Spreading and Investigation of Containment Worms

[1]Mohammad Nazeer, [2]Md. Shareef

[1,2]Dept. of Computer Science, Swarna Bharathi Institute of Tech. & Science, Khammam, AP, India

Abstract

In this paper, we investigate a new class of active worms, referred to as Containment 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 containments 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 non-worm 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, we 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.

Keywords

Worm, Containment, Anomaly Detection

I. Introduction

An active worm refers to a malicious software program that propagates itself on the Internet to infect other computers. The propagation of the worm is based on exploiting vulnerabilities of computers on the Internet. Many real-world worms have caused notable damage on the Internet. These worms include “Code-Red” worm in 2001 [1], “Slammer” worm in 2003 [2], and “Witty”/“Sasser” worms in 2004 [3]. Many active worms are used to infect a large number of computers and recruit them as bots or zombies, which are networked together to form botnets [4]. These botnets can be used to: (a) launch massive Distributed Denial-of-Service (DDoS) attacks that disrupt the Internet utilities [5], (b) access confidential information that can be misused [6] through large scale traffic sniffing, key logging, identity theft etc, (c) destroy data that has a high monetary value [7], and (d) distribute large-scale unsolicited advertisement emails (as spam) or software (as malware). Due to the substantial damage caused by worms in the past years, there have been significant efforts on developing detection and defense mechanisms against worms.

A network-based worm detection system plays a major role by moni-toring, collecting, and analyzing the scan traffic (messages to identify vulnerable computers) generated during worm attacks.

In this system, the detection is commonly based on the self-propagating behavior of worms that can be described as follows: after a worm-infected computer identifies and infects a vulnerable computer on the Internet, this newly infected computer1 will automatically and continuously scan several IP addresses to identify and infect other vulnerable computers.

As such, numerous existing detection schemes are based on a tacit assumption that each worm-infected computer keeps scanning the Internet and propagates itself at the highest possible speed. Furthermore, it has been shown that the worms can traffic volume and the number of worm-infected computers exhibit exponentially increasing patterns [2, 11-14].

Nevertheless, the attackers are crafting attack strategies that intend to defeat existing worm detection systems. In particular, ‘stealth’ is one attack strategy used by a recently-discovered active worm called “Atak” worm [15] and the “self-stopping” worm [16], circumvent detection by hibernating (i.e., stop propagating) with a pre-determined period. Worm might also use the evasive scan [17] and traffic morphing technique to hide the detection [18].

This worm attempts to remain hidden by sleeping (suspending scans) when it suspects it is under detection. Worms that adopt such smart attack strategies could exhibit overall scan traffic patterns different from those of traditional worms. Since the existing worm detection schemes will not be able to detect such scan traffic patterns, it is very important to understand such smart-worms and develop new countermeasures to defend against them .In this paper, we conduct a systematic study on a new class of such smart-worms denoted as Containment Worm(C-Worm in short). The C-Worm has a self-propagating behavior similar to traditional worms, i.e., it intends to rapidly infect as many vulnerable computers as possible.

The containment is achieved by manipulating the scan traffic volume of worm-infected computers. We note that the propagation controlling nature of the C-Worm (and similar smart-worms, such as “Atak”) cause a slow down in the propagation speed. However, by carefully controlling its scan rate, the C-Worm can: (a) still achieve its ultimate goal of infecting as many computers as possible before being detected, and (b) position itself to launch subsequent attacks [4-7].

We comprehensively analyze the propagation model of the C-Worm and corresponding scan traffic in both time and frequency domains. We observe that although the C-Worm scan traffic shows no noticeable trends in the time domain, it demonstrates a distinct pattern in the frequency domain.

Specifically, there is an obvious concentration within a narrow range of frequencies. This concentration within a narrow range of frequencies is inevitable since the C-Worm adapts to the dynamics of the Internet in a recurring manner for manipulating and controlling its overall scan traffic volume.

The above recurring manipulations 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 or such that the scan traffic volume does not cross thresholds that could reveal the C-Worm propagation.

Based on the above observation, we adopt frequency domain analysis techniques and develop a detection scheme against wide-spreading of the C-Worm. Particularly, we develop a novel spectrum-based detection scheme that uses the Power Spectral Density (PSD) distribution of scan traffic volume in the frequency domain and its corresponding Spectral Flatness Measure (SFM) to distinguish the C-Worm traffic from non-worm traffic (background traffic).

II. Background and Related Work

A. Active Worms

Active worms are similar to biological viruses in terms of their infectous and self-propagating nature. They identify vulnerable computers, infect them and the worm-infected computers propagate the infection further to other vulnerable computers. In order to understand worm behavior, we first need to model it. With this understanding, effective detection and defense schemes could be developed to mitigate the impact of the worms.

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 effec-tively than PRS worms using various methods, e.g., network port scanning, email, file sharing, Peer-to-Peer (P2P) networks, and Instant Messaging (IM) Different from the above worms, which attempt to accelerate the propagation with new scan schemes, the Containment Worm (C-Worm) studied in this paper aims to elude the detection by the worm defense system during worm propagation.

B. Worm Detection

Worm detection has been intensively studied in the past and can be generally classified into two categories: "host-based" detection and "network-based" detection. Host-based detection systems 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 contrast, network-based detection systems detect worms primarily by monitoring, collecting, and analyzing the scan traffic (messages to identify vulner-able computers) generated by worm attacks. However, while vulnerabilities exist and pose threats of large-scale damage, it is critical to also focus on network-based detection, as this paper does, to detectwide-spreadingworms.

In order to rapidly and accurately detect Internet-wide large scale propagation of active worms, it is imperative to monitor and analyze the traffic in multiple locations over the Internet to detect suspicious traffic generated by worms.

Network-based detection schemes commonly analyze the collected scanning traffic data by applying certain decisionrules for detecting the worm propagation.

III. Modeling of The C-Worm

A. C-Worm

The C-Worm containments its propagation by controlling scan traffic volume during its propagation. The simplest way to manipulate scan traffic volume is to randomly change the number of worm instances conducting port-scans. As other Alternatives, 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. Nevertheless, the open-loop control approach raises some issues of the invisibility of the attack. First, as we know, worm propagation over the Internet can be considered a dynamic system. When an attacker launches worm propagation, it is very challenging for the attacker to know the accurate parameters for worm propagation dynamics over the Internet. Given the inaccurate knowledge of worm propagation over the Internet, the open-loop control system will not be able to stabilize the scan traffic. This is a known result from control system theory .Consequently, the overall worm scan traffic volume in the open-loop control system will expose a much higher probability to show an increasing trend with the progress of worm propagation .As more and more computers get infected, they, in turn, take part in scanning other computers.

Hence, we consider the C-worm as a worst case attacking scenario that 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 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.

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 C-Worm 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, where all worm instances actively participate in the propagation. For the C-Worm, P (t) needs not be a constant value and can be set as a time varying function.

In order to achieve its containment behavior, the C-Worm needs to obtain an appropriate P(t) to manipulate its scan traffic. Specifically, the C-Worm will regulate its overall scan traffic volume such that: (a) it is similar to non-worm scan traffic in terms of the scan traffic volume over time, (b) it 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, and (c) 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 Internet3.

We assume that a worm attacker intends to manipulate scan traffic volume so that the number of worm instances participating in the worm propagation follow a random distribution with mean MC. This MC can be regulated in a random fashion during worm propagation in order to containment the propogationofC-worm.

To regulate MC , it is obvious that P (t) must be decreased over time since M (t) keeps increasing during the wormpropagation. We can express P(t) using a simple function MCM as follows:

P (t) = min(M(t) , 1), where (t) represents the

estimation of M (t) at time t. From the above expression, we know that the C-Worm needs to obtain the value of M (t) (as close to M(t) as possible) in order to generate an effective P (t). Here, we discuss one approach for the C-Worm to estimate M (t). The basic idea is as follows: A C-Worm could estimate the percentage of computers that have already been infected over the total number of IP addresses as well as M (t), through checking a scan attempt as a new hit (i.e., hitting an uninfected vulnerable computer) or a duplicate hit (i.e., hitting an already infected vulnerable computer). This method requires each worm instance (i.e., infected computer) to be marked indicating that this computer has been infected. Thus, when a worm instance (for example, computer A) scans one infected computer (for example, computer B), then computer A will detect such a mark, thereby becoming aware that computer B has been infected.

B. Propagation Model of the C-Worm

To analyze the C-Worm, we adopt the epidemic dynamic model for disease propagation, which has been extensively used for worm propagation modeling [2, 12]. Based on existing results [2, 12], this model matches the dynamics of real worm propgation over the Internet quite well. For this reason, similar to other publications, we adopt this model in our paper as well. Since our investigated C-Worm is a novel attack, we modified the original Epidemic dynamic formula to model the propagation of the C-Worm by introducing the P (t) - the attack probability that a worm-infected computer participates in worm propagation at time t. We note that there is a wide scope to notably improve our modified model in the future to reflect several characteristics that are relevant in real-world practice.

Particularly, the epidemic dynamic model assumes that any given computer is in one of the following states: immune, vulnerable, or infected. An immune computer is one that cannot be infected by a worm; a vulnerable computer is one that has the potential of being infected by a worm; an infected computer is one that has been infected by a worm. The simple epidemic model for a finite population of traditional PRS worms can be expressed as[4],

$$\frac{dM(t)}{dt} = \beta \cdot M(t) \cdot [N - M(t)], \qquad (1)$$

where M(t), is the number of infected computers at time t; N (= T • P_1 • P_2) is the number of vulnerable computers on the Internet; T is the total number of IP addresses on the Internet; P_1 is the ratio of the total number of computers on the Internet over T; P_2 is the ratio of total number of vulnerable computers on the Internet over the total number of computers on the Internet; β = S/V is called the pairwise infection rate S is the scan rate defined as the number of scans that an infected computer can launch in a given time interval. We assume that at t = 0, there are M (0) computers being initially infected and N −M(0) computers being susceptible to further worm infection.

The C-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,

$$\frac{dM(t)}{dt} = \beta \cdot M(t) \cdot P(t) \cdot [N - M(t)].$$

$$M_c \quad M \qquad (2)$$

Recall that P (t) = M(t), (t) is the estimation of M (t) at time t, and assuming that M (t) = (1 + ϵ) • M (t), where ϵ, is the estimation error, the Formula (2) can be rewritten as,

$$\frac{dM(t)}{dt} = \frac{\beta \cdot \bar{M}^{c}}{1+\epsilon(t)} \cdot [N - M(t)]. \qquad (3)$$

With Formula (3), we can derive the propagation model for the C-Worm as M(t) = N − e− 1+ϵ (t)•t(N − M (0)), where M(0) is the number of infected computers at time 0. Assume that the worm detection system can monitor Pm ($P_m \in [0, 1]$) of the whole Internet IP address space. Without loss of generality, the probability that at least one scan from a worm-infected computer (it generates S scans in unit time on average) will be observed by the detection system is $1-(1-P_m)^{P(t) \cdot S}$. We define that $M_A(t)$ is the number of worm instances that have been observed by the worm detection system at time t, then there are M (t) − $M_A(t)$ unobserved infected instances at time t. At the worm propagation early stage, M (t) − $M_A(t) \simeq$ M (t). The expected number of newly observed infected instances at t + δ (where δ is the interval of monitoring) is (M(t) − $M_A(t)$) • $[1-(1-P_m)^{P(t) \cdot S}] \simeq M(i)[1-(1-P_m)_{P(t) \cdot S}]$. Thus, we have $M_A(t+\delta) = M_A(t) + M(t)[1-(1-P_m)^{P(t) \cdot S}]$. Using simple mathematical manipulations, the number of worm instances observed by the worm detection system at time t is,

$$M_A(t) = P(t) \cdot M(t) \cdot P_m = \frac{P^m \cdot \bar{M}^{c}}{1+\epsilon(t)}. \qquad (4)$$

C. Effectiveness of the C-Worm

We now demonstrate the effectiveness of the C-Worm in evading worm detection through controlling P(t). Given random selection of M_c, we generate three C-Worm attacks (viz., C-Worm 1, C-Worm 2 and C-Worm 3) that are characterized by different selections of mean and variance magnitudes for M_C. In our simulations, we assume that the scan rate of the traditional PRS worm follow a normal distribution S_n = N(40, 40) (note that if the scan rate generated by above distribution is less than 0, we set the scan rate as 0). We also set the total number of vulnerable computers on the Internet as 360,000, which is the total number of infected computers in "Code-Red" worm incident [1].

Fig. 1 shows the observed number of worm-infected computers over time for the PRS worm and the above three C-Worm attacks. Fig. 2, shows the infection ratio for the PRS worm and the above three C-Worm attacks.

For the C-Worm, the trend of observed number of worm instances over time ($M_A(t)$) (defined in Formula (4)) is much different from that of the traditional PRS worm as shown in fig. 2. This clearly demonstrates how the C-Worm success-fullycontainments its increase in the number of worm instances ($M_A(t)$) and avoids detection by worm detection systems that expect exponential increases in worm instance numbers during large-scale worm propagation. Fig. 3, shows the number of scanning computers from normal non-worm port-scanning traffic (background traffic) for several well-known ports, (i.e., 25, 53, 135, and 8080) obtained over several months by the ISC. Comparing fig. 3 with fig. 1, we can observe that it is hard to distinguish the C-Worm port traffic from background port-scanning traffic in the time domain.

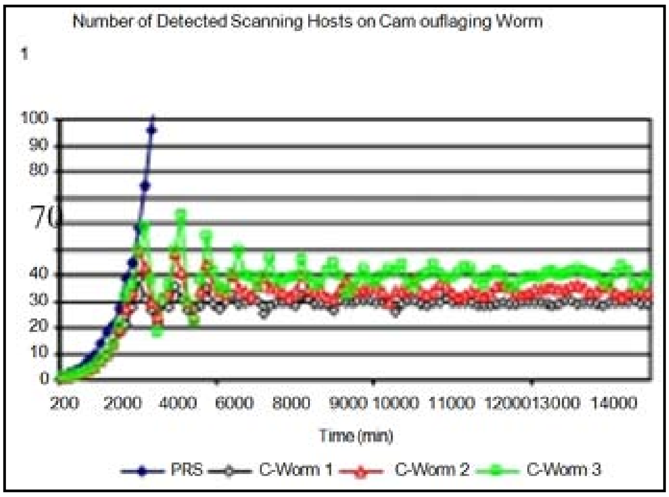

Fig. 1: Observed Infected Instance Number for the C-Worm and PRS Worm

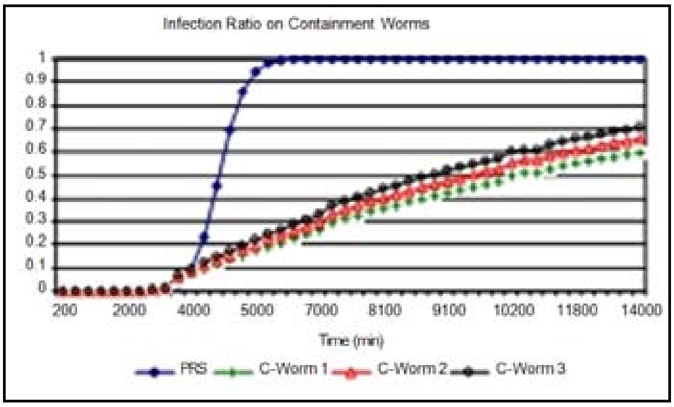

Fig. 2: Infected Ratio for the C-Worm and PRS-Worm

From above figs. 1 and 2, we also observe that the C-Worm is still able to maintain a certain magnitude of scan traffic so as to cause significant infection on the Internet. As a note regarding the speed of C-Worm propagation, we can observe from fig. 1 that the C-Worm takes approximately 10 days to infect 75% of total vulnerable hosts in comparison with the 3.3 days taken by a PRS worm5. We discussed the "Atak" worm in Section I and mentioned that it is similar to the C-Worm since it tries to avoid being detected, when it suspects that it is being detected by antiworm software. However, it differs from the C-Worm in its behavior. The "Atak" worm attempts to hide only during times it suspects its propagation will be detected by anti-worm software. Whereas, the C-Worm proactively containments itself at all times. In addition, the "Self-stopping" worm attempts to hide by co-ordinating with its members to halt propagation activity only after the vulnerable population is subverted [16]. This behavior leaves enough evidence for worm detection systems to recognize its propagation. The C-Worm, on the other hand, hides itself even during its propagation and thus keeps the worm detection schemes completely unaware of its propagation. The C-Worm also has some similarity in spirit with polymorphic worms that manipulate the byte stream of worm payload in order to avoid the detection of signature (payload)-based detection scheme. The manipulation of worm payload can be achieved by various mechanisms: (a) interleaving meaningful instructions with NOP (no operation), (b) Using different instructions to achieve the same results, (c) Shuffling the register set in each worm propagation program Code copy and (d) using cryptography mechanisms to change worm payload signature with every infection attempt.

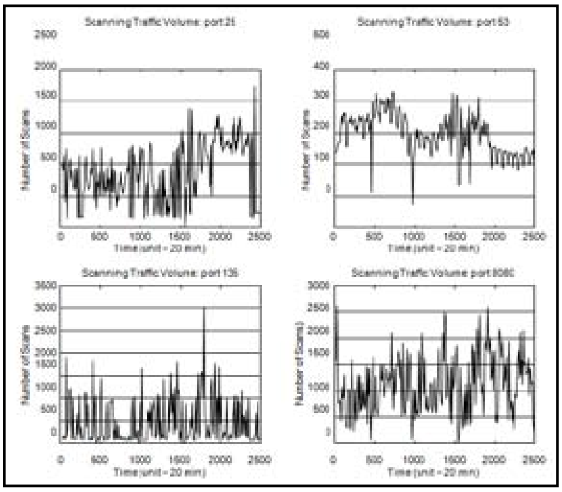

Fig. 3: Observed Infected Instance Number for Back-Ground Scanning Reported by ISC

In contrast, the C-Worm tries to manipulate the scan traffic pattern to avoid detection.

IV. Detecting the C-Worm

A. Design Rationale

In this section, we develop a novel spectrum-based detection scheme. Recall that the C-Worm goes undetected by detection schemes that try to determine the worm propagation only in the time domain. Our detection scheme captures the distinct pattern of the C-Worm in the frequency domain, and thereby has the potential of effectively detecting the C-Worm propagation. In order to identify the C-Worm propagation in the frequency domain, we use the distribution of Power Spectral Density (PSD) and its corresponding Spectral Flatness Measure (SFM) of the scan traffic. Particularly, PSD describes how the power of a time series is distributed in the frequency domain. Mathematically, it is defined as the Fourier transform of the auto-correlation of a time series. In our case, the time series corresponds to the changes in the number of worm instances that actively conduct scans over time. The SFM of PSD is defined as the ratio of geometric mean to arithmetic mean of the coefficients of PSD. The range of SFM values is [0, 1] and a larger SFM value implies flatter PSD distribution and vice versa.

To illustrate SFM values of both the C-Worm and normal non-worm scan traffic, we plot the Probability Density Func-tion (PDF) of SFM for both C-Worm and normal non-worm scan traffic as shown in fig. 5 and fig. 6, respectively. The normal non-worm scan traffic data shown in fig. 6, is based on real-world traces collected by the ISC 6. Note that we only show the data for port 8080 as an example, and other ports show similar observations. From this figure, we know that the SFM value for normal non-worm traffic is very small (e.g., SFM ∈ (0.02, 0.04) has much higher density compared magnitudes). The C-Worm data shown in fig. 5 is based on 800 C-Worms attacks generated by varying attack parameters defined in Section III, such as P(t)and $M_c(t)$.

From this fig., we know that the SFM value of the C-Worm attacks is high (e.g., SFM ∈ 0.5, 0.6 has high density). From the above two figures, we can observe that there is a clear demarcation range of SFM ∈ (0.3, 0.38) between the C-Worm and normal non-worm scan traffic. As such, the SFM can be used to sensitively detect the C-Worm scan traffic.

Using port 135 reported by SANs ISC as an example, we analyze the traces and obtain the traffic target distribution in a window lasting 10 mins. Following existing work], we use entropy as the metric to measure the attack target distribution. Fig. 4 shows the Probability Density Function (PDF) of background traffic's entropy values. We also simulate the worm propagation traffic, which allocates a portion of scan traffic bound for IP addresses monitored by the ITM system. Following this, we obtain the PDF of the entropy value for combined traffic including both worm propagation and background traffic.

From fig. 4, we know that when the attacker uses a portion of attack traffic to manipulate the target distribution, the entropy-based detection scheme can degrade significantly. For example, when the attacker uses 10% traffic to manipulate the traffic's entropy value, the false positive rate of entropy-based detection scheme is 14%.

When the attacker uses 30% traffic to manipulate the traffic's entropy value, the false positive rate becomes 40%. Hence, in order to preserve the performance, entropy-based detection scheme needs to evolve correspondingly and integrate with other detection schemes. We will perform a more detailed study of this aspect in our future work.

The large SFM values of normal non-worm scan traffic can be explained as follows. The normal non-worm scan traffic does not tend to concentrate at any particular frequency since its random dynamics is not caused by any recurring phenomenon. The small value of SFM can be reasoned by the fact that the power of C-Worm scan traffic is within a narrow-band frequency range. Such concentration within a narrow range of frequencies is unavoidable since the C-Worm adapts to the dynamics of the Internet in a recurring manner for manipulating the overall scan traffic volume.

B. Spectrum-Based Detection Scheme

We now present the details of our spectrum-based detection scheme. Similar to other detection schemes [19, 21], we use a "destination count" as the number of the unique destination IP addresses targeted by launched scans during worm propaga-tion. To understand how the destination count data is obtained, we recall that an ITM system collects logs from distributed monitors across the Internet. On a side note, Internet Threat Monitoring (ITM) systems are a widely deployed facility to detect, analyze, and characterize dangerous Internet threats such as worms. In general, an ITM system consists of one centralized data center and a number of monitors distributed across the Internet.

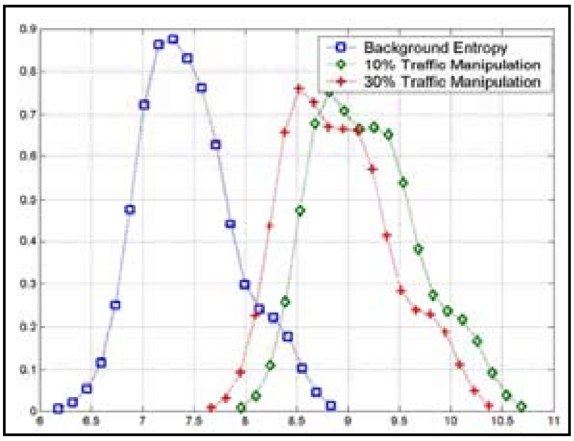

Fig. 4: Manipulation of Attack Target Distribution Entropy

Each monitor records traffic that addressed to a range of IP addresses (which are not commonly used IP address also called the dark IP addresses) and periodically sends the traffic logs to the data center. The data center then analyzes the collected traffic LOGS and publishes reports (e.g., statistics of monitored traffic) to ITM system users. Therefore the baseline traffic in our study is scan traffic. With reports in a sampling window Ws, the source count X(t) is obtained by counting the unique source IP addresses in received logs.

In our spectrum-based detection scheme, the distribution of PSD and its corresponding SFM are used to distinguish the C-Worm scan traffic from the non-worm scan traffic. Recall that the definition of PSD distribution and its corresponding SFM are introduced in Section IV(a). In our worm detection scheme, the detection data (e.g., destination counter), is further pro-cessed in order to obtain its PSD and SFM. In the following, we detail how the PSD and SFM are determined during the processing of the detection data.

1. Power Spectral Density (PSD)

To obtain the PSD distribution for worm detection data, we need to transform data from the time domain into the frequency domain. To do so, we use a random process X(t), t ∈ [0, n] to model the worm detection data. Assuming X(t) is the source count in time period [t − 1, t] (t ∈ [1, n]), we define the auto-correlation of X(t) by RX (L) = E[X(t)X(t + L)].

In Formula (5), RX (L) is the correlation of worm detection data in an interval L. If a recurring behavior exists, a Fourier transform of the auto-correlation function of RX (L) can reveal such behavior. Thus, the PSD function (also represented by SX (f); where f refers to frequency) of the scan traffic data is determined using the Discrete Fourier Transform (DFT) of its auto-correlation function as follows,

$$\psi\,(R_X\ [L],\ K) = \sum_{n=0} (R_X\ [L]) \cdot e^{-\ j2\pi\ Kn/N}\ ,$$

$$\text{Where } K = 0, 1 . . . N - 1 \qquad (6)$$

As the PSD inherently captures any recurring pattern in the frequency domain, the PSD function shows a comparatively even distribution across a wide spectrum range for the normal non-worm scan traffic.

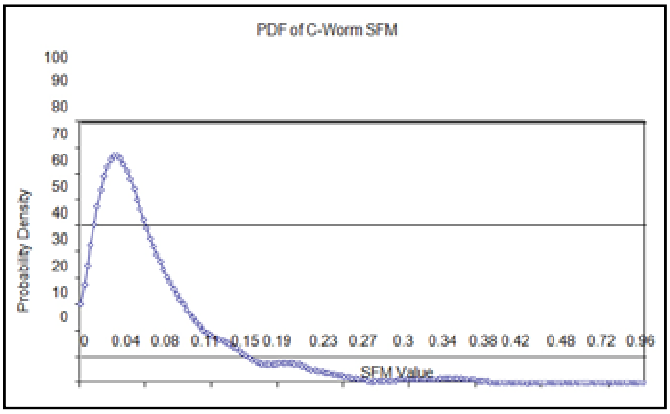

Fig. 5: PDF of SFM on C-Worm Traffic

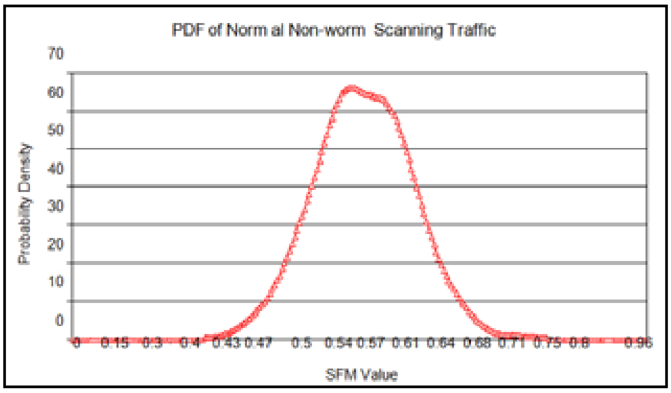

Fig. 6: PDF of SFM on Normal Non-Worm Traffic

The PSD of C-Worm scan traffic shows spikes or noticeably higher concentrations at a certain range of the spectrum.

2. Spectral Flatness Measure (SFM)

We measure the flatness of P SD to distinguish the scan traffic of the C-Worm from the normal non-worm scan traffic. For this, we introduce the Spectral Flatness Measure (SFM), which can capture anomaly behavior in certain range of frequencies.

The SFM is defined as the ratio of the geometric mean to the arithmetic mean of the PSD coefficients. It can be expressed as,

$$SFM = \left[\frac{1}{n}\sum_{k=1}^{n} \frac{S(f_k)}{S(f_k)}\right]_n , \qquad (7)$$

where S(fk) is an PSD coefficient for the PSD obtained from the results in Formula (6). SFM is a widely existing measure for discriminating frequencies in various applications such as voiced frame detection in speech recognition. In general, small values of SFM imply the concentration of data at narrow frequency spectrum ranges.

Note that the C-Worm has unpreventable recurring behavior in its scan traffic; consequently its SFM values are comparatively smaller than the SFM values of normal non-worm scan traffic. To be useful in detecting C-Worms, we introduce a sliding window to capture a noticeably higher concentrations at a small range of spectrum.

When such noticeably concentration is recognized, we derive the SFM within a wider frequency range. From fig. 5, we can observe that the SFM value for the C-Worm is very small (e.g., with a mean value of approximately 0.075).

V. Performance Evaluation

In this section, we report our evaluation results that illustrate the effectiveness of our spectrum-based detection scheme against both the C-Worm and the PRS worm in comparison with existing representative detection schemes for detecting wide-spreading worms. In addition, we also take into consideration destination distribution based detection schemes and evaluate their performance against the C-Worm.

A. Evaluation Methodology

1. Evaluation Metrics

In order to evaluate the performance of any given detection scheme against the C-Worm, we use the following three metrics listed in Table II. The first metric is the worm Infection Ratio (IR), which is defined as the ratio of the number of infected computers to the total number of vulnerable computers, assuming there is no worm detection/defense system in place. The other two metrics are the Detection Time (DT) and the Maximal Infection Ratio (MIR). DT is defined as the time taken to successfully detect a wide-spreading worm from the moment the worm propagation starts. It quantifies the detection speed of a detection scheme. MIR defines the ratio of an infected computer number over the total number of vulnerable computers up to the moment when the worm spreading is detected. It quantifies the damage caused by a worm before being detected. The objective of any detection scheme is to minimize the damage caused by a rapid worm propagation. Hence, MIR and DT can be used to quantify the effectiveness of any worm detection scheme. The higher the values, the more effective the worm attack and the less effective the detection. In addition, we use two more metrics -Detection Rate (PD) and False Positive Rate (PF). The PD is defined as the probability that a detection scheme can correctly identify a worm attack. The PF is defined as the probability that a detection scheme mistakenly identifies a non-existent worm attack.

2. Simulation Setup

In our evaluation we considered both experiments with real-world "non-worm" traffic and simulated c-worm traffic. To make our experiments reflect real-world practice, some key parameters that we used to generate C-worm traffic in our simulation were based on previous results from a real-worm incidence - "Code-Red" worm in 2001 [1]. Specifically, we set the total number of vulnerable computers on the Internet as 360,000, which is the maximum number of computers which could be infected by "Code-Red" worm.

Additionally, we set the scan rate S (number of scans per minute) to be variable within a range, this allows us to emulate the infected computers in different network environments. In our evaluation, the scan rates are predetermined and follow a Gaussian distribution S = N (S^m, $S^σ$), where S^m and $S^σ$ are in [(20, 70], similar to those used in [19]. In our evaluation, we merged the simulated C-worm attack traffic into replayed "non-worm" traffic traces and carried out evaluation study.

3. Detection Decision Rule

We now describe the method of applying an appropriate detection rule to detect C-Worm propagation. As the SFM value can be used to sensitively distinguish the C-Worm and normal non-worm scan traffic, the worm detection is performed by comparing the SFM with a predefined threshold Tr. If the SFM value is smaller than a predefined threshold Tr, then a C-Worm propagation alert is generated. The value of the threshold Tr used by the C-Worm detection can be fittingly set based on the knowledge of statistical distribution (e.g., PDF) of SFM values that correspond to the non-worm scan traffic. Notice that the Tr value for the non-worm traffic can be derived by analyzing the historical data provided by SANs Internet Storm Center (ISC). In the worm detection systems, monitors collect port-scan traffic to certain area of dark IP addresses and periodically reports scan traffic log to the data center. Then the data center aggregates the data from different monitors on the same port and publishes the data. Based on the historical data for different ports, we can build the statistical profiles of port-scan traffic on different ports and then derive the Tr value for the non-worm traffic. Based on the continuous reported data, the value of Tr will be tuned and adaptively used to carry out worm detection. If we can obtain the PDF of SFM values for the C-Worm through comprehensive simulations and even real-world profiled data in the future, the optimal threshold can be obtained by applying the Bayes classification. If the PDF of SFM values for the C-Worm is not available, based on the PDF of SFM values of the normal non-worm scan traffic, we can set an appropriate Tr value. For example, the Tr value can be determined by the Chebyshev inequality in order to obtain a reasonable false positive rate for worm detection.

Hence in Section V, we evaluate our spectrum-based detection scheme against the C-Worm on two cases: (a) the PDF of SFM values are known for both the normal non-worm scan traffic. We simulate the C-Worm attacks by varying the attack parameters, such as attack probability (P(t)) and the number of worm instances participating in the scan (M¯C) defined in Section III. The MC follows the Gaussian distribution N (m, σ) and are changed dynamically by the C-Worm during its propagation. Particularly, for N(m, σ), m is randomly selected in (12000, 75000) and σ is randomly selected in (0.2, 100). We simulate different C-Worm attacks by varying the values of m and σ.

In practice, since detection systems analyze port scan traffic blended with the non-worm scan traffic, we replay the real-world traces as non-worm scan traffic (background noise to attack traffic) in our simulations. In particular, we used the ISC real-world trace (Shield logs dataset) from 01/01/2005 to 01/15/2005.

B. Performance of Detection Schemes

We evaluate our proposed spectrum-based detection scheme by comparing its performance with three existing representative traffic volume-based detection schemes. The first scheme is the volume mean-based (MEAN) detection scheme which uses mean of scan traffic to detect worm propagation [20]; the second scheme is the trend-based (TREND) detection scheme which uses the increasing trend of scan traffic to detect worm propagation [19]; and the third scheme is the victim number variance based (VAR) detection scheme which uses the variance of the scan traffic to detect worm propagation [21].

We define our spectrum-based detection scheme as SPEC. We evaluate two types of SPEC: one has no knowledge of any C-Worm attacks or C-Worm scan traffic (denoted by SPEC(W)) and the other has knowledge of C-Worm attacks through an off-line training process (denoted by SPEC). For the off-line training, we use 1000 worm attacks that include both the C-Worm (800 C-Worm attacks) and PRS worms (200 PRS worm attacks).

1. Detection Performance for C-Worm Attacks

Table 2, shows the detection results of different detection schemes against the C-Worm. The results have been averaged over 500 C-Worm attacks. From this table, we can observe that existing detection schemes are not able to effectively detect the C-Worm and their detection rate (PD) values are significantly lower in comparison with our spectrum-based detection schemes (SPEC and SPEC(W)). For example, SPEC achieves the detection rate of 99%, which is at least 3-4 times more accurate than detection schemes such as VAR and MEAN that achieve detection rate values of only 48% and 14%, respectively.

Our SPEC and SPEC(W) detection schemes also achieve good detection time (DT) performance in addition to the high detection rate values indicated above. In contrast, the detection time of existing detection schemes have relatively larger values. As a consequence of the detection time values, we can see that the C-Worm propagation is effectively contained by SPEC and SPEC(W), as demonstrated by the lower values of maximal infection ratio (MIR) for the SPEC and SPEC(W).

Since the detection rate values for the existing detection schemes are relatively small, obtaining low values of M IR for those schemes are not as significant as those for SPEC and SPEC(W). Furthermore, we can notice that the detection performance of the SPEC(W) is worse than the SPEC.

2. Detection Performance for Traditional PRS Worms

We evaluate the detection performance of different detection schemes for traditional PRS worm attacks. The detection performance results have been averaged over 500 PRS worm attacks. We observe that both our SPEC and SPEC(W) schemes achieve 100% detection rate (PD) while detecting traditional PRS worms in comparison with the existing worm detection schemes that have been specifically designed for detecting the raditional PRS worms.

In view of emphasizing the relative performance of our SPEC and SPEC(W) schemes with the existing worm detection schemes, we plot the MIR and DT results in figs. 7 and 8 for different scan rates S. We can observe from these figures that the MIR and DT results of our spectrum based scheme (shown only for SPEC(W)) are comparable or better than the existing worm detection schemes. For a mean scan rate of 70/min, our SPEC(W) scheme achieves a detection time of 1024 mins, which is faster than that of VAR and MEAN schemes, whose values are 1239 min and 1161 min, respectively. For the same mean scan rate of 70/min, SPEC(W) achieves a maximal infection ratio of 0.03, which is comparable to TREND's MIR value and is less than 50% of the MIR value for the VAR and MEAN detection schemes. The effectiveness of our spectrum-based scheme is based on the fact that traditional PRS worm scanning traffic increase.

VI. Conclusion

In this paper, we studied a new class of smart-worm called C-Worm, which has the capability to containment its propagation and further avoid the detection. Our investigation showed that, although the C-Worm successfully containments its propagation in the time domain, its containment nature inevitably manifests as a distinct pattern in the frequency domain. Based on observation, we developed a novel spectrum-based detection scheme to detect the C-Worm.

Our evaluation data showed that our scheme achieved superior detection performance against the C-Worm in comparison with existing representative de-tection schemes. This paper lays the foundation for on going studies of "smart" worms that intelligently adapt their propa-gation patterns to reduce the effectiveness of countermeasures.

References

[1] D. Moore, C. Shannon, J. Brown,"Code-red: a case study on the spread and victims of an internet worm", Proceedings of the 2-th Internet Measurement Workshop (IMW), Marseille, France, November 2002.

[2] D. Moore, V. Paxson, S. Savage,"Inside the slammer worm", IEEE Magazine of Security and Privacy, July 2003.

[3] CERT, CERT/CC advisories, [Online] Available: http://www.cert.org/advisories/.

[4] P. R. Roberts, Zotob Arrest Breaks Credit Card Fraud Ring", [Online] Available: http://www.eweek.com/article2/0, 1895, 1854162,00.asp.

[5] W32/MyDoom.B Virus, [Online] Available: http://www.us-cert.gov/cas/techalerts/TA04-028A.html.

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

[7] Worm.ExploreZip, [Online] Available: http://www.symantec.com/avcenter/venc/data/worm.explore.zip.html.

[8] R. Naraine,"Botnet Hunters Search for Command and Control Servers", [Online] Available: http://www.eweek.com/article2/0,1759,1829347,00.asp.

[9] T. Sanders, Botnet operation controlled 1.5m PCs Largest zom-biearmyevercreated, botnet-operation-ruled-million, 2005.

[10] R. Vogt, J. Aycock, M. Jacobson,"Quorum sensing and self-stopping worms", in Proceedings of 5th ACM Workshop on Recurring Malcode (WORM), Alexandria VA, October 2007.

[11] S. Staniford, V. Paxson, N. Weaver,"How to own the internet in yourspare time", in Proceedings of the 11-th USENIX Security Symposium(SECURITY), San Francisco, CA, August 2002.

[12] 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.

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

[14] C. C. Zou, W. Gong, D. Towsley,"Code-red worm propagation modeling and analysis," Proceedings of the 9-th ACM Conference on Computer and Communication Security (CCS), Washington DC, November 2002.

[15] Zdnet, Smart worm lies low to evade detection, [Online] Available: http://news.zdnet.co.uk/internet/security/0,39020375,39160285,00.htm.

[16] J. Ma, G. M. Voelker, S. Savage,"Self-stopping worms", Proceedings of the ACM Workshop on Rapid Malcode (WORM), Washington D.C, November 2005.

[17] Min Gyyng Kang, Juan Caballero, Dawn Song, "Distributed evasive scan techniques and countermeasuress", Proceedings of International Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA), Lucerne, Switzerland, July 2007.

[18] Charles Wright, Scott Coull, Fabian Monrose,"Traffic morphing: An efficient defense against statistical traffic analysis", Proceedings of the 15th IEEE Network and Distributed System Security Symposium (NDSS), San Diego, CA, Febrary 2008.

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

[20] S. Venkataraman, D. Song, P. Gibbons, A. Blum,"New streaming algorithms for superspreader detection", Proceedings of the 12th IEEE Network and Distributed Systems Security Symposium (NDSS), San Diego, CA, Febrary 2005.

[21] J. Wu, S. Vangala, L. X. Gao,"An effective architecture and algorithm for detecting worms with various scan techniques", Proceedings of the 11-th IEEE Network and Distributed System Security Symposium (NDSS), San Diego, CA, Febrary 2004.

[22] Dshield.org, Distributed Intrusion Detection System, [Online] Available: http://www.dshield.org/, 2005.

[23] SANS, Internet Storm Center, [Online] Available: http://isc.sans.org/.

[24] C. C. Zou, W. Gong, D. Towsley,"Worm propagation modeling and analysis under dynamic quarantine defense", in Proceedings of the 1-th ACM CCS Workshop on Rapid Malcode (WORM), Washington DC, October 2003.

[25] C. C. Zou, D. Towsley, W. Gong,"Modeling and simulation study of the propagation and defense of internet e-mail worm", IEEE Transactions on Dependable and Secure Computing, Vol. 4, No. 2, pp. 105-118, 2007.

[26] C. Zou, Don Towsley, Weibo Gong,"Email worm modeling and defense", Proceedings of the 13-th International Conference on Computer Communications and Networks (ICCCN), Chicago, IL, October 2004.

[27] W. Yu, S. Chellappan C. Boyer, D. Xuan,"Peer-to-peer system-based active worm attacks: Modeling and analysis", Proceedings of IEEE International Conference on Communication (ICC), Seoul, Korea, May 2005.

[28] Dynamic Graphs of the Nimda Worm, [Online] Available: http://www.caida.org/dynamic/analysis/security/nimda.

[29] S. Staniford, D. Moore, V. Paxson, N. Weaver,"The top speed of flash worms", in Proceedings of the 2-th ACM CCS Workshop on Rapid Malcode (WORM), Fairfax, VA, October 2004.

[30] Yubin Li, Zesheng Chen, Chao Chen,"Understanding divide-conquer-scanning worms", Proceedings of International Performance Computing and Communications Conference (IPCCC), Austin, TX, December 2008.

[31] D. Ha, H. Ngo,"On the trade-off between speed and resiliency of flash worms and similar malcodes", Proceedings of 5th ACM Workshop on Recurring Malcode (WORM), Alexandria VA, October 2007.

[32] Y. Yang, S. Zhu, G. Cao,"Improving sensor network immunity under worm attacks: A software diversity approach", Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), Hong Kong, May 2008.

Mohammad Nazeer received his B.Tech degree in Information Technology in the year 2005 from JNT University, Hyderabad and also pursuing his M.Tech in COMPUTER SCIENCE from 2012 and about to complete his M.Tech at Swarana Bharathi Institute of Science and Technology, Khammam, JNT University, Hyderabad. At present working as Assistant Professor in CSE department at Swarana Bharathi College of Engineering, Khammam. His research interest includes Computer Networks, Information Security and Secure Computing.

Md.Shareef received his B.Tech degree in Computer Science Engineering in the year 2004 from kakatiya University, and also received his M.Tech in COMPUTER SCIENCE ENGINEERING in the year 2007 from JNT University, Ananthapur. At present working as an Assistant Professor in CSE department at Swarana Bharathi Institute of Science and Technology, Khammam, AP, His research interests includes Computer Networks, Network Security, Data mining.