Open access Journal International journal of Emerging Trends in Science and Technology

Detection of Non-overlapping C-Worms: A Survey

Authors

Khushboo Joshi[1] & Hemant Dhamecha[2]

[1]M.E,System Software Shri Ram Institute of Technology, Jabalpur
Email: joshi_khushboo@rocketmail.com
[2]Assistant Professor,Computer Science Department,SRIT, Jabalpur
Email: hemant.dhamecha@gmail.com

Abstract
Internet worms place a major security threats to the Internet. This is due to the aptitude of worms to propagate in an automated fashion as they progressively compromise computers on the Internet. Internet worms develop gradually during their propagation and thus place great challenges to preserved against them. In this paper, we examine a new class of active worms, referred to as Non-overlapping Camouflaging Worm .The Non-overlapping C-Worm is different from traditional worms because of its ability to intelligently manipulate its scan traffic volume over time. Thereby, the Non-overlapping C-Worm camouflages its propagation from existing worm detection systems based on analyzing the propagation traffic generated by worms. We analyze characteristics of the Non-overlapping 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 NOC worm. Motivated by our observations, we design a detection method that uses two-step procedures that combines a first stage change point detection with a second stage growth rate inference to confirm the existence of a worm. This scheme is better than the NOC-worm
Keywords-Worms , Propagation speed, Camouflage, Non-overlapping scanning.

1. INTRODUCTION

Worms and their variants have been an insisting security threat in the Internet from the late1980s, causing large parts of Internet becoming not accessible huge amount of financial loss and social shatter during the past decade. For example, "Code red" worm in 2001, "Slammer" worm in 2003 and "Witty"/ "Sasser" worms in 2004 infected millions of host and caused memorable damage on the internet [1]. Worms and viruses are frequently placed in same category, however there is a technical discrimination. On the basis of spreading behavior of worms, can be categorized in passive worms and active worms. A passive worms does not search for the suffered machine, it wait for possible sufferer to contact the worm or depend on user behavior to discover new target and active worms is the vicious software program that self propagate on internet to infect other computer.

Due to the considerable damage caused by worms in past few years, there has been an important effort on evolving detection and protecting mechanisms against worms. An Internet worm is a self-contained program that spread actively by coping itself from one system to other on the Internet. Generally, the worm frequently exploits the system vulnerability and bug detected through scrutinizing and investigation. In order to advert worms from spreading into a large scale, researchers have been focus on modeling their propagation

Table.1 Existing Worm Implementation

Worms	Target finding scheme	Propagation Scheme	Transmission Scheme	Payload format
Morris	Blind	Self Carried	TCP	Monomorphic
Code Red	Blind	Self Carried	TCP	Monomorphic
NImda	Blind	Self Carried	TCP and UDP	Monomorphic
Slammer	Blind	Self Carried	UDP	Monomorphic
Sasser	Blind	Second channel	TCP	Monomorphic
Witty	Blind	Botnet	UDP	Monomorphic

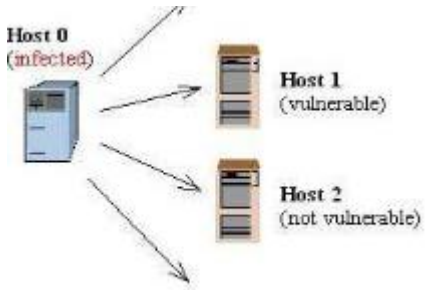
and on that basis examine the perfect countermeasures. Worms was first invented by John Bruner 1975" However the attackers have cunning attack tactics that aimed to subdue existing worm detection systems. In precise 'Stealth' does recently invented active worm called "ATAK" Worm use one attack tactics and "Self-Stopping" worm to elude detection by hibernating (i.e., stop propagating) with a predestined period. The worm tries to remain hidden by sleeping when it supposes it is under detection. As the persistence worm detection scheme will unable to detect such types of scan traffic patterns, so it is very necessary to be aware of such smart worms and invent new countermeasures preserved against them .[3]

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 worm scans traffic volume and the number of worm-infected computers exhibits exponentially increasing patterns. 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"Attack"worm and the "self-stopping"
worm circumvent detection by hibernating (i.e., stop propagating) with a pre-determined period. Worm might also use the evasive scan and traffic morphing technique to hide the detection.[4]

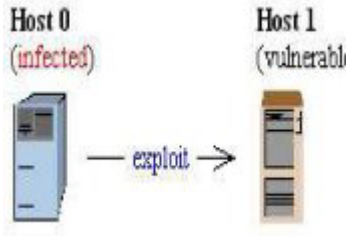
There are many models have been proposed for detecting different problem in network. The two types of model are deterministic and probabilistic .The deterministic model are acceptable for large scale network with high rates dynamic whereas probabilistic model are acceptable for small scale network and low rates dynamics[5]. Deterministic model are basically used for modeling internet worm because they occur in large scale network with thousand of hosts such as SI and SIR. SI [3]is a worm propagation model in which hosts stay in one of two states at any time: *susceptible* (denoted by 'S') or *infectious* (denoted by 'I'). SIR [3] is another worm propagation model that extends the SI model by adding a *removed* (denoted by 'R') state.[6]

1.1 The Life cycle of simple worm

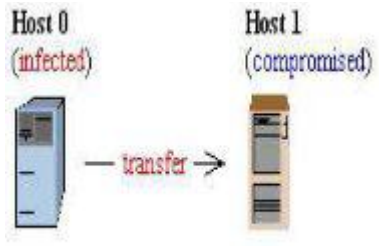
- Investigate for a victim.
- Utilizing the victim.
- Imitating it onto the victim.
- Running imitates to forward spread infection.
- Clandestine techniques used to hide itself



A. Target Discovery Stage



B. Exploitation Stage



C. Infection stage

D. Propagation stage

Worm Life Cycle Stages

2. Related Work

There are many studies on analyzing the behavior of worms and spreading nature Recently, some research have been done on new types of internet worm called nonoverlapping camouflaging worm that makes a trade-off between stealth and propagation speed on the internet.also some research have introduced and studied the propagation of internet worm .for example permutation scanning[7] worms avoid redundant scanning.In addition, worms use different scan strategies during different stages of propagation. In order to increase propagation efficiency, they use a local network or hit list to infect previously identified vulnerable computers at the initial stage of propagation [8] [9]. They may also use DNS, network topology and routing information to identify active computers instead of randomly scanning IP addresses. They split the target IP address space during propagation in order to avoid duplicate scans. Li *et al.* [10] studied a divide-conquer scanning technique that could potentially spread faster and stealthier than a traditional random-scanning worm. Ha *et al.* [8] formulated the problem of finding a fast and resilient propagation topology and propagation schedule for Flash worms. Yang *et al.* [9 studied the worm propagation over the sensor networks.Different from the above worms, which attempt to accelerate the propagation with new scan schemes, the Non-overlapping Camouflaging Worm studied in this paper aims to elude the detection by the worm defense system during worm propagation. Closely related, but orthogonal to our work, are the evolved active worms that are polymorphic in nature. Polymorphic worms are able to change their binary representation or signature as part of their propagation process. This can be achieved with self-encryption mechanisms or semantics-preserving code manipulation techniques. The Non-overlapping C-Worm also shares some similarity with stealthy port-scan attacks. Such attacks try to find out available services in a target system, while avoiding detection [11], [12]. It is accomplished by decreasing the port scan rate, hiding the origin of attackers, etc. Due to the nature of self-propagation, the Non-overlapping C-Worm must use more complex mechanisms to manipulate the scan traffic volume over time in order to avoid detection.

2.2 Worm Scanning Method

Random Scanning - In random scanning method, worm randomly searches the entire IP addresses to find vulnerable machines and select target IP addresses randomly, which leads to a fully connected topology with identical infection probability β for every edges .This method is not very useful because some hosts may be scanned more than once whereas some may not at all.[5][6]

1.2.Camouflaging Scanning- Camouflaging worm is a type of active worm.This is different from other scanning method such as random scan ,because it has a ability to control the worm over all scan rate over time.Therefore the propagation controlling nature of C-worm cause a slow down in propagation speed.Nevertheless ,by carefully controlling the c-worm can still infect as many computers as possible before it has been detected.[5]

1.2.3.Nonoverlapping Scanning- The non-overlapping scanning aims to evade repetitive scanning and so increase the stealth and speed of the worm. To this end, it divides the entire IP address space into several partitions, each of which is scanned by a different infectious host. One way to implement the non-overlapping scanning is as follows: each infectious host sequentially scans a logical address ring clockwise from its own address and infects each

susceptible host. When a host is infected, it does a jump to a random location on the ring and starts to scan IP addresses clockwise from this location. The reason for the jumping action is to avoid

duplicate scanning of IP addresses. An infectious host stops scanning after hitting a certain number of infected
hosts. Another way for implementing non-overlapping scanning is to use the divide and conquer strategy. In this strategy, the corrupted host divides its unscanned address space into two partitions and assigns one of them to the new infected host and keeps the other for itself[5].

2.3 Non-overlapping Camouflaging Worm

Non-overlapping worm makes corrupted hosts to collaborate with each other to ignore repetitive scanning of the same hosts . NOC-worm controls scan traffic volume during its propagation Camouflaging is a class of active worm and it has a capability to manipulate the scan traffic volume over time,non overlapping C- worm is to avoid the repetitive scanning and increase the stealth and propagation speed of worm.It has a advantages of both camouflaging and non- overlapping scanning. Corrupted hosts use the camouflaging scanning to remain the number of scanning less than the detection threshold of ITM or other worm detection
systems. It also engaged the non-overlapping scanning to avoid unemployed scans of the same hosts[5][4]

2.4 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 executable plays an important role in host- based detection systems. Many detection schemes fall under this category . In contrast, network-based detection systems detect worms primarily by monitoring, collecting, and analyzing the scan traffic (messages to identify vulnerable computers) generated by worm attacks. Many detection schemes fall under this category. Ideally, security vulnerabilities must be prevented to begin with, a problem, which must addressed by the programming language community. However, while vulnerabilities exist and pose threats of large-scale damage, it is critical to also focus on network-based detection. 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. The widely adopted worm detection framework consists of multiple distributed monitors and a worm detection center that controls the former . This framework is well adopted and similar to other existing worm detection systems, such as the Cyber center for disease controller , Internet motion sensor , SANS ISC (Internet Storm Center) [13], Internet sink , and network telescope . The monitors are distributed across the Internet and can be deployed at end- hosts, router, or firewalls etc. 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. For example, Venkataraman *et al.*[14] and Wu *et al.* in [15], proposed schemes to examine statistics of scan traffic volume, Zou *et al.* [16]presented a trend-based detection scheme to examine the exponential increase pattern of scan traffic , Lakhina *et al*[17] proposed schemes to examine other features of scan traffic, such as the distribution of destination addresses. Other works study worms that attempt to take on new patterns to avoid detection .Besides the above detection schemes that are based on the global scan traffic monitor by detecting traffic anomalous behavior, there are other worm detection and defense schemes such as sequential hypothesis testing for detecting worm-infected computers, payload-based worm signature detection[4] . In addition, Cai *et al.* in [18] presented both theoretical modeling and experimental results on a collaborative worm signature generation system that employs distributed fingerprint filtering and aggregation and multiple edge networks. In a state-space feedback control model that detects and control the spread of these viruses or worms by measuring the velocity of the number of new connections an infected computer makes. Despite the different approaches described above, we believe that detecting widely scanning anomaly behavior

continues to be a useful weapon against worms, and that in practice .

3. Detecting The C-Worm

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 Function* (PDF) of *SFM* for both C-Worm and normal non-worm scan traffic , respectively. The normal non-worm scan traffic data shown is based on real-world traces collected by the *ISC* · 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., $SFM \in (0.02, 0.04)$ has much higher density compared with other magnitudes). The C-Worm data is based on 800 C-Worms attacks generated by varying attack parameters such as P(t) and Mc(t)., we know that the *SFM* value of the C-Worm attacks is high (e.g., $SFM \in 0.5, 0.6$ has high density). From the above two figures, we can observe that there is a clear demarcation range of $SFM \in (0.3, 0.38)$ 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.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. In reality, the above recurring manipulations involve steady increase followed by a decrease in the scan traffic volume. Notice that the frequency domain analysis will require more samples in comparison with the time domain analysis, since the frequency domain analysis technique such as the Fourier transform, needs to derive power spectrum amplitude for different frequencies. In order to generate the accurate spectrum amplitude for relatively high frequencies, a high granularity of data sampling will be required. In our case, we rely on Internet threat monitoring (ITM) systems to collect traffic traces from monitors (motion sensors) in a timely manner. As a matter of fact, other existing detection schemes based on the scan traffic rate [20], variance [21] or trend [19] will also demand a high sampling frequency for ITM systems in order to accurately detect worm attacks. Enabling the ITM system with timely data collection will benefit worm detection in real-time.

3.2 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 propagation. 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. 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.

To conduct spectrum analysis, we consider a detection-sliding window Wd in the worm detection system. Wd consists of q (> 1) continuous detection sampling windows and each sampling window lasts Ws. The detection-sampling window is the unit time interval to sample the detection data (e.g., the destination count). Hence, at time i, within a sliding window Wd, there are q samples denoted by (X(i − q − 1), X(i − q−2),...,X(i)), where X(i−j−1) (j ∈(1,q)) is the j-th destination count from time i − j − 1 to i − j.

In 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 4.1. In our worm detection scheme, the detection data (e.g., destination counter), is further processed 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.

3.2.1 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

$R_X(L) = E[X(t)X(t + L)]$.

In Formula, 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,

$$\psi(R_X[L], K) = \sum_{n=0}^{N-1} (R_X[L]) \cdot e^{-j2\pi Kn/N},$$

Where K = 0, 1, . . . , N − 1. 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. The *PSD* of C-Worm scan traffic shows spikes or noticeably higher concentrations at a certain range of the spectrum.

3.2.2 Spectral Flatness Measure (SFM)

We measure the flatness of P.S.D. 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 [62], [63]. It can be expressed as,

$$SFM = \frac{[\prod_{k=1}^{n} S(f_k)]^{\frac{1}{n}}}{\frac{1}{n}\sum_{k=1}^{n} S(f_k)}$$

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 [63], [64]. 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 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).

CONCLUSION

An Internet worm is a program or algorithm that replicates itself over a computer network and invariably performs malicious actions such as shutting a machine down or using up its resources. No network of computers is impenetrable or immune to attacks of this kind. An active worm refers to a malicious software program that propagates itself on the Internet to infect other hosts. The propagation of the worm is based on exploiting vulnerabilities of hosts on the Internet. The Non-overlapping camouflaging worm(NOC-Worm) is a new type of worm.The NOC-Worm has a self-propagating behavior similar to traditional worms, i.e., it intends to rapidly infect as many vulnerable computers as possible. However, the NOC- Worm is quite different from

traditional worms in which it camouflages any noticeable trends in the number of infected computers over time. The camouflage is achieved by manipulating the scan traffic volume of worm-infected computers. We present a NOC- Worm detection method that uses two step procedures that combines a first stage change point detection with a second stage growth rate inference to confirm the existence of a worm.

REFERENCES

[1] Wei Yu, Xun Wang, Prasad Caylam, Dong Xuan ,and Wei Zhao, "Modeling and Detection of Camouflaging Worm",IEEE Transactions on Dependable and Secure Computing, vol. 8, no. 3, pp. May-June, 2011.

[2] Yini Wang, Sheng Wen, Yang Xiang and wanlei Zhou,"Modeling the Propagation of Worms in Network :A Survey,IEEE Communication Survey and Tutorials ,June 2013

[3] S.Preetha ,"Modeling and Detection of Camouflaging Worm using IP Traceback",International Journal of Computer Science & Communication Networks,Vol 2(2), 190-193

[4] Jeevakatiravan, D.Hema priyadarshani,C.Chellapan ,R Dhanalakshmi," A Novel Approach for Detecting Smart Camouflaging Worm" ,in Proceeding on Theoretical and applied information technology. Vol.47, no.2, 2013

[5][1] Ahad Azarian and Mahdi Abadi," On the Trade-off between Stealth and Propagation Speed of Internet Worm", in Proceeding of IEEE International ISC Conference on Information Security and Cryptology (ISCISC), Yazd,Iran ,August 2013

[6]Juan Wang, Chengyi Xia ,Qifeng Liu," A Novel Model For the Internet Worm Propagation" in Proceedings of the IEEE Conference on Natural Computation (ICNC 2010),China ,2010.

[7] P.K.Manna, S. Chen, and S.Ranka,"Exact Modeling of propagation for permutation-scanning worms,"pp.1696-1704, in Proceedings of the 27th IEEE International Conference on Computer Communication (INFOCOM '08), Phoenix,AZ,USA,2008

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

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

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

[11] Linux.com, Understanding Stealth Scans: Forewarned is Forearmed, http://security.itworld.com/4363/LWD010321vcontrol3/page1.html.

[12] Solar Designer, Designing and Attacking Port Scan Detection Tools, http://www.phrack.org/phrack/53/P53- 13.

[13] SANS, Internet Storm Center, http://www.dshield.org.

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

[15] J. Wu, S. Vangala, and 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.

[16] C. Zou, W. B. Gong, D. Towsley, and 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.

[17] A.Lakhina, M.Crovella, and C.Diot, "Mining anomalies using traffic feature distribution,"in Proceedings of ACM SIGCOMM, Philadelphia, PA,August 2005.

[18] M. Cai, K, Hwang, J. Pan, and C. Papadopoulos, "Wormshield: Fast worm signature generation with distributed fingerprint aggregation." IEEE Transaction on Dependable and Secure Computing, vol.4,no.2, pp. 88-104-2007