Modeling and Simulation of Scanning Worms: Network Traffic Analysis

Shin-Hun Kang and Jae-Hyun Kim
School of Electrical and Computer Engineering
Ajou University,
Suwon, Korea
E-mail: {cnonyk, jkim}@ajou.ac.kr

Abstract

Scanning worms increase network traffic load because they randomly scan network addresses to find vulnerable hosts that are susceptible to infection. Since propagation speed is faster than human reaction, scanning worms cause severe network congestion. So we propose an algorithm which can detect scanning worms based on statistical characteristics of network traffic. We modeled applications and profiles of scanning worms using OPNET and evaluated statistical characteristics such as variance, variance to mean ratio (VMR) and correlation coefficient of the network traffic. The proposed algorithm not only reduced computational complexity but also improved detection accuracy compared with existing algorithm.

I. Introduction

A worm is a self-replicating computer program that uses a network to send copies of itself to other computers without any user intervention. Scanning worms send packets with randomly generated addresses to find vulnerable hosts that are susceptible to infection. Since propagation speed is faster than human reaction, scanning worms increase network traffic load and result in severe network congestion. So an early detection system which can automatically detect scanning worms is needed to protect network from those attacks. Although many studies are conducted to detect scanning worms, most of them are focusing on the method using packet header information. And there have been few studies on the method using general traffic characteristics of scanning worms. The method using packet header information has high accuracy, but the detection delay is long since it must examine the header information of all packets entering or leaving the network. On the other hand, the method using traffic characteristics has somewhat low accuracy, but it can detect scanning worms efficiently in a short time because it does not need to examine all the packets. Therefore, we propose an algorithm to detect scanning worms using network traffic characteristics and verified the proposed algorithm by computer simulation, and compared with existing algorithm.

This paper is organized as follows. In section II we introduce some related works on scanning worm detection and in section III we describe the traffic characteristics of scanning worm. In section IV, we describe our detection algorithm. In section V, we evaluate the performance of proposed algorithm and present simulation results. Concluding remarks are offered in section VI.

II . Related Work

There have been two kinds of method to detect scanning worms: the method using packet header information and the method using network traffic volume. Generally, scanning worm traffic has random destination IP addresses to propagate more widely. And port numbers are fixed because scanning worms use vulnerability of specific service. So we can detect scanning worms using packet header information such as source and destination IP address, source and destination port number. There are some researches based on this idea. One approach is plotting a packet using its source IP address, destination IP address, and the destination port number in a 3-dimensional space graphically and observe whether a regular pattern appears[2]. Noh observes randomness of IP address and port number by computing entropy of IP address and port number[3]. Kim proposed a flow based detection that examine properties of flow such as flow size and packet count using packet header information[4]. Zou estimates infection rate by Kalman filter using scan monitoring system[5]. These methods are accurate because much packet header information is available. But the detection system is complex and delay is long because all packets are examined. On the other hand, the method using traffic characteristics is fast and efficient because it does not examine all packets. Roh proposed an algorithm which calculates packet count to traffic volume ratio(CVR)[6]. This method is fast and efficient but the detection accuracy is relatively low due to insufficient information. In order to detect scanning worms efficiently and accurately, we propose an algorithm using statistical characteristics of network traffic.

III. Traffic Characteristics of Scanning Worm

Scanning worms propagate so fast that cause severe network congestion than any other worms. For instance, in July 2001, the CodeRedIv2 infected over 359,000 computers within 14 hours[7]. In January 2003, the Slammer infected more than 75,000 computers in less than 10 minutes[1]. During the Blaster attack of August 2003, more than 500,000 computers were infected within a few hours[8]. In March 2004, the Witty infected over 12,000 computers in less than 45 minutes[9].

Table I represents the traffic characteristics of scanning worms. Since CodeRedII broke out while CodeRedIv2 was spreading, it is not possible to know exact scan rate. Thus the scan rate of CodeRedII is omitted from the table. From the table, we can see that the source or destination port number is fixed and destination IP address is random or partially random. CodeRedIv2, CodeRedII and Blaster have large packet size and slow scan rate while Slammer has small packet size and fast scan rate. If packet size and scan rate are considered individually, it is

- "This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement)" (IITA-2006-(C1090-0602-0011))
- This work was supported by the IT R&D program of MIC/IITA. [2006-S058-01, Development of Network/Service Control Technology in All-IP based Converged network]

difficult to understand the properties of scanning worm. In this paper we consider traffic volume which is product of packet size and scan rate because it is always large. Therefore we analyze the change of network traffic volume when a scanning worm breaks out. And we propose a detection algorithm using statistical characteristics of network traffic volume.

Name	Destination IP address	Destination port number	Packet size (byte)	Scan rate (packets/ sec)
CodeRedIv2	Random	80	3569	11
CodeRedII	12.5% : random 50% : in the same class B 37.5% : in the same class C	80	3818	.
Slammer	Random	1434	404	4000
Blaster	40% : in the same class C 60% : random	135	6176	15
Witty	Random	Random (Source port number:4000)	796~1307	357

Table I . The properties of scanning worms

IV. Scanning Worm Detection Algorithm

It is possible to distinguish scanning worm traffic from normal traffic by monitoring the variation of network traffic volume. A computer infected by CodeRedIv2 sends 11 packets per second and the packet size is 4 Kbytes. Since the packet size is large and the inter-arrival time is relatively long, the traffic volume variation is large. Thus the variance of traffic volume becomes large. In the case of Slammer, scan rate is 4000 packets/sec and the packet size is 404 bytes. Although the packet size is small, it generates a lot of packets and increases the variance. Based on the facts that scanning worms increase the traffic volume much more than normal traffic does, we can detect scanning worms by calculating the variance of traffic volume. By computer simulation, we verified that the variance of traffic volume increases when a scanning worm breaks out. Therefore we suggest three criteria to detect scanning worms: variance, variance to mean ratio(VMR), and correlation coefficient.

Variance

We consider variance as the first criterion because it is the simplest way to analyze network traffic. Since scanning worms result in a sudden increase of traffic volume, the variance of traffic volume during specific time periods also increase. When we measure the traffic volume from a network link, we define $X(t)$ as sum of the number of bits of packets between $t-1$ and t second. And W is defined as the window size, which determines how many samples will be used to calculate. Then the variance at the time t, $\text{var}(t)$ is given by

$$\text{var}(t) = \begin{cases} \dfrac{\sum_{k=t-W+1}^{t} \left[X(k) - \mu(t)\right]^2}{W}, & if\ k \geq 0 \\ 0, & if\ k < 0 \end{cases} \qquad (1)$$

where

$$\mu(t) = \begin{cases} \dfrac{\sum_{k=t-W+1}^{t} X(k)}{W}, & if\ k \geq 0 \\ 0, & if\ k < 0. \end{cases} \qquad (2)$$

At time t, if $\text{var}(t)$ is α times greater than the average value of variance from $t-W-1$ to $t-1$, then we consider that scanning worm is detected.

Variance to Mean Ratio(VMR)

Since variance is a simple measure of statistical dispersion, variance can increase when a scanning worm does not exist. And the variance can be different even though the same amount change happens because it is dependent on the whole traffic volume. VMR is defined as the ratio of the variance to the mean and can be written as

$$\text{vmr} = \frac{\sigma^2}{\mu}. \qquad (3)$$

VMR is independent of the whole traffic volume. So we can focus on the variation without influence of whole traffic volume. VMR can be obtained by using (1), (2)

$$\text{vmr}(t) = \frac{\text{var}(t)}{\mu(t)}. \qquad (4)$$

At time t, if $\text{vmr}(t)$ is β times greater than the average value of variance from $t-W-1$ to $t-1$, then we consider that scanning worm is detected.

Correlation Coefficient

Since normal traffic can also increase variation of network traffic volume, simple measure of variation of network traffic volume is not enough to determine the existence of scanning worm. Thus we considered correlation coefficient as a third criterion, which indicates the strength of a linear relationship between two random variables. We keep two sliding windows overlapping each other, and observe correlation coefficient between two windows. Correlation coefficient is 0 in the case of an independent relationship, 1 in the case of an increasing linear relationship, and -1 in the case of a decreasing linear relationship. If correlation coefficient keeps large value near 1 over some period, it means that network traffic is increasing continuously and we can think that scanning worm is propagating. Let σ_X, σ_Y be the standard deviation of random variables X, Y respectively, and $\text{cov}(X,Y)$ means covariance of X, Y. The correlation coefficient of X and Y, $\text{corr}(X,Y)$ is defined by

$$\text{corr}(X,Y) = \frac{\text{cov}(X,Y)}{\sigma_X \sigma_Y} \qquad (5)$$

where

$$\text{cov}(X,Y) = E\left[(X-\mu_X)(Y-\mu_Y)^T\right]. \qquad (6)$$

So correlation coefficient can be obtained by

$$\text{corr}(t) = \frac{E\left[(X-\mu_X)(Y-\mu_Y)^T\right]}{\sqrt{E(X^2)-E^2(X)}\sqrt{E(Y^2)-E^2(Y)}} \qquad (7)$$

where

$$X = \left[X\left(t-dW+1\right) \;\; X\left(t-dW+2\right) \;\; \cdots \;\; X\left(t-dW+W\right) \right],$$
$$Y = \left[X\left(t-W+1\right) \;\; X\left(t-W+2\right) \;\; \cdots \;\; X\left(t\right) \right]$$

where d is a parameter which decides amount of overlapping period. And A^T is transpose of a matrix A and W is even for computational convenience.

At time t, if $\mathrm{corr}(t)$ is greater than the threshold value γ, then we consider that scanning worm is detected.

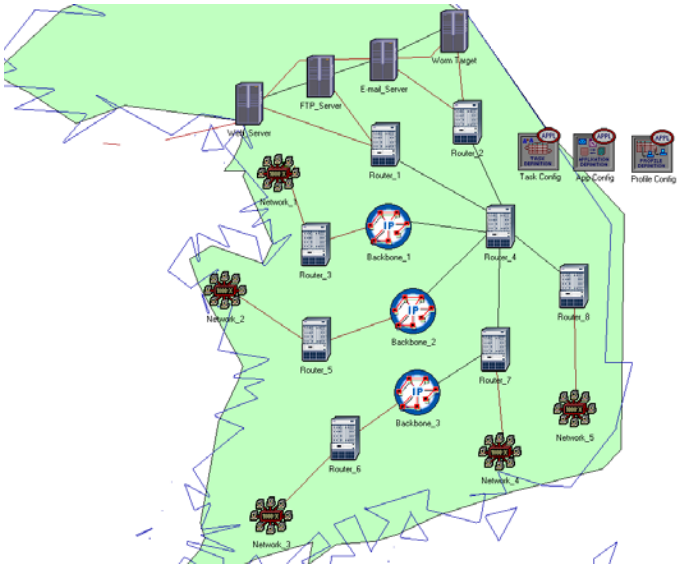


Figure 1. Reference network architecture

V. Performance Analysis

Simulation Model

We designed network model that generates normal traffic and scanning worm traffic using OPNET. We select CodeRedIv2, Slammer and Witty as scanning worm traffic because CodeRedIv2 has packets of large size and the lowest scan rate, and Slammer has the smallest packet size and the highest scan rate, and Witty has middle packet size and middle scan rate. If we can detect these worms that have various properties, we can probably detect other worms too. We analyze network traffic characteristics when only normal traffic exist and when scanning worms are propagating.

The reference network architecture in this paper is shown in Fig. 1. There are three networks that generate traffic and six corresponding servers. Normal traffics such as web browsing, e-mail, and FTP services start 100 seconds after simulation start. And CodeRedIv2 or Slammer or Witty worm traffic start 400 seconds after simulation start. We measured network traffic volume at the backbone link, and analyzed by methods proposed in chapter Ⅳ.

Detail Settings

Each network generates 8 kinds of traffic types: HTTP, FTP, E-mail, Video, Voice, CodeRedIv2, Slammer and Witty. Detail settings are summarized in Table II. Traffic models are from the standard models provided by OPNET. We use “Task Config” node to define scanning worm applications and profiles. Fig. 2 shows task specification of scanning worms. We referred to [1], [7] and [9] to configure CodeRedIv2, Slammer and Witty worm traffic models.

Traffic Measurement and Analysis

We set up a four case scenario: only normal traffic, normal traffic with CodeRedIv2, normal traffic with Slammer and normal traffic with Witty. Fig. 3 shows the number of bits per second measured at backbone link. Normal traffics start at 100 second and worm traffics start from 400 second in sequence and stop after 300 seconds. While CodeRedIv2 increases the whole traffic volume negligibly, Slammer increases the whole traffic substantially. For each case, we evaluated variance, VMR and correlation coefficient of network traffic volume. The detection parameters are set as follows: W =60, d =5/3, α =1.4, β =1.5, γ =0.15.

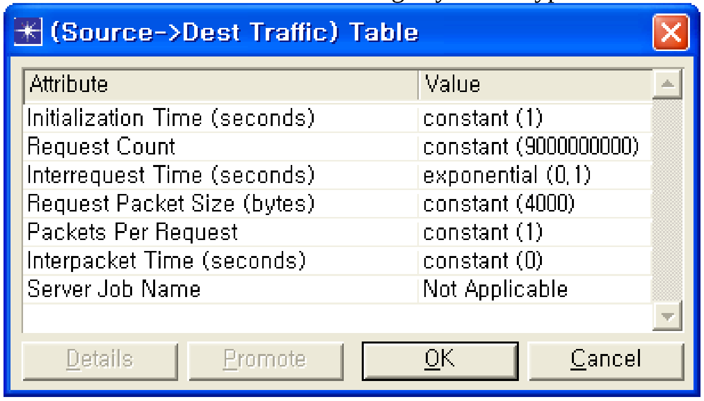
Fig. 4 shows the CVR which is used in the existing algorithm proposed in [6]. Slammer shows higher CVR than normal traffic because it has high scan rate and small packet size. The CVR of CodeRedIv2 and Witty does not increase or decrease noticeably during worm propagation period. So we can detect Slammer by using CVR, but CodeRedIv2 and Witty is not detectable.

In Fig. 5, we can see that variances of scanning worm traffics increase substantially when the worms break out. Therefore it is possible to detect scanning worms using variance. Before 200 second, it is unstable because of simulation initiation process. So we are not interested in the values before 200 second.

Attribute		Value	
Voice	Silence Length (sec)	Exponential (0.65)	
	Talk Spurt Length (sec)	Exponential (0.352)	
	Encoder Scheme	G.711	
	Voice Frames per Packet	1	
	Start Time (sec)	Uniform (100,110)	
Video	Frame Inter-arrival Time	30 frames/sec	
	Frame Size Information	352 x 240 pixels	
	Start Time (sec)	Uniform (100,110)	
FTP	Inter-Request Time (sec)	Exponential (360)	
	File Size (bytes)	Constant(50000)	
	Start Time (sec)	Uniform (100,110)	
E-mail	Send Inter-arrival Time (sec)	Exponential (360)	
	Receive Inter-arrival Time (sec)	Exponential (360)	
	E-mail Size (bytes)	Constant(2000)	
	Start Time (sec)	Uniform (100,110)	
HTTP	HTTP Specification	HTTP 1.1	
	Page Inter-arrival Time (sec)	Exponential (60)	
	Page Properties	Object Size(bytes)	Number of Object
		Constant(1000)	Constant(1)
		Uniform(500,2000)	Constant(5)
	Start Time (sec)	Uniform (100,110)	
Code RedI v2	Inter-Request Time (sec)	Exponential (0.1)	
	Request Packet Size (bytes)	Constant(4000)	
	Duration (sec)	Constant (300)	
	Start Time (sec)	Constant (400, 430, 454, 473, 488)	
Slammer	Inter-Request Time (sec)	Exponential (0.00025)	
	Request Packet Size (bytes)	Constant (404)	
	Duration (sec)	Constant (300)	
	Start Time (sec)	Constant (400, 430, 454, 473, 488)	
Witty	Inter-Request Time (sec)	Exponential (0.00025)	
	Request Packet Size (bytes)	Uniform (796,1307)	

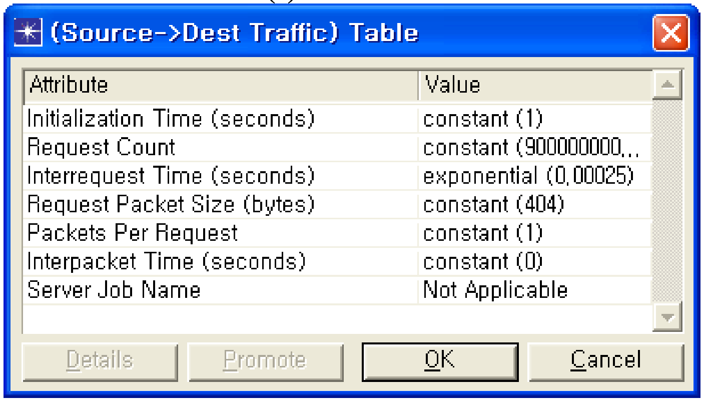
	Duration (sec)	Constant (300)
	Start Time (sec)	Constant (400, 430, 454, 473, 488)

Table II . Detail settings by traffic type


(Source->Dest Traffic) Table

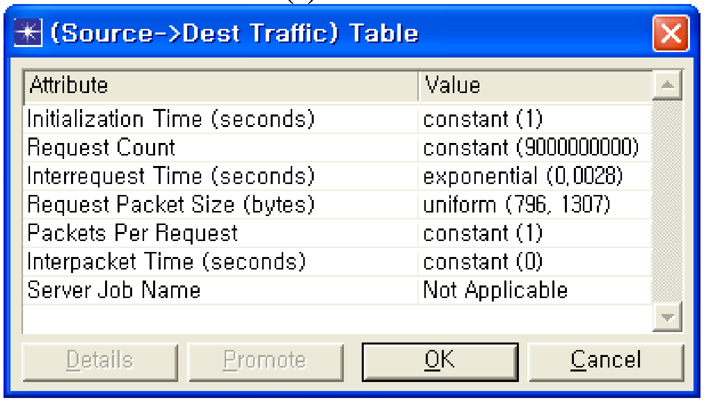
Attribute	Value
Initialization Time (seconds)	constant (1)
Request Count	constant (9000000000)
Interrequest Time (seconds)	exponential (0.1)
Request Packet Size (bytes)	constant (4000)
Packets Per Request	constant (1)
Interpacket Time (seconds)	constant (0)
Server Job Name	Not Applicable

(a) CodeRedIv2


(Source->Dest Traffic) Table

Attribute	Value
Initialization Time (seconds)	constant (1)
Request Count	constant (900000000...
Interrequest Time (seconds)	exponential (0.00025)
Request Packet Size (bytes)	constant (404)
Packets Per Request	constant (1)
Interpacket Time (seconds)	constant (0)
Server Job Name	Not Applicable

(b) Slammer


(Source->Dest Traffic) Table

Attribute	Value
Initialization Time (seconds)	constant (1)
Request Count	constant (9000000000)
Interrequest Time (seconds)	exponential (0.0028)
Request Packet Size (bytes)	uniform (796, 1307)
Packets Per Request	constant (1)
Interpacket Time (seconds)	constant (0)
Server Job Name	Not Applicable

(c) Witty

Figure 2. Task Specification of scanning worms

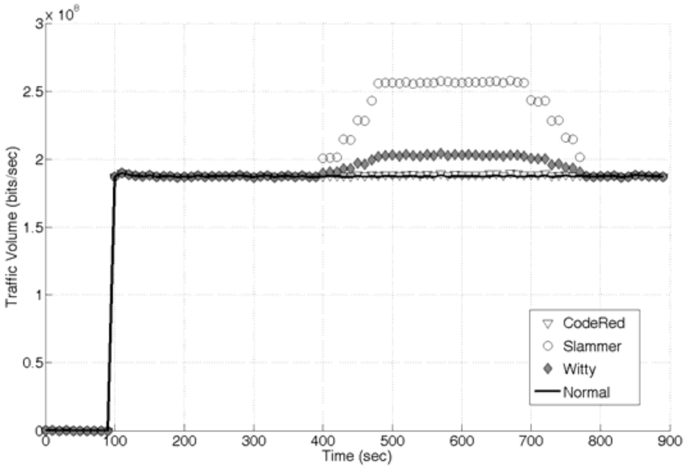


Figure 3. Measured traffic volume

Fig. 6 shows the VMR of network traffic. The VMRs of worm traffic increase drastically when the worms break out. Fig. 7 represents the correlation coefficient of network traffic and the correlation coefficient of normal traffic does not exceed an upper limit while worm traffics increase the correlation coefficient drastically during worm propagation period. We applied our detection algorithm to the values evaluated above. Detection results are shown in Fig. 8 and Table III.

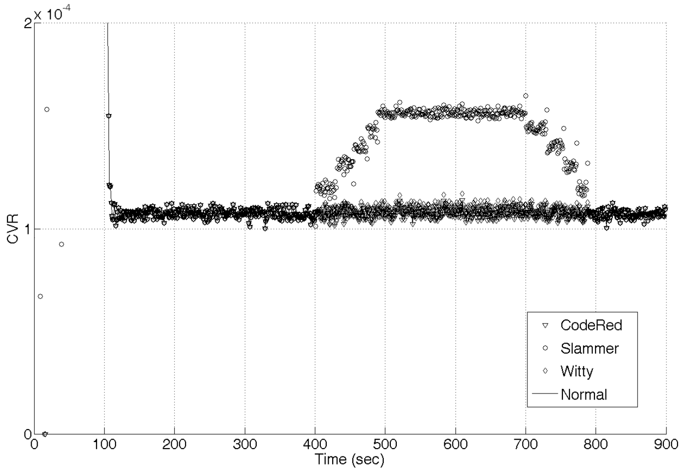


Figure 4. CVR of network traffic volume

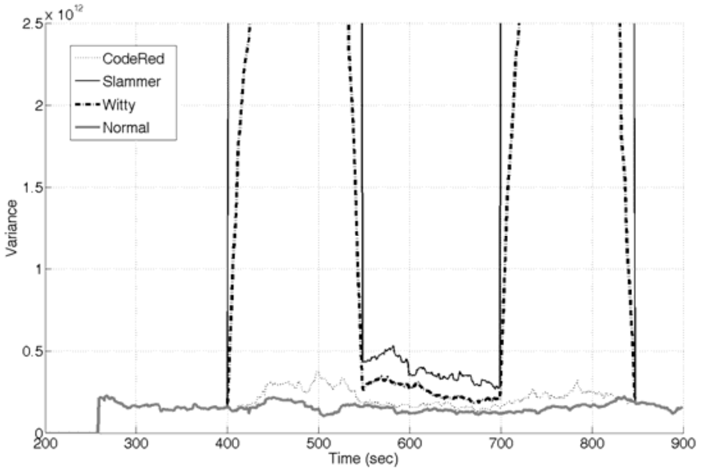


Figure 5. Variance of network traffic volume

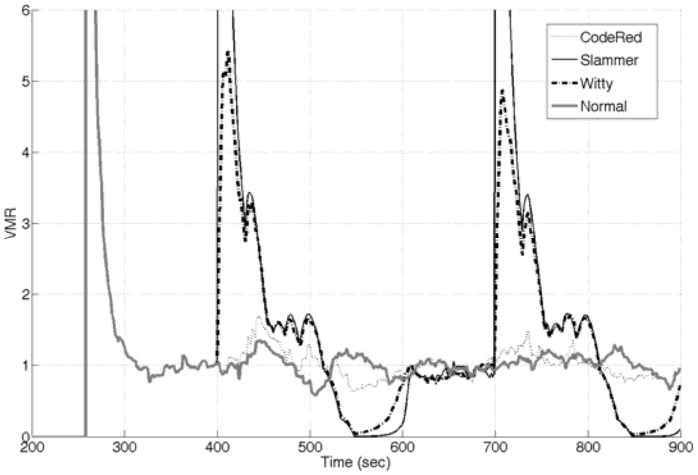


Figure 6. VMR of network traffic volume

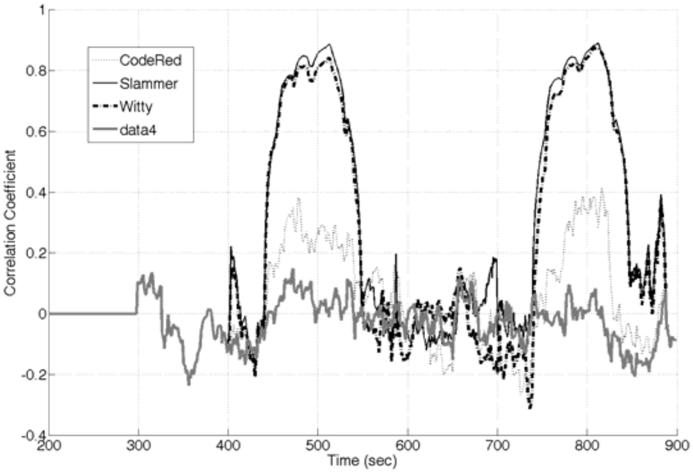


Figure 7. Correlation coefficient of network traffic volume

Detection delay is defined as the time from when a worm breaks out to when a worm is detected. We assume that there is no processing delay. If there are only normal traffics, no worm is detected. In other words, our algorithm has no false positive probability. If there are worms propagating through the network, all worms are detected by our algorithm. In other words, there is no false negative probability. In the point of detection speed, detection using variance is the fastest and using correlation coefficient is the slowest. It is difficult to detect CodeRedIv2 and detection delay is relatively long. Slammer is detected within 4 seconds for all criteria because slammer increases network traffic substantially.

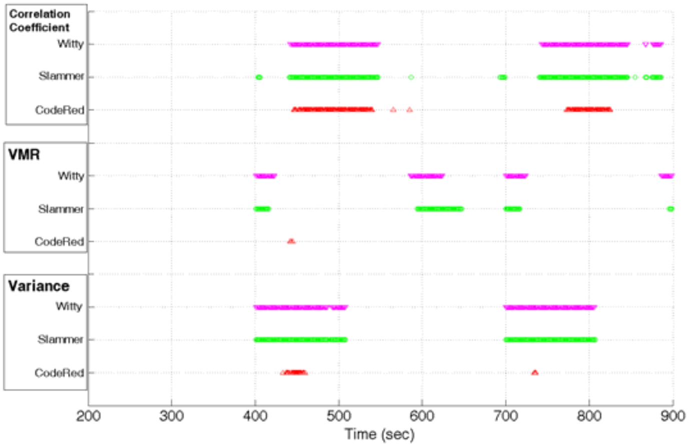


Figure 8. Detection results

Criteria	Normal	Normal + CodeRedIv2	Normal + Slammer	Normal + Witty
Variance	No detection	34	2	2
VMR	No detection	43	2	2
Correlation coefficient	No detection	47	4	43

Table III . Detection delay (sec)

V. Conclusion

It is important to detect scanning worms as fast as possible. Detection using packet header information is accurate but slower than detection using traffic volume. In this paper we proposed a detection algorithm which analyzes statistical characteristics of network traffic volume. We performed computer simulation to verify our algorithm, and computed variance, VMR, and correlation coefficient of network traffic volume. The results show that our algorithm can detect scanning worms accurately within a short time.

The following points are left as future problems. First, traffic data used here is generated by computer simulation and is different from real network traffic. Thus we are measuring real network traffic to verify our algorithm. Second, we analyzed only traffic volume(bits per second) in this paper. We can use more factors such as packets per second and inter-arrival-time. Third, detection using traffic characteristics is fast and efficient, but not sufficient to quarantine worms because we cannot know source or destination of attack. Therefore we are planning to combine two methods that using packet header information and traffic characteristics.

References

[1] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver, “Inside the Slammer Worm,” IEEE Security & Privacy, pp. 33-39, Jul./Aug. 2003.

[2] H. Kim, I. Kang, and S. Bahk, "Real-Time Visualization of Network Attacks on High-Speed Links,” IEEE Network, pp. 30-39, Sep./Oct. 2004.

[3] S. Noh, C. Lee, K. Ryu, K. Choi, and G. Jung, "Detecting Worm Propagation Using Traffic Concentration Analysis and Inductive Learning,” Lecture Notes in Computer Science, 3177(1), pp. 402-408, 2004.

[4] M. Kim, H. Kang, S. Hong, S. Chung, and W. Hong, "A Flow-based Method for Abnormal Network Traffic Detection," in Proc. IEEE/IFIP NOMS, 2004, pp. 599-612.

[5] C. Zou, W. Gong, D. Towsley, and L. Gao, "The Monitoring and Early Detection of Internet Worms," in Proc. 10th ACM conference on Computer and communication security, 2003, pp. 190-199.

[6] B. Roh and S. Yoo, "A Novel Detection Methodology of Network Attack Symptoms at Aggregate Traffic Level on Highspeed Internet Backbone Links," Lecture Notes in Computer Science, 3124, pp. 1226-1235, Aug. 2004.

[7] D. Moore, C. Shannon, and J.Brown, "Code-Red: a case study on the spread and victims of an Internet worm," in Proc. Second Internet Measurement Workshop, 2002, pp. 273-284.

[8] "W32.Blaster.Worm," [Online]. Available: securityresponse.symantec.com/avcenter/venc/data/w32.blaster.worm.html

[9] C. Shannon and D. Moore, "The Spread of the Witty Worm," IEEE Security & Privacy, pp. 46-50, Jul./Aug. 2004.

[10] J. Kim and S. Kang, "Detection Algorithm of Scanning Worms using network traffic characteristics," in Proc. WISC, 2006, pp. 71-82.