garetto, W. B. Gong, D. Towsley, “Modeling malware spreading dynamics”, in Proceedings of the IEEE Conference on Computer Communications (INFOCOM), San Francisco, CA, March 2003.

[5] C. Zou, W. B. Gong, D. Towsley, L. X. Gao,“Monitoring and early detection for internet worms”, in Proceedings of the 10-th ACM Conference on Computer and Communication Security (CCS), Washington DC, October 2003.

[6] J. Wu, S. Vangala, L. X. Gao,“An effective architecture and algorithm for detecting worms with various scan techniques”,

in Proceedings of the 11-th IEEE Network and Distributed System Security Symposium (NDSS), San Diego,CA, Febrary 2004
[7] SANS, Internet Storm Center, [Online] Available: http://isc.sans.org/.
[8] D. J. Daley, J. Gani, Epidemic Modeling: an Introduction, Cambridge University Press, 1999.
[9] G. F. Gu, D. Dagon, M. I. Sharif X. Z. Qin, W. Lee, G. F. Riley,"Worm detection, early warning, and response based on local victim information", in Proceedings of Proceedings of the 20-th Annual Computer Security Applications Conference (ACSAC2004), Tucson, Arizona, December 2004.
[10] G. M. Voelker J. Ma, S. Savage,"Self-stopping worms", in Proceedings of the ACM Workshop on Rapid Malcode (WORM), Washington D.C, November 2005.

Mr. Yugandhar Tirumani is a student of Akula Gopayya College of Engineering & Technology, Tadepalligudem. Presently he is pursuing his M.Tech (C.S.E) from this college and he received his B.Tech(CSE) from Swarnandhra College of Engineering & Technology , Seetharamapuram. His area of interest includes Computer Networks , Data Warehouse and Data Mining, Network Security and other advances in Computer Applications.

Mr. Raju Gumpula, excellent teacher Received B.Tech (IT) from BVC Engineering College, Odalarevu, JNTU Hyderabad and M.Tech (CSE) from Vignan's Institute of Information Technology, Visakhapatnam, JNTU Kakinada is working as an Associate Professor in Department of C.S.E, Akula Gopayya college of Engineering and Technology. He has 5 years of teaching experience. He has published many papers in both National & International Journals. His area of Interest includes Data Communications & Networks , Data Warehouse and Data Mining, Database Management Systems and other advances in computer Applications.