Optimization of Nachi Spreads

Satoshi Onoda s1080057 Supervised by Hiroshi Toyoizumi

Abstract

This paper presents a method to find the optimum scan rate of NACHI with some conditions. For this purpose, the relation between the worms NACHI and MSBLAST, which NACHI can kill were modeled by using a mathematical method and the data obtained by an experiment in which NACHI was run. As a result, we obtained a method to find it.

1 Introduction

Damage to PCs by computer viruses has caused significant trouble for several years. Today, several countermeasures exist to combat viruses such as an antivirus software and firewalls. Now, there is another method for protection, which is to release a good worm to kill malicious worms in the network system. However, this new method is not commonly used, since there are particular problems with it. The first problem relates to law, because a good worm can spread regardless of a user's will. Next, a good worm has to spread equal to or more quickly than a malicious worm, so a good worm may need to send a lot of packets. NACHI, which is one of such good viruses and is used in this research, also sends a great many packets. Therefore, a good worm can also have a bad influence on the network. But this method has also an advantage. For example, a good worm can automatically update PC security and remove a malicious worm, so people who don't worry about faulty security can be protected from malicious worms. In other words, it can control the spread of malicious worms. Actually, there are some good worms, for example, CODEGREEN and NACHI.

At the University of Aizu, a graduate student has researched the relationship between CODEREDI I and CODEGREEN [1]. His research focused on whether CODEGREEN is effective for preventing GODEREDII from spreading. CODEREDII is a ferocious worm. CODEGREEN is able to destroy CODEREDII. It was proved effective by use of a mathematical model, but he had not referred to a defect of good worm. So, we guided NACHI to be a better worm without sending a lot of packets. And, another graduate student, at the University of Aizu, studied about MSBLAST [7]. The author obtained some data and information of the MSBLAST from his study. As Figure 1, first, MSBLAST spreads on the network. Next, NACHI spreads, and if the computer infected by NACHI has already infected by MSBLAST, NACHI kill MSBLAST. Moreover, there is Predators [5] that is good will mobile codes combat against computer viruses. If you program Predators for MSBLAST, multiplication number may be determined by the method found in this paper. And, we obtained some knowledge about computer viruses by a book, Malicious Mobile Code [10].

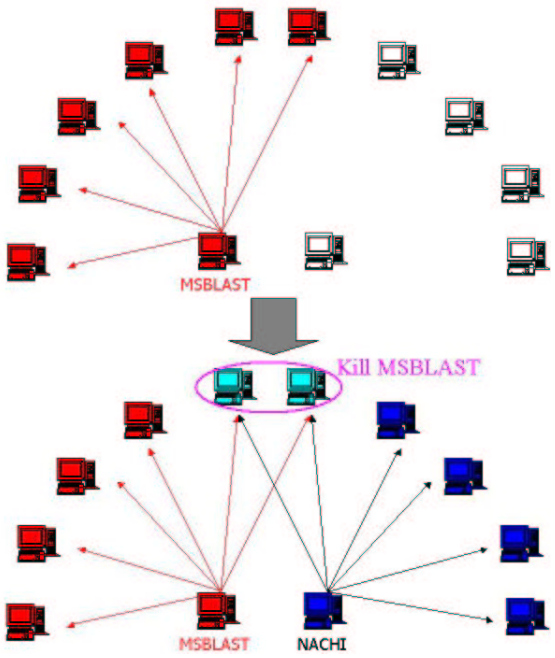

Figure 1: Spread of MSBLAST and NACHI

The purpose of this study was to amend NACHI. The optimum rate was found to scan newly infective computers. In this paper, optimum means at least of what NACHI can terminate MSBLAST with some conditions. The obtained rate is effective in containing the activities of MSBLAST without causing a bad influence on the network.

2 Target Computer Viruses

We explain about MSBLAST and NACHI, the target of our study, as follows. Figure 2 shows procedure of MSBLAST in infecting and Figure 3 shows procedure of NACHI by a flow chart.

Table 1: MSBLAST

Official name	MSBLAST.A
Type	Worm
Size of virus	6,176 bytes
Platform	Windows2000, WindowsXP
Discovered	August 11, 2003

2.1 MSBLAST

Spreading throughout a network, MSBLAST sends TCP SYN packets to IP addresses that are made by two algorithms [2]. Discovering computer, which is alive, MSBLAST exploits the RPC DCOM Buffer Overflow and instructs a target computer to download its copy using TFTP. However, successful probability of a download is about 20% [7]. MSBLAST has to send suitable data to Windows2000 and WindowsXP in order for it to exploit. For example, if MSBLAST sends data for WindowsXP to Windows2000, MSBLAST cannot infect the computer. Actually, the ratio of sending data for Windows2000 to WindowsXP is 1:4. MSBLAST launches a thread that performs a DDoS attack against windowsupdate.com in certain conditions [2].

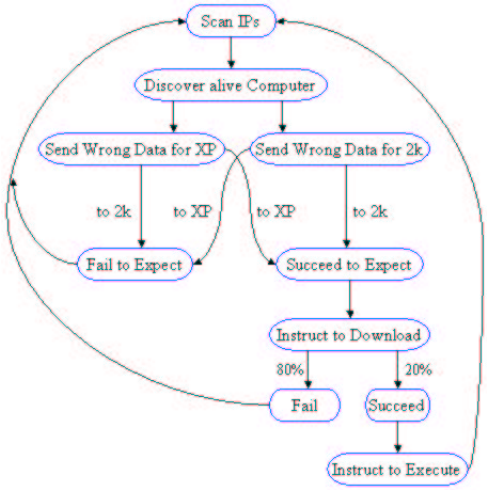

Figure 2: Flow Chart of MABLAST

2.2 NACHI

Table 2: NACHI

Official name	NACHI.A
Type	Worm
Size of virus	10,240 bytes
Platform	Windows2000, WindowsXP
Discovered	August 18, 2003

NACHI acts as Figure 3. First, NACHI terminates the MSBLAST process and then deletes the file, MSBLAST.EXE, from the Windows system folder. Next, NACHI patches the system against the RPC DCOM Buffer Overflow Exploit [8] that MSBLAST exploits in order to spread in the network. For this purpose, NACHI checks the operating system version and local information. If the operating system is Windows 2000 or XP and has no patch, NACHI downloads the appropriate patch from the designated Microsoft Web site and reboots the system. Next, NACHI does spreading activities: ICMP ECHO request packets are sent by running 300 threads to look for computers to infect by the different four algorithms [2]. Discovering a computer, which is alive, NACHI exploits the RPC DCOM Buffer Overflow and a WebDAV exploit [8], and instructs target computers to download the worm copy from the infected computer using TFTP. NACHI then resumes looking for the computers to infect [3].

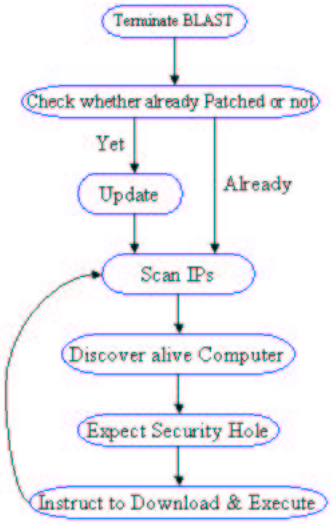

Figure 3: Flow Chart of NACHI

3 Mathematical Method

Now, we would like to model the relation between MSBLAST and NACHI. Let $x(t)$ be the number of the computers infected by MSBLAST at time t, $y(t)$ be the number of the computers infected by NACHI at time t, r be the propagation rate of MSBLAST, a be the propagation rate of NACHI, and b be the rate that a NACHI kills MSBLAST per second, then we obtain the following system of differential equations:

$$\frac{dx}{dt} = rx - by, \tag{1}$$

$$\frac{dy}{dt} = ay, \quad \text{and} \tag{2}$$

$$(x(0), y(0)) = (x_0, y_0).$$

The following equation can explain the meaning of formula (1) and (2):

$$x(t + \Delta t) = x(t) + r\Delta t x(t) - b\Delta t y(t). \quad (3)$$

Transforming formula (1) by definition of differentiation, we obtained formula (3). The left-hand side of formula (3) refers to the number of infected computers by MSBLAST at time $t + \Delta t$. The right side refers to the sum of the number of infected computers by MSBLAST at t, the increase of infected computers from t to $t + \Delta t$, and the decrease of infected computers from t to $t + \Delta t$. We can explain formula (2) in the same way as formula (1). Solving these simultaneous equations, we obtain formulas, which refer to the number of computers infected by MSBLAST and NACHI was modeled at time t. Furthermore, the scan rate of NACHI was found that could control the propagation of MSBLAST before MSBLAST propagate to optional value *max*. To solve formula (2), the following formula, which models NACHI was obtained:

$$y(t) = y_0 e^{at}. \quad (4)$$

Next, when formula (4) was substituted for formula (1) and the obtained a got differential equation was solved, the following formulas were obtained which models MSBLAST.

i)$a \neq r$

$$x(t) = x_0 e^{rt} + \frac{by_0(e^{rt} - e^{at})}{a - r} \quad (5)$$

and ii)$a = r$

$$x(t) = (x_0 - by_0)e^{rt}. \quad (6)$$

To know the actual propagation rate of NACHI and MSBLAST, the experiment in the next section was conducted.

4 Experiment

An experiment was conducted by running MSBLAST and NACHI as in the Figure 4 environment in order to capture the data.

First, NACHI or MSBLAST ran in a client belonging to VLAN1, and we captured packets which the client sent by capturing packets software Sniffer [4]. The following Table 3 shows the scanning data. We obtained a scan rate data of NACHI was 41.084, and MSBLAST was 10.991 per second on average.

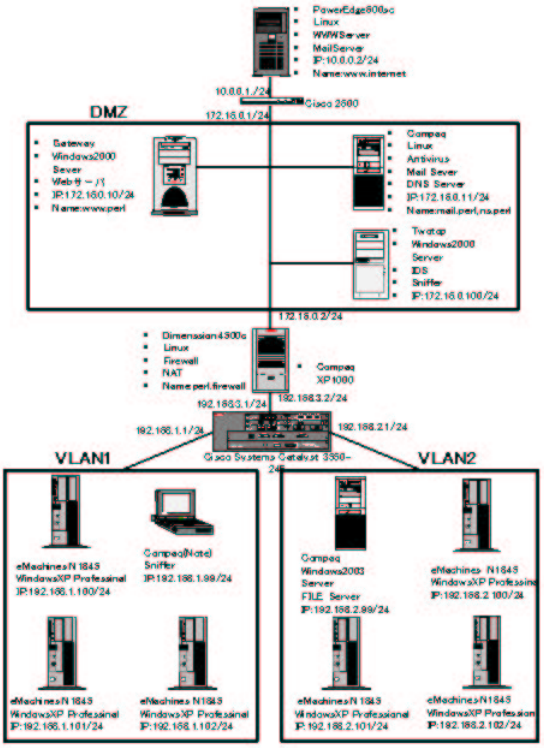

Figure 4: Experiment Network

Table 3: Result of Experiment

	Range of Scanning IP	Required Time[sec]
NACHI	192.168.0.0 - 192.168.255.255	4495
	192.165.0.0 - 192.165.255.255	3050
	61.157.0.0 - 61.157.255.255	1018
	(256 × 256 random IPs)	1008
BLAST	203.78.0.0 - 203.82.254.254	29582

5 Model of NACHI

We considered that the infecting probability, which is probability of computers having the exploit, of NACHI was 71/65536 from an occurrence when NACHI ran on the network of the University of Aizu [9]. Therefore from the product of scan rate and infecting probability, the propagation rate of NACHI, a, was obtained. We assume that NACHI was run when the number of infected computers by MSBLAST was 1,000, the infecting probability of MSBLAST was the same as NACHI, the ratio of the number of Windows2000 to WindowsXP computers was 1:1, and b is 10% of a of the propagation rate of NACHI. The result of the model of NACHI and MSBLAST is shown in Figure 5, where the constants take each value of Table 4.

Table 4: Each Constants Value

a	$41.084 \times \frac{71}{65536}$
r	$10.991 \times (\frac{4}{5} \times \frac{1}{2} \times \frac{71}{65536} + \frac{1}{5} \times \frac{1}{2} \times \frac{71}{65536})/5$
b	$a/10$
x_0	1000
y_0	1

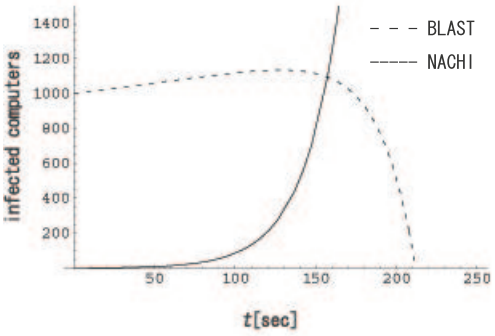

Figure 5: Model of NACHI and MSBLAST

6 Optimum Design of NACHI

6.1 Infection Rate

We obtained $x'(t)$, which refers to the global maximum of $x(t)$, by differentiating the formula (1) with respect to t. Then, by solving the function $x'(t) = 0$, t' was obtained, which is the time it takes for take the global maximum of $x(t)$. One of the condition of $x(t)$, $r < a, b > 0$ refers to that scan rate of NACHI is more than one of MSBLAST. And, another condition, $x_0 - \frac{by_0}{r-a} < 0$ refers to that scan rate of NACHI is less than one of MSBLAST, but the default number of NACHI is more than several times as the default number of MSBLAST.

$$t' = \frac{log\left(\frac{r(by_0 - x_0(a-r))}{aby_0}\right)}{a-r}, \tag{7}$$

where

$$r < a, \quad b > 0$$

or

$$r > a, \quad x_0 - \frac{by_0}{r-a} < 0.$$

Substituting t' for $x(t)$, we obtain following formula (8).

$$x(t') = x_0 f^r + \frac{by_0(f^r - f^a)}{a-r}, \tag{8}$$

where

$$f = \left(\frac{r(by_0 - x_0(a-r))}{aby_0}\right)^{\frac{1}{a-r}}. \tag{9}$$

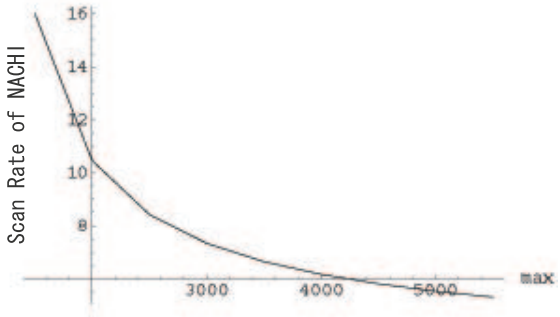

Figure 6: Optimum Scan Rate, where *max* is 1,500 - 5,500 and other constants are same as Table 4.

6.2 Algorithm

The following step is the algorithm to find the optimum scan rate of NACHI with some conditions, where *max* is the maximum number of infected computers by MSBLAST.

1. Decide the constants of formula (8) and (9), depending on the condition.
2. Decide a value of *max*.
3. Solve $x(t') = max$ for a.
4. Divide a by infecting probability.

6.3 Examples

We can obtain optimum scan rate of NACHI against several conditions by the upper algorithm. Figure 6-9 show the optimum scan rate of NACHI with some example conditions. For example, from Figure 6, we know that 8 is sufficient for scan rate of NACHI and 41 of actual scan rate is too much to terminate the MSBLAST before MSBLAST spreads more than 3,000 computers.

6.4 Terminate Time of MSBLAST

It is important that we know terminated time of MSBLAST. So, we found the time from formula (10). In solving the function $x(t) = 0$, the following t was obtained.

$$t = \frac{log\left(\frac{x_0(a-r)}{by_0}\right)}{a-r}, \tag{10}$$

where

$$r < a, \quad b > 0$$

or

$$r > a, \quad x_0 - \frac{by_0}{r-a} < 0$$

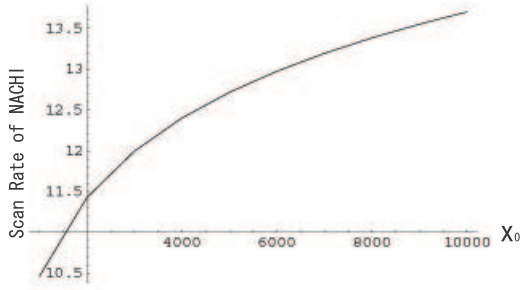

Figure 7: Optimum Scan Rate, where x_0 is 1,000 - 10,000, *max* is twice x_0, and other constants are same as Table 4.

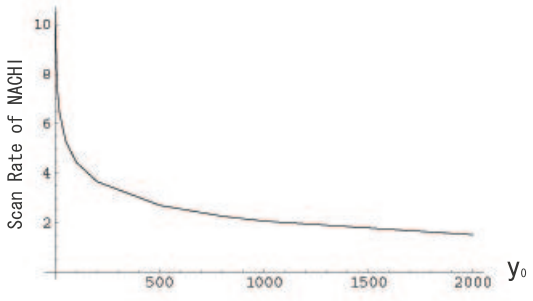

Figure 8: Optimum Scan Rate, where y_0 is 1 - 2000, *max* is 2000, and other constants are same as Table 4.

Figure 10 shows the terminate time of MSBLAST, where the constants are same as the Table 4.

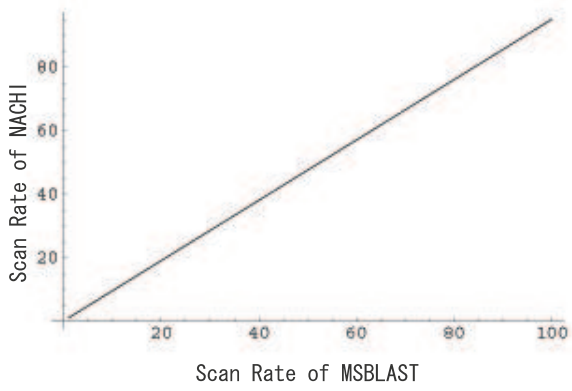

Figure 9: Optimum Scan Rate of NACHI, where one of MSBLAST is 1 - 100, *max* is 2000, and other constants are same as Table 4.

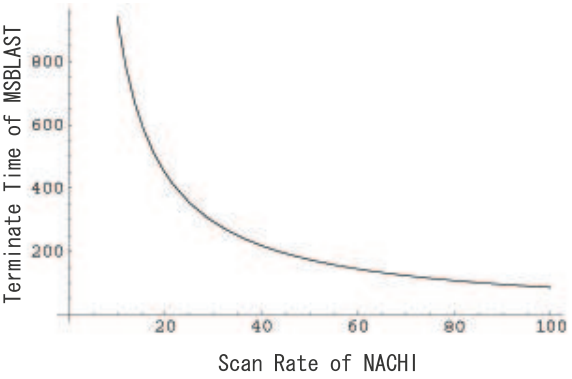

Figure 10: Terminate Time of MSBLAST, where the constants are same as the Table 4.

7 Conclusion and Future Work

We obtained a method to determine the scan rate, so that propagation of good worm can be controlled. This is a very useful method, but we must advance it to the next step. We need to contrive a new algorithm such as Predator [5], since good worms can keep spreading even if the malicious worms are eradicated.

Acknowledgement

I sincerely thank Prof. Toyoizumi for his entire support and also the Performance Evaluation Lab members of the University of Aizu for building the experiment network and conducting seminars about probability, network, and computer virus.

References

[1] *Yuuzou Kobayashi*, Modeling of Computer Viruses, University of Aizu, Graduation Thesis. March, 2003. /home/committee/aac/Thesis2002/s1070082

[2] *Trend Micro*, http://www.trendmicro.com/

[3] *Peter Ferry*, NETWORK WORLD 2004 January pp.198-201, IDG JAPAN, 2004.

[4] *TOYO Corporation*, http://www.toyo.co.jp/sniffer/

[5] *H Toyoizumi and A Kara*, Predator: Good Will Mobile Codes Combat against Computer Viruses, ASM Sigsac New Security Paradigms Workshop, 2002.

[6] *Symantec*, http://www.symantec.com/

[7] *Tatehiro Kaiwa*, Optimaization of Blaster worms by Stochastic modeling, University of Aizu, Graduation Thesis. March, 2003. /home/committee/aac/Thesis2003/s1080060

[8] *Microsoft TechNet*, http://www.microsoft.com/technet/

[9] *The University of Aizu Information Processing Center*, Security Report, 2003.12.5

[10] *Roger A. Grimes*, Malicious Mobile Code, O'Reilly & Associates, INC. 2001