On Effectiveness of Defense Systems against Active Worms

Zesheng Chen, Lixin Gao, and Chuanyi Ji

Abstract— **Active worms use self-propagating malicious code, and have been a persistent security threat to the Internet since 1988. Recent worm outbreaks have caused parts of Internet inaccessible temporarily, and cost millions of dollars to recover. Effective defense systems, however, have been lacking for fighting against worms. It is thus important to provide a basic understanding of how efficient the current systems defend against worms, what key factors determine the effectiveness of a defense system, and the guidelines that can be drawn for developing future defense systems. In this paper, we investigate these questions through modeling and analysis. Using a discrete-time model, we show that three key characteristics of worm propagation are exploited by the current defense systems: number of vulnerable machines, scanning rate, and time to complete infection. We first define the performance and resources of defense systems. We then derive and analyze the relationship between the performance and the resources for four widely-used or promising defense systems focusing on the worms that employ random scanning. We find that the existing defense systems can be categorized into two groups. One exploits the number of vulnerable machines, and the other focuses on the scanning rate. Our analysis shows that a significant amount of resources is required for the existing systems to fight effectively against active worms. When a single system can not acquire enough resources to contain worms, a combined use of all defense systems provides a hope to fight against worm propagation efficiently. To our knowledge, this is the first attempt on understanding the essence of different host-based defense systems and their combination quantitatively.**

I. Introduction

Internet worms have been a persistent security threat since the Morris worm arose in 1988. After the Code Red and Nimda worms were released into the Internet in 2001, Sapphire worm was unleashed with a 376-byte UDP packet and infected at least $160,000$ computers worldwide on January 25, 2003 [1], [2], [3]. These active worms caused parts of the Internet inaccessible temporarily, and cost both the public and private sectors millions of dollars to recover. Using self-propagating malicious code, active worms spread rapidly by infecting computer systems and by using infected nodes to disseminate the worms in an automated fashion. The frequency and virulence of active worm outbreaks have increased dramatically in the last few years, presenting a significant threat to today's Internet. It is therefore of great importance to investigate effective defense systems against worms.

Defense systems are used to slow down or even stop the propagation of active worms. Currently, a basic technique to defend against worms is patching, which repairs the security holes of a computer. Besides patching, there are three other widely-used or promising defense systems. LaBrea, presented by Liston, slows the growth of TCP-based worms such as the Code Red worm [10]. Worm Propagation Detection and Defense (WPDD) system, developed by CERIAS intrusion detection research group, concentrates on the worm propagation and uses port-scanning detection to search for infected machines [15]. Virus Throttle tool, put forward by Williamson, exploits the characteristics of local correlation in normal traffic to suppress high-rate malicious traffic generated by worms [11]. These systems are representatives of currently available methods to fight against worms.

What are the common characteristics of the different defense systems? How efficient are the existing defense systems? What are the key factors that determine the effectiveness of a defense system? How can a defense system make use of the key characteristics to improve its performance? In this paper, we attempt to investigate these issues. Our goal is two-fold: (a) to provide a basic understanding of underlying principles governing the existing defense systems, and (b) to develop an analytical approach for investigating the performance of a defense systems systematically. As effective defense systems are still lacking, guidelines can hopefully be drawn for developing future systems in fighting against worms.

Prior research on defense systems focuses mostly on developing approaches to defend against active worms. The performance and resources required have not been investigated systematically. We define the *performance* as the ability of a defense system to either contain or stop the spread of a worm. Such a performance can be characterized by the number of infected machines. We define the *resource* or the *cost* needed as the number of computers that are either patched or installed with a defense system. We focus on investigating the relationship between the performance and the amount of resources needed for defense systems. For example, if a

defense tool is installed in 25% of computers, how many machines would still be infected? The performance and cost together can be used to evaluate the effectiveness of a system.

To quantify the performance of defense systems, we first characterize the spread of active worms. Analytical Active Worm Propagation (AAWP) model, developed by Chen et. al. [5], can capture the propagation of active worms that employ random scanning. Using this analytical model, we identify three key parameters of worms propagation exploited by current systems: number of vulnerable machines, scanning rate, and time to complete infection. The severity of worm propagation can be mitigated greatly, if a defense system can reduce the number of vulnerable machines significantly, decrease the scanning rate dramatically, and prolong the time that worms need to infect a machine. Taking the Code-Red-v2-like worm as an example, we provide a quantitative analysis on how systems defend against worms through exploiting these parameters. We show that the current defense systems share many commonalities, and can thus be divided into two groups. One exploits the number of vulnerable machines, and the other focuses on the scanning rate. Our analysis shows that a significant amount of resources is required for the available systems to fight against active worms effectively. While a single system may not acquire enough resources to contain worms, the combination of all defense systems provides a hope to fight against active worms.

The motivation of our work is to develop simple mathematical models that can be used to illustrate and illuminate the essence of different defense systems. Our approach provides a modeling framework which allows one to assess a mass of defense systems.

The remainder of this paper is structured as follows. Section II gives a brief review of worm propagation and the related work. Section III describes the AAWP model and the key parameters of worm propagation that can be exploited by defense systems. Section IV evaluates and compares four widely-used or promising defense systems. Section V concludes a paper with a brief summary and an outline of future work.

II. Background

Self-propagation is a key characteristic of an active worm. For example, when a worm is released into the Internet, it starts out on a single host and scans randomly for other vulnerable machines. When the scan finds a host that can be compromised, the worm sends out a probe to infect the target. After a new host is compromised, the worm transfers a copy of itself to this host. This new host then begins to run the worm and infects other targets. Another example is Sapphire worm. Such a worm uses a single UDP packet to probe, compromise, and spread the worm to targets [1], [2], [3]. One other example is “hitlist” scanning worm investigated by Weaver [8]. Before a worm is released, the worm author gathers a “hitlist” of potentially vulnerable machines with good connections. The worm, when unleashed into the Internet, begins scanning down the list. After this list has been exhausted, the worm turns to infect other vulnerable machines.

Active worms can employ different scanning mechanisms to spread, such as random, localized, permutation, and topological scanning [9]. A worm that employs random scanning selects target IP addresses at random. Therefore, every vulnerable machine is equally likely to be infected. In this paper, we focus on random-scanning worms, for the following reasons. First, random scanning is used by the most widespread Internet worms, such as Code Red v2 and Sapphire. Second, many “sophisticated” scanning mechanisms still require certain forms of random scanning. For example, a worm that employs localized scanning scans the machines in a subnet uniformly. Last, the study of the defense system against random-scanning worms provides a benchmark for the study of the defense systems against other “sophisticated” worms.

When a worm spreads, some machines may stop functioning properly, forcing the users to reboot these machines or kill some of the processes exploited by the worm. This results in a death rate of worm propagation. When an infected computer is detected, a patch may be used to rescue the computer. This process results in a patching rate of worm propagation.

There have been only a handful studies on active worms since worm outbreaks have been rare until recently. One closely related work is “Internet Quarantine” by Moore et. al. [6]. This work investigates the requirements for containing the self-propagation code. The focus there is on two network-based defense approaches: content filtering and address blacklisting. Such approaches may require participation of network service providers. The focus of this work is on host-based or end-network-based defense approaches. Such an approach does not require participation of network service providers.

There are several quantitative studies of modeling the spread of active worms that employ random scanning. The first model is the Epidemiological model, which is grafted from traditional epidemiology by Kephart and White [7]. Another model is the two-factor worm model extended from the Epidemiological model by Zou, which takes into consideration of the human countermeasure and the worm’s impact on Internet traffic and infrastructure [4]. One other model is the Analytical Active Worm Propagation (AAWP) model, which uses a discrete time model [5]. Comparing with the Epidemiological model, the

AAWP model considers more parameters, such as the patching rate and the time that it takes the worm to infect a machine. In this paper we choose the AAWP as our basic model.

III. Model

To quantify the performance of defense systems, it is important to characterize the worm spread. In this section, we first review the AAWP model for worm propagation. We then present the parameters that are critical for defending against active worms. We finally provide a general characterization of the performance of defense systems.

A. Modeling the Spread of Active Worms

Active worms often spread through random scanning. Analytical Active Worm Propagation (AAWP) model captures this feature using a discrete-time model and a deterministic approximation [5]. This model shows that the speed of worms spreading is determined by such parameters as the size of a hitlist [8], the total number of vulnerable machines, the size of entry addresses that worms scan, the scanning rate, the death rate, the patching rate, and the time to complete infection.

The model assumes that worms can simultaneously scan many machines and do not re-infect an infected machine. The model also assumes that the machines on the hitlist are already infected at the start of the worm propagation.

The model is derived as follows. Suppose that a worm scans N entry addresses and needs one time tick to infect a machine. For random scanning, the probability that a machine is hit by one scan is $\frac{1}{N}$. Specially, when the worm scans 2^{32} entry addresses, this probability becomes $\frac{1}{2^{32}}$. Assume that currently there are n_i infected machines and m_i vulnerable machines, where i is the index of time tick. Then the infected machines send out $n_i s$ scans to find the vulnerable machines, where s is the scanning rate. On average, there are $(m_i - n_i)[1 - (1 - \frac{1}{N})^{n_i s}]$ newly-infected machines on the next time tick. Meanwhile, given death rate d and patching rate p, at the next time tick, pm_i vulnerable machines are patched, and $dn_i + pn_i$ infected machines change to either vulnerable machines without being patched (dn_i) or invulnerable machines (pn_i). Therefore, the number of infected machines is $n_{i+1} = n_i + (m_i - n_i)[1 - (1 - \frac{1}{N})^{sn_i}] - (d+p)n_i$ on the next time tick. In addition, $m_{i+1} = (1-p)m_i$, giving $m_i = (1-p)^i m_0 = (1-p)^i M$, where M is the total number of vulnerable machines. Putting the above equations together, and letting k_i and e_i be the average number of scans and the number of newly infected machines at time tick i $(i \geq 1)$ respectively, the AAWP model can be derived as:

$$m_{i+1} = (1-p)^{i+1} M \quad (1)$$

$$k_{i+1} = n_i s \quad (2)$$

$$e_{i+1} = (m_i - n_i)[1 - (1 - \frac{1}{N})^{k_{i+1}}] \quad (3)$$

$$n_{i+1} = (1 - d - p)n_i + e_{i+1} \quad (4)$$

where $i \geq 0$, $n_0 = h =$ *size of hitlist*, and $m_0 = M$. The recursion stops when there are no more vulnerable machines left or when the worm can not increase the total number of infected machines. AAWP model thus characterizes the active worms spreading (see [5] for more details). Table I summarizes all the notations.

The Code Red v2 worm is a typical example of worms that employ random scanning. The AAWP model can be used to simulate a Code-Red-v2-like worm that scans 2^{32} entry addresses with the following parameters: 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection [5]. In this paper, we mainly focus on the effect of defense systems on the Code-Red-v2-like worm that employs random scanning.

B. Important Parameters

AAWP model reveals the key parameters that constrain the speed of worms spreading and an ultimate prevalence of the worms in general. These parameters include the total number of vulnerable machines, the scanning rate, and the time to complete infection.

1) Total number of vulnerable machines: To understand the impact of this parameter, Figure 1 shows the propagation of the Code-Red-v2-like worm with different sizes of vulnerable machines[1]. As the size of vulnerable machines decreases, it takes the worm a longer time to spread. This is because that the scans from the worm are less likely to hit the vulnerable machines. For example, if $500,000$ vulnerable machines decrease to half, the time that the worm takes to reach the peak of the curve in Figure 1 increases from 24.8 hours to 54.3 hours. Therefore, reducing the number of vulnerable machines can be used by defense systems against worms spreading.

One example is the address blacklisting defense system [6]. When an IP address has been identified as being infected, packets arriving from this address are dropped when received by the routers with this defense system. In this way, an infected

[1]The curves show the transient behavior of the number of infected machines with respect to time, and are obtained from the recursive relation given by AAWP model.

TABLE I

NOTATION OF THE AAWP MODEL

Notation	Explanation
M	total number of vulnerable machines
N	size of entry addresses that worms scan
h	size of hitlist (the number of infected machines at the beginning of the spread of active worms)
s	scanning rate (the average number of machines scanned by an infected machine per unit time)
d	death rate (the rate at which an infection is detected on a machine and eliminated without patching)
p	patching rate (the rate at which an infected or vulnerable machine becomes invulnerable)
n_i	number of infected machines at time tick i
m_i	number of vulnerable machines at time tick i
k_i	number of scans at time tick i
e_i	number of newly infected machines at time tick i

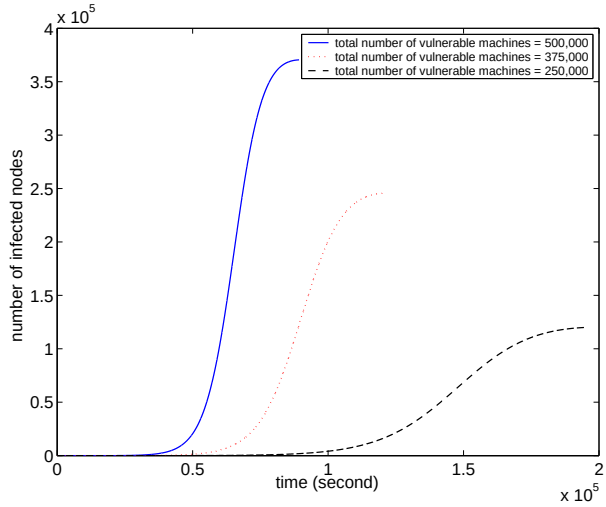

Fig. 1. Effect of size of vulnerable machines. All cases are for starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection.

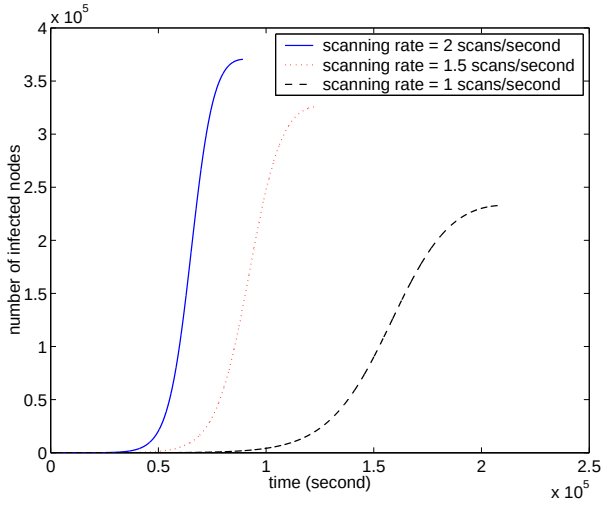

Fig. 2. Effect of scanning rate. All cases are for 500,000 vulnerable machines, starting on a single machine, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection.

host can be "isolated" from the Internet, and the number of vulnerable machines is thus reduced. The other examples are patching and WPDD system, which we describe in detail in Section IV.

2) Scanning rate: Figure 2 demonstrates the effect of the scanning rate on worm propagation. The Code-Red-v2-like worm spreads slowly when the scanning rate decreases. For example, the simulated Code-Red-v2-like worm propagates with a scanning rate of 2 scans/second and infects about $370,000$ machines in 25 hours, while the worm with a scanning rate of 1 scan/second infects about $230,000$ machines in 58 hours.

One example is the content filtering defense system [6]. When a worm's signature has been identified, packets containing this signature are dropped when received by the routers with this defense system. In this way, the system can block the scans or the worm copy transmissions from the infected machines, and therefore the scanning rate is reduced. Other examples are LaBrea and Virus Throttle, which we describe in detail in Section IV.

3) Time to complete infection: Figure 3 describes the effect of time to complete infection on worm propagation. In the future, worms can become more virulent by utilizing any of the following such methods: scanning the vulnerable machines only, increasing the scanning rate, and exploiting the vulnerability that many computers may have. One famous example is "Flash Worm" [9], which can flood the Internet within seconds. It is difficult to defend against this kind of rapidly-spreading worms. However, prolonging the time to complete infection can slow down the spread of these worms. As shown in Figure 3, the worm tries to infect $1,000,000$ vulnerable machines with a scanning rate of 100 scans/second. The worm with a time period of 10 seconds to infect a machine can compromise about $523,660$ machines in 18 minutes,

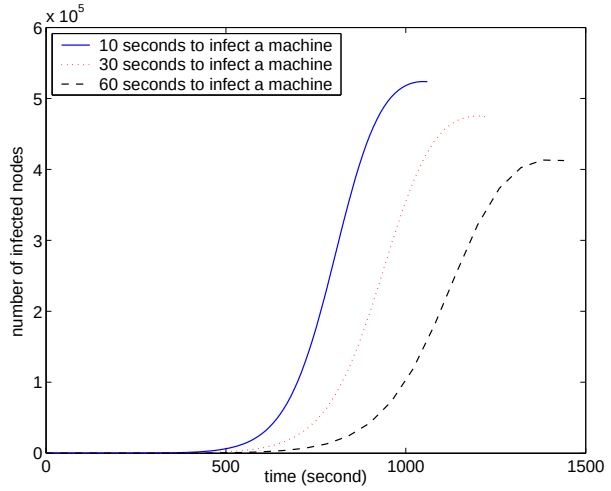

Fig. 3. Effect of time to complete infection. All cases are for 1,000,000 vulnerable machines, starting on a single machine, a scanning rate of 100 scans/second, a death rate of 0.001 /second, and a patching rate of 0.0005 /second.

while the worm with a time period of 60 seconds to infect a machine can compromise about $412,400$ machines in 25 minutes. Therefore, the worm spreading can be slowed down significantly when the time required to infect a machine is prolonged.

There are two research works on modeling the timing parameters [17], [18]. Both of these works show that prolonging the time to complete infection can slow down the spread of the worms. However, there is no existing defense system yet that makes use of this parameter against worm propagation.

C. Performance of Defense Systems

The performance and the needed resource of a defense system can then be defined based on these parameters.

1) Definition:

- The **performance** of a defense system is defined as the maximum number of infected machines that active worms can achieve under the containment of the system. The fewer the number of infected machines, the better the performance.
- The **resource** or **cost** of a defense system is defined as the number of machines which are either patched or installed with the defense system.

A defense system is considered to be effective when it causes a worm to infect less than half of the total vulnerable machines. The particular choice of "one half" seems to be arbitrary, but it quantifies a reasonable standard for evaluating defense systems[2].

[2]If another fraction is chosen as a criterion, the effectiveness in terms of the number of infected machines can be computed accordingly using AAWP.

2) Performance: To quantify the performance, let us imagine that a defense system can perform at least one of the following tasks: reducing the number of vulnerable machines significantly, decreasing the scanning rate dramatically, or prolonging the time that worms take to infect a machine. Then the number of scans, $n_i s$, is much less than N, in a time duration (e.g. a day) after the burst of the worms. The number of newly-infected machines, e_{i+1}, can thus be approximated as:

$$e_{i+1} \approx (m_i - n_i)(1 - e^{-\frac{n_i s}{N}}) \approx \frac{(m_i - n_i)n_i s}{N}. \quad (5)$$

This shows that although the original worm spreading is to grow exponentially, an ideal defense system can contain the growth polynomially. Therefore, defense systems that have a good performance can either stop or slow down worms spreading effectively.

IV. Effectiveness of Defense Systems

The performance measure can now be applied to evaluating the effectiveness of defense systems. In particular, we evaluate and compare the performance of four available defense systems: patching, Worm Propagation Detection and Defense (WPDD), LaBrea, and Virus Throttle. AAWP model is used as a unified approach to study the functionality and effectiveness of each defense system in reducing either the number of vulnerable machines or the scanning rate.

A. Effectiveness of Patching

Patching vulnerable machines is the most direct method and is widely-used for defending against worms.

1) Patching: A patch repairs a security hole of a host, which equivalently reduces the total number of vulnerable machines. Statistics show that few worms exploit vulnerabilities that are new and unknown. Popular worms, such as Code Red and Sapphire, attack well-known vulnerabilities. However, the prevalence of those worms reflects a fact that many people are reluctant to update patches in time. Then a question rises: How many vulnerable machines should be patched before worms' release to defend against them effectively?

2) Performance of Patching: To answer this question, we begin evaluating the performance of patching. We assume that V machines have been patched before a worm is released. Then there are $M-V$ vulnerable machines left. M in Equation (1) can be replaced by $M - V$, i.e.,

$$m_{i+1} = (1 - p)^{i+1}(M - V). \quad (6)$$

Since other parameters (see Section III-A) are not altered, Equations (2)∼(4) remain the same.

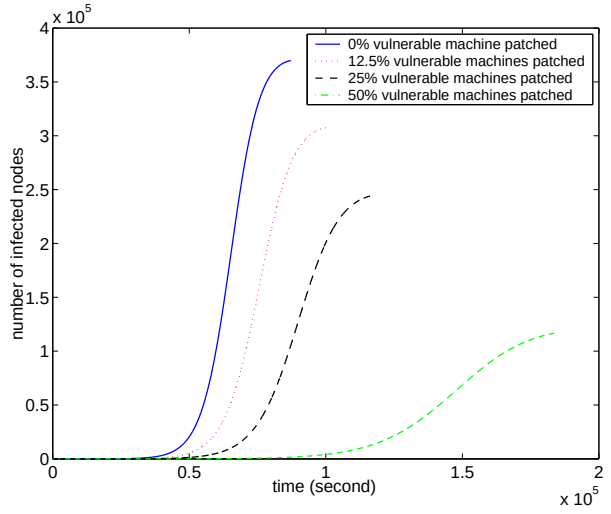

Fig. 4. Performance of patching. All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection.

The above recursive relation is a modified AAWP model due to patching, and is used to demonstrate the performance. Figure 4 shows the performance of patching for a Code-Red-v2-like worm spreading. The more machines are patched, the more slowly the worm spreads. But to defend against the worm effectively, the figure shows that at least 25% vulnerable machines should be patched. Therefore, the task of patching is challenging in a war between defenders and attackers.

B. Effectiveness of Worm Propagation Detection and Defense System

Worm Propagation Detection and Defense (WPDD) system is another approach which reduces the total number of vulnerable machines. Compared to patching, the WPDD system has an advantage of dynamically detecting infection. Once detected, infected machines can be disarmed.

1) Worm Propagation Detection and Defense (WPDD): This defense system is developed by an intrusion-detection research group at CERIAS (Center for Education and Research in Information Assurance and Security) [14], [15]. The main purpose of the WPDD system is to detect and defend against a class of worms which rapidly scan randomly-selected IP addresses on a fixed port (i.e., horizontal scan). This system acts as an "end-network firewall" that monitors outbound traffic of a network. The system monitors all scans leaving the scanning host, and looks for a certain number of horizontal scans that occur within a certain time period to detect abnormal traffic. When an infected machine monitored by the WPDD system begins to scan the Internet, all scans generated by this machine can be examined. Once a certain number of horizontal scans is counted from this machine within a certain period of time, the WPDD system generates an alert, and contains the abnormal traffic from the infected machine. This approach can quickly identify the infected machines, and stop the offending program. Here the infected machines that have been detected are assumed to be isolated from further infection or patched at once. Hence these machines become either traffic-broken or invulnerable as seen by the worm. However, WPDD systems might not be able to detect the all worms. Moreover, some worms may bypass the WPDD system that focuses on the horizontal scans.

2) Performance of WPDD: Such an ability of detecting infection and then isolating/patching infected machines can be characterized as follows. Let V be the number of vulnerable machines that are monitored by WPDD system. Let q be then probability that WPDD system detects a worm. At time tick i, there are e_i newly infected machines. Among these infected machines, on the average $q\frac{V}{M}\cdot e_i$ machines can be detected and then isolated/patched, while the rest $e_i - q\frac{V}{M}\cdot e_i$ machines begin infecting other machines. In this context, the worm spreading described by Equations (1) and (4) become

$$m_i = (1-p)(m_{i-1} - q\frac{V}{M}\cdot e_i) \quad (7)$$

$$n_{i+1} = (1-d-p)n_i + (1-q\frac{V}{M})e_{i+1}, \quad (8)$$

where $i \geq 0$, $e_0 = h$ and $m_{-1} = \frac{M}{1-p} + q\frac{V}{M}\cdot h$. Equations (2) and (3) remain the same. Note that if $q = 0$ or $V = 0$, i.e., if neither worms are detected nor machines are monitored by WPDD system, the above equations are the same as Equation (1) and (4).

Figure 5 shows the performance of the WPDD system for a Code-Red-v2-like worm spreading when the worms can be detected with probability 1. The more machines are monitored by the WPDD system, the more slowly the worm spreads. But to defend against the worm effectively, the figure shows that at least 20% vulnerable machines should be monitored. Figure 6 demonstrates the effect of the detection probability when 25% vulnerable machines are monitored by the WPDD system. This figure shows that the detection probability should be at least 0.8 to defend against the worm effectively. Conversely, if the WPDD system can detect a worm with the probability 0.8, it requires that the WPDD system monitors at least 25% vulnerable machines. Since the vulnerability exploited by the worms is unknown a priori, the WPDD system can be installed to monitor randomly-chosen machines. If $\frac{V}{M} = 25\%$ needs to be achieved, $\frac{1}{4}$ of the total number of machines in the Internet need to be monitored. That is, if there are 2^{30} computers in

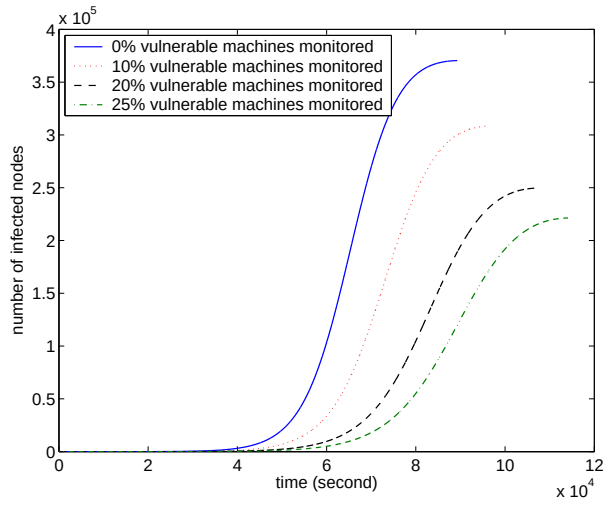

Fig. 5. Performance of the WPDD defense system ($q = 1$). All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection.

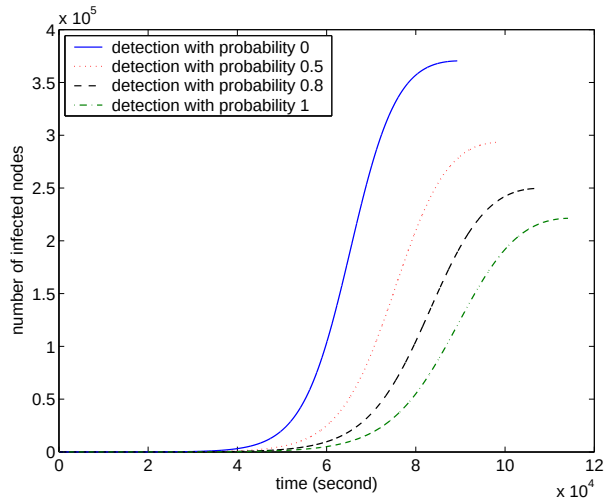

Fig. 6. Performance of the WPDD defense system ($\frac{V}{M} = 25\%$). All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection.

the Internet, the WPDD system needs to monitor more than 2^{28} hosts, in order to defend against the worm effectively. The amount of resources needed is overwhelming for installing the WPDD system.

C. Effectiveness of LaBrea

LaBrea[3] is developed to reduce the scanning rate of worms.

1) LaBrea: LaBrea is developed by Liston to slow down or even stop the spread of Code Red worm which broke out in 2001 [10], [5]. LaBrea takes advantage of the fact that many worms scan unused IP addresses and use TCP connection to propagate. Specifically, LaBrea can take over unused IP addresses on a network and create "virtual machines" that respond to TCP connection-requests. When a scan from an infected host hits one of these "virtual machines", LaBrea replies and establishes a connection with the infected machine. This connection can last for a long time. However, LaBrea can only defend against a worm that scans unused IP addresses and uses TCP connections. Such a tool is thus useless for a recent worm, Sapphire, that employs UDP packets.

2) Performance of LaBrea: To evaluate the performance of LaBrea, we need to address the following question. How many unused IP addresses should be monitored by the LaBrea tool to defend against active worms effectively?

Assume that LaBrea is installed in the Internet and is monitoring u unused IP addresses. These addresses are among entry addresses scanned by worms. Suppose that currently there are k_i scans from infected machines beginning to search the Internet. Because the LaBrea tool can trap the scanning threads, after one time tick, there are $\frac{u}{N}k_i$ scanning threads trapped, i.e., there are only $(1 - \frac{u}{N})k_i$ scanning threads left. Given death rate d and patching rate p, on the next time tick there are $(1-d-p)k_i(1-\frac{u}{N})$ "old" scans left and $s \cdot e_i$ "new" scans generated. Therefore, Equation (2) becomes

$$k_{i+1} = (1 - d - p)k_i(1 - \frac{u}{N}) + s \cdot e_i, \qquad (9)$$

where $i \geq 0$, $k_0 = 0$, and $e_0 = h$. Since LaBrea does not alter the total number of vulnerable machines, Equations (1), (3), and (4) remain the same. It should be noted that if $u = 0$, i.e., no unused IP addresses are monitored, Equation (9) is the same as Equation (2). But as soon as $u > 0$, the scanning rate can be reduced by the LaBrea tool.

Figure 7 shows the spreading of a simulated Code-Red-v2-like worm using the AAWP model. The figure shows that when LaBrea monitors fewer than 2^{16} unused IP addresses, the worm spread is changed slightly. But when more than 2^{18} unused IP addresses are monitored, the total number of infected machines stops increasing before the worm acquires a half of the vulnerable machines. Therefore, 2^{18} seems to be the number of unused IP addresses that needed to be monitored for the LaBrea tool to effectively defend against the worm propagation [4]. However, it might not be easy to get so many unused IP addresses.

[3]The effectiveness of LaBrea has been studied in [5]. We introduce it here for the completeness of discussion. Moreover, we will consider a combination of LaBrea with other defense systems in Section IV-F.

[4]A more rigorous approach for obtaining this quantity requires solving the non-linear difference equations which is beyond the scope of this work.

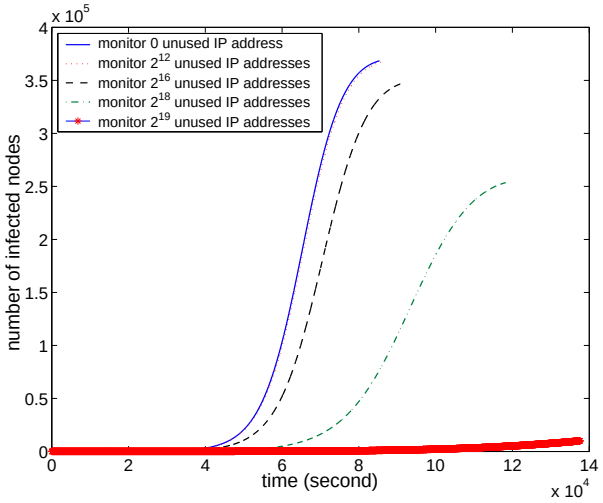

Fig. 7. Performance of the LaBrea tool defense system. All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection.

D. Effectiveness of Virus Throttle

The previous three defense systems reduce either the total number of vulnerable machines or the scanning rate, but not both. Virus Throttle reduces both the scanning rate of worms' spreading and the number of vulnerable machines.

1) Virus Throttle: Virus Throttle is a "personal firewall" like software designed by Williamson to defend against a worm which contacts as many machines as possible and spreads as fast as possible [11], [12]. When a machine sends out a connection request, the Virus Throttle tool installed on the machine first determines whether this request is for a new host based on a short list of recent connections. If so, the request is put into a delay queue. Otherwise, the request is processed immediately. A timer is set in the delay queue so that only one request is processed during a timeout period. In this way, most normal traffic is unaffected since it is locally correlated (i.e., it is likely to have repeated connections to recently accessed machines). Malicious traffic generated by worms is heavily penalized since such traffic has a much higher rate than that of normal traffic and is not locally correlated. At the same time, this tool can detect and disarm worms by monitoring the size or the increasing rate of the delay queue. When the machine is infected by a worm with a high attack rate, the delay queue grows fast and becomes long. This spreading behavior of the worm can be quickly detected. Therefore, the infected machine with Virus Throttle installed can be isolated subsequently, and then immunized. Here, we assume that infected machines that have been detected can be either isolated from further infection or patched at once. Hence, these machines become traffic-broken or invulnerable machines to the worm. However, not all machines installed the Virus Throttle tool can detect a worm for that different machines configure the tool differently. Moreover, well-designed worms can bypass the detection of Virus Throttle.

2) Performance of Virus Throttle: The ability of Virus Throttle can be characterized for detecting worms and reducing the scanning rate as follows. Assume that V vulnerable machines are installed the Virus Throttle tool, and have a probability q of detecting a worm. At time tick i, there are e_i newly infected machines. Among these infected machines, average $q\frac{V}{M} \cdot e_i$ machines can be detected and then isolated or patched, while $e_i - q\frac{V}{M} \cdot e_i$ machines are left undetected. Among the n_i infected machines, average $\frac{(1-q)V}{M-q\cdot V} n_i$ infected machines are controlled by the tool and $\left[1 - \frac{(1-q)V}{M-q\cdot V}\right] n_i$ infected machines are not installed with the tool. For an infected machine without the tool, the scanning rate is s. For an infected machine with the tool, all requests generated by the worm are assumed to be put into the delay queue[5]. Let T, O, and A denote the time for the worm to complete infection, the duration between timeouts in the delay queue, and the number of scanning threads generated by the worm, respectively. The Virus Throttle tool can restrict the scanning rate of undetected machines to $\frac{T}{O} \cdot \frac{s}{A}$. For active worms that spreads as fast as possible, $A > 50$ and $T < 10$. Meanwhile, to delay the malicious traffic, the Virus Throttle tool requires $O > 0.2$. Therefore, $\frac{T}{O} \cdot \frac{s}{A} < s$.

Taking into consideration that the Virus Throttle tool reduces the total number of vulnerable machines and decreases the number of scans to $\left[1 - \frac{(1-q)V}{M-q\cdot V}\right] \cdot s + \frac{(1-q)V}{M-q\cdot V} \cdot \frac{T}{O}\frac{s}{A}$, Equations (1), (2) and (4) become

$$m_i = (1-p)(m_{i-1} - q\frac{V}{M} \cdot e_i) \quad (10)$$

$$k_{i+1} = \left[1 - \frac{(1-q)V}{M-q\cdot V}\right] n_i \cdot s + \frac{(1-q)V}{M-q\cdot V} n_i \cdot \frac{T}{O}\frac{s}{A} \quad (11)$$

$$n_{i+1} = (1-d-p)n_i + (1-q\frac{V}{M})e_{i+1}, \quad (12)$$

where $i \geq 0$, $e_0 = h$ and $m_{-1} = \frac{M}{1-p} + q\frac{V}{M} \cdot h$. Equation (3) remains the same. It should be noted that if $V = 0$, i.e., no machines install Virus Throttle, the above three equations reduce to Equations (1), (2) and (4) of the original AAWP.

Three cases can be considered below.

1) *When no machines can detect the worm, i.e.,* $q = 0$. Equations (10) and (12) are the same as Equations (1)

[5]Since the list of recent connections is short ($5 \sim 20$) and the worm selects targets randomly.

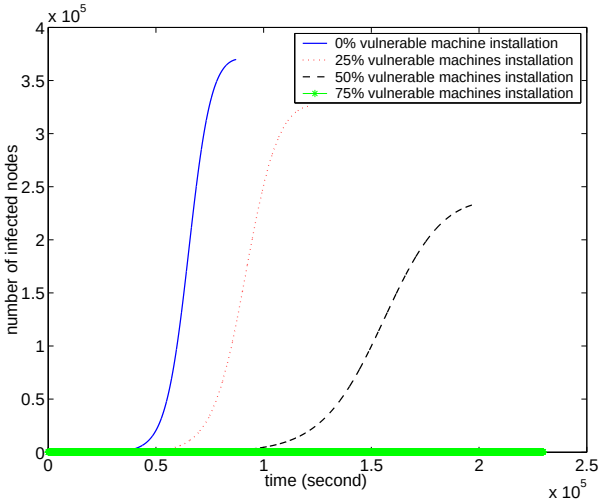

Fig. 8. Performance of the Virus Throttle tool defense system ($q = 0$). All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, a time period of 1 second to complete infection, and a time period of 1 second between timeouts for the delay queue.

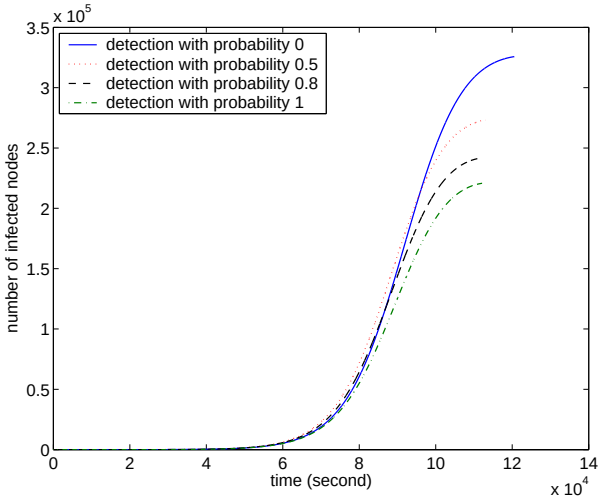

Fig. 9. Performance of the Virus Throttle tool defense system ($\frac{V}{M} = 25\%$). All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, a time period of 1 second to complete infection, and a time period of 1 second between timeouts for the delay queue.

and (4), and Equation (11) reduces to $k_{i+1} = (1 - \frac{V}{M})n_i s + \frac{VT}{MOA} n_i s$. It shows that the Virus Throttle tool decreases the scanning rate from s to $[(1-\frac{V}{M})+\frac{VT}{MOA}]s$. Figure 8 shows the spread of a Code-Red-v2-like worm under the containment of Virus Throttle. The figure demonstrates the growth of the number of infected nodes with the time for different percentages of the vulnerable machines installed Virus Throttle. The more vulnerable machines with the tool installed, the more slowly the worm spreads and the fewer machines are actually infected. Therefore, the Virus Throttle tool can retain the spread of active worms. However, at least 50% vulnerable machines need to install the tool to defend against the worm effectively. For the Code-Red-v2-like worm, $A = 99$ [13], $T = 1$, and $s = 2$. If $O = 0.5 \sim 1$, then $k_{i+1} \approx (1 - \frac{V}{M})n_i s$, for $i \geq 0$. Indeed, the tool decreases the scanning rate from s to $(1 - \frac{V}{M})s$. Therefore, $\frac{V}{M}$ should be sufficiently large (e.g. at least 0.5) in order to reduce the scanning rate effectively.

2) *When all machines with the tool can detect the worm with the probability 1, i.e.,* $q = 1$. Then, Equations (10), (11) and (12) are the same as the equations for the WPDD system. Figrue 5 demonstrates the performance in this case. Similar to the WPDD system, at least 20% machines need to be installed the Virus Throttle tool to defend against the worm effectively.
3) *Effect of detection probability, i.e.,* $0 \leq q \leq 1$. Figure 9 demonstrates the effect of detection probability when 25% vulnerable machines are installed with the Virus Throttle tool. Similar to the WPDD system, the detection probability should be at least 0.8 to defend against the worm effectively. However, comparing to the WPDD system (Figure 6), the Virus Throttle has the ability to delay the worm propagation. That is, the worm needs more time to achieve the peak under the containment of Virus Throttle. It is because Virus Throttle reduces both the number of vulnerable machines and the scanning rate.

Since the vulnerability exploited by the worms is unknown a priori, the Virus Throttle tool can only be installed on randomly-chosen machines. To achieve $\frac{V}{M} = 25\% \sim 50\%$, a quarter to a half of the total number of machines in the Internet need to install the tool. If there are 2^{30} computers in the Internet, the Virus Throttle tool needs to be installed on more than $2^{28} \sim 2^{29}$ computers to defend against the worm effectively. The resource consumption may be overwhelming.

E. Comparison of Different Defense Systems

Based on the above analysis, we summarize the defense systems in Table II for comparison. The table shows that the defense systems can be divided into two groups:

1) *Systems that exploit the number of vulnerable machines:* patching, WPDD, and Virus Throttle.
2) *Systems that exploit the scanning rate:* LaBrea and Virus Throttle.

We find that in order to fight against Code-Red-v2-like worm effectively, Group 1 requires at least 25% vulnerable machines

TABLE II

COMPARISON OF DIFFERENT DEFENSE SYSTEMS

Defense systems	Parameters mitigated	Resource requirements
Patching	number of vulnerable machines	at least 25% vulnerable machines
WPDD	number of vulnerable machines	at least 25% vulnerable machines ($q = 0.8$)
LaBrea	scanning rate	at least 2^{18} unused IP addresses
Virus Throttle	number of vulnerable machines and scanning rate	at least 25% $\sim$ 50% vulnerable machines

to be patched, monitored, or installed with the tool. Group 2 needs either more than 2^{18} unused IP addresses monitored or $25\% \sim 50\%$ vulnerable machines with the tool installed. These data show a challenge for defending today's Internet.

F. Effectiveness of Combining Defense Systems

While the defense systems have been analyzed separately, one interesting question is whether different systems can be combined to defend against worms more effectively. We conduct an initial investigation of this question through two cases.

1) WPDD and LaBrea: When the WPDD system and LaBrea are used to defend against active worms at the same time, this combined defense system has the characteristics of both WPDD and LaBrea, reducing both the number of vulnerable machines and the scanning rate. Therefore, Equations (1), (2) and (4) become

$$m_i = (1-p)(m_{i-1} - q\frac{V}{M} \cdot e_i) \quad (13)$$

$$k_{i+1} = (1-d-p)k_i(1-\frac{u}{N}) + s \cdot (1-q\frac{V}{M})e_i \quad (14)$$

$$n_{i+1} = (1-d-p)n_i + (1-q\frac{V}{M})e_{i+1}, \quad (15)$$

where $i \geq 1$, $m_0 = M$, $k_1 = s \cdot h$, $e_1 = (M-h)[1-(1-\frac{1}{N})^{sh}]$, and $n_1 = (1-d-p)h + (1-q\frac{V}{M})e_1$. V is the number of computer patched and u is the number of unused IP addresses monitored by LaBrea. Figure 10 shows the performance of this combined system. One curve corresponds to a combination of monitoring 12.5% vulnerable machines using WPDD system and monitoring 2^{17} unused IP addresses using LaBrea. The other curve corresponds to monitoring 25% vulnerable machines using WPDD without LaBrea. The third curve is for monitoring 2^{18} addresses using LaBrea only. These three curves have the similar performance, showing that the combination does not improve the performance. However, while one specific defense system can not acquire enough resources to fight against worms, the combination reduces the resource requirement for each individual system. This provides a hope to use different types of defense systems to win the war between defenders and attackers.

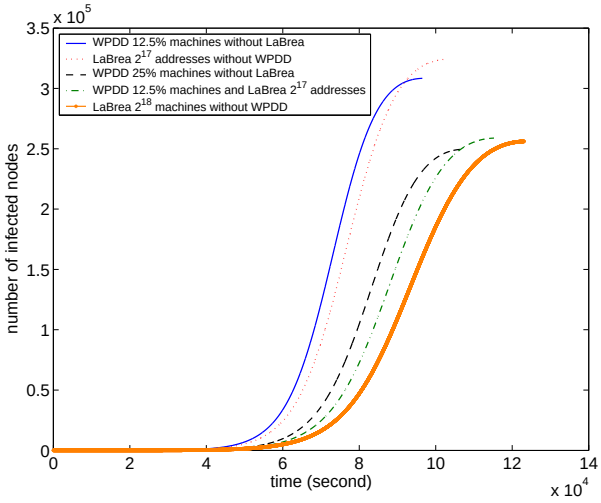

Fig. 10. Performance of combining WPDD ($q = 0.8$) and LaBrea. All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, and a time period of 1 second to complete infection.

2) LaBrea and Virus Throttle: When LaBrea and Virus Throttle are used to defend against active worms at the same time, both the outgoing scans (because of Virus Throttle) and the incoming scans (because of LaBrea) are reduced. Here we ignore the ability of the Virus Throttle to detect a worm, i.e. $q = 0$, in order to compare the ability of different systems to restrict the scanning rate. For the combined defense system, Equation (2) becomes

$$k_{i+1} = (1-d-p)k_i(1-\frac{u}{N}) + se_i[(1-\frac{V}{M}) + \frac{V}{M} \cdot \frac{T}{OA}], \quad (16)$$

where $i \geq 0$, $k_0 = 0$ and $e_0 = n_0 = h$. In the above equation, u is the number of unused IP addresses monitored by LaBrea and V is the number of vulnerable machines monitored by Virus Throttle. T and A denote the time for the worm to complete infection and the number of scanning threads generated by the worm. For the Code-Red-v2-like worm, $A = 99$ [13] and $T = 1$. O denotes the duration between timeouts in the delay queue of Virus Throttle. Figure 11 shows the performance of this combined system, which is a combination of monitoring 2^{17} unused IP address using LaBrea and monitoring 25% vulnerable machines using Virus

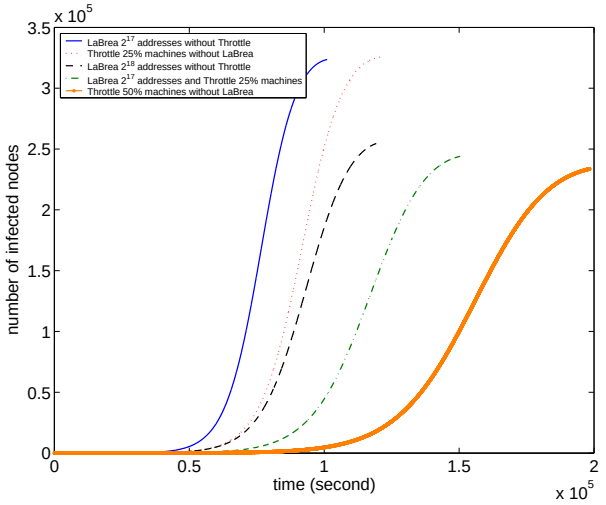

Fig. 11. Performance of combining LaBrea and Virus Throttle ($q = 0$). All cases are for 500,000 vulnerable machines, starting on a single machine, a scanning rate of 2 scans/second, a death rate of 0.00002 /second, a patching rate of 0.000002 /second, a time period of 1 second to complete infection, and a time period of 1 second between timeouts for the delay queue.

Throttle. We find that the performance of this combined system is similar to that of the system monitoring 2^{18} unused IP addresses using LaBrea without Virus Throttle and that of the system monitoring 50% vulnerable machines using Virus Throttle without LaBrea. However, from the view of the time that a worm needs to reach the peak, the curves indicate that a system monitoring 50% vulnerable machines using Virus Throttle without LaBrea has the best performance to delay the spread of the worm, while the combined system has better performance than the system that monitors 2^{18} unused IP addresses using LaBrea without Virus Throttle. Therefore, the combined system has the feature of both LaBrea and Virus Throttle.

V. Conclusions

In this paper, we have investigated the performance of different host-based defense systems against active worms using a discrete-time (AAWP) model. We have shown that the ability of worm propagation is constrained by three parameters: number of vulnerable machines, scanning rate, and time to complete infection. We have found that most of the existing defense systems essentially exploit some of these parameters. Focusing on the Code-Red-v2-like worm, we have performed a quantitative study on how well a system can slow down the propagation of worms. Four available systems have been investigated: patching, WPDD system, LaBrea, and Virus Throttle. These systems are divided into two groups. One group exploits the number of vulnerable machines, and requires at least 25% vulnerable machines to be patched or monitored. The other group focuses on the scanning rate, and needs more than 2^{18} unused IP addresses or $25\% \sim 50\%$ vulnerable machines with the tool installed. These results show a challenge for current computer systems to possess enough resources for fighting against worms. We have explored the idea of combining different defense systems and found that while it is hard for a single system to acquire enough resources, one could combine all systems to make the effective defense possible.

As part of our ongoing work, we will further study the optimal combination of different defense systems. In addition, we will study the effectiveness of defense systems on a worm that employs other scanning methods, such as localized scanning.

References

[1] eEye Digital Security, "Microsoft SQL Sapphire Worm Analysis," http://www.eeye.com/html/Research/Flash/AL20030125.html.

[2] CERT, "CERT Advisory CA-2003-04 MS-SQL Server Worm," http://www.cert.org/advisories/CA-2003-04.html.

[3] B. Krebs, "Internet Worm Hits Airline, Banks," SecurityFocus News, http://online.securityfocus.com/news/2167.

[4] C. C. Zou, W. Gong, and D. Towsley, "Code Red Worm Propagation Modeling and Analysis", *9th ACM Conference on Computer and Communication Security (CCS'02)*, Nov. 18-22, Washington DC, USA, 2002.

[5] Z. Chen, L. Gao, and K. Kwiat, "Modeling the Spread of Active Worms," *in INFOCOM* 2003, San Francisco, CA, April 2003.

[6] D. Moore, C. Shannon, G. M. Voelker, and S. Savage, "Internet Quarantine: Requirements for Containing Self-Propagating Code," *in INFOCOM* 2003, San Francisco, CA, April 2003.

[7] J. O. Kephart and S. R. White, "Directed-Graph Epidemiological Models of Computer Viruses," in *Proceedings of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy*, 1991, pp. 343-359.

[8] N. Weaver, "Warhol Worms: The Potential for Very Fast Internet Plagues," http://www.cs.berkeley.edu/∼nweaver/warhol.html.

[9] S. Staniford, V. Paxson, and N. Weaver, "How to 0wn the Internet in Your Spare Time," in *the Proceedings of the 11th USENIX Security Symposium (Security '02)*.

[10] T. Liston, "Welcome to My Tarpit The Tactical and Strategic Use of LaBrea," http://www.hackbusters.net/LaBrea/LaBrea.txt.

[11] M. M. Williamson, "Throttling Viruses: Restricting propagation to defeat malicious mobile code," *18th Annual Computer Security Applications Conference*, Las Vegas, Nevada, December, 2002.

[12] J. Twycross and M. M. Williamson, "Implementing and testing a virus throttle", *USENIX Security Symposium*, 2003.

[13] eEye Digital Security, ".ida "Code Red" Worm," Advisory AL20010717, July 2001, http://www.eeye.com/html/Research/Advisories/AL20010717.html.

[14] CERIAS Intrusion Detection Research Group, Purdue University, http://www.cerias.purdue.edu/.

[15] CERIAS Intrusion Detection Research Group, Purdue University, "Digging For Worms, Fishing For Answers," *18th Annual Computer Security Applications Conference*, Las Vegas, Nevada, December, 2002.

[16] T. Liston, "LaBrea," http://www.hackbusters.net/LaBrea/.

[17] C. C. Zou, W. Gong, and D. Towsley, "Worm Propagation Modeling and Analysis under Dynamic Quarantine Defense", *ACM CCS Workshop on Rapid Malcode (WORM'03)*, Oct. 27, Washington DC, USA, 2003.

[18] Y. Wang and C. Wang, “Modeling Timing Parameters for Virus Propagation on the Internet”, *ACM CCS Workshop on Rapid Malcode (WORM’03)*, Oct. 27, Washington DC, USA, 2003.