MODELING AND DEFENDING AGAINST INTERNET WORM ATTACKS

A Thesis
Presented to
The Academic Faculty

by

Zesheng Chen

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
May 2007

MODELING AND DEFENDING AGAINST INTERNET WORM ATTACKS

Approved by:

Professor Chuanyi Ji,
Committee Chair
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor George Riley
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Mostafa H. Ammar
College of Computing
Georgia Institute of Technology

Professor Wenke Lee
College of Computing
Georgia Institute of Technology

Professor John Copeland
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: 3 April 2007

To my parents,

Yisa Chen and Naice Sheng,

and to my wife,

Chao Chen.

ACKNOWLEDGEMENTS

This work would not have been possible without the invaluable advise, encouragement, and support that I received from various people in the past five years. First and foremost, I would like to express my sincere thanks to my advisor, Dr. Chuanyi Ji, for her kind guidance and continuous support throughout my doctoral study. Much of what lies in this dissertation can be credited to her patient advise and persistent encouragement.

I would like to acknowledge Dr. Mostafa H. Ammar, Dr. George Riley, and Dr. John Copeland for being on my dissertation proposal committee and defense committee. Their enlightening suggestions and invaluable comments have helped improve the quality of this thesis. Many thanks are due to Dr. Ammar for his great teaching and scientific spirit. I would also like to thank Dr. Wenke Lee for serving on my dissertation defense committee and giving valuable feedback.

I would like to thank the present and past members of the Communication Networks and Machine Learning Laboratory. Special thanks go to Dr. Guanglei Liu, Sung-eok Jeon, Rajesh Narasimha, Supaporn Erjongmanee, and Renjie Yang. I would also like to thank Cordai Farrar and Patricia Dixon in the School of Electrical and Computer Engineering for their help on all administrative matters.

Next, I would like to deeply thank my parents for their unconditional love. I am greatly thankful to my wife, Chao Chen, for her patient care and love. I am also graceful to the brothers and sisters in Westminster Christian Fellowship (WCF) and Atlanta Chinese Christian Campus Fellowship (ACCCF) for their friendship and encouragement. Finally, I thank Jesus, who gave me his amazing grace, for leading me through ups and downs during my doctoral study.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

2LLS	Two-Level Localized Scanning
AAWP	Analytical Active Worm Propagation
AS	Autonomous System
BA	Barabási-Albert
BGP	Border Gateway Protocol
BRITE	Boston university Representative Internet Topology gEnerator
CAIDA	Cooperative Association for Internet Data Analysis
CCDF	Complementary Cumulative Distribution Function
CCN	Correlated Complex Network
CDF	Cumulative Distribution Function
CIDR	Classless Inter-Domain Routing
DFLS	Decision-First Localized Scanning
DNS	Domain Name System
DoS	Denial of Service
ER	Erdös-Rényi
FLS	Feedback Localized Scanning
IANA	Internet Assigned Number Authority
IDS	Intrusion Detection System
IP	Internet Protocol
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IRC	Internet Relay Chat
IS	Importance Scanning

ISS	Internet Security Systems
LS	Localized Scanning
MSE	Mean Squared Error
MSS	Modified Sequential Scanning
NLANR	National Laboratory for Applied Network Research
PP	Proactive Protection
PPLS	Ping-Pong Localized Scanning
RS	Random Scanning
SI	Susceptible → Infected
SIS	Susceptible → Infected → Susceptible
SQL	Structured Query Language
SSH	Secure SHell
STD	STandard Derivation
UCN	Uncorrelated Complex Network
URL	Uniform Resource Locator
VT	Virus Throttling
WWW	World Wide Web

SUMMARY

As computer and communication networks become prevalent, the Internet has been a battlefield for attackers and defenders. One of the most powerful weapons for attackers is the Internet worm. Specifically, a worm attacks vulnerable computer systems and employs self-propagating methods to flood the Internet rapidly. As a result, worms, such as Code Red, Slammer, and Witty, have infected hundreds of thousands of hosts and become a significant threat to network security and management. Moreover, the attacking methods generated by worms' designers have become increasingly sophisticated, which poses considerable challenges to defenders.

The objective of this research is to characterize worm attack behaviors, analyze Internet vulnerabilities, and develop effective countermeasures. More specifically, some fundamental factors that enable a worm to be designed with advanced scanning methods are presented and investigated through mathematical modeling, simulations, and real measurements.

First, one factor is an uneven vulnerable-host distribution that leads to an optimal scanning method called importance scanning. Such a new method is developed from and named after importance sampling in statistics and enables a worm to spread much faster than both random and routable scanning. The information of vulnerable-host distributions, however, may not be known before a worm is released. To overcome this, worms using two sub-optimal methods are then investigated. One is a self-learning worm that can accurately estimate the underlying vulnerable-host distribution while propagating. The other is a localized-scanning worm that has been exploited by Code

Red II and Nimda worms. The optimal localized scanning and three variants of localized scanning are also studied. To fight against importance-scanning, self-learning, and localized-scanning worms, defenders should scatter applications uniformly in the entire IP-address space from the viewpoint of game theory. Next, a new metric, referred to as the non-uniformity factor, is presented to quantify both the unevenness of a vulnerable-host distribution and the spreading ability of network-aware worms. This metric is essentially the Renyi information entropy and better characterizes the non-uniformity of a distribution than the Shannon entropy. With the help of the non-uniformity factor, five data sets from real measurements show that vulnerable hosts are indeed highly unevenly distributed in the Internet.

Finally, another fundamental factor is topology information that enables topological-scanning worms. The spreading dynamics of topological-scanning worms are modeled through a spatial-temporal random process and simulated with both real and synthesized topologies.

CHAPTER I

INTRODUCTION

Since the Morris worm arose in 1988, Internet worms have been a persistent security threat. For example, the Code Red worm compromised at least 359,000 machines in 24 hours on July 19, 2001 [39]. The Slammer worm was unleashed with a 376-byte user datagram protocol (UDP) packet and infected more than 90% of vulnerable hosts in 10 minutes on January 25, 2003 [41]. These active worms caused large parts of the Internet to be temporarily inaccessible and cost both public and private sectors millions of dollars. Moreover, the frequency and the virulence of active-worm outbreaks have been increasing dramatically in the last few years, presenting a significant threat to today's Internet. Therefore, it is imperative to characterize the worm attack behaviors, analyze Internet vulnerabilities, and study countermeasures accordingly.

1.1 Internet Worm Attacks

A key characteristic of an Internet worm is self-propagation. That is, active worms can spread rapidly by infecting computer systems and by using infected hosts to disseminate the worms in an automated fashion. Based on the target-search process, we can divide Internet worms into two types: scan-based and topology-based worms.

1.1.1 Scan-Based Worms

A scan-based worm probes the entire IPv4 address space or the routable address space. For example, when a worm is released into the Internet, it simultaneously scans many hosts in an attempt to find a vulnerable host. When a target is found, the worm sends out a probe to infect it. After this target is compromised, the worm transfers a copy of itself to this host. The newly infected host then begins to run

the worm program and to compromise other targets. All these steps are combined into one for the Slammer worm [41]. That is, the Slammer worm uses a single UDP packet to scan, compromise, and spread the worm to targets.

1.1.2 Topology-Based Worms

A topology-based worm spreads through topological neighbors. For example, the Morris worm retrieves the neighbor list from the local Unix files */ect/hosts.equiv* and */.rhosts* and in individual users' *.forward* and *.rhosts* files. Another topological worm is a SSH worm, which locates new targets by searching its current host for the names and the addresses of other hosts that are likely to be susceptible to infection [55]. An email virus is another example of topological worms. When an email user receives an email message and opens the attachment containing a virus program, the virus infects the user's machine and uses the recipient's address book to send copies of itself to other email addresses. The addresses in the address book disclose the neighborhood relationship.

1.2 Research Objectives and Solutions

The objective of this thesis is to model and defend against worm attacks that use different advanced scanning methods. Specifically, we investigate the fundamental factors that enable a worm to be designed with advanced scanning methods. We attempt to answer the following important questions:

- What factors can help a worm spread faster and why?
- When worms take advantage of a specific factor, what is the "best-case scenario" for worm attacks?
- How can we analyze quantitatively the relationship between the spreading speed that worms can achieve and the factors that worms can use?

- How can we defend against such fast-spreading worms?

To investigate these questions, we apply mathematical modeling methodology and verify analytical results through simulations and real measurements. Mathematical models can provide quantitative analysis on the propagation dynamics of worms and the effectiveness of defense systems. Specifically, our mathematical models ignore the details of the infection process inside a single computer and focus on key characteristics of worm-spreading dynamics. For example, we consider a vulnerable host to be in one of two possible discrete statuses, *infected* or *susceptible*. A susceptible host can be infected by other infectious hosts, while an infected host can be recovered and become susceptible. Combining infection and recovery provides one of the simplest models, the *susceptible* $\rightarrow$ *infected* $\rightarrow$ *susceptible (SIS)* model. Here, the *susceptible* $\rightarrow$ *infected (SI)* model, which further ignores recovery, is regarded as a special case of the SIS model. Although simple, the SIS (or SI) model captures the most important characteristics of worm-scanning methods. Meanwhile, simulations and real measurements are used to verify our analytical results and approximations.

In this thesis, the following four topics are investigated:

1. **Designing an optimal worm-scanning method:** Most Internet worms use random scanning. The distribution of vulnerable hosts on the Internet, however, is highly non-uniform over the IP-address space. This implies that random scanning wastes many scans on invulnerable addresses and more virulent scanning schemes may take advantage of the non-uniformity of a vulnerable-host distribution. An optimal scanning method, *importance scanning*, is presented. Importance scanning is developed from and named after importance sampling in statistics and scans the IP-address space according to an empirical distribution of vulnerable hosts. Furthermore, a game-theory approach is applied to counteract importance-scanning worms.

2. **Analyzing sub-optimal worm-scanning methods:** The use of side information by an attacker can help a worm speed up the propagation. This philosophy has been the basis for advanced worm-scanning mechanisms such as hitlist scanning, routable scanning, and importance scanning. Some of these scanning methods use information on vulnerable hosts. Such information, however, may not be easy to collect before a worm is released. As the optimal scanning strategy is difficult to implement, two practical sub-optimal scanning methods are investigated. Specifically, a self-learning worm using importance scanning is presented. The self-learning worm is demonstrated to have the ability to accurately estimate the underlying vulnerable-host distribution if a sufficient number of infected hosts are observed. Another sub-optimal scanning method is localized scanning that has been used by Code Red II and Nimda worms. The optimal localized scanning and three variants of localized scanning are also studied.

3. **Evaluating the vulnerability of the Internet:** Five data sets from real measurements show the clustered vulnerable-hosts distributions consistently. The information on the highly uneven distributions of vulnerable hosts is exploited by network-aware worms, such as importance-scanning and localized-scanning worms. It is not well understood, however, how to characterize the relationships between vulnerable-host distributions and network-aware worms. A new metric, referred to as the *non-uniformity factor*, is presented to quantify both the unevenness of a vulnerable-host distribution and the spreading ability of network-aware worms. This metric is essentially the Renyi information entropy and better characterizes the non-uniformity of a distribution than the Shannon entropy.

4. **Modeling the spread of topological-scanning worms:** It is important that defenders understand how topological worms spread quantitatively. The spread of topological-scanning worms, however, is especially hard to model. The difficulty lies in characterizing the impact of topologies and the interactions among nodes in both space and time. A spatial-temporal model for worm propagation in networks is proposed. As the spatial dependence is particularly difficult to characterize, we propose the independent model and the Markov model as simple approximations. Our models are motivated by probabilistic graphs, which have been widely investigated in machine learning.

1.3 Thesis Outline

The rest of this thesis is organized as follows. Chapter 2 surveys the related work. Chapter 3 presents an optimal worm-scanning method using vulnerable-host distributions and the corresponding countermeasure from the viewpoint of game theory. Since the optimal worm-scanning method is difficult to implement, Chapter 4 and 5 focus on worms exploiting two sub-optimal scanning methods: a self-learning worm and a localized-scanning worm. Chapter 6 further studies a new metric, the non-uniformity factor, to quantify the unevenness of a vulnerable-host distribution and the spreading ability of network-aware worms. Next, Chapter 7 models topological-scanning worm propagation in network, using a spatial-temporal random process. Finally, Chapter 8 summarizes the research contributions and identifies several future research directions.

CHAPTER II

RELATED WORK

In this thesis, we focus on two types of worms: scan-based and topology-based worms. A scan-based worm probes the entire IPv4 address space or the routable address space, while a topology-based worm spreads through topological neighbors. These two types of worms are the most important worms and have been widely studied.

2.1 Scan-Based Worms

A scan-based worm spreads by employing distinct scanning mechanisms such as random, selective random, and localized scanning [59, 70]. *Random scanning* selects target IPv4 addresses at random and is used by such famous worms as Code Red and Slammer. *Selective random scanning* reduces the scanning space, using the information such as the Bogon list [62] and/or the IANA's IPv4 address allocation map [87], and is used by the Slapper worm. *Localized scanning* preferentially scans for hosts in the "local" address space and is used by Code Red II and Nimda worms.

Some advanced scanning methods have been developed in the research community. For example, Weaver presented the *hitlist-scanning* idea [68] to speed up the spread of worms at the initial stage. There, a list of vulnerable machines is built beforehand and targeted first when the worm is released. An extreme case for the hitlist-scanning worms is called *flash worms* [60], where IP addresses of all vulnerable machines are known in advance and gathered into the list. The flash worms are considered the fastest possible worms, as every worm scan can hit a vulnerable host. One other scanning method to improve the spread of worms is to use the information provided by BGP routing tables. This scanning method is called *routable scanning* [70, 79] and is a special case of selective random scanning. Zou et al. designed two types of

routable-scanning worms (also called *routing worms*) [79]. One is based on class-A (x.0.0.0/8) address allocations and is thus called a "Class-A routing worm." Such a worm can reduce the scanning space to 45.3% of the entire IPv4 address space. The other is based on BGP routing tables and is thus called a "BGP routing worm." Such a worm can reduce the scanning space to only about 28.6% of the entire IPv4 address space. One other strategy that a worm can potentially employ is *DNS random scanning* [26], where a worm uses the DNS infrastructure to locate likely targets by guessing DNS names instead of IP addresses. Such a worm in an IPv6 Internet is shown to exhibit a propagation speed comparable to that of an IPv4 random-scanning worm.

Most of these advanced worms can propagate far faster than a traditional random-scanning worm. When these advanced worms are studied, however, vulnerable hosts are assumed to be uniformly distributed in either the entire IPv4 address or the scanning space. Hence, the information on a vulnerable-host distribution is not exploited by the worms.

Moreover, these advanced scanning mechanisms have been developed based on the philosophy: *The use of side information by an attacker can help a worm speed up the propagation.* In the Internet, however, it may not be easy for attackers to collect the information on vulnerable hosts. For example, Windows SQL database servers do not advertise their addresses [41, 79]. Therefore, it is difficult for the Slammer worm to obtain the list of vulnerable hosts or the underlying distribution of vulnerable hosts before the worm is released.

Only a handful of works have been carried out on localized scanning. Chen et al. pointed out that if the vulnerable hosts are uniformly distributed in the IPv4 address space, localized scanning spreads at a slightly slower rate than random scanning [8]. Zou et al. showed that if the vulnerable hosts are uniformly distributed only in the

routable address space, localized scanning has a spreading speed comparable to Class-A routable scanning [81]. Rajab et al. further demonstrated that if the vulnerable hosts follow a power law distribution, localized scanning can propagate much faster than random scanning [49]. The prior work, however, focuses on simulation comparisons between localized scanning and random scanning. The mathematical reasoning on these comparisons has not been studied.

2.2 Topology-Based Worms

Topology-based worms (or *topological-scanning worms*) rely on the "address" information contained in the victim machines to locate new targets and is used by the Morris worm.

Several approaches have been proposed to model and simulate worm spreading in different topologies. Kephart and White presented the Epidemiological model, which is suitable for analyzing virus spreading in random graphs [32]. This work points out the difficulty in applying the Epidemiological model to study arbitrary topologies. Garetto et al. analyzed worm spreading in small-world topologies using a variation of the influence model, where the influence of neighbors is constrained to take a multilinear form [28]. Boguñá et al. studied epidemic spreading in complex networks [6], and Wang et al. proposed a model for virus propagation in arbitrary topologies [67]. Both works [6, 67] are proposed to obtain the epidemic threshold of virus infection. Zou et al. and Wang et al. investigated the effect of topologies and immunization on the propagation of computer viruses through simulation [80, 66]. Ganesh et al. modeled the spread of an epidemic as a contact process [36] to study what makes an epidemic either weak or potent [27]. The model assumes that a vulnerable node can be infected by its infectious neighbors at a rate that is proportional to the number of infected neighbors. Some recent investigations focus on random-scanning worms. Zou et al. modeled the spread of the Code Red worm, taking into consideration of the

human countermeasures and the worm's impact on the Internet infrastructure [78]. Chen et al. studied the propagation of active worms employing random scanning and extended the proposed modeling method to investigate the spread of localized-scanning worms [8]. Moore et al. applied the Epidemiological model to investigate the requirements for containing the self-propagation worm with random target selection [40]. The prior work, however, has not incorporated the spatial dependence on worm propagation in networks. This motivates the development of mathematical models to capture the spatial dependence and the use of spatial models to characterize both the transient and equilibrium behaviors of worm propagation with different scanning methods in arbitrary topologies. Furthermore, based on the models proposed in this thesis we studied the significance of the spatial dependence in determining epidemic thresholds and the speed of propagation [42].

CHAPTER III

OPTIMAL WORM-SCANNING METHOD

3.1 Introduction

As the number of computers and communication networks increases, Internet worms have become increasingly prevalent [39, 41, 56]. Using malicious, self-propagating codes, worms spread rapidly by infecting computer systems and disseminating themselves in an automated fashion using the infected hosts.

Most worms employ random scanning to select target IP addresses. Since the density of vulnerable hosts is low, a random scan hits a vulnerable machine with a small probability. For example, the Code Red worm infected a vulnerable population of 360,000 machines among 2^{32} IP addresses [77]. Thus, the probability that a random scan will hit a vulnerable target is only $\frac{360{,}000}{2^{32}} = 8.38 \times 10^{-5}$. Therefore, random scanning wastes many scans on invulnerable addresses.

Future worms, however, are likely to employ more effective scanning strategies to identify their targets. Hence, it is important that advanced scanning strategies that can potentially be used to access worst-case scenarios be studied. This chapter proposes such an optimal scanning method referred to as *importance scanning*. Importance scanning is inspired by importance sampling in statistics [72, 31, 57]. The basic idea of importance sampling is to make rare events occur more frequently and thus reduce the number of samples needed for accurately estimating the corresponding probability. Rare events for worm scanning correspond to hitting a target in a large population. Thus, importance scanning allows attackers to focus on the most relevant parts of an address space so that the probability of hitting vulnerable hosts increases.

Importance scanning relies on a certain statistic of an underlying vulnerable-host

distribution. An attacker can potentially obtain such information by querying a database of parties to the vulnerable protocol, stealthy scanning the (partial) target address space, and/or searching the records of old worms [59].

In view of the amount of information an attacker can obtain, random, flash [60], and routing [79] worms can be regarded as special cases of importance-scanning worms. In particular, a random worm has no information about the vulnerable-host distribution and thus regards the distribution as uniform in the IPv4 space. A flash worm acquires all knowledge, and the target distribution is uniform only in the vulnerable-population space. A routing worm has the knowledge from BGP routing tables about the space of existing hosts, and the corresponding distribution can be considered as uniform in the routing space.

In this chapter, we assume that a probability distribution of vulnerable hosts is available/obtainable. We then intend to answer the following questions:

- How can an attacker design a fast importance-scanning worm by taking advantage of the knowledge of the vulnerable-host distribution?
- How can we quantitatively analyze the relationship between the speed that worms can achieve and the knowledge that attackers can obtain?
- How can a defender counteract such importance-scanning worms?

To answer these questions, we focus on two quantities: the infection rate that characterizes how fast worms can spread at an early stage and the scanning strategy that is used to locate vulnerable hosts. We first derive relationships between the infection rate and scanning strategies. We then model the spread of importance-scanning worms, using the analytical active worm propagation (AAWP) model [8]. We derive the optimal scanning strategy that maximizes the infection rate. That is, the optimal strategy corresponds to the best-case scenario for attackers and the worst-case scenario for defenders. As the optimal strategy is difficult to achieve in reality, we

derive a sub-optimal scanning strategy as an approximation. To assess the virulence, we compare importance scanning with random and routable scanning. We take the empirical distributions of Witty-worm victims and Web servers as examples of the vulnerable-host distribution. We show that an importance-scanning worm based on parameters chosen from real measurements can spread nearly twice as fast as a routing worm before the victim population becomes saturated.

Moreover, we demonstrate, from the viewpoint of game theory, that a defense mechanism against importance-scanning worms requires the uniform distribution of an application. Under this defense strategy, the best strategy of importance scanning is equivalent to the random-scanning strategy.

Our designed importance scanning is inspired by importance sampling [72, 31, 57]. Our work, however, is different from [72] in that [72] is on estimating the density of Web servers and we focus on optimal scanning worms that use an uneven vulnerable-host distribution. Hence, while [72] studies a static quantity as the density of Web servers, we consider a dynamic process as the worm propagation. Moreover, [72] uses the variance of an estimator as the performance indicator, and we employ the worm propagation speed, such as the infection rate, as the objective function.

The remainder of this chapter is structured as follows. Section 3.2 provides the background on vulnerable-host distributions and a random worm propagation model. Section 3.3 describes the problem. Section 3.4 characterizes the importance-scanning strategy through the theoretical analysis. Section 3.5 shows the propagation speed of importance-scanning worms empirically. Section 3.6 further discusses the defense strategy and Section 3.7 concludes the chapter.

3.2 Preliminaries

3.2.1 Vulnerable-Host Distributions

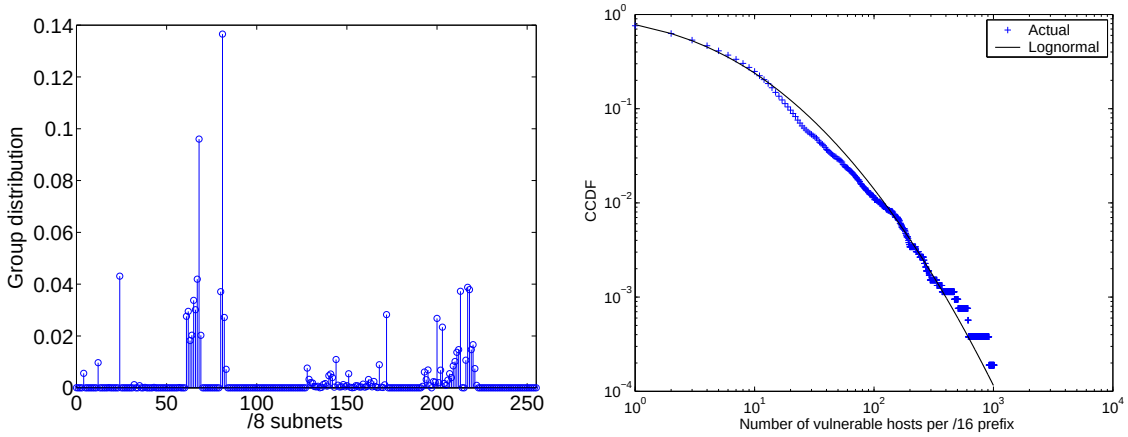
The distribution of vulnerable hosts in the Internet is not uniform. This is evident as no hosts can exist in reserved or multicast IPv4 address ranges assigned by the Internet Assigned Number Authority (IANA) [87, 73]. More importantly, the vulnerable-host distribution may be highly non-uniform over the registered IPv4 address space as indicated by our two collected traces.

The first trace is a traffic log of the Witty worm obtained from CAIDA [92]. The Witty worm attacks ISS firewall products and carries a destructive payload [56]. CAIDA used a *Network Telescope* to record the packets from the victims of the Witty worm. Since the network telescope approximately contains 2^{24} addresses, the collected trace can accurately reflect the distribution of hosts that are vulnerable to the Witty worm [56]. The collected victim addresses are then used to form a *group distribution* in /8 subnets, where

$$p_g(i) = \frac{\text{number of addresses with the first byte equal to } i}{\text{total number of collected addresses}}, \tag{1}$$

where $i = 0, 1, \cdots, 255$. The results are shown in Figure 1(a). It is observed that the distribution of vulnerable hosts is far from uniform. We further plot the complementary cumulative distribution function (CCDF) of the distribution of Witty-worm victims in /16 subnets in log-log scales in Figure 1(b) for collected data. The CCDF, denoted by $F(d)$, is defined as the fraction of the /16 subnets with the number of vulnerable hosts greater than d. We find that a lognormal distribution with mean 1.2 and standard deviation 1.55 closely fits these measurement data. This indicates that the distribution of Witty-worm victims nearly follows a *power law* distribution.

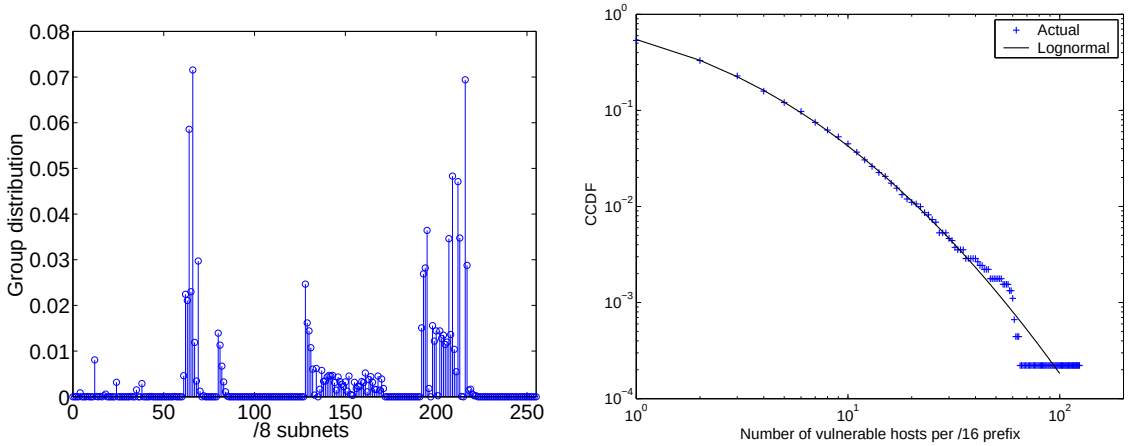
The second trace is the Web-server (port 80) distribution. To estimate the distribution of Web servers, we exploited a random uniform resource locator (URL)



(a) Group distribution of vulnerable hosts (/8 subnets).

(b) Vulnerable-host distribution (/16 subnets) in log-log scales and lognormal curve fitting.

Figure 1: Uneven distribution of hosts vulnerable to the Witty worm.



(a) Group distribution of Web servers (/8 subnets).

(b) Web-server distribution (/16 subnets) in log-log scales and lognormal curve fitting.

Figure 2: Uneven distribution of Web servers.

generator from UROULETTE (http://www.uroulette.com/) to collect 13,866 IP addresses of Web servers on January 24, 2005. Figures 2(a) and 2(b) show the group distribution in /8 subnets and the CCDF in /16 subnets for Web servers. The lognormal distribution has mean 0.15 and standard deviation 1.25, and closely fits the measurement data.

To summarize, the distributions on Web servers and Witty-worm victims are both non-uniform. These two distributions can both be approximated by lognormal

distributions but with different means and variances. In particular, the distribution of Witty-worm victims has a larger mean and a larger standard deviation than that of Web servers. This means that the Witty-worm victim distribution is more non-uniform than the Web-server distribution.

3.2.2 Random Worm Propagation Model

We now review a worm propagation model as preparation for relating the rate of worm spreading with the distribution of vulnerable hosts. A simple model, known as the *susceptible* $\rightarrow$ *infected* (SI) model, has been used to model the spread of random-scanning worms in various earlier works [59, 79, 26]. The model assumes that each host has only two states: susceptible or infected. Once infected, a host remains infected.

As importance scanning (sampling) is usually performed in discrete time [72], we adopt a discrete-time SI model. In particular, we use the analytical active worm propagation (AAWP) model, developed by Chen et al. in [8]. In the AAWP model, the spread of random-scanning worms is characterized as follows:

$$I_{t+1} = I_t + (N - I_t)[1 - (1 - \frac{1}{\Omega})^{sI_t}], \tag{2}$$

where I_t is the number of infected hosts at time t ($t \geq 0$); N is the number of vulnerable hosts; s is the scanning rate of the worm; and Ω is the scanning space. At $t = 0$, I_0 represents the number of hosts on the hitlist.

When a worm begins to spread, $I_t << N$ and $sI_t << \Omega$. The AAWP model can be approximated by

$$I_{t+1} \approx I_t + N \cdot \frac{sI_t}{\Omega} = (1 + \alpha)I_t, \tag{3}$$

where $\alpha = \frac{sN}{\Omega}$ is the *infection rate* [79]. The infection rate represents the average number of vulnerable hosts that can be infected per unit time by one infected host during the early stage of worm propagation. Based on Equation (3), $I_t \approx (1 + \alpha)^t I_0$, i.e., the number of infected hosts increases exponentially. Therefore, to speed up

the spread of worms at the early stage, attackers should design effective scanning methods to increase the infection rate. For instance, a traditional random worm scans the entire IPv4 address space, and thus $\Omega = 2^{32}$. The infection rate of this worm is $\alpha_0 = \frac{sN}{2^{32}}$. In contrast, a BGP and a Class-A routing worm can achieve faster infection rates with the same scanning rate and the same number of targets [79]: $\alpha_1 = \frac{sN}{0.286\times 2^{32}} = 3.5\alpha_0$ and $\alpha_2 = \frac{sN}{0.453\times 2^{32}} = 2.2\alpha_0$.

3.3 Problem Description

We now describe the problems studied in this chapter. Let s be the scanning rate or the number of scans that an infected host sends per unit time. Define A_n $(1 \le n \le s)$ as an IPv4 address probed by the nth scan from an infected host at the early stage of worm propagation. Thus, A_n is a random variable, and $A_n \in \{1, 2, \cdots, 2^{32}\}$. Let $I(A_n)$ denote the vulnerability of address A_n,

$$I(A_n) = \begin{cases} 1, & \text{if address } A_n \text{ is vulnerable to a worm;} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Thus, $\sum_{A_n} I(A_n) = N$. Let $p(A_n)$ denote the actual vulnerable-host distribution, i.e., the probability that $I(A_n) = 1$.

$$p(A_n) = \frac{I(A_n)}{N} = \begin{cases} \frac{1}{N}, & \text{if } I(A_n) = 1; \\ 0, & \text{if } I(A_n) = 0. \end{cases} \tag{5}$$

It is noted that $\sum_{A_n} p(A_n) = 1$.

Let $p^*(A_n)$ denote the probability that the worm scans address A_n. Note that $\sum_{A_n} p^*(A_n) = 1$. $p^*(A_n)$ can be a uniform distribution as in random-scanning worms or a non-uniform biasing distribution as in flash worms. $p^*(A_n)$ is chosen by an attacker. The choice of the scanning distribution $p^*(A_n)$ is essential to the effectiveness of importance scanning. As we shall see, $p^*(A_n)$ depends on the actual probability distribution $p(A_n)$.

In this chapter, we intend to answer the following questions:

- Given complete knowledge about $p(A_n)$, what is the optimal choice of $p^*(A_n)$ that maximizes infection rate α?
- Given partial knowledge about $p(A_n)$, what is the optimal choice of $p^*(A_n)$ that maximizes α?
- What are the spread dynamics of importance-scanning worms using the optimal or the practical choice of $p^*(A_n)$?
- How much faster can an importance-scanning worm spread than a random or a routing worm?
- How can we defend against such importance-scanning worms by customizing $p(A_n)$?

Table 1 shows the notations used in this thesis.

3.4 Importance Scanning

We begin by answering the first three of these five questions in this section. This suffices to deriving the infection rate of importance-scanning worms and modeling the spread of importance-scanning worms.

3.4.1 Infection Rate

Let R be the number of hosts that can be infected per unit time by one infected host during the early stage of worm propagation. R can be expressed as

$$R = \sum_{n=1}^{s} I(A_n), \tag{6}$$

where we assume that different scans do not hit the same target at the early stage of worm propagation, i.e., if $i \neq j$, then $A_i \neq A_j$. Therefore, the infection rate is given

Table 1: Notations used in this thesis.

Notation	Explanation
s	Scanning rate: Number of scans that an infected host sends per unit time
N	Total number of vulnerable hosts
Ω	Scanning space: address space that a worm scans
$p(A_n)$	Actual vulnerable-host distribution: Probability of address A_n being vulnerable to a worm
$p^*(A_n)$	Scanning distribution: Probability of a worm scan hitting address A_n
R	Number of vulnerable hosts that can be infected per unit time by one infected host during the early stage of worm propagation
α	Infection rate: $\alpha = E[R]$
I_t	Expected number of infected host at time t
m	Number of groups in the Internet
N_i	Number of vulnerable hosts in group i
Ω_i	Size of the address space in group i
D_i	Set of addresses in network i
$I_{t,i}$	Expected number of infected hosts in group i at time t
$p_g(i)$	Group distribution: Percentage of vulnerable hosts in group i
$p_g^*(i)$	Group scanning distribution: Probability of a worm scan hitting group i
$p_i(b)$	Interface distribution: Probability of finding a vulnerable host with the interface equal to b, given that the host is in network i
$p_i^*(b)$	Interface scanning distribution: Probability of scanning interface b, given that a scan hits network i
v_i	Vulnerable-host density: $v_i = \frac{p_g(i)}{\Omega_i}$

by

$$\begin{aligned} \alpha &= E_*[R] && (7)\\ &= \sum_{n=1}^{s} E_*[I(A_n)] && (8)\\ &= \sum_{n=1}^{s}\sum_{A_n} I(A_n)p^*(A_n) && (9)\\ &= N\sum_{n=1}^{s}\sum_{A_n} p(A_n)p^*(A_n) && (10)\\ &= sN\sum_{A_n} p(A_n)p^*(A_n), && (11) \end{aligned}$$

where $E_*[\cdot]$ denotes the expectation with respect to the scanning distribution $p^*(A_n)$. It is noted that

$$\alpha \le \sum_{n=1}^{s}\sum_{A_n} p^*(A_n) = s, \quad (12)$$

for any $p^*(A_n)$.

Hence, the infection rate is strongly influenced by the choice of scanning distribution $p^*(A_n)$. A choice of $p^*(A_n)$ determines a scanning strategy, and a good choice, in the view of an attacker, should maximize infection rate α. Two special cases have been observed on "choosing" $p^*(A_n)$. The first case is the random-scanning worms, in which $p^*(A_n) = \frac{1}{2^{32}}$. Thus, $\alpha = \frac{sN}{2^{32}} = \alpha_0$. The second case is the flash worms, in which $p^*(A_n) = p(A_n)$. In this case, $p^*(A_n)$ obtains the optimal scanning strategy $p^*_{opt}(A_n)$, which leads to $\max_{p^*(A_n)}\{\alpha\} = s$, indicating that every scan from the worm would hit a vulnerable host.

One interpretation of $p^*_{opt}(A_n)$ suggests that a good worm scanning strategy should concentrate the scans on the areas that are more likely to find a vulnerable host. The vulnerable-host probability distribution $p(A_n)$, however, cannot be obtained without probing the entire IP address space or gathering a complete database of parties to the vulnerable protocol. Therefore, attackers may not acquire the entire knowledge of $p(A_n)$. However, partial knowledge can be obtained, e.g., by aggregating the subspaces of IP addresses.

3.4.2 Group Distributions

Such partial information is referred to as *group distributions*, which capture the statistics of groups of addresses rather than individual addresses. The vulnerable-host probability distribution in groups is essentially the marginal of the actual distribution $p(A_n)$. Such groups of addresses can be formed in several ways. For example, IP addresses can be grouped by using the conventional 4-byte description. In [72], this approach is applied to measure the size of the Internet via importance sampling. Here, we extract relevant groups in a more general setting by defining the networks. In particular, we regard a *network* as a group of IP addresses that can be identified by such diverse methods as either the first byte of IP addresses (/8 subnets) or IP

prefixes in classless inter-domain routing (CIDR).

We assume that the Internet is partitioned into m networks. Let D_i $(i = 1, 2, \cdots, m)$ denote the partition set of addresses in network i, which has Ω_i $(\Omega_i \geq 0)$ addresses. Thus, $\sum_{i=1}^{m} \Omega_i = \Omega = 2^{32}$. We define the group distribution $p_g(i)$ $(i = 1, 2, \cdots, m)$ as the proportion of vulnerable hosts in network i, i.e.,

$$p_g(i) = \frac{N_i}{N} = \sum_{A_n \in D_i} p(A_n), \tag{13}$$

where N_i is the population of vulnerable hosts in network i.

The partition of networks reflects the knowledge that attackers can obtain. For example, in one extreme case of random-scanning worms, $m = 1$ and $\Omega_1 = 2^{32}$. In the other extreme case of flash worms, $m = 2^{32}$ and $\Omega_i = 1$ $(i = 1, 2, \cdots, 2^{32})$. Another choice of partitioning networks is based on the first byte of IP addresses (/8 subnets), where $m = 2^8$ and $\Omega_i = 2^{24}$ $(i = 1, 2, \cdots, 2^8)$. The amount of knowledge collected by the worm with the /8 subnet distribution is only partial, somewhere between that by the random worm and that by the flash worm.

Recall that the goal of importance scanning is to maximize the infection rate. From Equation (11), we have the infection rate

$$\alpha = sN \sum_{i=1}^{m} \sum_{A_n \in D_i} p(A_n) p^*(A_n). \tag{14}$$

Refer to the location of an address A_n that is in network i as the *interface* denoted by b $(b = 0, 1, \cdots, \Omega_i - 1)$. Let $p_i(b)$ denote the actual probability of finding a vulnerable host with the interface equal to b, given that the host is in network i, i.e., $p_i(b) = \frac{I(A_n)}{N_i}$. Similarly, define *group scanning distribution* $p_g^*(i)$ as the probability of scanning network i and *interface scanning distribution* $p_i^*(b)$ as the probability of scanning interface b, given that a scan hits network i for the scanning distribution $p^*(A_n)$. We can obtain

$$p(A_n) = p_g(i) \cdot p_i(b) \tag{15}$$

$$p^*(A_n) = p_g^*(i) \cdot p_i^*(b), \tag{16}$$

where A_n is in network i with interface b. From Equations (15) and (16), the infection rate becomes

$$\alpha = sN \sum_{i=1}^{m} \sum_{b=0}^{\Omega_i - 1} p_g(i) p_i(b) p_g^*(i) p_i^*(b) \tag{17}$$

$$= sN \sum_{i=1}^{m} \left[p_g(i) p_g^*(i) \sum_{b=0}^{\Omega_i - 1} p_i(b) p_i^*(b) \right]. \tag{18}$$

We assume that attackers can obtain information only about group distribution $p_g(i)$ and cannot acquire further knowledge about interface distribution $p_i(b)$. Therefore, if a scan hits network i, the Ω_i hosts in this network are targeted by that scan with the same likelihood, i.e., $p_i^*(b) = \frac{1}{\Omega_i}$. Hence, Equation (18) yields

$$\alpha = sN \sum_{i=1}^{m} \frac{p_g(i) p_g^*(i)}{\Omega_i}. \tag{19}$$

Equation (19) provides the relationships among the infection rate, the group distribution, and the group scanning distribution. Let $v_i = \frac{p_g(i)}{\Omega_i}$, referred to as the *vulnerable-host density* in group i, then

$$\alpha = sN \sum_{i=1}^{m} v_i p_g^*(i) \tag{20}$$

$$\leq sN \sum_{i=1}^{m} \max_k \{v_k\} p_g^*(i) \tag{21}$$

$$= sN \max_k \{v_k\}. \tag{22}$$

The equality holds when

$$p_g^*(j) = \begin{cases} 1, & j = \arg\max_k \{v_k\}; \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

This means that the optimal importance scanning of a worm is to scan only the network with the largest vulnerable-host density.

3.4.3 Importance-Scanning Worm Propagation Model

We now model the spreading dynamics of importance-scanning worms based on the information of a group distribution.

At time t ($t \geq 0$), let $I_{t,i}$ denote the average number of infected hosts in network i. Thus, the total number of infected hosts $I_t = \sum_{i=1}^{m} I_{t,i}$. The rate at which network i is scanned is $sI_t p_g^*(i)$. As an importance scanning worm employs random scanning *within* each network, on the next time epoch, the number of infected hosts in network i can be derived by the AAWP model, i.e.,

$$I_{t+1,i} = I_{t,i} + (N_i - I_{t,i})[1 - (1 - \frac{1}{\Omega_i})^{sI_t p_g^*(i)}], \tag{24}$$

where $i = 1, 2, \cdots, m$ and $t \geq 0$. $I_{0,i}$ is the number of initially infected hosts in network i. The above equation yields

$$I_{t+1,i} = I_{t,i} + sI_t \frac{(N_i - I_{t,i})p_g^*(i)}{\Omega_i} - O(\frac{1}{{\Omega_i}^2}). \tag{25}$$

Since $\frac{1}{\Omega_i} << 1$, we ignore item $O(\frac{1}{{\Omega_i}^2})$. Summing over $i = 1, 2, \cdots, m$ on both sides, we obtain

$$\begin{aligned} I_{t+1} &= I_t + sI_t \sum_{i=1}^{m} (\frac{N_i - I_{t,i}}{\Omega_i}) p_g^*(i) && (26) \\ &\leq I_t + sI_t \sum_{i=1}^{m} \max_k \{\frac{N_k - I_{t,k}}{\Omega_k}\} p_g^*(i) && (27) \\ &= [1 + s \cdot \max_k \{\frac{N_k - I_{t,k}}{\Omega_k}\}] I_t. && (28) \end{aligned}$$

The equality holds when

$$p_g^*(j) = \begin{cases} 1, & j = \arg\max_k \{\frac{N_k - I_{t,k}}{\Omega_k}\}; \\ 0, & \text{otherwise.} \end{cases} \tag{29}$$

When $t = 0$, $N_i >> I_{t,i}$ and then $\max_k \{\frac{N_k - I_{t,k}}{\Omega_k}\} \approx N \max_k \{v_k\}$, which leads to $\alpha = sN \max_k \{v_k\}$. The above derivation results in an optimal importance-scanning strategy that maximizes the infection rate.

Optimal importance scanning:

1. At each time step t, the worm first finds the network that has the largest value of the left vulnerable-host density, i.e., $j = \arg\max_k \{\frac{N_k - I_{t,k}}{\Omega_k}\}$.

2. Then all infected hosts concentrate on scanning this network. That is, $p_g^*(j) = 1$ and $p_g^*(i) = 0$, $\forall i \neq j$.

This optimal importance scanning strategy, however, is difficult to implement. First, N may not be known in advance. Second, the network that has the largest value of the left vulnerable-host density changes with time, and therefore, the optimal assignment of $p_g^*(i)$ is time-varying. Even when N were given, it would require that each infected host knows $I_{t,i}$, which leads to numerous information exchanges between infected hosts. However, the essence of optimal importance scanning is that it provides the best scenario of worm scanning using the vulnerable-host distribution, which can be used as the baseline for a sub-optimal selection of $p_g^*(i)$.

A simple strategy for sub-optimal importance scanning is to assume $p_g^*(i) = \frac{p_g(i)/\Omega_i}{\sum_{j=1}^{m} p_g(j)/\Omega_j}$. That is, the probability that a worm scans network i is proportional to the vulnerable-host density of this network. If $\Omega_1 = \Omega_2 = \cdots = \Omega_m$, then $p_g^*(i) = p_g(i)$. For this scanning strategy, Equation (24) becomes

$$I_{t+1,i} = I_{t,i} + (N_i - I_{t,i})[1 - (1 - \frac{1}{\Omega_i})^{sI_t \frac{p_g(i)/\Omega_i}{\sum_{j=1}^{m} p_g(j)/\Omega_j}}]. \tag{30}$$

Sub-optimal importance scanning:

1. Before a worm is released, an attacker first obtains vulnerable-host group distribution $p_g(i)$ and then encodes group scanning distribution $p_g^*(i) = \frac{p_g(i)/\Omega_i}{\sum_{j=1}^{m} p_g(j)/\Omega_j}$ in the worm code.

2. At each time step t, the worm scans network i with probability $p_g^*(i)$.

3.5 Experiments

In this section, we study the propagation speed of importance-scanning worms based on parameters chosen from the real measurements. We first introduce the experimental set-up. We then show the effect of knowledge and vulnerable-host distributions

on the propagation of importance-scanning worms. Finally, we compare importance scanning with random and routable scanning.

3.5.1 Experimental Set-up

In our experiments, we use the model in Equation (2) to imitate the spread of random-scanning and routing worms. Meanwhile, we employ the model in Equations (24) and (29) to study propagation as a result of the optimal importance-scanning strategy. We also use the model in Equation (30) to simulate the spread of sub-optimal importance-scanning worms. To implement the models in Equations (24), (29), and (30), we need to obtain group distribution $p_g(i)$. Here, we use the Witty-worm victim and the Web-server distributions as examples of the vulnerable-host distribution. In other words, we assume that worms attack vulnerable hosts with the same group distribution as that of Witty-worm victims or Web servers. Our collected trace of Web servers does not include all Web servers. However, we assume that the estimated results obtained by Equation (1) are the actual group distribution of Web servers.

The parameters we use in simulated worms are comparable to those in Witty and Code Red worms for evaluating propagation. The Witty worm has a vulnerable population $N = 12,000$ and a scanning rate $s = 1,200$ per second [56, 81]. The Code Red worm has parameters $N = 360,000$ and $s = 358$ per minute [77]. The victims of the Code Red worm is assumed to have the same group distribution as Web servers. We then refer to such an importance-scanning worm as the importance-scanning (IS) Witty or Code Red. Since the experimental results of the Code Red worm are similar to those of the Witty worm, we mainly present the observations from the Witty worm.

3.5.2 Knowledge Effect

The amount of knowledge about a vulnerable-host distribution affects the rate of spread of importance-scanning worms. Figure 3 shows the propagation comparison among sub-optimal importance-scanning Witty worms with different amounts

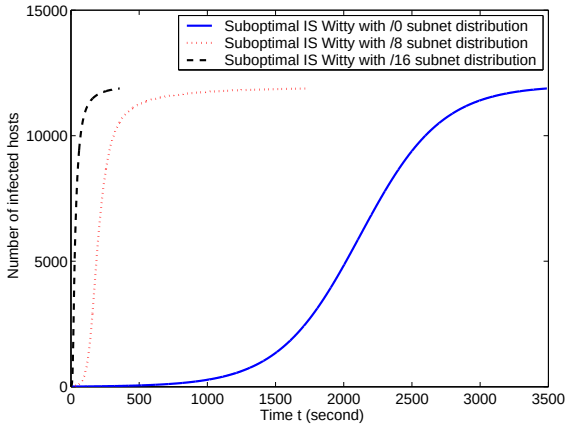


Figure 3: Effect of knowledge.

of knowledge about the vulnerable-host distribution, assuming a hitlist of 10 (i.e., $I_0 = 10$). If a worm has the /0 subnet distribution, it knows nothing about the distribution and thus has to use random scanning. We assume that all three Witty worms have the same scanning rate, although a worm that contains more information about the group distribution might slow down for a larger payload. It takes the Witty worm with a /0 subnet distribution 46.3 minutes to infect 90% of vulnerable hosts, whereas the Witty worms with a /8 subnet distribution and a /16 subnet distribution take only 6.6 minutes and 1.6 minutes, respectively. Therefore, more information about the vulnerable-host distribution may help an attacker design a faster worm.

3.5.3 Vulnerable-Host Distribution Effect

A vulnerable-host distribution also affects the rate of propagation of importance-scanning worms. Figure 4 demonstrates the spread of the sub-optimal importance-scanning Witty worms using the /8 subnet distribution, in which vulnerable hosts follow different distributions, assuming a hitlist of 10 (i.e., $I_0 = 10$). A uniform distribution in IPv4 can slow down the worm at least six times than the Witty-worm victim distribution before the victim population becomes saturated. Therefore, the distribution of vulnerable hosts strongly affects the rate of spread of importance-scanning worms.

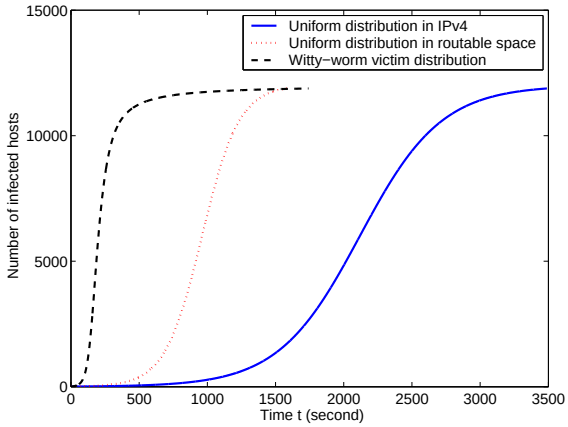
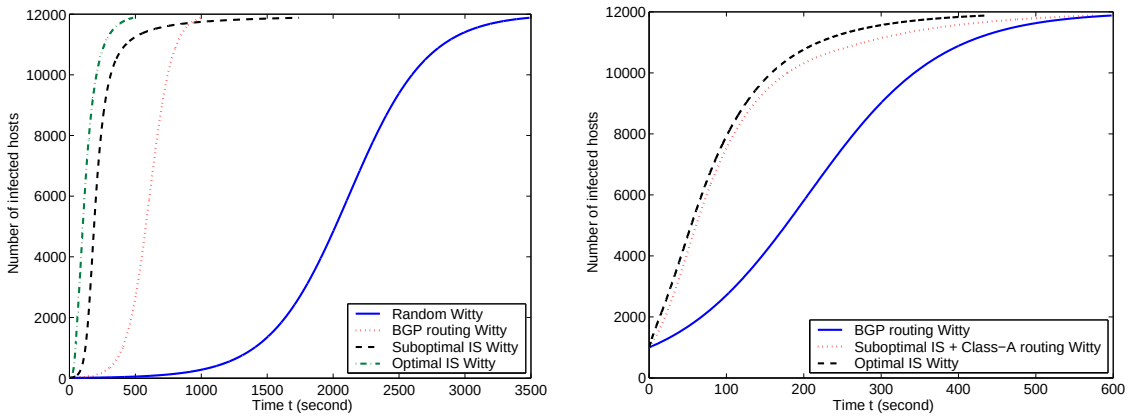


Figure 4: Effect of distributions.

3.5.4 Propagation Comparisons

Importance scanning also helps hasten the propagation of a worm. Figure 5(a) shows how propagation as a result of importance-scanning Witty worms compares with that of random and BGP routing Witty worms, assuming a hitlist of 10 (i.e., $I_0 = 10$). The rate of spread of importance-scanning Witty worms increases significantly by using the information on the /8 subnet distribution of vulnerable hosts. The optimal importance-scanning Witty worm can infect 90% vulnerable hosts in as few as 4.2 minutes, whereas the BGP routing Witty worm requires 13.3 minutes. The sub-optimal importance-scanning Witty worm spreads more slowly than the optimal worm, but only takes 6.6 minutes to infect the same number of hosts. A BGP routing worm obtains the refined information about the routable space than the worm using the /8 subnet distribution. The BGP routing worm, however, employs random scanning in the BGP routable space. Hence, such a worm, most of time, spreads more slowly than the importance-scanning worms with the /8 subnet distribution, which exploits the underlying uneven distribution of vulnerable hosts.

Once most of the vulnerable hosts are infected, the spread of the sub-optimal importance-scanning Witty worm slows down. This is because the sub-optimal strategy always uses the same group scanning distribution. As the infected hosts become



(a) Witty worm with a hitlist of 10

(b) Witty worm with a hitlist of 1,000

Figure 5: Witty worm propagation comparisons.

saturated, a network that initially has more vulnerable hosts actually contains fewer uninfected vulnerable machines. To overcome this problem, sub-optimal importance scanning can choose to switch to the routable scanning when only a few uninfected vulnerable hosts are left. Figure 5(b) shows the results for the same experiments, assuming a hitlist of 1,000. Sub-optimal importance scanning switches to Class-A routable scanning when 90% vulnerable hosts are infected. Compared with the propagation of a BGP routing worm, importance-scanning worms spread faster before the victim population becomes saturated.

Figure 6 shows the propagation comparison among an optimal importance-scanning Code Red worm, a sub-optimal importance-scanning Code Red worm, a Class-A routing Code Red worm, and a random Code Red worm, assuming $I_0 = 10$. The importance-scanning Code Red worms use the /8 subnet distribution. The sub-optimal importance-scanning Code Red worm can propagate nearly twice as fast as the Class-A routing Code Red worm before the victim population becomes saturated.

With regard to the storage requirement for /8 subnet group-distribution information, each $p_g(i)$ requires 4 bytes, and each /8 prefix 1 byte. Therefore, the total number of bytes is $5 \times 256 = 1280$. We can reduce this payload by removing the entries with $p_g(i) = 0$, where $i \in \{0, 1, \cdots, 255\}$. Since there are only 97 entries with

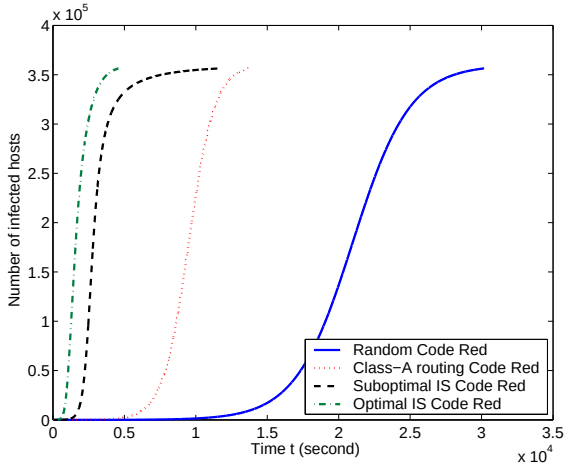


Figure 6: Code Red worm propagation comparisons.

non-zero $p_g(i)$'s according to the empirical distribution in Figure 1(a), the table can be stored in a $97 \times 5 = 485$ byte payload. Hence, the scanning rate of importance-scanning worms will not decrease much.

3.6 Game Theory for Attackers and Defenders

Defense against such importance-scanning worms can be modeled by relating it to the interaction between attackers and defenders in game theory. Assume that when an application is introduced to the Internet, defenders can choose how to deploy this application in networks. That is, group distribution $p_g(i)$ can be controlled by defenders, thus leading to a game between attackers and defenders. The attackers attempt to maximize the infection speed (characterized by infection rate α in Equation (19)) by choosing optimal group scanning distribution $p_g^*(i)$, while the defenders endeavor to minimize the worm propagation speed by customizing group distribution $p_g(i)$. Let $V = \{p_g^* : \sum_{i=1}^m p_g^*(i) = 1\}$ stand for the set of group scanning probability vectors p_g^*. Let $U = \{p_g : \sum_{i=1}^m p_g(i) = 1\}$ represent the set of feasible probability assignments for the application distribution. An attacker fears that if a defender knows about the worm-scanning strategy, the defender would then choose a strategy that $\min_{p_g \in U}\{\alpha\}$. Therefore, the objective of an attacker is to choose group scanning distribution $p_g^*(i)$

that maximizes the minimum value, i.e.,

$$\max_{p_g^* \in V} \min_{p_g \in U} sN \sum_{i=1}^{m} \frac{p_g(i)p_g^*(i)}{\Omega_i}. \tag{31}$$

In a similar argument, the objective of a defender is

$$\min_{p_g \in U} \max_{p_g^* \in V} sN \sum_{i=1}^{m} \frac{p_g(i)p_g^*(i)}{\Omega_i}. \tag{32}$$

This is a classical two-person zero-sum game, and the following well-known theorem [45] gives an optimal solution.

Theorem 1 *There exists an optimal solution to the worm-scanning game, where*

$$\alpha_{opt} = \max_{p_g^* \in V} \min_{p_g \in U} sN \sum_{i=1}^{m} \frac{p_g(i)p_g^*(i)}{\Omega_i} \tag{33}$$

$$= \min_{p_g \in U} \max_{p_g^* \in V} sN \sum_{i=1}^{m} \frac{p_g(i)p_g^*(i)}{\Omega_i}, \tag{34}$$

where α_{opt} is the value of the game.

The solution of this *minmax* problem is derived in the following theorem.

Theorem 2 *The value of the worm-scanning game is $\alpha_{opt} = \frac{sN}{2^{32}}$, and the best strategy for a defender is to distribute the application uniformly in the Internet, i.e., $p_g(i) = \frac{\Omega_i}{2^{32}}$, where $i = 1, 2, \cdots, m$.*

PROOF: From Equation (22), we have

$$\max_{p_g^* \in V} \alpha = sN \max_k \{\frac{p_g(k)}{\Omega_k}\}. \tag{35}$$

Set $J = \max_k \{\frac{p_g(k)}{\Omega_k}\}$. The optimal choice of $p_g(i)'s$ requires that J be minimized. Since $\frac{p_g(i)}{\Omega_i} \leq \max_k \{\frac{p_g(k)}{\Omega_k}\} = J$, $p_g(i) \leq J\Omega_i$ for $\forall i$. Thus,

$$1 = \sum_{i=1}^{m} p_g(i) \leq \sum_{i=1}^{m} J\Omega_i = J\Omega, \tag{36}$$

which leads to $J \geq \frac{1}{\Omega}$. The inequality holds when $\frac{p_g(i)}{\Omega_i} = J = \frac{1}{\Omega}$ for $\forall i$. That is, $p_g(i) = \frac{\Omega_i}{\Omega} = \frac{\Omega_i}{2^{32}}$, where $i = 1, 2, \cdots, m$, i.e., the defenders should deploy the application uniformly in the entire IP-address space.

Combining $p_g(i) = \frac{\Omega_i}{2^{32}}$ with Equation (35), the game value is $\alpha_{opt} = \frac{sN}{2^{32}}$. ∎

From Theorem 2, we note that when the defender uses the optimal strategy, the best strategy that the attacker exploits is equivalent to the random-scanning strategy. Meanwhile, Figure 4 demonstrates that the vulnerable-host distribution has a strong effect on worm propagation. Therefore, the design of the future Internet should consider how to distribute an application in security engineering.

In the current Internet, however, the application distributor may not control how to deploy the application in the entire IPv4 address space. Although not applicable for the entire Internet, the best strategy of defenders can still apply for enterprise networks. That is, if an enterprise network attempts to defend against importance-scanning worms, the administrator of this network should distribute the application uniformly in the entire enterprise network from the viewpoint of game theory.

3.7 Summary

In order to effectively defend against Internet worms, we must study potential scanning techniques that attackers may employ. In this chapter, we present an optimal worm-scanning method, called *importance scanning*, using the information of a vulnerable-host distribution. This scanning strategy then provide a best-case scenario for attackers when the vulnerable-host distribution is available. Importance scanning can be combined with other scanning methods such as hitlist scanning. Moreover, the division of groups can be very general, such as domain name system (DNS) top-level domains, countries, Autonomous Systems, IP prefixes in CIDR, the first byte of IP addresses (/8 subnets), or the first two bytes of IP addresses (/16 subnets). For example, when naming distribution information is exploited, importance scanning can also be applied to DNS worms [26], which is worth further investigation. In addition, when IPv4 is updated to IPv6, an importance-scanning worm will not be slowed down

very much if vulnerable hosts are still distributed in a clustered fashion. A game-theoretical approach suggests that the best strategy for defenders is to distribute the applications evenly in the entire address space or in each enterprise network.

CHAPTER IV

SUB-OPTIMAL WORM-SCANNING METHOD: A SELF-LEARNING WORM

4.1 Introduction

Some advanced scanning mechanisms, such as hitlist scanning, routable scanning, and importance scanning, have been developed based on the philosophy: *The use of side information by an attacker can help a worm speed up the propagation.* In the Internet, however, it may not be easy for attackers to collect information on vulnerable hosts. For example, Windows SQL database servers do not advertise their addresses [41, 79]. It is therefore difficult that the Slammer worm obtain a list of vulnerable hosts or an underlying vulnerable-host distribution before the worm is released. Nevertheless, future worms can become more intelligent and potentially learn a certain knowledge about, e.g. the vulnerable-host distribution, while propagating. For example, attackers can estimate the distribution using measurements[1]. In this work, we study worm behaviors that utilize information on the vulnerable-host distribution. In particular, we focus on self-learning worms and intend to answer the following questions:

- How can a worm self-learn about a vulnerable-host distribution from measurements and make use of such information?
- What is the performance of a self-learning worm?

Here the *performance* refers to the propagation speed of worms. If a worm spreads faster, it has a better performance and is thus more virulent.

When a group vulnerable-host distribution is available, the optimal way for worms to scan is to perform importance sampling, resulting in importance-scanning worms

[1] Yes, attackers can use measurement-based approaches in a similar way to networking researchers.

as studied initially in the previous chapter and our prior work [10, 17]. When the knowledge of the vulnerable-host group distribution is unavailable before spreading, we design a self-learning worm. Such a worm begins with random scanning or routable scanning and collects information on the IP addresses of infected hosts while propagating. The key capability of this worm is to learn an underlying vulnerable-host group distribution. The resulting worm spreading then consists of two stages: a learning stage where worms (attackers) obtain an empirical vulnerable-host distribution from measurements and a sampling stage where worms scan vulnerable hosts using the group distribution. The virulence of such a worm can be characterized through a dynamic worm-propagation model.

We show analytically and empirically that the self-learning worm can accurately estimate the group distribution in /8 subnets through a simple and unbiased proportion estimator using as few as 500 samples (IP addresses of infected hosts). After estimating the vulnerable-host distribution, this self-learning worm switches to importance scanning. The optimal importance-scanning method has been proposed in our prior work [10]. This optimal approach, however, is difficult to implement, since it requires numerous information exchanges between infected hosts. Therefore, we derive a practical importance-scanning strategy that optimizes a new metric on the effectiveness of scanning. This metric reflects the average number of worm scans required until the first scan hits a randomly chosen vulnerable host. We demonstrate the optimality of such importance scanning through analysis and simulation.

To evaluate the performance of our proposed self-learning worms, we use two data sets, the distributions of Web servers and Witty-worm victims, as the examples of the vulnerable-host distribution. We show that a self-learning worm based on parameters chosen from the real measurements of the Code Red v2 worm spreads nearly five times faster than a random-scanning worm, four times faster than a permutation-scanning worm, and two times faster than a Class-A routing worm, after collecting 500 samples

and estimating the group distribution in /8 subnets.

The remainder of this chapter is structured as follows. In Section 4.2, we give a problem description. In Section 4.3, we characterize the optimal static importance-scanning strategy through theoretical analysis. We then design a self-learning worm in detail in Section 4.4 and show the performance of such a self-learning worm in Section 4.5.We further discuss some guidelines for detecting and defending against such self-learning worms in Section 4.6. We conclude this chapter in Section 4.7 with a brief summary.

4.2 Problem Description

Assume that the Internet is composed of m groups. Let N_i and Ω_i denote the number of vulnerable hosts and the size of the address space in group i ($i = 1, 2, \cdots, m$), respectively. Thus, $\sum_{i=1}^{m} N_i = N$ and $\sum_{i=1}^{m} \Omega_i = \Omega$. Define the *group distribution* of vulnerable hosts, $p_g(i)$ ($i = 1, 2, \cdots, m$), as the ratio between the number of vulnerable hosts in group i and the total number of vulnerable hosts, i.e., $p_g(i) = \frac{N_i}{N}$. Define the *group scanning distribution*, $p_g^*(i)$ ($i = 1, 2, \cdots, m$), as the probability that a worm scan hits group i. Thus, $\sum_{i=1}^{m} p_g(i) = 1$ and $\sum_{i=1}^{m} p_g^*(i) = 1$.

There are two types of importance scanning: *dynamic importance scanning* if $p_g^*(i)$'s vary with time and *static importance scanning* if $p_g^*(i)$'s are fixed at all time. For static importance-scanning strategies, assuming that $\Omega_1 = \Omega_2 = \cdots = \Omega_m = \frac{2^{32}}{m}$, we can relate the group scanning distributions $p_g^*(i)$ with the group distributions $p_g(i)$ in the following formula:

$$p_g^*(i) = \frac{(p_g(i))^n}{\sum_{k=1}^{m} (p_g(k))^n} \propto (p_g(i))^n . \tag{37}$$

In our study, $p_g(i)$'s represent an underlying group probability distribution of vulnerable hosts[2]. $p_g(i)$ may or may not be available in advance. An intelligent

[2]For example, if $p_g(i)$ is a uniform distribution, vulnerable hosts are uniformly distributed in the entire IP address space.

worm, however, can learn this distribution by observing the measurements (e.g. IP addresses of vulnerable hosts). Given a set of L measurements for estimating $p_g(i)$, we study a self-learning worm through

1. learning, where attackers (worms) estimate $p_g(i)$ using given measurements;
2. sampling, where worms scan the IP-address space based on an optimal use of the learned vulnerable-host distribution;
3. accessing the performance, where we obtain the speed of worm propagation through dynamic models for worm spreading;
4. deriving defense strategies for self-learning worms.

4.3 Optimal Static Importance-Scanning Strategy

In this section, we derive an optimal static importance-scanning strategy, assuming that a vulnerable-host distribution is given.

As stated in the previous chapter, the optimal dynamic importance-scanning strategy is difficult to implement. One alternative selection is the static importance scanning that avoids information exchanges between infected hosts. We design the optimal strategy for static importance scanning through a new metric. The metric is the average number of worm scans required until the first scan hits a randomly chosen vulnerable host. Such a metric is motivated by the intuition that a fewer scans a worm uses to hit a vulnerable host, the faster the worm spreads.

When a worm scan hits group i ($i \in \{1, 2, \cdots, m\}$), Ω_i hosts in this group are targeted by that scan with the same likelihood. That is, when considering a vulnerable host in group i, it has a probability of $\frac{1}{\Omega_i}$ to be hit by a worm scan given that the scan hits the group. Thus, a vulnerable host in group i is hit by an importance-scanning worm scan with probability

$$p_h(i) = p_g^*(i) \cdot \frac{1}{\Omega_i}. \tag{38}$$

Since the events of a vulnerable host being hit are independent in static importance scanning, the number of scans required until the first scan hits an appointed vulnerable host in group i, denoted by F_i, follows a geometric distribution [53]

$$P(F_i = j) = p_h(i)\,(1 - p_h(i))^{j-1}, \quad j = 1, 2, \cdots. \tag{39}$$

Then, the expected number of scans needed until this vulnerable host is hit is

$$E[F_i] = (p_h(i))^{-1} = \frac{\Omega_i}{p_g^*(i)}. \tag{40}$$

Therefore, if we randomly choose a vulnerable host in the Internet, the average number of scans required until the first scan hits this host, denoted by Y, is

$$Y = \frac{1}{N}\sum_{i=1}^{m} N p_g(i)\frac{\Omega_i}{p_g^*(i)} = \sum_{i=1}^{m}\frac{\Omega_i p_g(i)}{p_g^*(i)}, \tag{41}$$

where $Np_g(i)$ is N_i, the number of vulnerable hosts in group i. Intuitively, a good metric for measuring the effectiveness of scanning strategies is the average number of scans required for hitting all vulnerable hosts divided by the number of vulnerable hosts. The expression of this metric, however, is complex and difficult to obtain. Instead, Y gives an alternative metric for the effectiveness of scanning strategies. A better static importance-scanning strategy leads to a smaller Y. Thus, the goal of the static importance scanning is to minimize Y. The optimal static importance-scanning strategy can be found by the Lagrangian optimization of Y as shown in the following theorem.

Theorem 3 *Among all possible static importance-scanning strategies, the group scanning distribution $\tilde{p}_g^*(i)$ is optimal in minimizing Y subject to $\sum_{i=1}^{m} p_g^*(i) = 1$, where*

$$\tilde{p}_g^*(i) = \frac{\sqrt{\Omega_i p_g(i)}}{\sum_{k=1}^{m}\sqrt{\Omega_k p_g(k)}}. \tag{42}$$

PROOF: The optimal static importance-scanning strategy can be found by minimizing Y. Let the Lagrangian objective function be

$$J = \sum_{i=1}^{m}\frac{\Omega_i p_g(i)}{p_g^*(i)} + \lambda\left(\sum_{i=1}^{m} p_g^*(i) - 1\right). \tag{43}$$

For each group i, differentiating with respect to $p_g^*(i)$ and setting the result equal to zero yield $\tilde{p}_g^*(i) = \sqrt{\frac{\Omega_i p_g(i)}{\lambda}}$. The constraint $\sum_{i=1}^m \tilde{p}_g^*(i) = 1$ gives $\lambda = \left(\sum_{i=1}^m \sqrt{\Omega_i p_g(i)}\right)^2$, which leads to Equation (42). Since $\nabla^2 J\left(p_g^*(i)\right) \geq 0$, $\tilde{p}_g^*(i)$ is the optimal static importance-scanning strategy that minimizes Y. ∎

Putting $\tilde{p}_g^*(i)$ into Equation (41), we obtain

$$\tilde{Y}_{min} = \left(\sum_{i=1}^{m} \sqrt{\Omega_i p_g(i)}\right)^2. \tag{44}$$

When $\Omega_1 = \Omega_2 = \cdots = \Omega_m = \frac{2^{32}}{m}$, $\tilde{p}_g^*(i) = \frac{\sqrt{p_g(i)}}{\sum_{k=1}^m \sqrt{p_g(k)}}$ and $\tilde{Y}_{min} = \frac{2^{32} \times \sum_{i=1}^m \sqrt{p_g(i)}^{\ 2}}{m}$. For example, using $p_g(i)$'s from the /8 subnet distribution of Witty-worm victims, $\tilde{Y}_{min} = 8.6 \times 10^8$, and using $p_g(i)$'s from the /8 subnet distribution of Web servers, $\tilde{Y}_{min} = 1.1 \times 10^9$. Thus, the /8 subnet distribution of Witty-worm victims is more vulnerable to a static importance-scanning worm than that of Web servers.

Optimal static importance scanning results in $n = \frac{1}{2}$ in Equation (37) and can be described as:

1. Before a worm is released, attackers first obtain the group distribution of vulnerable hosts $p_g(i)$ and then encode the group scanning distribution $p_g^*(i) = \frac{\sqrt{\Omega_i p_g(i)}}{\sum_{k=1}^m \sqrt{\Omega_k p_g(k)}}$ in the worm code.
2. At each time step t, the worm scans the group i with the probability $p_g^*(i)$.

This optimal static importance scanning can be exploited by a self-learning worm that is described in the next section.

4.4 A Self-Learning Worm Without the Group Distribution

We now assume that the knowledge of the group distribution is not available before a worm begins to spread. We focus on a self-learning worm that learns the distribution while propagating.

4.4.1 Algorithm

For practicality, we assume that learning takes place, using as few information exchanges between hosts as possible. Such a worm system is shown in Figure 7. A host with a high Internet bandwidth capacity, called the *worm server*, is responsible for collecting and processing information about the IP addresses of infected hosts. An infected host is called a *worm client* and may communicate with the worm server, but not with other infected hosts. If the communication uses Internet relay chat (IRC), this worm system forms a Botnet [18, 47].

The propagation process of this self-learning worm can be divided into two stages:

- **Learning stage:** Each infected host (worm client) performs random scanning or routable scanning [79, 70]. Once a vulnerable host is infected and becomes a new worm client, it reports its IP address to the worm server. The worm server records the clients' IP addresses in a list. When the worm server records a sufficient number of IP addresses, it estimates the group distribution of the vulnerable hosts ($p_g(i)$) based on collected data and sends the corresponding group scanning distribution ($p_g^*(i)$) to all worm clients on the list.

- **Importance-scanning stage:** Upon receiving $p_g^*(i)$, a worm client switches from either random scanning or routable scanning to static importance scanning using $p_g^*(i)$. The newly infected hosts at this stage do not need to communicate with the worm server, but perform static importance scanning directly.

This spreading algorithm of the self-learning worm is simple and effective, behaving in a similar way to the query process in the Napster peer-to-peer system [54].

4.4.2 Estimating the Group Distribution

The propagation speed of the self-learning worm strongly depends on how the worm server accurately estimates the group distribution of vulnerable hosts. Let L denote

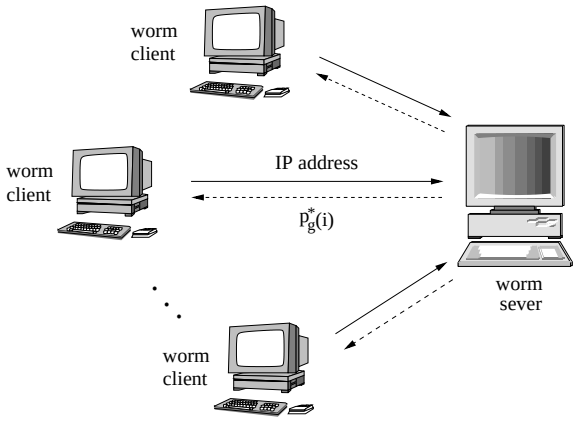


Figure 7: A self-learning worm system.

the number of measurements (clients' IP addresses) collected on the worm server. How large should L be for accurately estimating the group distribution? We answer this question by deriving the bias and the variance of an estimator.

Let L_i $(i = 1, 2, \cdots, m)$ denote the number of worm clients' IP addresses from group i among all L addresses. Then, a simple proportion estimator for group i's distribution is

$$\hat{p}_g(i) = \frac{L_i}{L}. \tag{45}$$

Let Z_j $(j = 1, 2, \cdots, L)$ denote the event that the jth worm client is in group i, i.e.,

$$Z_j = \begin{cases} 1, & \text{if the } j\text{th worm client is in group } i; \\ 0, & \text{otherwise.} \end{cases}$$

Thus, $\sum_{j=1}^{L} Z_j = L_i$. Since the worm uses random scanning or routable scanning in the learning stage of worm propagation, Z_j follows a Bernoulli distribution with parameter $p_g(i)$. Then, $E[Z_j] = p_g(i)$, and $Var[Z_j] = p_g(i)\,(1 - p_g(i))$. Thus,

$$E[\hat{p}_g(i)] = E\left[\frac{\sum_{j=1}^{L} Z_j}{L}\right] = \frac{1}{L}\sum_{j=1}^{L} E[Z_j] = p_g(i). \tag{46}$$

This means that the estimator is unbiased, which is desirable.

The variance of the estimator can now be calculated as follows. When $j \neq k$, $E[Z_j Z_k] = P(Z_j = 1, Z_k = 1) = P(Z_j = 1)P(Z_k = 1|Z_j = 1) = \frac{N_i}{N} \cdot \frac{N_i - 1}{N - 1}$, and

$E[Z_j]E[Z_k] = (p_g(i))^2 = (\frac{N_i}{N})^2$. Thus,

$$Cov[Z_j, Z_k] = E[Z_j Z_k] - E[Z_j]E[Z_k] = -p_g(i)\frac{1 - p_g(i)}{N - 1}, \tag{47}$$

which leads to

$$\begin{aligned} Var[\hat{p}_g(i)] &= Var\left[\frac{\sum_{j=1}^{L} Z_j}{L}\right] && (48)\\ &= \frac{\sum_{j=1}^{L} Var[Z_j] + 2\sum_{j<k} Cov[Z_j, Z_k]}{L^2} && (49)\\ &= \frac{1}{L} \cdot \frac{N - L}{N - 1} \cdot p_g(i)(1 - p_g(i)). && (50) \end{aligned}$$

The mean squared error (MSE) of the estimator is defined as in [5]:

$$MSE[\hat{p}_g(i)] = E\left[\sum_{i=1}^{m} (\hat{p}_g(i) - p_g(i))^2\right]. \tag{51}$$

That is,

$$MSE[\hat{p}_g(i)] = \sum_{i=1}^{m} Var[\hat{p}_g(i)] = \frac{1}{L} \cdot \frac{N - L}{N - 1} \cdot \left(1 - \sum_{i=1}^{m} p_g^2(i)\right). \tag{52}$$

Thus, the error of the estimator is the square root of $MSE[\hat{p}_g(i)]$. Note that $MSE[\hat{p}_g(i)]$ mainly depends on L and the vulnerable-host distribution. For example, if $L = 500$ and $N = 12,000$, then $MSE[\hat{p}_g(i)] = 0.00191$ for the /8 subnet distribution of Witty-worm victims, and $MSE[\hat{p}_g(i)] = 0.00194$ for the /8 subnet distribution of Web servers. Thus, even using a sample size of 500, a worm can estimate the group distributions with an error less than $\sqrt{2 \times 10^{-3}} = 4.5 \times 10^{-2}$ for these two cases. Meanwhile, as $MSE[\hat{p}_g(i)]$ is smaller for the Witty worm than for the Web-attacking worm, the proportion estimator works slightly better in the case of the Witty worm.

Since $\sum_{i=1}^{m} p_g^2(i) \cdot \sum_{i=1}^{m} 1^2 \geq (\sum_{i=1}^{m} p_g(i))^2$ by the Cauchy-Schwarz inequality, $\sum_{i=1}^{m} p_g^2(i) \geq \frac{1}{m}$. Therefore,

$$MSE[\hat{p}_g(i)] \leq \frac{1}{L} \cdot \frac{N - L}{N - 1} \cdot \frac{m - 1}{m} \leq \frac{1}{L}. \tag{53}$$

This means that we can choose the number of samples L to achieve a desired accuracy of the estimation. For example, if $L = 13,866$, which is the number of Web servers collected from UROULETTE, we have $MSE[\hat{p}_g(i)] < 10^{-4}$.

4.4.3 Final Size of Infection

The proportion estimator given in Equation (45) may miss some groups that contain vulnerable hosts. That is, for some i, the actual distribution $p_g(i) > 0$, but its estimator $\hat{p}_g(i) = 0$. If this happens, the self-learning worm based on such a estimator may never scans some groups that contain vulnerable hosts, resulting in a fewer number of infected hosts than the total population of vulnerable machines. Let the total number of hosts infected by the self-learning worm be the *final size of infection* N_f. We attempt to answer the question: What is the percentage of vulnerable hosts that would be missed by a self-learning worm, given L IP addresses of worm clients collected on the worm server? That is, we want to compute $p_l = \frac{N-N_f}{N}$, given L.

Let X_i $(i = 1, 2, \cdots, m)$ denote the event that the group i is **not** scanned by the self-learning worm, i.e.,

$$X_i = \begin{cases} 1, & \text{if the group } i \text{ is not scanned by the worm;} \\ 0, & \text{otherwise.} \end{cases}$$

Since the worm uses random scanning or routable scanning in the learning stage of worm propagation, $L_1, L_2, \cdots, L_m$ form a multinormial distribution, and X_i follows a Bernoulli distribution with parameter $(1-p_g(i))^L$. Then, $E[X_i] = (1-p_g(i))^L$, and $Var[X_i] = (1-p_g(i))^L \left[1 - (1-p_g(i))^L\right]$. Moreover, when $j \neq k$, $E[X_j X_k] = P(X_j = 1, X_k = 1) = (1 - p_g(j) - p_g(k))^L$. Thus, $Cov[X_j, X_k] = E[X_j X_k] - E[X_j]E[X_k] = (1 - p_g(j) - p_g(k))^L - (1 - p_g(j))^L (1 - p_g(k))^L$.

The percentage of vulnerable hosts missed by the worm is

$$p_l = \sum_{i=1}^{m} p_g(i) X_i. \tag{54}$$

Therefore, the expected value of p_l is

$$E[p_l] = \sum_{i=1}^{m} p_g(i) E[X_i] = \sum_{i=1}^{m} p_g(i)(1 - p_g(i))^L, \tag{55}$$

and the variance of p_l is

$$Var[p_l] = \sum_{i=1}^{m} p_g^2(i) Var[X_i] + 2 \sum_{j<k} p_g(j) p_g(k) Cov[X_j, X_k]. \quad (56)$$

Both $E[p_l]$ and $Var[p_l]$ only depend on the group distribution $p_g(i)$ and sample size L. For example, if $L = 500$, $E[p_l] = 0.0279$ and $Var[p_l] = 4.3080 \times 10^{-5}$ for the /8 subnet distribution of Witty-worm victims, and $E[p_l] = 0.0334$ and $Var[p_l] = 6.7692 \times 10^{-5}$ for the /8 subnet distribution of Web servers. Thus, even using a sample size of 500, a self-learning worm only misses about 3% vulnerable hosts in these two cases. Moreover, the worm misses a fewer vulnerable hosts for the Witty worm than for the Web-server attacking worm.

4.5 Performance Evaluation

In this section, we evaluate the performance of self-learning worms empirically, where the performance refers to the propagation speed of worms as described in Section 4.1. First, we introduce the simulation set-up. We then show the optimality of our proposed static importance scanning, which is used in the importance-scanning stage of self-learning worms. We also demonstrate that a self-learning worm can learn the underlying group distribution in /8 subnets, using as few as 500 samples. Finally, we compare a self-learning worm with a random-scanning worm, a permutation-scanning worm, and a Class-A routing worm.

4.5.1 Simulation Set-up

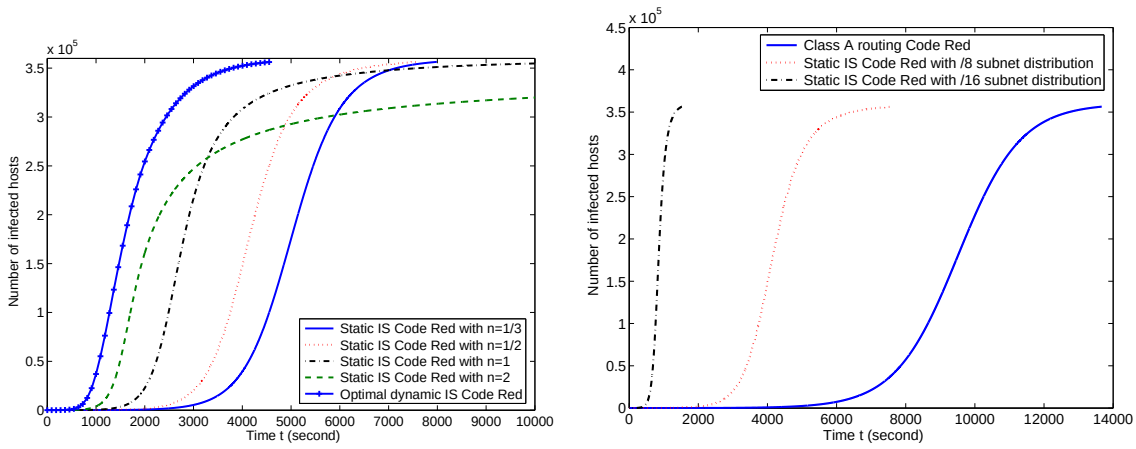
In our simulation, we employ the model in Equation (2) to study the spread of random-scanning and Class-A routing worms. Meanwhile, we use the model in Equations (24), (29), and (37) to imitate the propagation of dynamic and static importance-scanning worms, assuming that the group distribution $p_g(i)$ is given. For a self-learning worm, we adopt the model in Equation (2) to simulate the spread in

the learning stage and the model in Equations (24) and (37) to emulate the propagation in the importance-scanning stage. There in Equation (37), $p_g(i)$ changes to $\hat{p}_g(i)$, and $n = \frac{1}{2}$. To mimic the effect of the self-learning, we simulate 1000 runs for the importance-scanning stage with estimated group distribution $\hat{p}_g(i)$. To generate $\hat{p}_g(i)$, we write a random-scanning worm propagation simulator. When L hosts are infected, the number of infected hosts in each group (e.g., /8 subnet) is counted, and the proportion estimator is performed using Equation (45).

The simulated worms have parameters comparable to those of Code Red v2 and Witty worms. The Code Red v2 worm has a vulnerable population $N = 360,000$, a scanning rate $s = 358$ per minute, and a hitlist size $I_0 = 10$ [39, 77]. The Witty worm has a vulnerable population $N = 12,000$, a scanning rate $s = 1200$ per second, and a hitlist size $I_0 = 110$ [56, 76]. We ignore the effect of disk damage on the spread in the case of the Witty worm. We also assume that the victims of the Code Red v2 worm have the same distribution as the Web servers. Since the experimental results of the Witty worm are similar to those of Code Red v2, here we mainly present the observations for the Code Red v2 worm.

4.5.2 Static Important-Scanning Strategies

We first examine the propagation speed of static importance-scanning strategies. Figure 8(a) compares different static importance-scanning (IS) Code Red worms ($n = \frac{1}{3}, \frac{1}{2}, 1, 2$) as well as the optimal dynamic IS Code Red worm with the /8 subnet distribution. As expected, when $n = \frac{1}{2}$, static IS infects 99% vulnerable hosts in the shortest time duration among all static strategies. Therefore, we choose $n = \frac{1}{2}$ for a self-learning worm in the importance-scanning stage. One interesting observation is that if a static strategy (e.g. $n = 2$) spreads faster at the early stage, it will propagate slower at the late stage; or vise verse (e.g. $n = \frac{1}{3}$). This is because a static IS uses the same group scanning distribution all the time. A larger n corresponds to an IS



(a) Different choices of n (b) Different amount of knowledge

Figure 8: Comparison of static importance-scanning (IS) strategies.

worm that preferentially scans the groups containing more vulnerable hosts at the early stage, but unfavorably probes the groups having more left vulnerable hosts at the late stage. Therefore, attackers may choose a corresponding static IS strategy based on the purpose of attacks, e.g. infecting some amount of hosts as quickly as possible.

We also inspect the effect of the amount of knowledge on the spreading speed of static importance-scanning worms. Figure 8(b) shows the propagation comparison among a Class-A routing Code Red worm, a static IS Code Red worm with the /8 subnet distribution, and a static IS Code Red worm with the /16 subnet distribution. Here, static IS Code Red worms use the optimal strategy, i.e. $n = \frac{1}{2}$. We assume that all three worms have the same scanning rate, although an IS worm may slow down for a larger payload if it contains more information about the group distribution. The Class-A routing worm needs 227 minutes to infect 99% vulnerable hosts, while the static IS worm with the /8 subnet distribution and the /16 subnet distribution can use only 126 minutes and 25 minutes, respectively. A Class-A routing worm can be regarded as a worm that only has the knowledge about the routable space in /8 subnets. Therefore, more knowledge about the vulnerable-host distribution may help

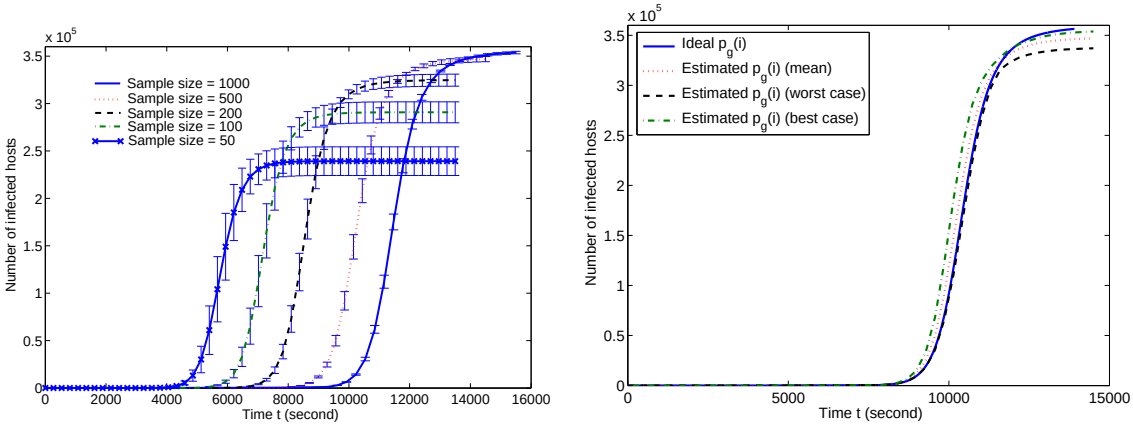
an attacker in designing a faster worm.

4.5.3 Sample Size

Next, we study the effect of the sample size (L) on the spread of a self-learning worm. Figure 9(a) compares different sample sizes ($L = 50, 100, 200, 500, 1000$). These self-learning worms employ random scanning to estimate the /8 subnet distribution in the learning stage and use the optimal static importance-scanning strategy (i.e. $n = \frac{1}{2}$) in the importance-scanning stage. In this figure, a curve expresses the average of experimental results over 1000 runs, while an error-bar represents the standard deviation of experimental results based on 1000 runs. If a self-learning worm uses fewer samples, it can usually spread faster at the early stage, but propagate with a larger variation and infect fewer vulnerable hosts at the late stage. This is because both the MSE in Equation (52) and the expected percentage of vulnerable hosts missed $E[p_l]$ in Equation (55) increase, when the sample size L decreases. We further plot an example of a self-learning worm with 500 samples in Figure 9(b). This figure shows the worst case and the best case among 1000 runs for worm propagation, as well as the average case and the propagation with the actual group distribution in the importance-scanning stage. It is observed that all four cases close to each other. Therefore, even with 500 samples, a self-learning worm can estimate the group distribution accurately.

4.5.4 Self-Learning Worms

Now, we are ready to compare a self-learning worm with a random-scanning worm, a permutation-scanning worm, and a Class-A routing worm. A self-learning worm uses either random scanning or routable scanning and collects 500 samples in the learning stage, and employs the optimal static importance-scanning strategy $p_g^*(i) = \frac{\sqrt{\Omega_i \hat{p}_g(i)}}{\sum_{k=1}^{m} \sqrt{\Omega_k \hat{p}_g(k)}}$ in the importance-scanning stage. In permutation scanning, all worms share the common pseudo random permutation of the IP address space and coordinate

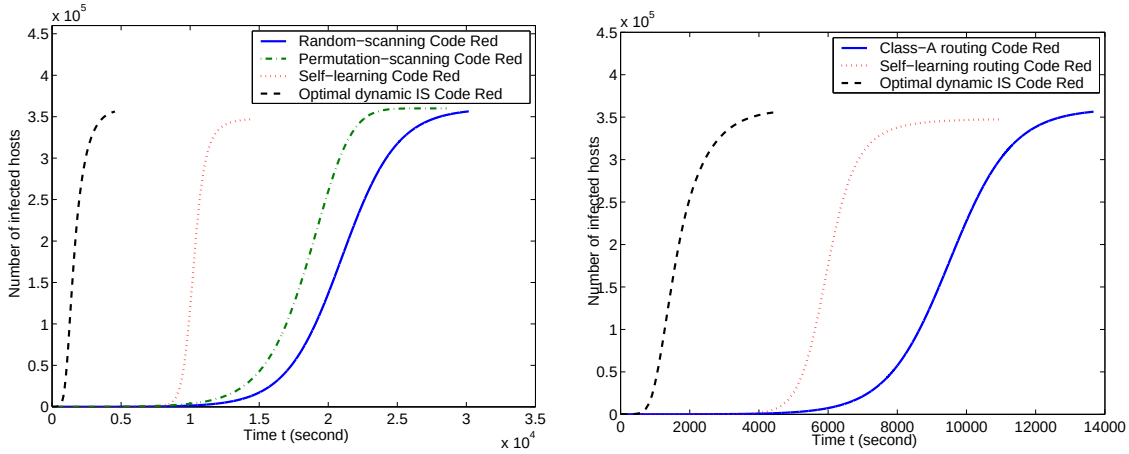


(a) Comparison of sample sizes

(b) A sample size of 500

Figure 9: Effect of sample size.

to provide comprehensive scanning [68]. Such permutation scanning is implemented by Weaver's simulator, which uses the 32-bit, 6-round variant of RC5 to generate all permutations and random numbers. Figure 10(a) shows the propagation comparison among a self-learning Code Red worm (average over 1000 runs), a permutation-scanning Code Red worm, and a random-scanning Code Red worm. It is observed that the self-learning Code Red worm spends about 50% of the spreading time to infect the first 500 hosts, but uses the left 50% of time to infect the other 3.5×10^5 hosts. Thus, the self-learning worm has an astounding spreading speed at the importance-scanning stage. Figure 10(b) demonstrates the spread of another self-learning Code Red worm if the worm uses the Class-A routable scanning in the learning stage. After collecting the information of 500 worm clients, the self-learning worms (exploiting random scanning or routable scanning in the learning stage) use only 64 minutes to infect the other 3.2×10^5 (90%) vulnerable hosts in the importance-scanning stage. In comparison, a random-scanning Code Red worm, a permutation-scanning Code Red worm, and a Class-A routing Code Red worm need 293 minutes, 254 minutes, and 133 minutes, respectively, to infect the same number of vulnerable hosts. Hence, a simple self-learning process can greatly increase worms' spreading speed.



(a) A self-learning Code Red.

(b) A self-learning routing Code Red.

Figure 10: Performance of self-learning Code Red worms.

4.6 Detecting and Defending Against Self-Learning Worms

How can we detect and defend against self-learning worms? Our study on self-learning worms provides the following guidelines:

- When a new application is introduced to the future Internet, how can we deploy this application? From Equation (41), attackers attempt to minimize Y by choosing the optimal static group scanning distribution $p_g^*(i)$, while defenders endeavor to maximize Y by customizing the group distribution $p_g(i)$. This is a classic two-person zero-sum game [45] between the attackers and the defenders, which leads to

$$Y_{opt} = \min_{p_g^*(i)} \max_{p_g(i)} \{Y\} = \max_{p_g(i)} \min_{p_g^*(i)} \{Y\}. \tag{57}$$

  Similar to the derivation in the previous chapter, we find that the optimal strategy for the defenders is to deploy a new application uniformly in the Internet for any grouping criteria, such as /8 subnets, /16 subnets, and DNS top-level domains [26]. Thus, the self-learning process cannot help the worm in speeding up the propagation. It is a common belief that IPv6 can slow down the spread of scanning worms effectively due to the large address space. An importance-scanning worm, however, can have an astonishing spreading speed, if vulnerable

hosts are still distributed in a non-uniform fashion and the group distribution can be obtained. On the other hand, current traffic engineering requires the non-uniform partition of the address space for routing aggregation. How to balance the tradeoff between traffic engineering and security engineering is a challenging task for designing the future Internet.

- Since a self-learning worm has an astounding spreading speed at the importance-scanning stage, defenders need to detect the worm during the learning stage of worm propagation. Scan or probe detection can be combined with content-based anomaly detection to improve the speed and the accuracy of detection. Moreover, a good detection system should be distributed as proposed in [49]. Interestingly, the effectiveness of this worm monitoring system [49] strongly depends on obtaining the information of the underlying vulnerable-host group distribution in /8 subnets and /16 subnets. Thus, the weapon race between the attackers and the defenders relies on how each side can collect and process the information of the vulnerable-host distribution. The cooperation between the defenders from different domains provides information sharing and therefore a possibly more effective detection system [37].

- For the self-learning worm system proposed in this chapter, a key issue in defense is to detect and disable the worm server before the importance-scanning stage. One possible method to detect the worm server, for example, is to use the host contact graph presented in [71]. After detecting the worm server, different mechanisms can be applied to disable the worm server, for example, putting the IP address of the worm server in the *address blacklist* [40]; providing the false information of worm clients to the worm server; or even performing the denial of service (DoS) attack on the worm server.

4.7 Summary

We have characterized attack behaviors through both analysis and simulation. Our designed "self-learning worm" has the intelligence to gather and process the measurements while propagating and thus increases the propagation speed. Our findings include

- A worm can learn the group distribution in /8 subnets well, using a proportion estimator and as few as 500 samples. The estimator is unbiased, and the MSE of the estimator approximately decreases in reverse proportion to the number of measurements.

- An optimal yet practical importance-scanning method can be derived based on static importance sampling to speed up the propagation of a worm.

- A self-learning worm based on parameters chosen from real measurements of the Code Red v2 worm spreads nearly five times faster than a random-scanning worm, four times faster than a permutation-scanning worm, and two times faster than a Class-A routing worm, after collecting 500 samples and estimating the group distribution in /8 subnets.

CHAPTER V

SUB-OPTIMAL WORM-SCANNING METHOD: A LOCALIZED-SCANNING WORM

5.1 Introduction

In this chapter, our focus is on localized scanning, which has been used by such famous worms as Code Red II and Nimda. *Localized scanning* preferentially searches for vulnerable hosts in the "local" address space. For example, the Code Red II worm selects target IP addresses as follows [82]:

- 50% of the time, an address with the same first byte is chosen as the target,
- 37.5% of the time, an address with the same first two bytes is chosen as the target,
- 12.5% of the time, a random address is chosen.

Song et al. showed that Nimda and Code Red II worms accounted for 90% infection attempts in the seven-week period from September 19 to November 3, 2001 [58]. Why is such a localized strategy so effective? It has been observed that in the current Internet, a sub-network intends to have many computers with the same operating systems and applications for easy management [49]. Hence, vulnerable hosts usually form clusters [8]. Once a vulnerable host in such a subnet is infected, a localized-scanning worm can rapidly compromise all the other local vulnerable hosts.

The goal of this work is to better understand the spreading ability and characteristics of localized-scanning worms. Specifically, we attempt to answer the following questions:

- What is the effect of vulnerable-host distributions on the spread of localized-scanning worms? The prior work has studied this effect empirically [8, 81, 49].

In this work, we use mathematical reasoning to show the relationships between vulnerable-host distributions and localized-scanning worms. Specifically, it is shown analytically that localized-scanning worms spread slower than random-scanning worms if vulnerable hosts are uniformly distributed, or faster if highly unevenly distributed. Moreover, if infected hosts are uniformly distributed, localized-scanning worms can speed up the propagation with nearly a rate of the non-uniformity factor that quantifies the non-uniformity of a vulnerable-host distribution [11].

- What is the propagation capacity of a localized-scanning worm? We design an optimal localized-scanning strategy that maximizes the localized-scanning worm propagation speed. Such a strategy dynamically adapts the parameters used for scanning the local sub-network and the global Internet, based on the distribution of uninfected vulnerable hosts. Although the optimal localized scanning is difficult to implement, it provides an upper bound on the spreading speeds of the currently used localized scanning and its variants. Moreover, we empirically show that the propagation speed of the currently used localized scanning can approach that of the optimal strategy.

- What are some possible variants of localized-scanning worms? We study three variants of localized scanning that can be easily implemented. The first one makes an infected host focus on scanning either locally or globally. Such a variant, however, is shown empirically to spread slower and have a larger variance than localized scanning. Therefore, it may not be a good candidate for worm attacks. The second variant is inspired by the optimal localized scanning. Specifically, an infected host initiates to scan the local sub-network and switches to scanning the global Internet when it probes a local host that has been already infected. Such a strategy makes an infected host adapt scanning

strategies dynamically, based on the feedback from the probed host. We show that this simple variant can spread faster than localized scanning and has a smaller variance. Therefore, this scanning method is a potential tool for attackers. The second variant is easily extended to a "ping-pong" algorithm, which further improves the worm spreading speed at the late stage.

The remainder of this chapter is structured as follows. Section 5.2 provides the background on localized scanning and vulnerable-host distributions. Section 5.3 shows the effect of vulnerable-host distributions on localized scanning analytically. Sections 5.4 and 5.5 design the optimum and the variants of localized scanning. Section 5.6 concludes the chapter.

5.2 Preliminaries

5.2.1 Localized Scanning

Localized scanning preferentially scans for targets in the address space that is close to the victim. The basic idea of such a scanning method is that if vulnerable hosts are clustered, an infected host searching for local hosts would have a higher probability to find a target than random guessing. Localized scanning has been exploited by Code Red II and Nimda worms [82, 83]. Moreover, the Blaster worm also uses localized scanning to select its starting point [85]. The successes of these worms indicate the effectiveness of such a simple scanning strategy.

In this work, we consider two types of localized scanning (LS). The first type is a simplified version of LS, called */l LS*, which scans the Internet as follows:

- p_a $(0 \leq p_a \leq 1)$ of the time, an address with the same first l bits is chosen as the target,
- $1 - p_a$ of the time, a random address is chosen.

When $p_a = 0$, $/l$ LS is identical to random scanning (RS). Here, we use the classless inter-domain routing (CIDR) notation. The IPv4 address space is partitioned into

subnets according to the first l bits of IP addresses, i.e., $/l$ prefixes or $/l$ subnets, where $l \in \{0, 1, , \cdots, 32\}$. Thus, each $/l$ subnet i ($i = 1, 2, \cdots, 2^l$) has 2^{32-l} addresses.

The second type is called *two-level LS* (2LLS), which has been used by the Code Red II and Nimda worms. 2LLS scans the Internet as follows:

- p_b ($0 \leq p_b \leq 1$) of the time, an address with the same first byte is chosen as the target,
- p_c ($0 \leq p_c \leq 1 - p_b$) of the time, an address with the same first two bytes is chosen as the target,
- $1 - p_b - p_c$ of the time, a random address is chosen.

For example, for the Code Red II worm, $p_b = 0.5$ and $p_c = 0.375$ [82]; for the Nimda worm, $p_b = 0.25$ and $p_c = 0.5$ [83].

5.2.2 Vulnerable-Host Distribution

The prerequisite for localized scanning is that vulnerable hosts are non-uniformly distributed in the Internet. The non-uniformity of vulnerable-host distributions has been observed in prior work [3, 39, 41, 56, 49, 10]. Taking the distribution of Witty-worm victims among /16 subnets as an example, we process the data provided by CAIDA [92] as follows. First, the /16 subnets are sorted decreasingly according to the number of vulnerable hosts. Then, the empirical cumulative distribution function (CDF) of the percentage of vulnerable hosts in the sorted /16 subnets is computed and plotted in Figure 11. We find that 1,573 (2.4%) /16 subnets contain 80% vulnerable hosts, whereas 2,453 (3.7%) /16 subnets hold 90% vulnerable hosts. Therefore, only a small percentage of /16 subnets contain a large portion of vulnerable hosts, and the distribution of Witty-worm victims is highly non-uniform.

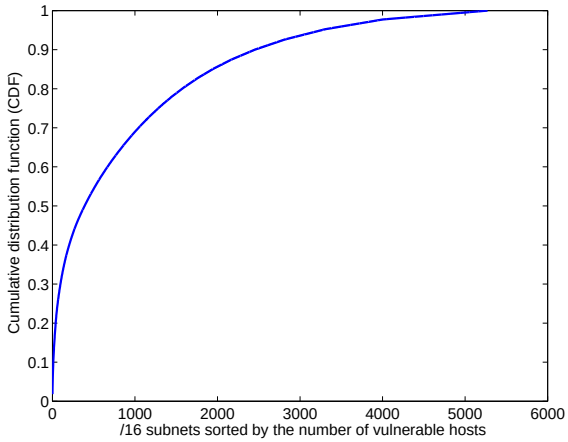


Figure 11: CDF of the percentage of Witty-worm victims in sorted /16 subnets.

5.2.3 Non-Uniformity Factor

How can we quantify the non-uniformity of a vulnerable-host distribution? In our prior work [11], we use a simple metric, called the *non-uniformity factor*, to measure the non-uniformity of a distribution.

Let N be the total number of vulnerable hosts and $N_i^{(l)}$ be the number of vulnerable hosts in $/l$ subnet i ($i = 1, 2, \cdots, 2^l$). Define $p_g^{(l)}(i) = \frac{N_i^{(l)}}{N}$, which is called the group distribution in $/l$ subnets. Then, the non-uniformity factor in $/l$ subnets is defined as

$$\beta^{(l)} = 2^l \sum_{i=1}^{2^l} \left(p_g^{(l)}(i)\right)^2. \tag{58}$$

A larger non-uniformity factor indicates a more non-uniform distribution. When a vulnerable-host distribution is uniform among $/l$ subnets, $\beta^{(l)} = 1$. For the Witty-worm victim distribution, $\beta^{(8)} = 12.0$ and $\beta^{(16)} = 126.7$. We will further discuss the non-uniformity factor in the next chapter.

5.3 Effect of Vulnerable-Host Distributions on Localized Scanning

In this section, we study the effect of vulnerable-host distributions on localized scanning and compare the spreading dynamics of localized-scanning (LS) worms with

those of random-scanning (RS) worms by modeling their propagation. As a dynamic worm-propagation model is non-linear, it is difficult to result in a close-form solution. Hence, we gain some insights through exploring extreme cases of vulnerable-host and infected-host distributions among subnets. Specifically, we consider three extreme cases: (1) Vulnerable hosts are evenly distributed, (2) Vulnerable hosts are highly unevenly distributed, and (3) Infected hosts are uniformly distributed.

A simple abstract model, known as the *susceptible* $\rightarrow$ *infected* (SI) model, has been exploited to model the spread of worms in various earlier work [59, 79]. The SI model assumes that each host has two states: susceptible and infected. Once infected, a host stays in the infected state. Here, we adopt a discrete-time SI model. In particular, we employ the analytical active worm propagation (AAWP) model, which was proposed by Chen et al. in [8] and has been applied in [49, 30, 70].

5.3.1 Random Scanning

In the AAWP model, the spread of RS worms is characterized as follows [10]:

$$I_{t+1} = I_t + (N - I_t)\left[1 - \left(1 - \frac{1}{\Omega}\right)^{I_t s}\right] \quad (59)$$

$$= I_t + (N - I_t)\frac{I_t s}{\Omega} - O\left(\frac{1}{\Omega^2}\right), \quad (60)$$

where I_t is the average number of infected hosts at time t ($t \geq 0$); N is the total number of vulnerable hosts; s is the scanning rate; and Ω is the scanning space. Since $\Omega = 2^{32} >> 1$, we ignore $O\left(\frac{1}{\Omega^2}\right)$ and have

$$I_{t+1} = I_t + \frac{(N - I_t)I_t s}{\Omega}. \quad (61)$$

5.3.2 /l Localized Scanning

The AAWP model can be extended to model the spread of /l LS worms:

$$I_{t+1,i}^{(l)} = I_{t,i}^{(l)} + (N_i^{(l)} - I_{t,i}^{(l)})\left[1 - \left(1 - \frac{1}{\Omega_i}\right)^{S_{t,i}}\right] \quad (62)$$

$$= I_{t,i}^{(l)} + (N_i^{(l)} - I_{t,i}^{(l)})\frac{S_{t,i}}{\Omega_i} - O\left(\frac{1}{\Omega_i^2}\right), \quad (63)$$

where $i = 1, 2, \cdots, 2^l$; $I_{t,i}^{(l)}$ is the expected number of infected hosts in $/l$ subnet i at time t ($t \geq 0$); $N_i^{(l)}$ is the number of vulnerable hosts in $/l$ subnet i; Ω_i is the size of the address space in $/l$ subnet i; and $S_{t,i}$ is the average number of scans hitting $/l$ subnet i during time period $(t, t+1]$. Since $\Omega_i = 2^{32-l} >> 1$, we ignore $O\left(\frac{1}{\Omega_i^2}\right)$. Summing over $i = 1, 2, \cdots, 2^l$ on both sides of Equation (63), we have

$$I_{t+1} = I_t + \sum_{i=1}^{2^l} \frac{\left(N_i^{(l)} - I_{t,i}^{(l)}\right) S_{t,i}}{2^{32-l}}, \tag{64}$$

where $I_t = \sum_{i=1}^{2^l} I_{t,i}^{(l)}$.

The average number of scans that fall into $/l$ subnet i during the time period $(t, t+1]$ (i.e., $S_{t,i}$) consists of two parts: (a) $p_a I_{t,i}^{(l)} s$ scans from local infected hosts within subnet i and (b) $\frac{(1-p_a)I_t s}{2^l}$ scans from all infected hosts. That is,

$$S_{t,i} = \left(p_a I_{t,i}^{(l)} + \frac{1-p_a}{2^l} I_t\right) s, \qquad i = 1, 2, \cdots, 2^l. \tag{65}$$

Putting Equation (65) into Equation (64), we have

$$I_{t+1} = I_t + (1-p_a)\frac{(N-I_t)I_t s}{\Omega} + p_a \frac{s}{\Omega} \cdot 2^l \sum_{i=1}^{2^l} \left(N_i^{(l)} - I_{t,i}^{(l)}\right) I_{t,i}^{(l)}. \tag{66}$$

On the right-hand side of the above equation, the second term represents the random-scanning component in the $/l$ LS, while the third term corresponds to the preference of scanning the local $/l$ subnet. If $\frac{(N-I_t)I_t s}{\Omega} \leq \frac{s}{\Omega} \cdot 2^l \sum_{i=1}^{2^l} \left(N_i^{(l)} - I_{t,i}^{(l)}\right) I_{t,i}^{(l)}$, a $/l$ LS worm should choose a large value of p_a to speed up the propagation.

As a close-form expression for I_t is difficult to obtain, we consider three extreme cases of vulnerable-host and infected-host distributions. The first case assumes that vulnerable hosts are uniformly distributed, i.e., $N_1^{(l)} = N_2^{(l)} = \cdots = N_{2^l}^{(l)}$. Then, when $I_{t,i}^{(l)} > I_{t,j}^{(l)}$, $N_i^{(l)} - I_{t,i}^{(l)} < N_j^{(l)} - I_{t,j}^{(l)}$, $i, j \in \{1, 2, \cdots, 2^l\}$. This results in

$$2^l \sum_{i=1}^{2^l} \left(N_i^{(l)} - I_{t,i}^{(l)}\right) I_{t,i}^{(l)} < \left[\sum_{i=1}^{2^l} \left(N_i^{(l)} - I_{t,i}^{(l)}\right)\right] \left(\sum_{i=1}^{2^l} I_{t,i}^{(l)}\right) = (N - I_t) I_t, \tag{67}$$

assuming that the numbers of infected hosts among subnets are not all equal. The above relation is obtained by the Chebyshev sum inequality [69] or the rearrangement

inequality [90]. When applying the result of Equation (67) to Equation (66), we obtain that $I_{t+1} < I_t + \frac{(N-I_t)I_t s}{\Omega}$. Therefore, the uniform distribution of vulnerable hosts leads to a low value of p_a for an effective $/l$ LS worm. Moreover, the spread of $/l$ LS worms is slower than that of RS worms in this case.

The second case assumes that vulnerable hosts are highly unevenly distributed so that when a $/l$ subnet has more infected hosts, it would also contain more uninfected vulnerable hosts. That is, when $I_{t,i}^{(l)} > I_{t,j}^{(l)}$, $N_i^{(l)} - I_{t,i}^{(l)} > N_j^{(l)} - I_{t,j}^{(l)}$, $i, j \in \{1, 2, \cdots, 2^l\}$. We can then derive

$$2^l \sum_{i=1}^{2^l} \left(N_i^{(l)} - I_{t,i}^{(l)} \right) I_{t,i}^{(l)} > (N - I_t) I_t, \tag{68}$$

assuming that the numbers of infected hosts among subnets are not all equal. The above relation is obtained by the Chebyshev sum inequality. When applying the result of Equation (68) to Equation (66), we obtain that $I_{t+1} > I_t + \frac{(N-I_t)I_t s}{\Omega}$. Therefore, for such an extreme case, a large value of p_a is preferred for an effective $/l$ LS worm. Moreover, the spread of $/l$ LS worms is faster than that of RS worms.

The last case assumes a uniform distribution of infected hosts among subnets. That is, the number of infected hosts in $/l$ subnet i is proportional to the number of vulnerable hosts in this subnet, i.e., $I_{t,i}^{(l)} = I_t \cdot p_g^{(l)}(i)$, $i = 1, 2, \cdots, 2^l$. This assumption changes Equation (66) to

$$I_{t+1} = I_t + \left(1 - p_a + p_a \beta^{(l)}\right) \frac{(N - I_t) I_t s}{\Omega}, \tag{69}$$

where $\beta^{(l)}$ is the non-uniformity factor as defined in Equation (58). Thus, compared with RS (Equation (61)), $/l$ LS can increase the propagation speed with a rate of $1 - p_a + p_a \beta^{(l)}$. For example, when $p_a = 0.75$, a /8 LS Witty worm can increase the spreading speed with a factor of 9.25, whereas a /16 LS Witty worm can increase the speed with a factor of 95.28.

5.3.3 Two-Level Localized Scanning

For 2LLS, Equation (64) still holds when $l = 16$. The average number of scans hitting /16 subnet i during time period $(t, t+1]$ is

$$S_{t,i} = \left(p_c I_{t,i}^{(16)} + \frac{p_b}{2^8} \sum_{j \in A_i^{(8)}} I_{t,j}^{(16)} + \frac{1 - p_b - p_c}{2^{16}} I_t \right) s, \tag{70}$$

where $i = 1, 2, \cdots, 2^{16}$; $A_i^{(8)}$ denotes the set of /16 subnets that have the same first byte of the subnet address as /16 subnet i; and $\sum_{j \in A_i^{(8)}} I_{t,j}^{(16)}$ represents the expected number of the infected hosts in the Class-A subnet that has the same first byte of the address as the /16 subnet i. Putting Equation (70) into Equation (64) and setting $l = 16$, we have,

$$\begin{aligned} I_{t+1} &= I_t + (1 - p_b - p_c) \frac{(N - I_t) I_t s}{\Omega} + \frac{2^8 p_b s}{\Omega} \sum_{i=1}^{2^8} \left(N_i^{(8)} - I_{t,i}^{(8)} \right) I_{t,i}^{(8)} \\ &\quad + \frac{2^{16} p_c s}{\Omega} \sum_{i=1}^{2^{16}} \left(N_i^{(16)} - I_{t,i}^{(16)} \right) I_{t,i}^{(16)}. \end{aligned} \tag{71}$$

Similar to /l LS, 2LLS can be shown to spread slower (or faster) than RS if vulnerable hosts are uniformly (or highly unevenly) distributed[1]. Moreover, if infected hosts are uniformly distributed, the model for the 2LLS (i.e., Equation (71)) becomes

$$I_{t+1} = I_t + \left(1 - p_b - p_c + p_b \beta^{(8)} + p_c \beta^{(16)}\right) \cdot \frac{(N - I_t) I_t s}{\Omega}. \tag{72}$$

Comparing Equations (61) with (72), we find that when p_c is large and the uniformity condition of infected hosts holds, a 2LLS worm can speed up the propagation nearly $\beta^{(16)}$ times compared with an RS worm.

Our findings provide quantifications to some of the previous observations [8, 81, 49]. For example, when vulnerable hosts are uniformly distributed, an LS worm propagates slower than an RS worm [8]. On the other hand, when the underlying vulnerable-host distribution follows nearly a power law, an LS worm can spread much faster than an RS worm [49].

[1] We omit the details of derivation for brevity.

5.4 Optimal Dynamic Localized Scanning

What is the "best-case scenario" for LS worms? How different is the currently used LS from the optimal LS? To answer these questions, we study the optimal LS, focusing on /l LS for simplicity. The essential of the optimal LS is to choose the best parameters (i.e., p_a, p_b, and p_c) to maximize the propagation speed. Intuitively, the optimal LS should be dynamic and adjust its parameters during the scanning process. Hence, these parameters depend on the location of infected hosts and vary with time. We use $p_{t,i}^{(a)}$ to denote p_a at time t for an infected host in /l subnet i.

5.4.1 Optimal /l Localized Scanning

The optimal /l LS should determine $p_{t,i}^{(a)}$ $(0 \leq p_{t,i}^{(a)} \leq 1)$ to maximize the probability of finding an uninfected vulnerable host. To obtain this, we assume that the number of vulnerable hosts and the number of infected hosts in each subnet at time t (i.e., $N_i^{(l)}$'s and $I_{t,i}^{(l)}$'s) are known to the worm. Therefore, our problem reduces to obtaining the optimal $p_{t,i}^{(a)}$'s for worm propagation, given $N_i^{(l)}$'s and $I_{t,i}^{(l)}$'s.

For the dynamic /l LS, the average number of scans that fall into /l subnet i during time period $(t, t+1]$ (i.e., $S_{t,i}$) consists of two parts: (a) $p_{t,i}^{(a)} I_{t,i}^{(l)} s$ scans from local infected hosts within subnet i and (b) $\frac{1}{2^l}\sum_{j=1}^{2^l}(1-p_{t,j}^{(a)})I_{t,j}^{(l)}s$ scans from all infected hosts. That is,

$$S_{t,i} = \left[p_{t,i}^{(a)} I_{t,i}^{(l)} + \frac{\sum_{j=1}^{2^l}(1-p_{t,j}^{(a)})I_{t,j}^{(l)}}{2^l} \right] s, \tag{73}$$

where $i = 1, 2, \cdots, 2^l$. Putting Equation (73) into Equation (64), we have

$$I_{t+1} = I_t + \frac{s}{2^{32-l}} \sum_{i=1}^{2^l} I_{t,i}^{(l)} \left[p_{t,i}^{(a)}(N_i^{(l)} - I_{t,i}^{(l)}) + (1-p_{t,i}^{(a)})\frac{N-I_t}{2^l} \right]. \tag{74}$$

To maximize I_{t+1}, $p_{t,i}^{(a)}$ needs to satisfy

$$p_{t,i}^{(a)} = \begin{cases} 1, & \text{if } N_i^{(l)} - I_{t,i}^{(l)} > \frac{N-I_t}{2^l}; \\ 0, & \text{otherwise.} \end{cases} \tag{75}$$

That is, if the number of uninfected vulnerable hosts in subnet i is larger than the average number of uninfected vulnerable hosts among 2^l subnets at time t, the infected hosts in sunbet i should scan only the local subnet; otherwise, the infected hosts should use random scanning. Thus, the propagation model for the optimal dynamic $/l$ LS is

$$I_{t+1} = I_t + \frac{s}{2^{32-l}} \sum_{i=1}^{2^l} I_{t,i}^{(l)} \max \left\{ N_i^{(l)} - I_{t,i}^{(l)}, \frac{N - I_t}{2^l} \right\}. \tag{76}$$

Using this optimal scanning method, a worm starting from a subnet that contains many vulnerable hosts would first scan locally. The infected hosts in this subnet then switch from scanning locally to scanning globally later when few uninfected vulnerable hosts remain. The key is that the worm switches the scanning strategy when it is aware of the change of the distribution of uninfected vulnerable hosts.

It should be noted that implementing such optimal LS is difficult. First, $N_i^{(l)}$'s may not be known in advance. Second, to perform this LS, each infected host needs to know $I_{t,i}^{(l)}$'s, which leads to numerous information exchanges between infected hosts. The optimal dynamic LS, however, provides the best scenario of LS and can be used as the baseline for designing some realistic LS worms.

5.4.2 Optimal Two-Level Localized Scanning

We can easily extend the above derivation to the optimal dynamic 2LLS and conclude the results here. Similar to $p_{t,i}^{(a)}$, let $p_{t,i}^{(b)}$ and $p_{t,i}^{(c)}$ $(0 \le p_{t,i}^{(b)} \le 1 - p_{t,i}^{(c)} \le 1)$ denote p_b and p_c at time t for an infected host in /16 subnet i. Assume that $N_i^{(16)}$ is the number of vulnerable hosts in /16 subnet i; $I_{t,i}^{(16)}$ is the number of infected hosts in /16 subnet i at time t; and $A_i^{(8)}$ is the set of /16 subnets that have the same first byte of the subnet address as /16 subnet i. Three items, $N_i^{(16)} - I_{t,i}^{(16)}$, $\frac{1}{2^8} \sum_{j \in A_i^{(8)}} (N_j^{(16)} - I_{t,j}^{(16)})$, and $\frac{1}{2^{16}}(N - I_t)$, are compared. The corresponding optimal 2LLS worm-scanning strategy is summarized in Table 2.

Table 2: Summary of the optimal 2LLS.

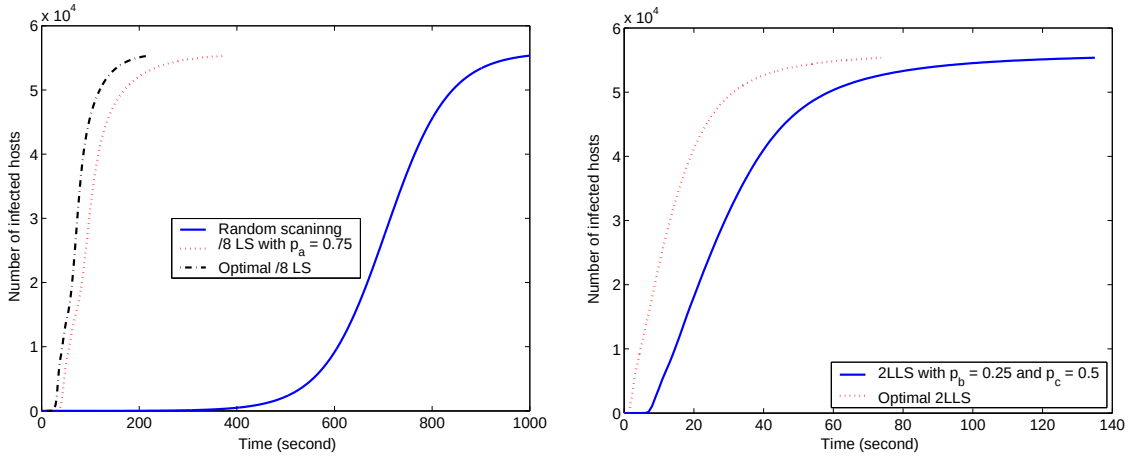
Compare three items	Result	Scanning strategy	Meaning
$\max\{N_i^{(16)} - I_{t,i}^{(16)}$, $\frac{1}{2^8}\sum_{j\in A_i^{(8)}}(N_j^{(16)} - I_{t,j}^{(16)})$, $\frac{1}{2^{16}}(N - I_t)\}$	first	$p_{t,i}^{(b)} = 0$, $p_{t,i}^{(c)} = 1$	scan the local /16 subnet
	second	$p_{t,i}^{(b)} = 1$, $p_{t,i}^{(c)} = 0$	scan the local /8 subnet
	third	$p_{t,i}^{(b)} = p_{t,i}^{(c)} = 0$	scan the global Internet

5.4.3 Experimental Results

In our experiments, we simulate the spread of a Witty worm, which has a vulnerable population $N = 55,909$ [92] and a scanning rate $s = 1,200$ per second [56]. The effect of disk damage on the Witty worm propagation is ignored. The worm is assumed to begin spreading from one initially infected host (i.e., $I_0 = 1$).

We evaluate the propagation speed of optimal LS worms by two methods. The first method is the numerical analysis of the worm propagation models. Specifically, the spread of /l LS worms is simulated by Equations (62) and (65), while the propagation of 2LLS worms is implemented by Equations (62) and (70). The optimal /l LS uses Equations (62), (73), and (75). RS is regarded as a special case of the /l LS when $p_a = 0$ and an extreme example of the 2LLS when $p_b = p_c = 0$. The initially infected host is assumed to be located in the subnet that contains the smallest number of vulnerable hosts. Figure 12(a) compares the propagation speeds of RS, optimal /8 LS, and the /8 LS with $p_a = 0.75$. Figure 12(b) compares the spreading speeds of optimal 2LLS and the 2LLS with $p_b = 0.25$ and $p_c = 0.5$. It is shown that LS can spread the worm much faster than RS, and the spreading speed of the currently used LS (i.e., 2LLS) can approach that of the optimal LS.

The second evaluation method uses a discrete event simulator to imitate the spread of LS worms. Our simulator implements each worm scan through a random number generator and simulates each scenario with 100 runs using different seeds. The initially infected host is located in the subnet that contains the largest number of vulnerable



(a) /8 LS. (b) 2LLS.

Figure 12: Numerical analysis of (optimal) LS worm propagation.

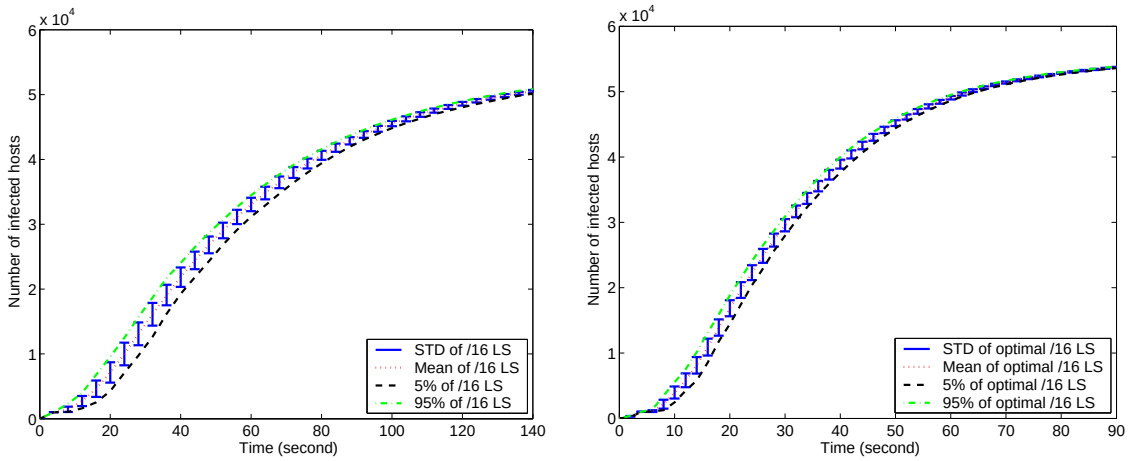
hosts. Figure 13(a) plots the mean and the variance of /16 LS worm propagation with $p_a = 0.75$. If a worm has a smaller variance, its spread is more predictable and stable. The "5%" (or "95%") propagation curve denotes that a worm spreads no slower (or faster) than this curve in 95 out of 100 simulation runs. The standard derivation (STD) error-bar reflects the variance of worm propagation among 100 simulation runs. It is observed that a /16 LS infected 50,318 (90%) vulnerable hosts in 138 seconds averagely. Figure 13(b) plots the simulation results of optimal /16 LS worm propagation. Such an optimal worm only takes 65 seconds to infected 90% vulnerable hosts. Moreover, the optimal /16 LS has a smaller variance compared with the /16 LS.

5.5 Variants of Localized Scanning

In this section, we study three variants of LS that can be easily implemented and do not require information exchanges between infected hosts.

5.5.1 Decision-First Localized Scanning

The first variant is called *decision-first localized scanning* (DFLS). Instead of combining local scanning and global scanning, DFLS makes an infected host focus on



(a) /16 LS. (b) Optimal /16 LS.

Figure 13: Simulations of /16 LS and optimal /16 LS worm propagation.

scanning either locally or globally. For example, when a host is infected, it flips a coin and makes a decision:

- Scan only the local /l subnet with probability p_a,
- Scan globally with probability $1 - p_a$.

This scanning strategy is called /l DFLS, which is the counterpart of /l LS. Since in a /l subnet p_a percentage of infected hosts scan locally and $1 - p_a$ percentage of infected hosts use random scanning, Equations (62) and (65) still hold for /l DFLS.

We write a simulator to imitate the spread of DFLS worms and use the same setting as in Figure 13. Figure 14 plots the mean and the variance of /16 DFLS worm propagation with $p_a = 0.75$. It is observed that /16 DFLS spreads slower than /16 LS and on average takes 140 seconds to infect 40,000 vulnerable hosts. Moreover, /16 DFLS has a large variance as shown in the figure. This is because each infected host scans only either locally or globally. The hosts scanning globally have a slower speed to find a target. On the other hand, the hosts scanning locally waste scans after the local infected hosts become saturated. Thus, DFLS lacks a randomized algorithm to search for targets both locally and globally and may not be a good candidate for

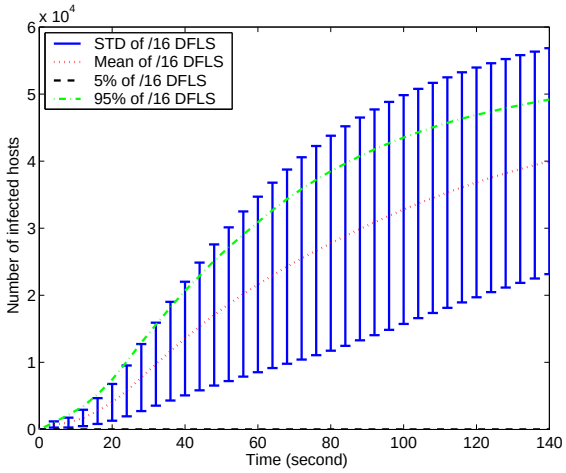


Figure 14: Simulations of /16 DFLS worm propagation.

worm attacks.

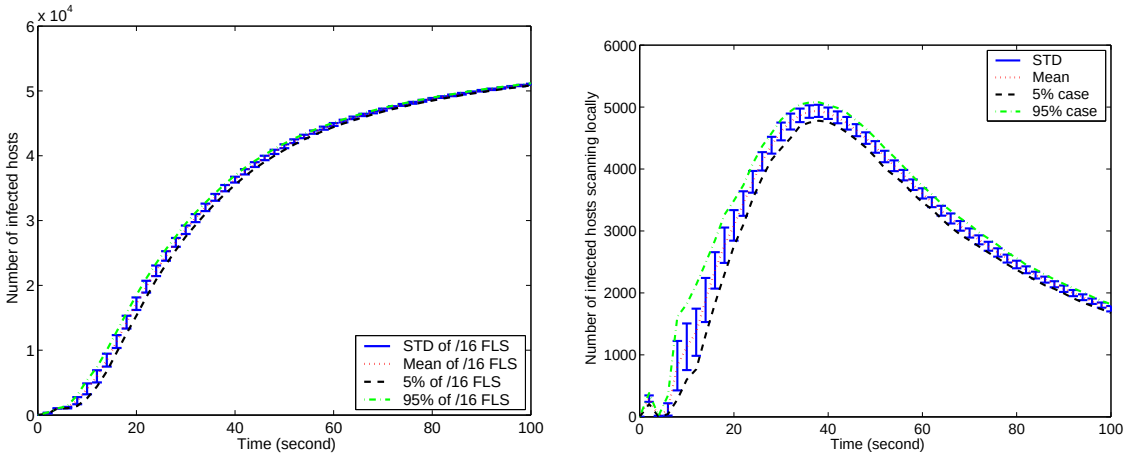
5.5.2 Feedback Localized Scanning and Ping-Pong Localized Scanning

The second variant is called *feedback localized scanning* (FLS), which is inspired by the optimal LS. The optimal strategy adapts the scanning methods, based on the local density of uninfected vulnerable hosts. In the similar way, we design a variant of LS, based on the feedback from the local probed host. For example, an infected host behaves as follows:

- First, initiates to scan the local $/l$ subnet until probing a local host that has been already infected,
- Then, switches from scanning locally to scanning the global Internet.

This scanning strategy is called $/l$ FLS. The basic idea is that when the infected host probes a local host that has been already infected, it realizes that the infected hosts in the subnet probably have become saturated and had better switch to scanning globally.

We also write a simulator for FLS and show the results in Figure 15. Figure 15(a) plots the mean and the variance of /16 FLS worm propagation. It is observed that /16



(a) /16 FLS. (b) Number of infected hosts scanning locally.

Figure 15: Simulations of /16 FLS worm propagation.

FLS takes only 93 seconds to infect 90% vulnerable hosts and further approaches the spreading capacity of the optimal /16 LS. Moreover, /16 FLS has a small variance. Figure 15(b) further plots how the number of infected hosts that scans locally changes with time. It is shown that the number first increases with time and reaches the maximum after about 40 seconds and then decreases with time. This indicates that in the beginning many infected hosts focus on scanning locally and later switch to scanning globally. Therefore, FLS shows a better performance than the original LS and can be a potential tool for attackers.

FLS can further be extended to a "ping-pong" localized scanning (PPLS) method by adding the following algorithm:

- An infected host that uses random scanning will switch to scanning the local /l subnet when it probes a host that has been already infected.

Thus, an infected host switches between local scanning and global scanning, in an attempt to adapt to the underlying distribution of uninfected vulnerable hosts. Figure 16 plots the mean and the variance of /16 PPLS worm propagation. /16 PPLS further improves worm propagation at the late stage and only takes 81 seconds to infected 90% vulnerable hosts.

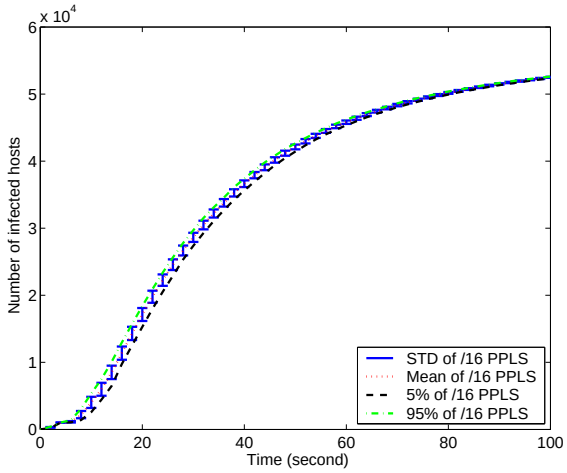


Figure 16: Simulations of /16 PPLS worm propagation.

5.6 Summary

In this chapter, we attempt to understand the behaviors of localized-scanning (LS) worms through both analysis and simulation. We have shown analytically that an LS worm spreads slower than a random-scanning (RS) worm if the vulnerable-host distribution is uniform, or faster if highly uneven. Moreover, if the infected hosts are uniformly distributed, the LS method can increase the spreading speed by nearly a non-uniformity factor compared with the RS scheme.

We have designed the optimal dynamic LS worms. The spreading speed of such optimal LS can be approached by the currently used LS, showing that the existing LS is near-optimal. We have also constructed three variants of LS. While the decision-first localized scanning (DFLS) shows a poor performance empirically, the feedback localized scanning (FLS) and the ping-pong localized scanning (PPLS) demonstrate better performances than the original LS and can be good candidates for worm attacks. The key of FLS and PPLS is that a worm adapts its scanning strategies based on the feedback from the probed host.

CHAPTER VI

NON-UNIFORMITY FACTOR: NETWORK-AWARE WORM ATTACKS AND DEFENSE

6.1 Introduction

Worm scanning has become more and more sophisticated since the initial attacks of Internet worms. Most of the real, especially "old" worms, such as Code Red [39], Slammer [41], and latter Witty [56], exploit naive random scanning that chooses target IP addresses uniformly and does not use any information on network vulnerabilities. Advanced scanning methods, however, have been developed that take the IP address structure into consideration. One example is routable scanning that selects targets only in the routable address space, using the information provided by the BGP routing table [70, 79]. Another example is evasive worms that exploit lightweight sampling to obtain the knowledge of *live* subnets of the address space and spread only in these networks [48].

This chapter focuses on a class of *network-aware worms*. Such worms exploit the information on the highly uneven distributions of vulnerable hosts. The vulnerable-host distributions have been observed to be bursty and spatially inhomogeneous by Barford et al. [3]. A non-uniform distribution of Witty-worm victims has been reported by Rajab et al. [49]. We have also found that a Web-server distribution is non-uniform in the IP address space [10]. These discoveries suggest that vulnerable hosts and Web servers may be "clustered" (i.e., non-uniform). The clustering/non-uniformity makes the network vulnerable since if one host is compromised in a cluster, the rest may be compromised rather quickly.

In our prior chapters, we have studied a class of "worst-case" worms, called *importance-scanning* worms, which exploit non-uniform vulnerable-host distributions

[10, 16]. Importance scanning is developed from and named after importance sampling in statistics. Importance scanning probes the Internet according to an underlying vulnerable-host distribution. Such a scanning method forces worm scans on the most relevant parts of an address space and supplies the optimal strategy[1]. Importance scanning thus provides a "what-if" scenario: When there are many ways for intelligent worms to exploit such a vulnerability, importance scanning is a worst-case threat-model. Hence, importance scanning can serve as a benchmark for studying real worms.

Are there any real network-aware worms? Code Red II and Nimda worms have used localized scanning [82, 83]. Localized scanning preferentially searches for vulnerable hosts in the "local" address space. The Blaster worm has used sequential scanning in addition to localized scanning [85]. Sequential scanning searches for vulnerable hosts through their closeness in the IP address space. It is not well understood, however, how to characterize the relationships between vulnerable-host distributions and these network-aware worms.

What has been observed is that real network-aware and importance-scanning worms spread much faster than random-scanning worms [49, 10]. This shows the importance of the problem. Does there exist a *generic* characteristic across different vulnerable-host distributions? If so, how do intelligent worms exploit such a vulnerability, and how can we defend against such worms?

Our goal is to investigate such a generic characteristic in vulnerable-host distributions, to quantify its relationship with network-aware worms, and to understand the effectiveness of defense strategies. In particular, we would like to answer the following questions:

- How to quantify the non-uniformity of a vulnerable-host distribution by a simple

[1]Hitlist scanning [60] can be regarded as a special case of importance scanning when the complete information of vulnerable hosts is known.

metric?

- How to measure the spreading ability of network-aware worms quantitatively?
- How to relate vulnerable-host distributions with network-aware worm spreading ability?
- What are the challenges to defense strategies on slowing down the spread of a network-aware worm?

To answer these questions, we first observe, from five measurement sets, common characteristics of non-uniform vulnerable-host distributions. We then derive a new metric as the *non-uniformity factor* to characterize the non-uniformity of a vulnerable-host distribution. A larger non-uniformity factor reflects a more non-uniform distribution of vulnerable hosts. We obtain the non-uniformity factors from the data sets on vulnerable-host distributions and show that all data sets have large non-uniformity factors. Moreover, the non-uniformity factor is a function of the Renyi entropy, a generalized entropy, of order two [50]. We show that the non-uniformity factor better characterizes the unevenness of a distribution than the Shannon entropy. Therefore, in view of information theory, the non-uniformity factor provides a quantitative measure of the unevenness/uncertainty of a vulnerable-host distribution.

Next, we analyze the spreading speed of network-aware worms, especially at an early stage. A worm that spreads faster at the early stage can in general infect most of the vulnerable hosts in a shorter time. The propagation ability of a worm at the early stage is characterized by the *infection rate* [79]. Therefore, we derive the infection rates of network-aware worms. We find that the infection rates of representative network-aware worms can be represented explicitly as a function of the non-uniformity factor. For example, localized scanning can increase the infection rate by nearly the non-uniformity factor, comparing to random scanning. Thus, the spreading speed of localized scanning can approach the capacity of sub-optimal importance scanning [10].

These analytical results on the relationships between vulnerable-host distributions and network-aware worm spreading ability are validated by simulation. Furthermore, to show the generality of our approach, we study sequential scanning. We demonstrate that a combination of sequential scanning and random scanning can increase the infection rate significantly.

Finally, we study new challenges to worm defense posed by network-aware worms. Using the non-uniformity factor, we show quantitatively that the host-based defense strategies, such as proactive protection [7] and virus throttling [63], should be deployed at almost all hosts to slow down network-aware worms at the early stage. A partial deployment would nearly invalidate such host-based defense. Moreover, we demonstrate that the infection rate of a network-aware worm in the IPv6 Internet can be comparable to that of the Code Red v2 worm in the IPv4 Internet. Therefore, fighting network-aware worms is a real challenge.

The remainder of this chapter is structured as follows. Section 6.2 presents our collected data sets. Sections 6.3 and 6.4 introduce a new metric called the non-uniformity factor and compare this metric to the Shannon entropy. Sections 6.5 and 6.6 characterize the spreading ability of network-aware worms through theoretical analysis and simulations. Section 6.7 further studies the effectiveness of defense strategies on network-aware worms. Section 6.8 concludes this chapter.

6.2 Measurements and Vulnerable-Host Distribution

How significant is the unevenness of vulnerable-host distributions? To answer this question, we study five data sets.

6.2.1 Measurements

DShield (D1): DShield collects intrusion detection system (IDS) logs [84]. Specifically, DShield provides the information of vulnerable hosts by aggregating logs from more than 1,600 IDSes distributed throughout the Internet. We further focus on the

following ports that were attacked by worms: 80 (HTTP), 135 (DCE/RPC), 445 (NetBIOS/SMB), 1023 (FTP servers and the remote shell attacked by W32.Sasser.E.Worm), and 6129 (DameWare).

iSinks (P1 and C1): Two unused address space monitors run the *iSink* system [75]. The monitors record the unwanted traffic arriving at the unused address spaces that include a Class-A network (referred to as "Provider" or P1) and two Class B networks at the campus of the University of Wisconsin (referred to as "Campus" or C1) [3].

Witty-worm victims (W1): A list of Witty-worm victims is provided by CAIDA [56]. CAIDA used a network telescope with approximate 2^{24} IP addresses to log the traffic of Witty-worm victims that are Internet security systems (ISS) products.

Web-server list (W2): IP addresses of Web servers were collected through UROULETTE (http://www.uroulette.com/). UROULETTE provides a random uniform resource locator (URL) generator to obtain a list of IP addresses of Web servers.

The first three data sets (D1, P1, and C1) were collected over a seven-day period from 10-16 December 2004 and have been studied in [3] to demonstrate the bursty and spatially inhomogeneous distribution of (malicious) source IP addresses. The last two data sets (W1 and W2) have been used in our prior work [10] to show the virulence of importance-scanning worms. The summary of our data sets is given in Table 3.

Table 3: Summary of the data sets.

Trace	Description	# of unique source addresses
D1	DShield	7,694,291
P1	Provider	2,355,150
C1	Campus	448,894
W1	Witty-worm victims	55,909
W2	Web servers	13,866

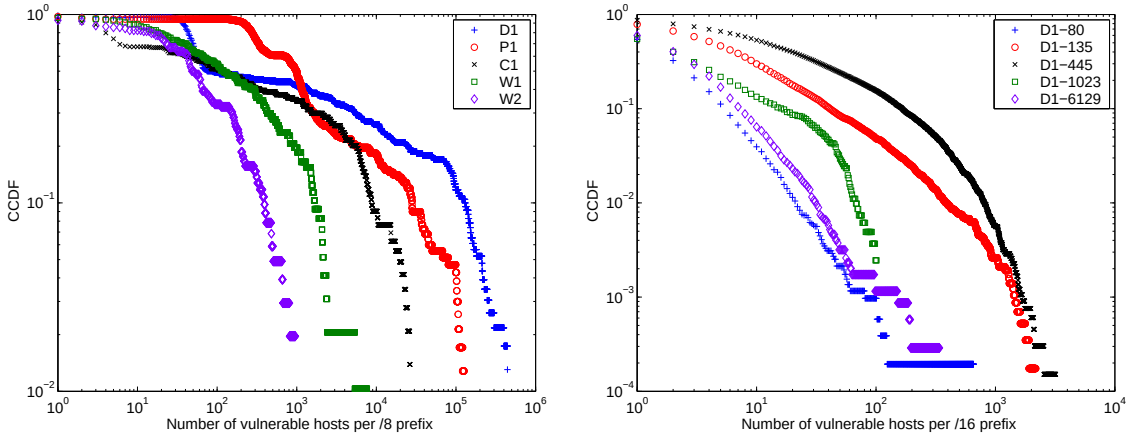
6.2.2 Vulnerable-Host Distribution

To obtain vulnerable-host group distributions, we use the classless inter-domain routing (CIDR) notation [34]. The Internet is partitioned into subnets according to the first l bits of IP addresses, i.e., $/l$ prefixes or $/l$ subnets. In this division, there are 2^l subnets, and each subnet contains 2^{32-l} addresses, where $l \in \{0, 1, \cdots, 32\}$. For example, when $l = 8$, the Internet is grouped into Class-A subnets (i.e., /8 subnets); when $l = 16$, the Internet is partitioned into Class-B subnets (i.e., /16 subnets).

We plot the complementary cumulative distribution functions (CCDF) of our collected data sets in /8 and /16 subnets in Figure 17 in log-log scales. CCDF is defined as the faction of the subnets with the number of hosts greater than a given value. Figure 17(a) shows population distributions in /8 subnets for D1, P1, C1, W1, and W2, whereas Figure 17(b) exhibits host distributions in /16 subnets for D1 with different ports (80, 135, 445, 1023, and 6129). Figure 17 demonstrates a wide range of populations, indicating highly inhomogeneous address structures. Specifically, the relatively straight lines, such as W2 and D1-135, imply that vulnerable hosts follow a power law distribution. Similar observations were given in [3, 49, 46, 39, 41, 10].

Why is the vulnerable-host distribution non-uniform in the IPv4 address space? First, no vulnerable hosts can exist in reserved or multicast address ranges [87]. Second, different subnet administrators make different use of their own IP address space. Third, a subnet intends to have many computers with the same operating systems and applications for easy management [59, 8]. Last, some subnets are more protected than others [3, 49].

How can we quantify the non-uniformity of a vulnerable-host distribution? One way is to use the population distribution such as CCDF plotted in Figure 17. But it is complex to compare the unevenness of two distributions.



(a) Population distributions in /8 subnets. (b) Population distributions in /16 subnets.

Figure 17: CCDF of collected data sets.

6.3 Non-Uniformity Factor

In this section, we derive a simple metric, called the *non-uniformity factor*, to quantify the non-uniformity of a vulnerable-host distribution.

6.3.1 Definition and Property

Let $p_g^{(l)}(i)$ $(i = 1, 2, \cdots, 2^l)$ denote the group distribution of vulnerable hosts in $/l$ subnets. Let $N_i^{(l)}$ be the number of vulnerable hosts in $/l$ subnet i and N be the total number of vulnerable hosts. Then, $p_g^{(l)}(i) = \frac{N_i^{(l)}}{N}$, which is the ratio between the number of vulnerable hosts in group i and the total number of vulnerable hosts. It is noted that $\sum_{i=1}^{2^l} p_g^{(l)}(i) = 1$ and $\sum_{i=1}^{2^l} N_i^{(l)} = N$.

Definition: The *non-uniformity factor* in $/l$ subnets is defined as

$$\beta^{(l)} = 2^l \sum_{i=1}^{2^l} \left(p_g^{(l)}(i)\right)^2. \tag{77}$$

It is noted that

$$\beta^{(l)} \geq \left(\sum_{i=1}^{2^l} p_g^{(l)}(i) \right)^2 = 1. \tag{78}$$

The above inequality is derived by the Cauchy-Schwarz inequality. The equality holds if and only if $p_g^{(l)}(i) = 2^{-l}$, for $i = 1, 2, \cdots, 2^l$. In other words, when the vulnerable-host distribution is uniform, $\beta^{(l)}$ achieves the minimum value 1. On the other hand,

since $p_g^{(l)}(i) \geq 0$,

$$\beta^{(l)} \leq 2^l \cdot \left(\sum_{i=1}^{2^l} p_g^{(l)}(i) \right)^2 = 2^l. \tag{79}$$

The equality holds when $p_g^{(l)}(j) = 1$ for some j and $p_g^{(l)}(i) = 0$, $i \neq j$, i.e., all vulnerable hosts concentrate on one subnet. This means that when the vulnerable-host distribution is extremely non-uniform, $\beta^{(l)}$ obtains the maximum value 2^l. Therefore, $\beta^{(l)}$ characterizes the non-uniformity of a vulnerable-host distribution. A larger non-uniformity factor reflects a more non-uniform distribution of vulnerable hosts.

How does $\beta^{(l)}$ vary with l? When $l = 0$, $\beta^{(0)} = 1$. In the other extreme where $l = 32$,

$$p_g^{(32)}(i) = \begin{cases} \frac{1}{N}, & \text{address } i \text{ is vulnerable to the worm;} \\ 0, & \text{otherwise,} \end{cases} \tag{80}$$

which results in $\beta^{(32)} = \frac{2^{32}}{N}$. More importantly, $\beta^{(l)}$ is a non-decreasing function of l, as shown below.

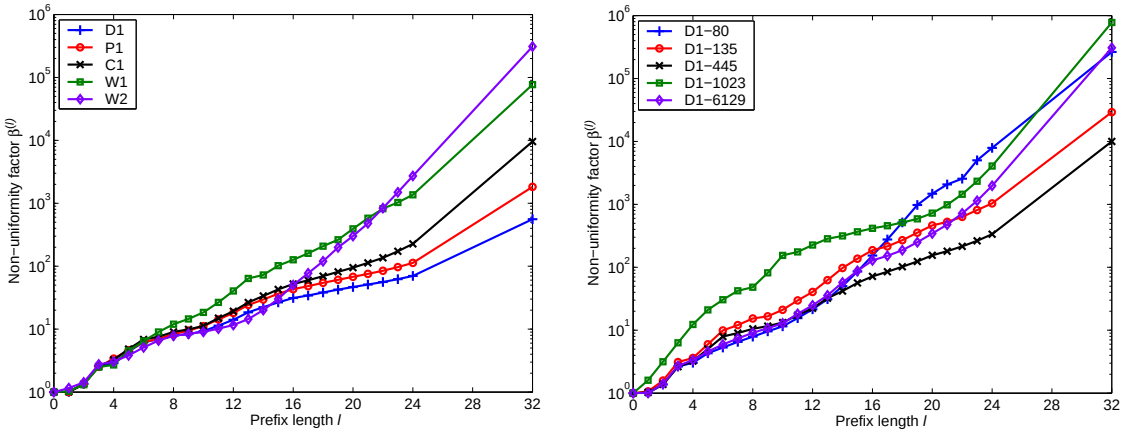
Theorem 4 *If $l > r$, $\beta^{(l)} \geq \beta^{(r)}$, where $l, r \in \{0, 1, \cdots, 32\}$.*

PROOF: Let $k = l - r$. Group i ($i = 1, 2, \cdots, 2^r$) of $/r$ subnets is partitioned into groups $2^k \cdot (i-1) + 1$, $2^k \cdot (i-1) + 2$, $\cdots$, $2^k \cdot (i-1) + 2^k$ of $/l$ subnets. Thus,

$$p_g^{(r)}(i) = \sum_{j=1}^{2^k} p_g^{(l)}(2^k \cdot (i-1) + j), \qquad i = 1, 2, \cdots, 2^r. \tag{81}$$

Then, $\beta^{(l)}$ is related to $\beta^{(r)}$ by the Cauchy-Schwarz inequality.

$$\begin{aligned} \beta^{(l)} &= 2^r \sum_{i=1}^{2^r} \left\{ \left(\sum_{j=1}^{2^k} 1^2 \right) \left[\sum_{j=1}^{2^k} \left(p_g^{(l)}(2^k \cdot (i-1) + j) \right)^2 \right] \right\} && (82) \\ &\geq 2^r \sum_{i=1}^{2^r} \left(\sum_{j=1}^{2^k} p_g^{(l)}(2^k \cdot (i-1) + j) \right)^2 && (83) \\ &= \beta^{(r)}. && (84) \end{aligned}$$



(a) Five data sets.

(b) D1 with different ports.

Figure 18: Non-uniformity factors of collected data sets. The y-axis uses a *log* scale.

The equality holds when $p_g^{(l)}(2^k \cdot (i-1)+j) = \frac{p_g^{(r)}(i)}{2^k}$, $j = 1, 2, \cdots, 2^k$, $i = 1, 2, \cdots, 2^r$. That is, in each $/r$ subnet, the vulnerable hosts are uniformly distributed in 2^k groups.

■

An intuitive explanation of this theorem is as follows. For $/l$ and $/(l+1)$ subnets, group i ($i = 1, 2, \cdots, 2^l$) of $/l$ subnets is partitioned into groups $2i-1$ and $2i$ of $/(l+1)$ subnets. If vulnerable hosts in each group of $/l$ subnets are equally divided into groups of $/(l+1)$ subnets (i.e., $p_g^{(l+1)}(2i-1) = p_g^{(l+1)}(2i) = \frac{1}{2}p_g^{(l)}(i)$, $\forall\ i$), then $\beta^{(l+1)} = \beta^{(l)}$. Otherwise, if the division of vulnerable hosts is uneven for a group (i.e., $p_g^{(l+1)}(2i-1) \neq p_g^{(l+1)}(2i)$, $\exists\ i$), then $\beta^{(l+1)} > \beta^{(l)}$.

6.3.2 Estimated Non-Uniformity Factor

Figure 18 shows the non-uniformity factors estimated from our data sets. The non-uniformity factors increase with the prefix length for all data sets. The y-axis is in a *log* scale. Thus, $\beta^{(l)}$ increases *almost exponentially* with a wide range of l. To gain intuition on how large $\beta^{(l)}$ can be, $\beta^{(8)}$ and $\beta^{(16)}$ are summarized for all data sets in Table 4. We observe that $\beta^{(8)}$ and $\beta^{(16)}$ have large values, indicating the significant unevenness of collected distributions.

Table 4: $\beta^{(8)}$ and $\beta^{(16)}$ of collected distributions.

$\beta^{(l)}$	D1	P1	C1	W1	W2
$\beta^{(8)}$	7.9	8.4	9.0	12.0	7.8
$\beta^{(16)}$	31.2	43.2	52.2	126.7	50.2
$\beta^{(l)}$	D1-80	D1-135	D1-445	D1-1023	D1-6129
$\beta^{(8)}$	7.9	15.4	10.5	48.2	9.1
$\beta^{(16)}$	153.3	186.6	71.7	416.3	128.9

6.4 Entropy and Non-Uniformity Factor

It is well-known that the Shannon entropy can be used to measure the non-uniformity of a distribution [20]. Why do we choose the non-uniformity factor instead?

Consider a general entropy, called the *Renyi entropy* [50], which is defined as

$$H_q\left(P^{(l)}\right) = \frac{1}{1-q}\log_2 \sum_{i=1}^{2^l} \left(p_g^{(l)}(i)\right)^q, \text{ for } q \neq 1, \tag{85}$$

where $P^{(l)} = \{p_g^{(l)}(1), p_g^{(l)}(2), \cdots, p_g^{(l)}(2^l)\}$. The non-uniformity factor can relate to the Renyi entropy of order two in the following equation:

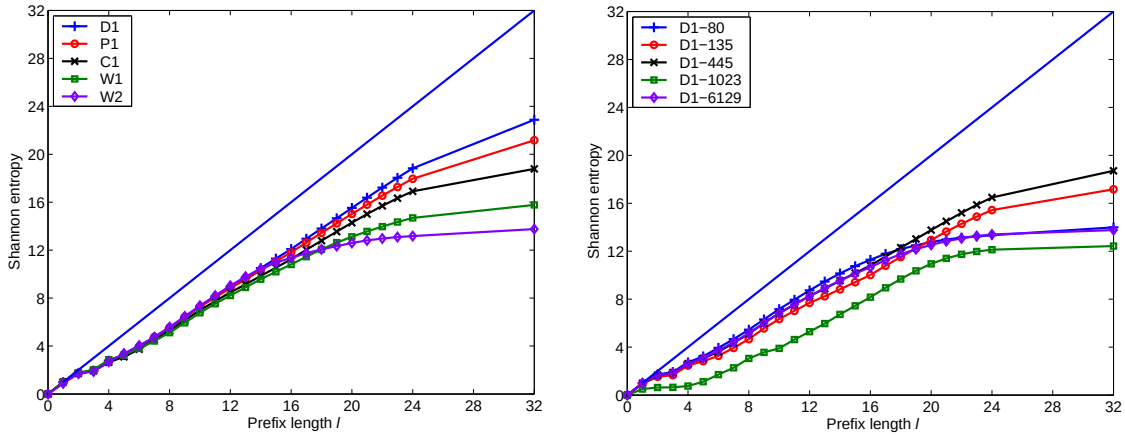
$$\beta^{(l)} = 2^{l - H_2\left(P^{(l)}\right)}. \tag{86}$$

Thus, the non-uniformity factor is essentially an entropy.

The Shannon entropy, $H\left(P^{(l)}\right) = -\sum_{i=1}^{2^l} p_g^{(l)}(i)\log_2 p_g^{(l)}(i)$, is a special case of the Renyi entropy [50], i.e.,

$$H\left(P^{(l)}\right) = \lim_{q\to 1} H_q\left(P^{(l)}\right). \tag{87}$$

Figure 19 shows the Shannon entropies of our empirical distributions from the data sets. If a distribution is uniform, $H\left(P^{(l)}\right) = l$ as denoted by the diagonal line in the figure. On the other hand, if a distribution is extremely non-uniform, e.g., all vulnerable hosts concentrate on one subnet, $H\left(P^{(l)}\right) = 0$. Hence, the distance between $H\left(P^{(l)}\right)$ and 0 in Figure 19 reflects how uniform a distribution is. Similarly, the distance between $\beta^{(l)}$ and the horizontal access 1 in Figure 18 measures the degree of unevenness. A larger $H\left(P^{(l)}\right)$ corresponds to a more even distribution,



(a) Five data sets. (b) D1 with different ports.

Figure 19: Shannon entropies of collected data sets.

whereas a larger $\beta^{(l)}$ corresponds to a more non-uniform distribution. Evidenced by Figure 18, the non-uniformity factor magnifies the unevenness of a distribution. In addition, if two distributions have different prefix lengths, we can directly apply the non-uniformity factor to compare the unevenness between them. Therefore, the non-uniformity factor provides a better measure for describing the non-uniformity of a distribution.

More importantly, the non-uniformity factor can directly reflect how much faster a network-aware worm spreads than a random-scanning worm, which is shown in the next section.

From an information theoretical viewpoint, the entropy provides a quantitative measure of uncertainty. The uncertainty of a vulnerable-host probability distribution is important for an attacker to design an intelligent network-aware worm. If there is no uncertainty about the distribution of vulnerable hosts (e.g., either all vulnerable hosts are concentrated on a subnet or all information about vulnerable hosts is known), the entropy is minimum, and the worm that uses the information on the distribution can spread fastest by employing the optimal importance scanning [10]. On the other hand, if there is maximum uncertainty (e.g., vulnerable hosts are uniformly distributed), the entropy is maximum. But the worm cannot take advantage of the information of the

distribution and can only use random scanning. Moreover, when an attacker obtains more information about the vulnerable-host distribution, in general, the resulting worm can spread faster.

6.5 Network-Aware Worm Spreading Ability

How to quantify the spreading speed of a network-aware worm with the information of a vulnerable-host distribution? We characterize the spread of a network-aware worm at an early stage by deriving the infection rate.

6.5.1 Infection Rate

The infection rate, denoted by α, is defined as the average number of vulnerable hosts that can be infected per unit time by one infected host during the early stage of worm propagation [79]. The infection rate is an important metric for studying network-aware worm spreading ability for two reasons. First, since the number of infected hosts increases exponentially with the rate $1 + \alpha$ during the early stage, a worm with a higher infection rate can spread much faster at the beginning and thus infect a large number of hosts in a shorter time [10]. Second, while it is generally difficult to derive a close-form solution for dynamic worm propagation, we can obtain a close-form expression of the infection rate for different worm-scanning methods.

Let R denote the (random) number of vulnerable hosts that can be infected per unit time by one infected host during the early stage of worm propagation. The infection rate is the expected value of R, i.e., $\alpha = E[R]$. Let s be the scanning rate or the number of scans sent by an infected host per unit time, N be the number of vulnerable hosts, and Ω be the scanning space (i.e., $\Omega = 2^{32}$).

For random scanning (RS) [79, 10], an infected host sends out s random scans per unit time, and the probability that one scan hits a vulnerable host is $\frac{N}{\Omega}$. Thus, R

follows a Binomial distribution B(s, $\frac{N}{\Omega}$)[2], resulting in

$$\alpha_{RS} = E[R] = \frac{sN}{\Omega}. \tag{88}$$

6.5.2 Importance Scanning

We derive the infection rates of importance scanning (IS) [10, 16]. An infected host scans /l subnet i with the probability $q_g^{(l)}(i)$. $q_g^{(l)}(i)$ is called the group scanning distribution and is to be chosen with respect to the group distribution $p_g^{(l)}(i)$. If a worm scan hits /l subnet i, it would have a probability of $\frac{Np_g^{(l)}(i)}{2^{32-l}}$ to find a vulnerable host. Thus, a worm scan hits a vulnerable host with a likelihood of $\sum_{i=1}^{2^l}\left(q_g^{(l)}(i)\cdot\frac{Np_g^{(l)}(i)}{2^{32-l}}\right)$. Similar to random scanning, R of IS follows a Binomial distribution B(s, $\sum_{i=1}^{2^l}\frac{Np_g^{(l)}(i)q_g^{(l)}(i)}{2^{32-l}}$), which leads to

$$\alpha_{IS} = E[R] = sN\sum_{i=1}^{2^l}\frac{p_g^{(l)}(i)q_g^{(l)}(i)}{2^{32-l}}. \tag{89}$$

The same result was derived in [10] but by a different approach.

We now consider a special case of IS, where the group scanning distribution $q_g^{(l)}(i)$ is chosen to be proportional to the number of vulnerable hosts in group i, i.e., $q_g^{(l)}(i) = p_g^{(l)}(i)$. This results in sub-optimal IS [10], called /l IS. Thus, the infection rate is

$$\alpha_{IS}^{(l)} = \frac{sN}{2^{32-l}}\sum_{i=1}^{2^l}\left(p_g(i)\right)^2 = \alpha_{RS}\cdot\beta^{(l)}. \tag{90}$$

Compared with RS, this /l IS can increase the infection rate by a factor of $\beta^{(l)}$. Such an infection rate can be considered as a benchmark for comparison with other network-aware worms.

6.5.3 Localized Scanning

Localized scanning (LS) has been used by such real worms as Code Red II and Nimda [49, 8]. We first consider a simplified version of LS, called /l LS, which scans the Internet as follows:

[2]In our derivation, we ignore the dependency of the events that different scans hit the same target at the early stage of worm propagation.

- p_a $(0 \leq p_a \leq 1)$ of the time, an address with the same first l bits is chosen as the target,
- $1 - p_a$ of the time, a random address is chosen.

Assume that an initially infected host is randomly chosen from the vulnerable hosts. Let I_g denote the subnet where an initially infected host locates. Thus, $P(I_g = i) = p_g^{(l)}(i)$, where $i = 1, 2, \cdots, 2^l$. For an infected host located in $/l$ subnet i, a scan from this host probes globally with the probability of $1 - p_a$ and hits $/l$ subnet j $(j \neq i)$ with the likelihood of $\frac{1-p_a}{2^l}$. Thus, the group scanning distribution for this host is

$$q_g^{(l)}(j) = \begin{cases} p_a + \frac{1-p_a}{2^l}, & \text{if } j = i; \\ \frac{1-p_a}{2^l}, & \text{otherwise,} \end{cases} \tag{91}$$

where $j = 1, 2, \cdots, 2^l$. Given the subnet location of an initially infected host, we can apply the results of IS. Specifically, putting Equation (91) into Equation (89), we have

$$E[R|I_g = i] = \frac{sN}{2^{32-l}} \left(p_a p_g^{(l)}(i) + \frac{1 - p_a}{2^l} \right). \tag{92}$$

Therefore, we can compute the infection rate of $/l$ LS as

$$\alpha_{LS}^{(l)} = E[R] = E[E[R|I_g]] = \sum_{i=1}^{2^l} p_g^{(l)}(i) E[R|I_g = i], \tag{93}$$

resulting in

$$\alpha_{LS}^{(l)} = \alpha_{RS} \left(1 - p_a + p_a \beta^{(l)} \right). \tag{94}$$

Since $\beta^{(l)} > 1$ ($\beta^{(l)} = 1$ is for a uniform distribution and is excluded here), $\alpha_{LS}^{(l)}$ increases with respect to p_a. Specifically, when $p_a \to 1$, $\alpha_{LS}^{(l)} \to \alpha_{RS} \beta^{(l)} = \alpha_{IS}^{(l)}$. Thus, $/l$ LS has an infection rate comparable to that of $/l$ IS. In reality, p_a cannot be 1. This is because an LS worm begins spreading from one infected host that is specifically in a subnet; and if $p_a = 1$, the worm can never spread out of this subnet. Therefore, we expect that the optimal value of p_a should be large but not 1.

Next, we further consider another LS, called two-level LS (2LLS), which has been used by the Code Red II and Nimda worms [82, 83]. 2LLS scans the Internet as follows:

- p_b ($0 \leq p_b \leq 1$) of the time, an address with the same first byte is chosen as the target,
- p_c ($0 \leq p_c \leq 1 - p_b$) of the time, an address with the same first two bytes is chosen as the target,
- $1 - p_b - p_c$ of the time, a random address is chosen.

For example, for the Code Red II worm, $p_b = 0.5$ and $p_c = 0.375$ [82]; for the Nimda worm, $p_b = 0.25$ and $p_c = 0.5$ [83]. Using the similar analysis for $/l$ LS, we can derive the infection rate of 2LLS:

$$\alpha_{2LLS} = \alpha_{RS} \left(1 - p_b - p_c + p_b\beta^{(8)} + p_c\beta^{(16)}\right). \tag{95}$$

Since $\beta^{(16)} \geq \beta^{(8)} \geq 1$ from Theorem 4, α_{2LLS} holds or increases when both p_b and p_c increase. Specially, when $p_c \to 1$, $\alpha_{2LLS} \to \alpha_{RS}\beta^{(16)} = \alpha_{IS}^{(16)}$. Thus, 2LLS has an infection rate comparable to that of /16 IS. Moreover, $\beta^{(16)}$ is much larger than $\beta^{(8)}$ as shown in Table 4 for the collected distributions. Hence, p_c is more significant than p_b for 2LLS.

6.5.4 Modified Sequential Scanning

The Blaster worm is a real worm that exploits sequential scanning in combination with localized scanning. A *sequential-scanning* worm studied in [81, 30] begins to scan addresses sequentially from a randomly chosen starting IP address and has a similar propagation speed as a random-scanning worm. The Blaster worm selects its starting point locally as the first address of its Class-C subnet with probability 0.4 [85, 81]. To analyze the effect of sequential scanning, we do not incorporate localized

scanning. Specifically, we consider our $/l$ modified sequential-scanning (MSS) worm, which scans the Internet as follows:

- Newly infected host A begins with random scanning until finding a vulnerable host with address B.
- After infecting the target B, host A continues to sequentially scan IP addresses $B+1$, $B+2$, $\cdots$ (or $B-1$, $B-2$, $\cdots$) in the $/l$ subnet where B locates.

Such a sequential worm-scanning strategy is in a similar spirit to the *nearest neighbor rule*, which is widely used in pattern classification [19]. The basic idea is that if the vulnerable hosts are clustered, the neighbor of a vulnerable host is likely to be vulnerable also.

Such a $/l$ MSS worm has two stages. In the first stage (called MSS_1), the worm uses random scanning and has an infection rate of α_{RS}, i.e., $\alpha_{MSS\_1} = \alpha_{RS}$. In the second stage (called MSS_2), the worm scans sequentially in a $/l$ subnet. The fist l bits of a target address are fixed, whereas the last $32-l$ bits of the address are generated additively or subtractively and are modulated by 2^{32-l}. Let I_g denote the sunbet where B locates. Thus, $P(I_g = i) = p_g^{(l)}(i)$, where $i = 1, 2, \cdots, 2^l$. Since a sequential worm scan in subnet i has a probability of $\frac{N_i^{(l)}}{2^{32-l}}$ to hit a vulnerable host, $E[R|I_g = i] = \frac{N_i^{(l)}}{2^{32-l}} s = \alpha_{RS} \cdot 2^l p_g^{(l)}(i)$, which leads to

$$\alpha_{MSS\_2} = E[R] = E[E[R|I_g]] = \alpha_{RS} \cdot \beta^{(l)}. \tag{96}$$

Therefore, the infection rate of $/l$ MSS is between α_{RS} and $\alpha_{RS}\beta^{(l)}$.

In Summary, the infection rates of all three network-aware worms (IS, LS, and MSS) can be far larger than that of an RS worm, depending on the non-uniformity factors.

6.6 Simulation and Validation

In this section, we validate our analytical results through simulations and the collected data sets.

6.6.1 Infection Rate

We first focus on validating infection rates. We apply the discrete event simulation to our experiments [52]. Specifically, we simulate the searching process of a worm using different scanning methods at the early stage. We use the C1 data set for the vulnerable-host distribution. The worm spreads over the C1 distribution with $N = 448,894$ and has a scanning rate $s = 100$. Note that the C1 distribution has the non-uniformity factors $\beta^{(8)} = 9.0$ and $\beta^{(16)} = 52.2$. The simulation stops when the worm has sent out 10^3 scans for RS, /16 IS, /16 LS, and 2LLS, and 65,535 scans for /16 MSS_2. Then, we count the number of vulnerable hosts hit by the worm scans and compute the infection rate. The results are averaged over 10^4 runs. Table 5 compares the simulation results (i.e., sample mean) with the analytical results (i.e., Equations (88), (90), (94), (95), and (96)). Here, a /16 LS worm uses $p_a = 0.75$, whereas a 2LLS worm employs $p_b = 0.25$ and $p_c = 0.5$. We observe that the sample means and the analytical results are almost identical.

Table 5: Infection rates of different scanning methods.

Scanning method	RS	/16 IS	/16 LS	2LLS	/16 MSS_2
Analytical result	0.0105	0.5456	0.4118	0.2989	0.5456
Sample mean	0.0103	0.5454	0.4023	0.2942	0.5489
Sample variance	0.0010	0.0543	0.2072	0.1053	0.3186

We observe that network-aware worms have much larger infection rates than random-scanning worms. LS indeed increases the infection rate with nearly the non-uniformity factor and approaches the capacity of sub-optimal IS. This is significant as LS only depends on one or two parameters (i.e., p_a for $/l$ LS and p_b, p_c for 2LLS),

while IS requires the information of the vulnerable-host distribution. On the other hand, LS has a larger sample variance than IS as indicated by Table 5. This implies that the infection speed of an LS worm depends on the location of initially infected hosts. If the LS worm begins spreading from a subnet containing densely populated vulnerable hosts, the worm would spread rapidly. Furthermore, we notice that the MSS worm also has a large infection rate at the second stage, indicating that MSS can indeed exploit the clustering pattern of the distribution. Meanwhile, the large sample variance of the infection rate of MSS_2 reflects that an MSS worm strongly depends on the initially infected hosts. We further compute the infection rate of a /16 MSS worm that includes both random-scanning and sequential-scanning stages. Simulation results are averaged over 10^6 runs and are summarized in Table 6. These results strongly depend on the total number of worm scans. When the number of worm scans is small, an MSS worm behaves similar to a random-scanning worm. When the number of worm scans increases, the MSS worm spends more scans on the second stage and thus has a larger infection rate.

Table 6: Infection rates of a /16 MSS worm.

# of worm scans	10	100	1000	10000	50000
Sample mean	0.0108	0.0190	0.0728	0.2866	0.4298
Sample variance	0.1246	0.1346	0.1659	0.2498	0.2311

6.6.2 Dynamic Worm Propagation

An infection rate only characterizes the early stage of worm propagation. We now employ the analytical active worm propagation (AAWP) model and its extensions to characterize the entire spreading process of worms [8]. Specifically, the spread of RS and IS worms is implemented as described in [10], whereas the propagation of LS worms is modeled according to [49]. The parameters that we use to simulate a worm are comparable to those of the Code Red v2 worm. Code Red v2 has a vulnerable

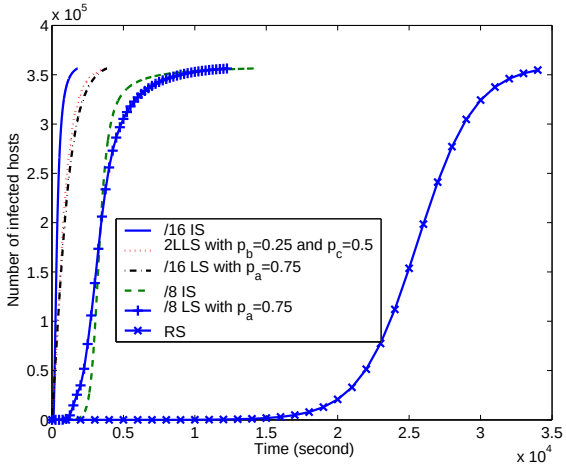


Figure 20: A network-aware worm spreads over the D1-80 distribution.

population $N = 360,000$ and a scanning rate $s = 358$ per minute [77]. We assume that the worm begins spreading from an initially infected host that is located in the subnet containing the largest number of vulnerable hosts.

We first show the propagation speeds of network-aware worms for the same vulnerable-host distribution from data set D1-80. From Section 6.5, we expect that a network-aware worm can spread much faster than an RS worm. Figure 20 demonstrates such an example on a worm that uses different scanning methods. It takes an RS worm 10 hours to infect 99% of vulnerable hosts, whereas a /8 LS worm with $p_a = 0.75$ or a /8 IS worm takes only about 3.5 hours. A /16 LS worm with $p_a = 0.75$ or a 2LLS worm with $p_b = 0.25$ and $p_c = 0.5$ can further reduce the time to 1 hour. A /16 IS worm spreads fastest and takes only 0.5 hour.

We also study the effect of vulnerable-host distributions on the propagation of network-aware worms. From Table 4, we observe that $\beta_{D1-1023}^{(16)} > \beta_{W1}^{(16)} > \beta_{C1}^{(16)} > \beta_{D1}^{(16)}$. Thus, we expect that a network-aware worm using the /16 D1-1023 distribution would spread faster than using other three distributions. Figure 21 verifies this through the simulations of the spread of a 2LLS worm that uses different vulnerable-host distributions (i.e., D1-1023, W1, C1, and D1). Here, the 2LLS worm employs the same parameters as the Nimda worm, i.e., $p_b = 0.25$ and $p_c = 0.5$. As expected, the

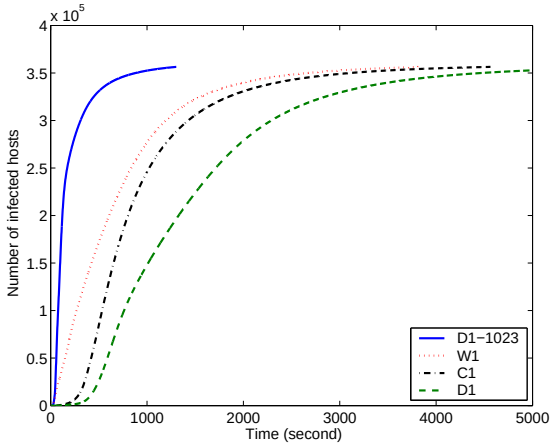


Figure 21: A 2LLS worm spreads over different distributions.

worm using the D1-1023 distribution spreads fastest, especially at the early stage of worm propagation.

6.7 Effectiveness of defense strategies

What are new requirements and challenges for a defense system to slow down the spread of a network-aware worm? We study the effectiveness of defense strategies through non-uniformity factors.

6.7.1 Host-Based Defense

Host-based defense has been widely used for random-scanning worms. Proactive protection and virus throttling are examples of host-based defense strategies.

A *proactive protection* (PP) strategy proactively hardens a system, making it difficult for a worm to exploit vulnerabilities [7]. Techniques used by PP include address-space randomization, pointer encryption, instruction-set randomization, and password protection. Thus, a worm requires multiple trials to compromise a host that implements PP. Specifically, let p ($0 \leq p \leq 1$) denote the protection probability or the probability that a single worm attempt succeeds in infecting a vulnerable host that implements PP. On the average, a worm should make $\frac{1}{p}$ exploit attempts to compromise the target. We assume that hosts with PP are uniformly deployed in the

Internet. Let d ($0 < d \leq 1$) denote the deployment ratio between the number of hosts with PP and the total number of hosts.

To show the effectiveness of the PP strategy, we consider the infection rate of a $/l$ IS worm. Since now some of the vulnerable hosts implement PP, Equation (90) changes to

$$\alpha_{IS}^{(l)} = \frac{sN}{2^{32-l}} \sum_{i=1}^{2^l} \left[dp \left(p_g^{(l)}(i) \right)^2 + (1-d) \left(p_g^{(l)}(i) \right)^2 \right] \quad (97)$$

$$= \alpha_{RS} \beta^{(l)} (1 - d + dp). \quad (98)$$

To slow down the spread of a sub-optimal IS worm to that of a random-scanning worm, $\beta^{(l)}(1 - d + dp) \leq 1$, resulting in

$$p \leq \frac{1 - (1-d)\beta^{(l)}}{d\beta^{(l)}}. \quad (99)$$

When PP is fully deployed, i.e., $d = 1$, p can be at most $\frac{1}{\beta^{(l)}}$. On the other hand, if PP provides perfect protection, i.e., $p = 0$, d should be at least $1 - \frac{1}{\beta^{(l)}}$. Therefore, when $\beta^{(l)}$ is large, Inequality (99) presents high requirements for the PP strategy. For example, if $\beta^{(16)} = 50$ (most of $\beta^{(16)}$'s in Table 4 are larger than this value), $p \leq 0.02$ and $d \geq 0.98$. That is, a PP strategy should be almost fully deployed and provide a nearly perfect protection for a vulnerable host.

We extend the model described in [10] to characterize the spread of sub-optimal IS worms under the defense of the PP strategy and show the results in Figure 22. Here, Code-Red-v2-like worms spread over the C1 distribution with $\beta^{(16)} = 52.2$. It is observed that even when the protection probability is small (e.g., $p = 0.01$) and the deployment ratio is high (e.g., $d = 0.8$), a /16 IS worm is slowed down a little at the early stage, compared with a /16 IS worm without the PP defense (i.e., $p = 1$ and $d = 0$). Moreover, when p is small (e.g., $p \leq 0.02$), d is a more sensitive parameter than p.

We next consider the *virus throttling* (VT) strategy that constrains the number of outgoing connections of a host [63]. Thus, VT can reduce the scanning rate of an

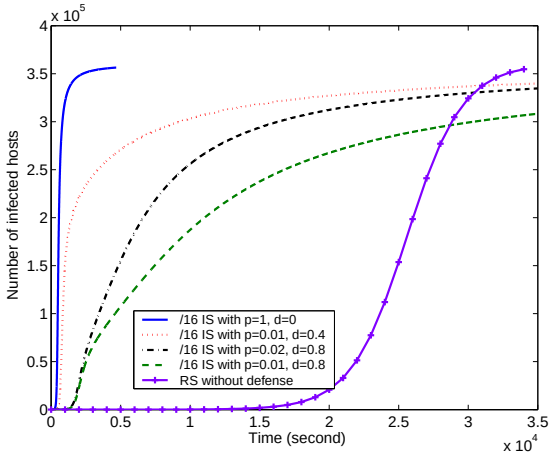


Figure 22: A /16 IS worm spreads under the defense of PP.

infected host. We find that Equation (98) also holds for this strategy, except that p is the ratio between the scanning rate of infected hosts with VT and that of infected hosts without VT. Therefore, VT also requires to be almost fully deployed for fighting network-aware worms effectively.

From these two strategies, we have learned that an effective strategy should reduce either α_{RS} or $\beta^{(l)}$. Host-based defense, however, is limited in such capabilities shown in this section.

6.7.2 IPv6

IPv6 can decrease α_{RS} significantly [79] by increasing the scanning space. But the non-uniformity factor would increase the infection rate if the vulnerable-host distribution is still non-uniform. Hence, an important question is whether IPv6 can counteract network-aware worms when both α_{RS} and $\beta^{(l)}$ are taken into consideration.

We study this issue by computing the infection rate of a network-aware worm in the IPv6 Internet. As pointed out by [4], a smart worm can first detect some vulnerable hosts in /64 subnets containing many vulnerable hosts, then release to the hosts on the hitlist, and finally spread inside these subnets. Such a worm only scans the local /64 subnet. Thus, we focus on the spreading speed of a network-aware worm

in a /64 subnet. From Figure 18, we extrapolate that $\beta^{(32)}$ in the IPv6 Internet can be in the order of 10^5 if hosts are still distributed in a clustered fashion. Using the parameters $N = 10^8$ proposed by [26] and $s = 4,000$ used by the Slammer worm [41], we derive the infection rate of a /32 IS worm in a /64 subnet of the IPv6 Internet: $\alpha_{IS}^{IPv6} = \frac{sN}{2^{64}} \cdot \beta^{(32)} = 2.2 \times 10^{-3}$. α_{IS}^{IPv6} is larger than the infection rate of the Code Red v2 worm in the IPv4 Internet, where $\alpha_{RS}^{CR} = \frac{360,000 \times 358/60}{2^{32}} = 5 \times 10^{-4}$.

Therefore, IPv6 can only slow down the spread of a network-aware worm to that of a random-scanning worm in IPv4. To defend against the worm effectively, we should further consider how to slow down the increase rate of $\beta^{(l)}$ as l increases when IPv4 is updated to IPv6.

6.8 Summary

In this chapter, we have observed and characterized non-uniform vulnerable-host distributions across five measurement sets from different sources. We have derived a simple metric, known as the non-uniformity factor, to quantify an uneven distribution of vulnerable hosts. The non-uniformity factors have been obtained using our collected data, and all of which demonstrate large values. This implies that the non-uniformity of the vulnerable-host distribution is significant and seems to be consistent across networks and applications. Moreover, the non-uniformity factor, shown as a function of the Renyi entropy of order two, better characterizes the uneven feature of a distribution than the Shannon entropy.

The importance of a non-uniformity factor is that it quantifies the spreading ability of network-aware worms. We have derived analytical expressions relating the non-uniformity factors with the infection rates of network-aware worms. We have empirically verified that localized scanning and modified sequential scanning can increase the infection rate by nearly the non-uniformity factor when compared to random scanning and thus approach the capacity of sub-optimal importance scanning.

Furthermore, we have evaluated the effectiveness of several commonly used defense strategies on network-aware worms. The host-based defense, such as proactive protection or virus throttle, requires to be almost fully deployed to slow down worm spreading at the early stage. This implies that host-based defense would be weakened significantly by network-aware scanning. More surprisingly, different from previous findings, we have shown that network-aware worms can be zero-day worms in the IPv6 Internet if vulnerable hosts are still clustered. These findings present a significant challenge to worm defense: Entirely different strategies may be needed for fighting network-aware worms.

CHAPTER VII

SPATIAL-TEMPORAL MODELING OF WORM PROPAGATION IN NETWORKS

7.1 Introduction

In this chapter, "worms" are used to cover an entire gamut of hostile softwares including viruses and network worms [88, 61]. There are mainly two types of worms categorized by how they spread. Active network worms such as Slammer and Morris exploit self-propagating malicious code [40], whereas viruses such as Melissa and Concept need human interactions to spread [28]. Spreading can take place rapidly, resulting in potential network damages and service disruptions. Hence, an important step towards preventing such catastrophic events is to study the dynamic behavior of worm spreading.

The recent investigations of worm propagation mostly focus on modeling the spread of worms employing a random scanning scheme [40, 78, 8]. Random scanning selects targets to infect randomly. Worms, however, can use other scanning methods. For example, the Morris worm exploits topological scanning that examines local configuration files to find potential neighbors [44]. Although only a few topological worms are known, topological scanning is a potential threat to the network routing infrastructure, World Wide Web (WWW) networks, and peer-to-peer systems [59], where topologies play an important role for worm propagation [80]. Only a handful of works, however, have been done on topological-scanning worms. For instance, a contact process is used to analyze the ease of propagation on different topologies [27]. The difficulty lies in characterizing the impact of topologies and the interactions among nodes in both space and time [32]. Such interactions result in a complex spatial-temporal dependence, which is especially hard to model.

The goal of this work is to develop a modeling framework and mathematical models that can characterize the spread of worms employing different scanning strategies and the impact of the underlying topology on worm propagation. To this purpose, we first abstract the problem of worm propagation using a graphical representation so that different scanning methods can be mapped to the corresponding topologies and parameters. With the help of the graphical representation, we then formulate worm propagation through a spatial-temporal random process based on the interactions among nodes. We take advantage of a discrete-time model and detailed topology information to describe the spatial and temporal statistical dependencies of worm propagation in arbitrary networks.

As the temporal dependence can be naturally modeled as Markov, the spatial dependence requires calculations with a multivariate probability distribution. When the number of random variables is large, an exact solution to the spatial dependence is computationally too expensive to obtain. The problem then becomes how to approximate the spatial dependence using a simple (i.e., biased) model in a general setting of machine learning. In particular, the spatial approximation is studied in light of the mean-field approximation [43]. The mean-field approximation is widely studied in machine learning [43] but usually for static networks where time is not involved. Exact mean-field solutions for dynamic networks are complex. Hence, we consider in this work simple approximations. The simplest approximation assumes spatial independence, which is asserted in our *independent model*. The spatial independence assumption factorizes an exact joint probability distribution into a form that only depends on one-node marginal probabilities. Although the independent model ignores the spatial dependence, it captures the temporal dependence and the detailed topology information. Simulation results show that the independent model performs better than the previous models in characterizing the transient behavior of worm propagation. A test on spatial correlation though indicates a strong spatial

dependence among nodes. We therefore present the *Markov model* that incorporates the simplest spatial dependence as the *conditional* independence, motivated by the Bethe approximation used in graphical models [74]. The spatial Markov assumption factorizes an exact joint probability distribution into a form that only depends on one-node and two-node marginal probabilities. We have conducted both theoretical analysis and extensive simulations on the real and synthesized topologies of large networks. Our results demonstrate that the Markov model equipped with the simple spatial dependence can achieve a greater accuracy than the independent model, especially in the sparse graphs. We then use a *relative entropy* to illustrate a performance gap between the Markov model and the reality, suggesting directions for further improvements.

We apply our proposed models to describe the *final size of infection* that corresponds to the equilibrium solution and characterizes the potential damage of worm propagation. Simulation results show that the Markov model can characterize the final size of infection no matter whether the underlying network is a homogeneous network or a complex network.

The rest of this chapter is organized as follows. In Section 7.2, we provide a problem formulation of worm propagation. In Section 7.3, we model the spread of worms accurately through a spatial-temporal random process. To approximate the spatial dependence, we present the independent model and the Markov model in Sections 7.4 and 7.5, respectively. In Section 7.6, we apply our proposed models to estimate the final size of infection. We conclude this chapter in Section 7.7 with a brief summary.

7.2 Worm Propagation in Networks

In this section, we first introduce worm propagation briefly. We then abstract the problem, using a susceptible → infected → susceptible (SIS) model and a graphical representation. Finally, we model different scanning mechanisms using graphical representations.

7.2.1 Worm Propagation

A computer is called *infected* if a worm is present there, and *susceptible* if it could be infected by the intrusion of the worm. If a worm cannot exist on the computer, we call this computer *insusceptible* to the worm. An infected computer is *cured* if it removes the copy of the worm and recovers to be susceptible. The *final size of infection* is defined as the number of initially susceptible computers that ultimately become infected in a network. The widespread occurrence of a worm is referred to as an *epidemic* [1]. Worm propagation is a procedure that the worm infects as many computers as possible through network connections. Those connections can be logical as to be described below.

A worm can propagate in many ways. For example, when a worm is released into the Internet, it scans many machines among its neighbors in an attempt to find a susceptible machine. When a vulnerable host is found, the worm sends out a probe to infect the target. If successful, a copy of this worm is transferred to the new host, which then begins to run the worm code and tries to infect other targets. The Morris worm is a typical self-propagation worm and moves from node to node, using only its own and the infected node's local information [44]. Specifically, the Morris worm retrieves the neighbor list from the local Unix files */ect/hosts.equiv* and */.rhosts* and in individual users' *.forward* and *.rhosts* files. Another topological worm is a SSH worm, which locates new targets by searching its current host for the names and the addresses of other hosts that are likely to be susceptible to infection

[55]. An email virus is another example of topological worms. When an email user receives an email message and opens the attachment containing a virus program, the virus infects the user's machine and uses the recipient's address book to send copies of itself to other email addresses. The addresses in the address book represent the neighborhood relationship. A *birth rate* (or an *infection rate*) is introduced to denote the rate at which an infected computer can infect a susceptible neighbor. The birth rate is affected by many factors. For example, for worms, the factors include the number of computer's susceptible neighbors, the payload size of a worm copy, the exploited computer vulnerability, and network congestion. For email viruses, the factors include the email checking frequency, user vigilance in opening an email attachment, and mailbox configuration. Some worms may have a large birth rate to flood the network as quickly as possible, whereas other worms spread slowly and surreptitiously to evade detection and thus have a small birth rate.

An infected computer might die for encountering an unexpected resource limit on the computer. Moreover, during the spreading of a worm, some infected computers may stop functioning properly, forcing the users to reboot these machines or kill some of the processes exploited by the worm. These computers are then cured, but subject to further infection. A *death rate* (or a *cure rate*) is introduced to denote the rate at which an infected computer becomes susceptible. The death rate is affected by many factors, such as resources on the computers, user alertness, the ability of a worm to disguise, and the performance of intrusion detection systems (IDS).

Combining infection and recovery, we have one of the simplest epidemiological models, the *susceptible* $\rightarrow$ *infected* $\rightarrow$ *susceptible (SIS)* model, which is widely used in epidemiological research [1]. Such a model neglects the details of infection inside a single computer, abstracts the worm transmission and removal as probabilities per unit time in the form of the birth rate and the death rate, and considers a computer to be in one of the two possible discrete statuses, *infected* or *susceptible*. Although

simple, the SIS model can capture key characteristics of worm spreading dynamics. The *susceptible → infected (SI)* model further ignores recovery and is regarded as a special case of the SIS model.

The SIS model assumes that an infected computer cannot be re-infected. The model also assumes that users do not become more vigilant after experiencing a worm infection. Therefore, the birth rate and the death rate do not change with time. Moreover, we ignore patching that is usually employed to repair security holes at the computers. This is because the spreading of worms can be much faster compared with traditional patching techniques that need human intervention, and a patch may not be available when some worm attacks unknown vulnerabilities. Nevertheless, our proposed models can be easily extended to take patching into consideration.

7.2.2 Graphical Representation

A *worm network* consists of all nodes in a network that are either infected or susceptible. The worm network can be constructed by removing insusceptible nodes and the edges associated with these nodes in the original network. Hence, a worm network is an abstraction of vulnerable nodes that can be either end-hosts, routers, and servers, or email addresses.

We use a *directed graph* $G(V, E)$ to represent the worm network, where V is the set of nodes and E is the set of edges. As defined in Section 7.2.1, each node has two statuses, susceptible or infected, as illustrated in Figure 23. Each edge (j, i) is associated with β_{ji}, the *birth rate* at which an infected node j can infect a susceptible neighbor i. Similarly, each node i is associated with δ_i, the *death rate* at which an infected node i becomes susceptible. The *neighborhood* of node i, denoted by N_i, is a subset of V such that every node j in this subset has an edge from node j to node i, i.e., $N_i = \{j|(j, i) \in E\}$. Figure 23 shows an example of a directed graph wherein the neighborhood of node 1 is given as $N_1 = \{3, 4, 5\}$.

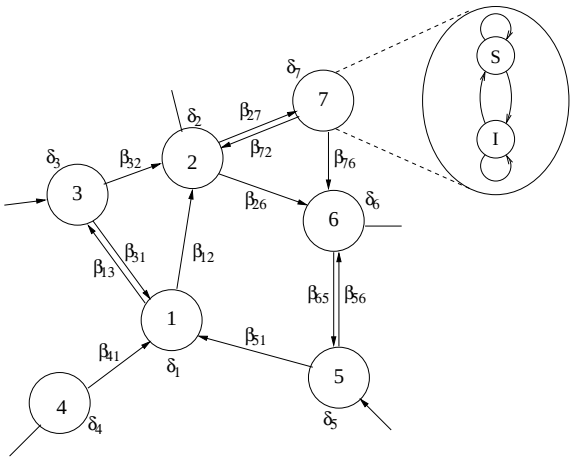
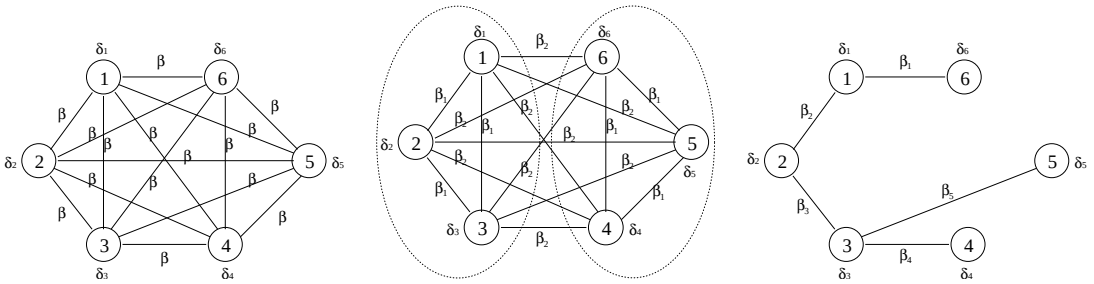


Figure 23: Directed graph (S=Susceptible, I=Infected).

We consider two widely used types of networks in the research of epidemic modeling: *homogeneous networks* and *complex networks* [6]. In a homogeneous network, each node has roughly the same nodal degree. A fully connected topology, a standard hypercubic lattice, and an Erdös-Rényi (ER) random network are three typical examples of homogeneous networks [23]. In a complex network, the nodal degree complies to a particular distribution. A widely studied representative complex network has a power law topology, where the nodal degree distribution is characterized as $P(k) \sim k^{-r}$ with $P(k)$ being the probability that a node has a degree of k [25]. It has been shown that the AS-level Internet topology, WWW networks, and some overlay topologies of peer-to-peer systems can be described by power law characteristics [2, 54]. Moreover, email groups and networks exhibit the power law distribution, which is observed in [80] and [22]. Hence, worm networks with a power law topology can be used to study potential worm propagation on those networks.

7.2.3 Scanning Methods

A worm spreads by employing distinct scanning mechanisms such as random, localized, and topological scanning [59]. Although the nature of each scanning method is different, they can be modeled using the same graphical representation.



(a) Random Scanning. (b) Localized Scanning. (c) Topological Scanning.

Figure 24: Graphical representations of scanning methods.

Random scanning is used by some well-known worms such as Code Red v2 and Sapphire worms. A worm that employs random scanning selects target IP addresses at random. If each IP address is visualized as a network node, random scanning results in a fully connected topology illustrated in Figure 24(a), where the birth rate (β) is identical for every edge.

Localized scanning is used by Code Red II and Nimda worms. Instead of selecting targets randomly, a worm preferentially scans for hosts in the "local" address space. Such a scanning scheme results in a fully connected topology such as the one illustrated in Figure 24(b), where nodes within a group (e.g., IP addresses with the same first two octets) infect one another with the same birth rate (β_1), whereas nodes in different groups infect one another with a different birth rate (β_2).

Topological scanning is used by email viruses and Morris/SSH worms. The worm relies on the information contained in the victim machine to locate new targets. The information may include routing tables, email addresses, a list of peers, and uniform resource locations (URLs). The topological-scanning scheme can result in an arbitrary topology such as an undirected power law topology illustrated in Figure 24(c), where β_i's and δ_i's ($i = 1, 2, \cdots, 5$) represent different birth rates and death rates.

Although only a few topological worms are known, topological scanning is worth investigating for the following reasons. First, the network routing infrastructure, WWW networks, and peer-to-peer systems are vulnerable to topological scanning.

For example, a worm attacking a Web site could look for neighboring Web sites in its URLs and use these Web sites as targets. Second, when IPv4 is upgraded to IPv6, the address space will be much sparser. It would be difficult for either random-scanning or localized-scanning worms to find a target in the IPv6 address space. Therefore, topological scanning may be preferred by attackers. Finally, models of topological-worm propagation would provide insights for the development of countermeasures, which are lacking for such worms.

7.3 Spatial-temporal Model

The problem of modeling worm propagation in networks can be stated as follows: Given a worm network topology, values of β_{ji}'s and δ_i's, and an initial infection node, what is the expected number of infected nodes at time t? To approach this problem, we formulate worm propagation through a spatial-temporal random process based on local interactions of nodes in networks.

Let $X_i(t)$ denote the status of node i at time t, where t represents discrete time, i.e.,

$$X_i(t) = \begin{cases} 1, & \text{if node } i \text{ is infected at time } t; \\ 0, & \text{if node } i \text{ is susceptible at time } t. \end{cases}$$

As node i can be infected only by its neighbors, $X_i(t)$ is statistically dependent on $X_i(t-1)$ and the statuses of its neighbors. Since the status of a neighbor also depends on its own neighbors, conceptually, the statuses of all nodes is statistically dependent in space and time. Let vector $\mathbf{X}(t)$ denote the statuses of all nodes at time t, i.e., $\mathbf{X}(t) = \{X_1(t), X_2(t), \cdots, X_M(t)\}$, where M represents the total number of nodes in the network. $\mathbf{X}(t)$ is then a *spatial-temporal process*.

If node i is susceptible, it can be compromised by any of its infected neighbors, e.g., node j, with a birth rate β_{ji}. Therefore, given the statuses of the neighbors of node i, at the next time step the susceptible node i can get infected with probability $\beta_i(t) = 1 - \prod_{j \in N_i}(1 - \beta_{ji})^{x_j(t)}$, where $x_j(t)$ is the realization of the status of node j

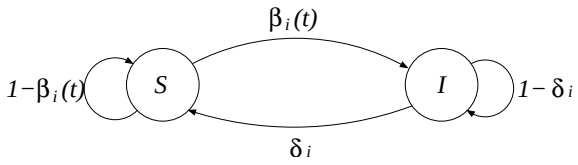


Figure 25: State diagram of a node i.

at time t and $x_j(t) = 0$ or 1. Otherwise, node i is infected and has a death rate δ_i to recover at the next time step. This procedure can be expressed by a Markov chain as in Figure 25. Therefore, the temporal dependence of node i can be shown as

$$P(X_i(t+1) = 0 | X_i(t) = 1) = \delta_i, \tag{100}$$

$$P(X_i(t+1) = 1 | X_i(t) = 0, \mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t)) = \beta_i(t), \tag{101}$$

where vector $\mathbf{X}_{N_i}(t)$ is used to denote the statuses of all neighbors of node i at time t and vector $\mathbf{x}_{N_i}(t)$ is the realization of $\mathbf{X}_{N_i}(t)$, i.e., $\mathbf{X}_{N_i}(t) = \{X_j(t), j \in N_i\}$ and $\mathbf{x}_{N_i}(t) = \{x_j(t), j \in N_i\}$. If for $\forall j$, $\beta_{ji} << 1$, the birth rate (β) is identical for every edge, and the death rate (δ) is identical for every node, then $\beta_i(t) \approx \sum_{j \in N_i} \beta_{ji} x_j(t) = \beta \sum_{j \in N_i} x_j(t)$ and $\delta_i = \delta$, which are assumptions used in the contact process [27].

The probability that node i recovers from the infected to the susceptible status at time $t+1$ is expressed by $R_i(t) = P(X_i(t+1) = 0, X_i(t) = 1)$. Thus, Equation (100) leads to

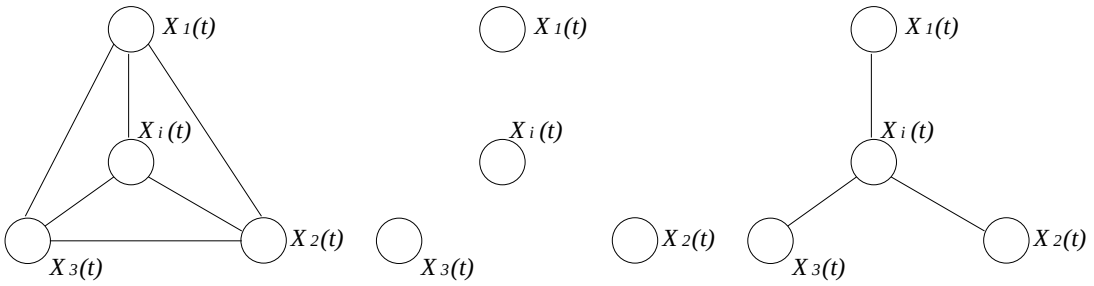
$$R_i(t) = \delta_i P(X_i(t) = 1). \tag{102}$$

Given node i is susceptible at time t, the probability that node i remains susceptible at the next time step can be defined as $S_i(t) = P(X_i(t+1) = 0 | X_i(t) = 0)$. From the local dependence of Equation (101), we have

$$S_i(t) = \sum_{\mathbf{x}_{N_i}(t)} \left[P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t) | X_i(t) = 0)(1 - \beta_i(t)) \right]. \tag{103}$$

Therefore, the definitions of $R_i(t)$ and $S_i(t)$ yield that for $\forall i \in \{1, 2, \cdots, M\}$,

$$P(X_i(t+1) = 1) = 1 - R_i(t) - P(X_i(t) = 0) S_i(t). \tag{104}$$



(a) Joint Probability. (b) Independent Model. (c) Markov Model.

Figure 26: Dependency graph.

Combined with Equations (102) and (103), Equation (104) provides a recursive relationship between $X_i(t+1)$ and $X_i(t)$, $X_j(t)$ for $j \in N_i$, and gives a formal stochastic model. This model explicitly characterizes the spatial and temporal statistical dependencies. In particular, the joint probability $P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t)|X_i(t) = 0)$ characterizes the spatial dependence as a result of network topologies and nodal interactions. The transition probabilities, $\beta_i(t)$ and δ_i, characterize the temporal evolution as a result of infection and recovery. Together, they describe the spatial-temporal process of worm propagation in networks. The expected number of infected nodes at time t, $n(t)$, can be easily computed from $P(X_i(t) = 1)$, i.e., $n(t) = E[\sum_{i=1}^{M} X_i(t)] = \sum_{i=1}^{M} P(X_i(t) = 1)$.

Although in principle Equation (104) can be used to study the behavior of worm propagation, it is challenging to model the spatial dependence. This is because the joint probability $P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t)|X_i(t) = 0)$ is computationally too expensive to obtain, especially when the size of the neighborhood is large. For example, if node i has k neighbors, the total number of statuses needed to describe this joint probability is $O(2^k)$. Therefore, we introduce approximations for the spatial dependence in Sections 7.4 and 7.5. An example of the dependency graph of the joint probability $P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t)|X_i(t) = 0)$ is shown in Figure 26(a), where node i has three neighbors (i.e., nodes 1, 2, 3) and all nodes are dependent on each other.

Remark: It is noted that mean-field methods are used to reduce the computational

complexity involved in typical calculations with multivariate probability distributions when the number of random variables is large [43]. The mean-field methods, however, are difficult to be employed directly to our problem. A typical context for a mean-field approximation is to compute marginal probabilities and expectations of a given joint distribution. Our problem, however, requires to obtain an accurate joint distribution based on the marginal probabilities. Moreover, in many cases the mean-field methods use a set of recursions to find a stationary solution of a corresponding optimization problem in space [65], whereas the topological worm propagation involves both space and time. Although the mean-field methods are currently difficult to be grafted directly to worm propagation problem, the spirit of the mean-field theory motivates our approach for approximating the spatial dependence. For example, the naive mean field assumes that each random variable acts independently and thus approximates the true distribution through a complete factorization [43]. This idea is adopted by our independent model.

7.4 Independent Model

The simplest spatial approximation is to assume independence, resulting in our independent model.

7.4.1 Model

In the independent model, we assume that the statuses of all nodes at time t ($t = 0, 1, 2, \cdots$) is spatially independent. That is,

$$P(\mathbf{X}(t) = \mathbf{x}(t)) = \prod_{i=1}^{M} P(X_i(t) = x_i(t)), \tag{105}$$

where $\mathbf{x}(t)$ is the realization of $\mathbf{X}(t)$, i.e., $\mathbf{x}(t) = \{x_1(t), x_2(t), \cdots, x_M(t)\}$. With the spatial independence assumption, the dependency graph shown in Figure 26(a) is reduced to the graph shown in Figure 26(b), which is a graph with no edges. Thus, the joint probability $P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t)|X_i(t) = 0)$ can be factorized into a form that

only depends on one-node marginal probabilities. This kind of the full factorization is also employed in the naive mean-field approach, where each factor is obtained through the mean-field equations [65].

Theorem 5 (Independent Model) *If the statuses of all nodes at the same time step is spatially independent, the state evolution of node i from Equation (104) satisfies*

$$P(X_i(t+1)=1) = 1 - R_i(t) - P(X_i(t)=0)S_i^{ind}(t), \tag{106}$$

where

$$S_i^{ind}(t) = \prod_{j\in N_i}[1-\beta_{ji}P(X_j(t)=1)]. \tag{107}$$

PROOF: Since the statuses of all nodes at time t is spatially independent, it is true that

$$P(\mathbf{X}_{N_i}(t)=\mathbf{x}_{N_i}(t)|X_i(t)=0) = \prod_{j\in N_i} P(X_j(t)=x_j(t)). \tag{108}$$

With this assumption, it follows from Equation (103) that

$$\begin{aligned} S_i^{\text{ind}}(t) &= \sum_{\mathbf{x}_{N_i}(t)}\prod_{j\in N_i}\left[P(X_j(t)=x_j(t))(1-\beta_{ji})^{x_j(t)}\right] && (109)\\ &= \prod_{j\in N_i}\sum_{x_j(t)}\left[P(X_j(t)=x_j(t))(1-\beta_{ji})^{x_j(t)}\right] && (110)\\ &= \prod_{j\in N_i}[P(X_j(t)=0)+P(X_j(t)=1)(1-\beta_{ji})] && (111)\\ &= \prod_{j\in N_i}[1-\beta_{ji}P(X_j(t)=1)], && (112)\end{aligned}$$

where the exchange of the summation and product signs is because: Set $f(x_j(t)) = P(X_j(t)=x_j(t))(1-\beta_{ji})^{x_j(t)}$ and $j=1,2,\cdots,K$, where K is the number of the

neighborhood of node i; thus

$$\sum_{\mathbf{x}_{N_i}(t)} \prod_{j=1}^{K} f(x_j(t)) = \sum_{x_1(t)} \sum_{x_2(t)} \cdots \sum_{x_K(t)} f(x_1(t)) f(x_2(t)) \cdots f(x_K(t)) \quad (113)$$

$$= \left(\sum_{x_1(t)} f(x_1(t)) \right) \left(\sum_{x_2(t)} f(x_2(t)) \right) \cdots \left(\sum_{x_K(t)} f(x_K(t)) \right) \quad (114)$$

$$= \prod_{j=1}^{K} \sum_{x_j(t)} f(x_j(t)). \quad (115)$$

∎

Such an independent model is intuitive. That is, node j, one of the neighbors of node i, can infect node i with probability $\beta_{ji} P(X_j(t) = 1)$. Thus, the probability that node i cannot be infected by its neighbors at time $t+1$ is $\prod_{j \in N_i} [1 - \beta_{ji} P(X_j(t) = 1)]$, according to the independence assumption. Although ignoring the spatial dependence, the independent model maintains the temporal dependence and the detailed topology information. Moreover, if node i has k neighbors, the total number of statuses needed to describe the joint probability $P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t) | X_i(t) = 0)$ is reduced from $O(2^k)$ to $O(k)$.

Remark: It should be noted that the spatial independence assumption is implicitly used in the prior work [67]. The independent model given here, however, is different from the model proposed in [67] in the following aspects. First, our proposed model is derived from the accurate spatial-temporal process and the explicit approximation on the spatial dependence. Second, our independent model only allows one event (i.e., susceptible → infected or infected → susceptible) in one single discrete time step, whereas the model in [67] grants the concurrence of infection and recovery. Finally, our model focuses on the transient behavior of worm propagation, whereas the model in [67] emphasizes on the steady-state solution and the epidemic threshold.

7.4.2 Performance

How accurate is this independent model? We compare the outcomes of the independent model with those of some well-known models and the simulation results in both homogeneous and complex networks. For simplicity, we consider the special cases where the birth rate (β) is identical for every edge and the death rate (δ) is identical for every node. Such assumptions are used in all previous models. Simulation results provide a benchmark for assessing the accuracy of models. For the simulation, we track each node's status (infected or susceptible) in discrete time. Each simulation has 100 individual runs and is averaged over the cases that the worm survives[1].

7.4.2.1 Homogeneous Networks

In homogeneous networks, the standard Epidemiological model uses a nonlinear differential equation to measure the worm population dynamics [32]:

$$\frac{dn(t)}{dt} = \beta\overline{k}n(t)[1 - \frac{n(t)}{M}] - \delta n(t), \tag{116}$$

where $\overline{k}$ is the average nodal degree. The solution to the above equation is

$$n(t) = \frac{n(0)M(1-\rho)}{n(0) + [M(1-\rho) - n(0)]e^{-(\beta'-\delta)t}}, \tag{117}$$

where $\beta' = \beta\overline{k}$ and $\rho = \frac{\delta}{\beta'}$. Another model used in homogeneous networks is the analytical active worm propagation (AAWP) model, which uses a discrete time equation [8]:

$$n(t+1) = (1-\delta)n(t) + [M - n(t)][1 - (1 - \frac{1}{M})^{sn(t)}], \tag{118}$$

where the scanning rate $s = \beta\overline{k}$ and the patching rate is ignored. Both the Epidemiological model and the AAWP model have been used to model the spread of active worms that employ random scanning and shown to perform accurately if the

[1]We focus on the transient behavior of epidemic worms and ignore the cases that the worms die out.

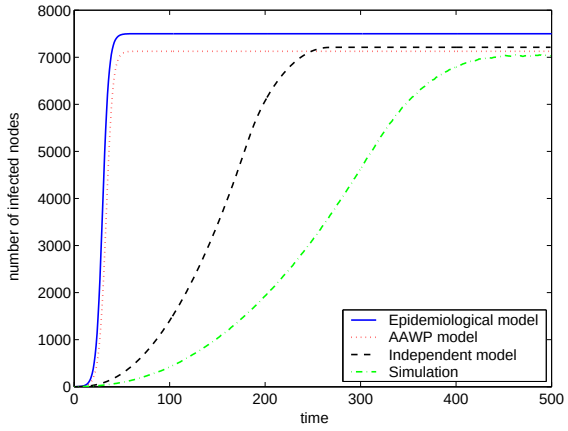


Figure 27: Worm propagation in a two-dimensional lattice with 10,000 nodes, $\beta = 0.1$, and $\delta = 0.1$.

underlying graph is an ER random graph with a large $\overline{k}$ or a fully connected topology [32, 8].

Figure 27 shows the evolution of the average number of infected nodes for the Epidemiological model, the AAWP model, the independent model, and the simulation on a four-neighbor two-dimensional lattice with 10,000 nodes, $\beta = 0.1$, and $\delta = 0.1$. The two-dimensional lattice is wrapped around in both dimensions to form a torus. It is observed that all three models over-predict the growth of infected nodes. The independent model, however, describes the transient behavior of worm propagation better than the other two models.

7.4.2.2 Complex Networks

Boguñá et al. classify complex networks into two types: uncorrelated and correlated complex networks, and present epidemic models for each type [6]. We name these two models as the *uncorrelated complex network (UCN)* model and the *correlated complex network (CCN)* model. In these models, the number of infected nodes with a degree of k at time t, $n_k(t)$, can be described by the following equation [6]:

$$\frac{dn_k(t)}{dt} = \beta k[1 - \frac{n_k(t)}{M_k}]\Theta_k(t) - \delta n_k(t), \tag{119}$$

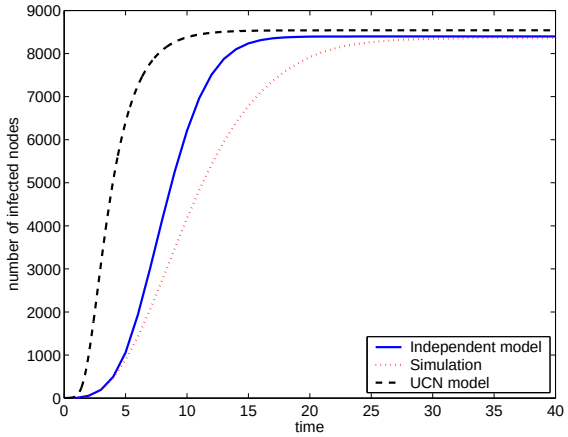


Figure 28: Worm propagation in a BA network with 10,000 nodes, $\overline{k} = 1.9998$, $\beta = 0.5$, and $\delta = 0.1$.

where M_k is the total number of nodes with a degree of k and $\sum_k M_k = M$. In the UCN model, $\Theta_k(t)$ is independent of k and defined as

$$\Theta_k(t) = \frac{M_k}{\overline{k}} \sum_{k'} k' P(k') \frac{n_{k'}(t)}{M_{k'}}; \tag{120}$$

whereas in the CCN model, the effect of the degree k is considered and the expression for $\Theta_k(t)$ is

$$\Theta_k(t) = M_k \sum_{k'} P(k'|k) \frac{n_{k'}(t)}{M_{k'}}. \tag{121}$$

Figure 28 compares the predictions of the independent model against the UCN model in a Barabási-Albert (BA) network, which is a type of power law networks [2]. BA networks are generated using the AS-level BA model in the BRITE simulator [38] that is a tool for topology generation. The BRITE simulator can provide good synthetic topologies that are the base of our simulations. In Figure 28, the BA network has 10,000 nodes, with $\overline{k} = 1.9998$, $\beta = 0.5$, and $\delta = 0.1$. The infection starts at a single node with a degree of 5. Since the BA networks lack correlations [64], we only consider the UCN model for BA networks. It is observed that both the independent model and the UCN model over-predict the spread of worm. When compared with the simulation results, however, the independent model yields a greater accuracy than the UCN model.

An intuitive explanation for the results in Figures 27 and 28 can be given as follows: The Epidemiological model, the AAWP model, and the UCN model express the propagation dynamics in terms of how many nodes are infected, without delving into the details of which nodes are infected [32], whereas the independent model considers the details of how nodes are connected to one another. Therefore, the topology information can help us obtain models better than the previous ones. Moreover, the independent model can be used in arbitrary graphs and with varying β_{ji}'s or δ_i's, whereas the other models are used in special graphs and assume that β_{ji} (or δ_i) is identical for every edge (or node).

7.4.3 Test of the Spatial Independence Assumption

As the independent model achieves a better performance than the previous models, Figures 27 and 28 still show obvious performance gaps between the independent model and the simulation results. Is the spatial independence a good enough assumption? To answer this question, we consider the *correlation coefficient* $\rho_{ij}(t)$ between the statuses of node i and node j, which is defined as

$$\rho_{ij}(t) = \frac{E[X_i(t)X_j(t)] - E[X_i(t)]E[X_j(t)]}{\sqrt{Var[X_i(t)]Var[X_j(t)]}}, \tag{122}$$

where $E[X_i(t)X_j(t)] = P(X_i(t) = 1, X_j(t) = 1)$, $E[X_i(t)] = P(X_i(t) = 1)$, and $Var[X_i(t)] = P(X_i(t) = 1)[1 - P(X_i(t) = 1)]$. If the status of node i is independent of that of node j, $\rho_{ij} = 0$. Otherwise, if the statuses of nodes i, j are positively (or negatively) correlated, $\rho_{ij} > 0$ (or $\rho_{ij} < 0$). We obtain the correlation coefficients through simulation on a four-neighbor two-dimensional lattice with 10,000 nodes, $\beta = 0.1$, $\delta = 0.1$, and 1,000 individual runs. In this two-dimensional lattice, each node is represented by its coordinate (x, y), where x, y are integers and $0 \leq x, y \leq 99$. Node (x, y) has four neighbors $(x - 1, y)$, $(x + 1, y)$, $(x, y - 1)$, and $(x, y + 1)$, where arithmetic operations are *modular* on 100. We assume that the worm begins to spread from node $(0, 0)$ and consider the correlation coefficients between the statuses of node

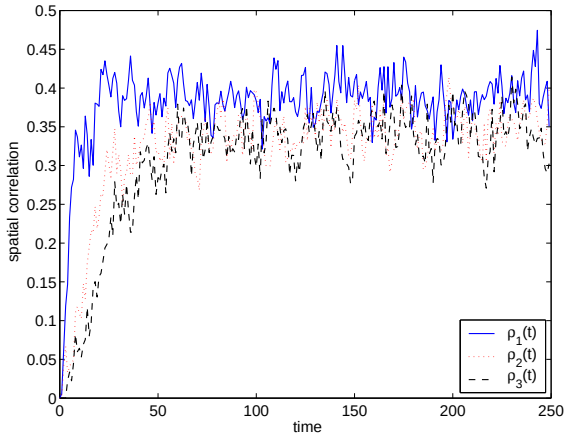


Figure 29: Spatial correlation in a two-dimensional lattice with 10,000 nodes, $\beta = 0.1$, and $\delta = 0.1$.

$(0, 0)$ and node $(0, i)$ (denoted by $\rho_i(t)$) for $i = 1, 2, 3$. Figure 29 shows how the correlation coefficients vary with time. It is observed that the correlation coefficients are initially close to 0, but increase with time. When $t > 50$, all coefficients are larger than 0.25. This shows a strong dependence in space among nodes and suggests a better model that accounts for the spatial dependence.

7.5 Markov Model

7.5.1 Model

Our Markov model assumes a conditional independence in space [35]. That is, at time t ($t = 1, 2, 3, \cdots$), given the status of node i, the statuses of its neighbors is (conditionally) independent,

$$P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t) | X_i(t) = x_i(t)) = \prod_{j \in N_i} P(X_j(t) = x_j(t) | X_i(t) = x_i(t)). \qquad (123)$$

With the spatial Markov assumption, the dependency graph shown in Figure 26(a) is changed to the graph shown in Figure 26(c), where the edges between the neighbors of node i are deleted. The spatial Markov assumption is motivated by the Bethe approximation [74], a way of deriving and correcting the mean-field theory, which has been widely investigated in the area of machine learning. The Bethe approximation

factorizes an exact joint probability distribution into a form that only depends on one-node and two-node marginal probabilities in a Markov network. Moreover, the Bethe approximation is shown to be equivalent to belief propagation in [74]. Here we adopt the spirit of the Bethe approximation by incorporating a simple spatial dependence into the Markov model.

Theorem 6 (Markov Model) *If the statuses of node i's neighbors at the same time step is spatially independent given the status of node i, then the state evolution of node i from Equation (104) satisfies*

$$P(X_i(t+1)=1) = 1 - R_i(t) - P(X_i(t)=0)S_i^{mar}(t), \tag{124}$$

where

$$S_i^{mar}(t) = \prod_{j\in N_i}[1-\beta_{ji}P(X_j(t)=1|X_i(t)=0)]. \tag{125}$$

PROOF: Since the statuses of node i's neighbors at time t is spatially independent given the status of node i, as shown by Equation (123), Equation (103) yields

$$\begin{aligned} S_i^{\text{mar}}(t) &= \sum_{\mathbf{x}_{N_i}(t)}\prod_{j\in N_i}\left[P(X_j(t)=x_j(t)|X_i(t)=0)(1-\beta_{ji})^{x_j(t)}\right] && (126)\\ &= \prod_{j\in N_i}\sum_{x_j(t)}\left[P(X_j(t)=x_j(t)|X_i(t)=0)(1-\beta_{ji})^{x_j(t)}\right] && (127)\\ &= \prod_{j\in N_i}\left[1-\beta_{ji}P(X_j(t)=1|X_i(t)=0)\right]. && (128) \end{aligned}$$

∎

The computation of the conditional probability $P(X_j(t)=1|X_i(t)=0)$ is calculated in the following way. We introduce a two-node joint probability $P(X_i(t)=1, X_j(t)=1)$. Thus,

$$P(X_j(t)=1|X_i(t)=0) = \frac{P(X_j(t)=1) - P(X_i(t)=1, X_j(t)=1)}{P(X_i(t)=0)}. \tag{129}$$

To simply the notation, we set $P_{uv}(t) = P(X_i(t+1)=1, X_j(t+1)=1|X_i(t)=u, X_j(t)=v)$, where $u, v \in \{0,1\}$. The two-node joint probability can be obtained by

the following equations:

$$P\left(X_i(t+1)=1, X_j(t+1)=1\right)=\sum_{u,v}\left[P(X_i(t)=u, X_j(t)=v)P_{uv}(t)\right], \tag{130}$$

where the total probability theorem is used and

$$P_{11}(t)=(1-\delta_i)(1-\delta_j), \tag{131}$$

since given that both node i and node j are infected at time t, they independently choose to stay in the *infected* status;

$$P_{01}(t)=(1-\delta_j)[1-S_i'(t)], \tag{132}$$

in that $S_i'(t)=P\left(X_i(t+1)=0|X_i(t)=0, X_j(t)=1\right)$ and thus

$$S_i'(t)=(1-\beta_{ji})\prod_{l\in N_i-\{j\}}[1-\beta_{li}P(X_l(t)=1|X_i(t)=0)], \tag{133}$$

where the spatial Markov assumption is used; similarly,

$$P_{10}(t)=(1-\delta_i)[1-S_j'(t)], \tag{134}$$

in that

$$S_j'(t)=(1-\beta_{ij})\prod_{l\in N_j-\{i\}}[1-\beta_{lj}P(X_l(t)=1|X_j(t)=0)]; \tag{135}$$

$$P_{00}(t)\approx[1-S_i''(t)][1-S_j''(t)], \tag{136}$$

where

$$S_i''(t)=\prod_{l\in N_i-\{j\}}[1-\beta_{li}P(X_l(t)=1|X_i(t)=0)], \tag{137}$$

$$S_j''(t)=\prod_{l\in N_j-\{i\}}[1-\beta_{lj}P(X_l(t)=1|X_j(t)=0)]. \tag{138}$$

Equation (136) uses an approximation to avoid the introduction of a three-node joint probability $P(X_i(t)=0, X_j(t)=0, X_l(t)=x_l(t))$ if nodes i, j, l construct a triangle. Equation (130) is obtained by replacing $P_{uv}(t)$ with the results from Equations (131) $\sim$ (136). Equations (124) and (130) provide a recursive relationship between $(X_i(t+1),$

$X_j(t+1))$ and $(X_i(t),\ X_j(t))$ for $j \in N_i$. It is assumed that the statuses of all nodes are independent at time 0.

The Markov model takes into account a part of the neglected correlations between random variables (i.e., node i and its neighbors at time t) and thus improves the approximation. The Markov model differs from the independent model only in the probability that one of node i's neighbors infects node i. This probability is $\beta_{ji}P(X_j(t)=1|X_i(t)=0)$ for the Markov model, whereas it is $\beta_{ji}P(X_j(t)=1)$ for the independent model. If the dependence between node i and its neighbors is ignored, the Markov model is reduced to the independent model. Moreover, with the spatial Markov assumption, if node i has k neighbors, the total number of statuses needed to describe the joint probability $P(\mathbf{X}_{N_i}(t)=\mathbf{x}_{N_i}(t)|X_i(t)=0)$ is $O(k)$.

Is it always beneficial to incorporate the spatial dependence? We investigate this issue by introducing the notion of *association* defined in [24].

Definition 1 *Random variables $T_1, \cdots, T_n$ are associated if*

$$Cov[f(\mathbf{T}), g(\mathbf{T})] = E[f(\mathbf{T})g(\mathbf{T})] - E[f(\mathbf{T})]E[g(\mathbf{T})] \geq 0 \tag{139}$$

for all nondecreasing functions f and g for which $E[f(\mathbf{T})]$, $E[g(\mathbf{T})]$, and $E[f(\mathbf{T})g(\mathbf{T})]$ exist, and $\mathbf{T} = \{T_1, \cdots, T_n\}$.

In most cases, if one neighbor of node i, e.g., node j, is infected, node i then has an increasing probability to be infected. That is, node i and node j are positively correlated as shown in Figure 29. Therefore, the statuses of nodes i and j, $X_i(t)$ and $X_j(t)$, are associated by definition. Furthermore, if $X_i(t)$ and $\mathbf{X}_{N_i}(t)$ are associated random variables, we can show in the following theorem that the Markov model indeed achieves a better performance than the independent model.

Theorem 7 (Performance Bound) *If $X_i(t)$ and $\mathbf{X}_{N_i}(t)$ are associated, then*

$$S_i^{ind}(t) \leq S_i^{mar}(t) \leq S_i(t). \tag{140}$$

PROOF: Since $X_i(t)$ and $X_j(t)$ ($j \in N_i$) are *associated*, $Cov[X_i(t), X_j(t)] \geq 0$. We can write

$$P(X_i(t) = 1, X_j(t) = 1) \geq P(X_i(t) = 1) \cdot P(X_j(t) = 1), \tag{141}$$

which leads to

$$P(X_j(t) = 1) \geq P(X_j(t) = 1 | X_i(t) = 0). \tag{142}$$

Therefore, $S_i^{\text{ind}}(t) \leq S_i^{\text{mar}}(t)$.

Given $X_i(t) = 0$, let $f(\mathbf{X}_{N_i}(t)) = -(1 - \beta_{li})^{X_l(t)}$ and $g(\mathbf{X}_{N_i}(t)) = -\prod_{j \in N_i - \{l\}}(1 - \beta_{ji})^{X_j(t)}$, where $l \in N_i$. Since $\mathbf{X}_{N_i}(t)$ are associated, from the definition of association we have

$$Cov\left[f(\mathbf{X}_{N_i}(t)),\ g(\mathbf{X}_{N_i}(t)) | X_i(t) = 0\right] \geq 0, \tag{143}$$

which leads to

$$E\left[f(\mathbf{X}_{N_i}(t)) | X_i(t) = 0\right] \cdot E\left[g(\mathbf{X}_{N_i}(t)) | X_i(t) = 0\right] \leq E\left[f(\mathbf{X}_{N_i}(t)) g(\mathbf{X}_{N_i}(t)) | X_i(t) = 0\right]. \tag{144}$$

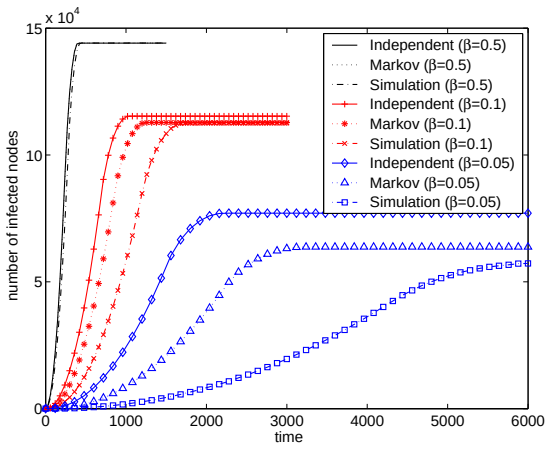
The repeated use of the above argument yields

$$\prod_{j \in N_i} E\left[(1 - \beta_{ji})^{X_j(t)} | X_i(t) = 0\right] \leq E\left[\prod_{j \in N_i} (1 - \beta_{ji})^{X_j(t)} | X_i(t) = 0\right]. \tag{145}$$

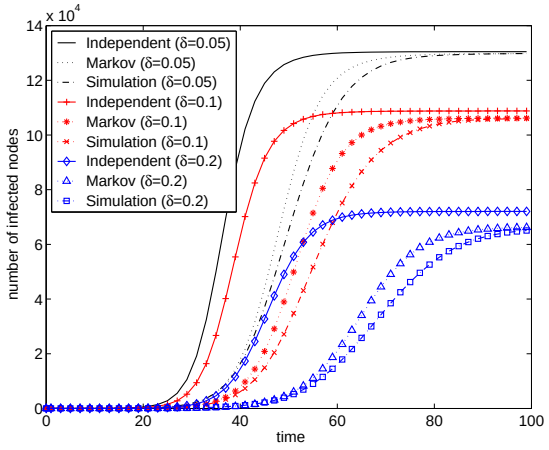
That is, $S_i^{\text{mar}} \leq S_i(t)$. ∎

7.5.2 Performance

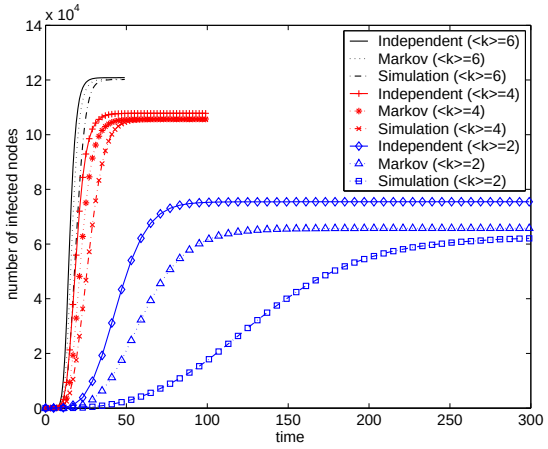
How much does the spatial Markov dependence help in improving the performance? We compare the performance of our proposed models with the simulation results in a two-dimensional lattice, an ER random graph, a BA power law network, a real topology, and a top-down hierarchical topology. Except for the two-dimensional lattice, which is a regular graph, we begin each simulation and models with a single, randomly chosen infected node on a given topology. Each plot considers 10 different initially infected nodes, and each simulation plot also has 10 individual runs for an initially infected node.



(a) A two-dimensional lattice with 160,000 nodes and $\delta = 0.1$.



(b) An ER random graph with 160,000 nodes, $\overline{k} = 4$, and $\beta = 0.1$.



(c) A BA network with 160,000 nodes, $\beta = 0.1$, and $\delta = 0.1$.

Figure 30: Worm propagation in different topologies.

Figure 30(a) compares the predictions of the independent model and the Markov model with the simulation results on a four-neighbor two-dimensional lattice. The number of nodes $M = 160,000$, and the death rate $\delta = 0.1$. For the case of the birth rate $\beta = 0.5$, the three curves nearly coincide with each other. When β decreases, however, the infection spreads at a faster rate in both the independent and the Markov models than the simulation. In all three cases ($\beta = 0.5$, $\beta = 0.1$, $\beta = 0.05$), the Markov model yields more accurate results than the independent model.

Figure 30(b) shows the predictions of two models with the simulation results on an

ER random graph, with 160,000 nodes, an average nodal degree $\overline{k} = 4$, and $\beta = 0.1$. When we constructed the ER random graph, the generated graph was disconnected for that $\overline{k}$ is small. Therefore, in this disconnected graph we chose the largest cluster with 156,763 nodes and an average degree of 4.07 as the target network. It can be seen that the Markov model yields a far better performance than the independent model when compared with the simulation results.

Figure 30(c) depicts the simulation results against two models on a BA network, with 160,000 nodes, $\beta = 0.1$, $\delta = 0.1$, and $< k >= \overline{k}$. For the case when $\overline{k} = 6$, both models give precise results. When $\overline{k}$ decreases, however, the predictions of both models become worse. In all three cases ($\overline{k} = 6$, $\overline{k} = 4$, $\overline{k} = 2$), the Markov model predicts worm propagation more accurately than the independent model.

It is observed that the parameters can affect the accuracy of the models. When β or $\overline{k}$ is large, both the independent model and the Markov model perform well. When both β and $\overline{k}$ are small, however, both models fail to predict the slow growth of worm propagation. Therefore, both models are suited for dense graphs, where each node fluctuates independently about its mean value. On the other hand, the Markov model outperforms the independent model in all cases with different parameters and underlying topologies. That is, Theorem 7 is confirmed by the results shown in Figure 30.

Another observation is that the underlying topology can affect the speed of worm propagation and the final size of infection. For the case of $\beta = 0.1$ and $\delta = 0.1$, although all three graphs (Figure 30) have the same number of nodes and edges, the worm spreading dynamics in these graphs are significantly different. It takes the worm about 1,716 time steps to enter an equilibrium stage in the two-dimensional lattice, whereas it needs about 100 time steps and 66 time steps in the ER random graph and the BA network, respectively. Moreover, after entering the equilibrium stage, the worm infects a total of 112,506 nodes in the two-dimensional lattice, 106,023 nodes in

the ER random graph, and 105,511 nodes in the BA network. This shows the effect of network structures on the dynamics of worm propagation.

Figure 31 shows worm propagation in a real topology, an ER random graph, a BA network, and a four-neighbor two-dimensional lattice for the special case when $\beta = 1$ and $\delta = 0.1$. The real topology is an AS graph collected at the Oregon router server route-views.oregon-ix.net, which is a site for collecting BGP data [91]. The dataset is selected on 1 June 2004 and contains 38,086 links among 17,653 ASes ($\overline{k} = 4.3$). The constructed ER random graph has a largest cluster with 17,648 nodes and $\overline{k} = 4.3$, and the worm only propagates in this largest cluster. The BA network has 17,652 links among 17,653 nodes ($\overline{k} = 2$). The generated BA network is connected and thus is a tree. The two-dimensional lattice is with 17,689 nodes and $\overline{k} = 4$. Among all these four topologies, the curves of both models overlap with the simulation results in this special case. Therefore, both models can achieve the best performance in the case of $\beta = 1$. Although the AS graph and the ER random graph have almost the same number of (connected) nodes and the average nodal degree, the worm takes only 6 time steps to enter an equilibrium stage in the AS graph, whereas it needs about 9 time steps in the ER random graph. This shows that these two topologies have different diameters and the AS graph is more vulnerable to worm propagation than the ER random graph. It is interesting to notice that although the dynamics of worm spreading in different topologies are distinct, the final sizes of infection are almost the same, i.e., $n(t) \approx 16,000$, when $t = 150$. This reflects that for the case when $\beta = 1$ and $\delta = 0.1$, the final size of infection is not dependent on the network structure, but on the total number of nodes.

Figure 32 demonstrates another special case when $\delta = 0$, which corresponds to the susceptible $\rightarrow$ infected (SI) model. The worm spreads in a top-down hierarchical topology generated by BRITE [38]. The top AS-level topology is from NLANR on 2

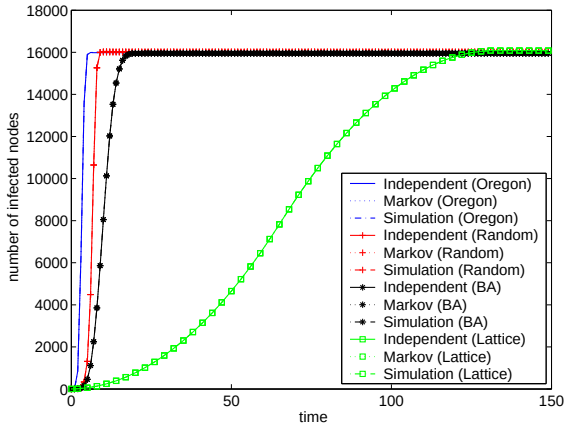


Figure 31: Worm propagation in a real topology, an ER random graph, a BA network, and a two-dimensional lattice with $\beta = 1$ and $\delta = 0.1$.

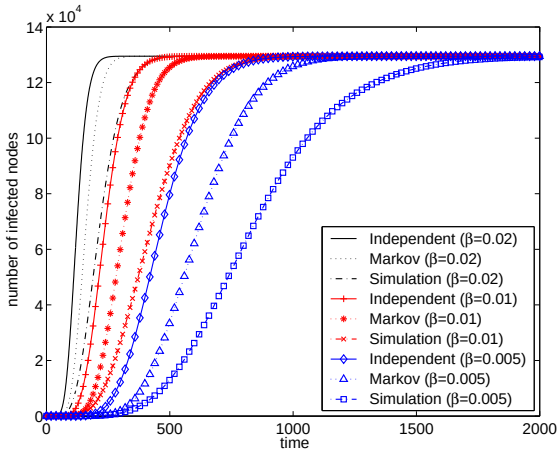


Figure 32: Worm propagation in a top-down hierarchical topology with 129,480 nodes, 266,005 edges, and $\delta = 0$.

January 2000 [89], with 6,474 ASes and 13,895 interconnections. The down router-level topology is generated by the BRITE router-level BA model, with 20 nodes per AS. The constructed top-down hierarchical topology has 129,480 nodes and 266,005 edges. The merit of the Markov model can also be observed in this special case when $\delta = 0$.

7.5.3 Test of the Spatial Markov Assumption

To further examine the goodness of the spatial Markov assumption, we use a *relative entropy* (or *Kullback-Leibler distance*) between two probability mass functions $p(x, t)$

and $q(x,t)$ as defined in [20]:

$$D(p\|q) = \sum_x p(x,t) log \frac{p(x,t)}{q(x,t)}. \tag{146}$$

The relative entropy is a measure of the distance between two distributions $p(x,t)$ and $q(x,t)$. If $q(x,t)$ is "closer" to $p(x,t)$, $D(p\|q)$ is smaller; and $D(p\|q) = 0$ if and only if $p = q$.

For our case, $p(x,t) = P(\mathbf{X}_{N_i}(t) = \mathbf{x}_{N_i}(t)|X_i(t) = 0)$ is the joint distribution of the statuses of node i's neighbors given node i is susceptible at time t. For the independent model, $q_1(x,t) = \prod_{j \in N_i} P(X_j(t) = x_j(t))$; for the Markov model, $q_2(x,t) = \prod_{j \in N_i} P(X_j(t) = x_j(t)|X_i(t) = 0)$. We obtain the relative entropies $D(p\|q_1)$ and $D(p\|q_2)$ through simulation on a four-neighbor two-dimensional lattice with 10,000 nodes, $\beta = 0.1$, $\delta = 0.1$, and 1,000 individual runs. As described in Section 7.4.3, each node is represented by its coordinate, and the worm begins to spread from node $(0,0)$. Node i is specified at $(1,1)$. Figure 33 shows how the relative entropies $D(p\|q_1)$ and $D(p\|q_2)$ change with time. It is observed that the relative entropies are initially close to 0, but increase with time. $D(p\|q_2)$ is smaller than $D(p\|q_1)$ for all time t, suggesting that the spatial Markov model is indeed a better approximation than the spatial independent model. On the other hand, when $t > 60$, $D(p\|q_2) > 0.5$. This explains the performance gap between the Markov model and the simulation observed in Figures 30 and 32. Hence, a model that incorporates the more spatial dependence than the Markov model may result in a smaller relative entropy.

7.6 Final Size of Infection

The final size of infection corresponds to the equilibrium state of a worm network that is the average number of infected nodes when time t approaches infinity, i.e., $\lim_{t \to +\infty} n(t)$. The final size of infection characterizes the potential damage as a result of worm propagation. If the final size of infection can be predicted at an early

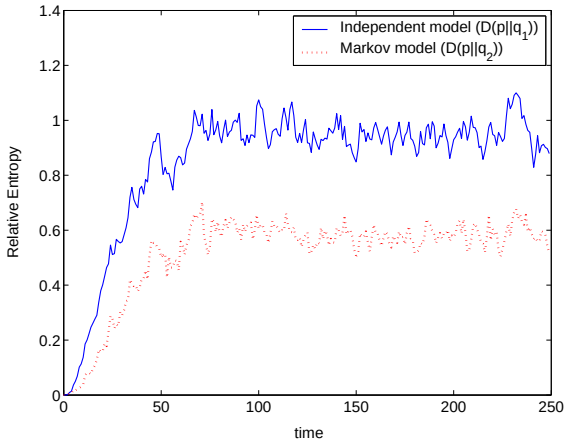
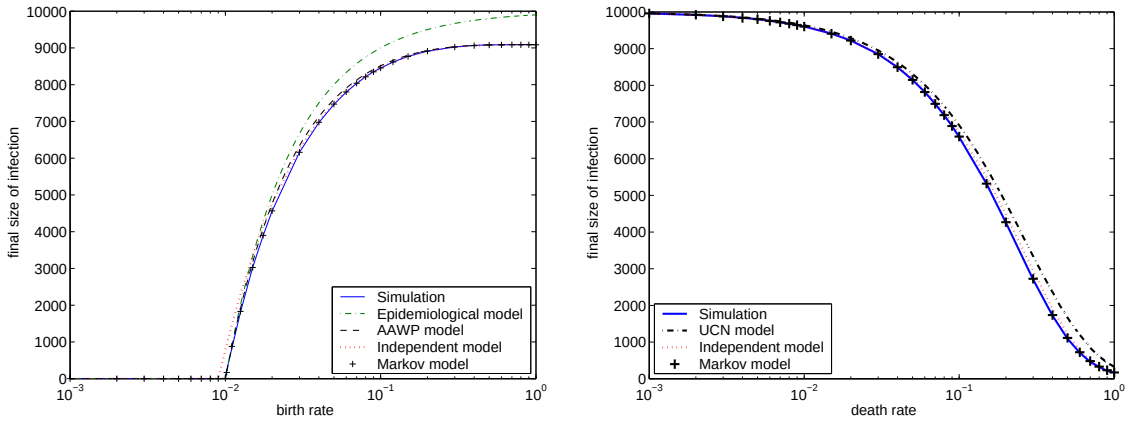


Figure 33: Relative entropies in a two-dimensional lattice with 10,000 nodes, $\beta = 0.1$, and $\delta = 0.1$.

stage of worm spreading, the potential damage can be assessed, and preventive actions can be taken accordingly. In this section, we compare our proposed models with the simulation results and the other models in estimating the final size of infection in homogeneous and complex networks. Each simulation scenario has 100 individual runs and is averaged over the cases that worms survive. The final size of infection is sampled at time $t = 2000$.

Figure 34(a) shows a comparison of the Epidemiological model, the AAWP model, the independent model, the Markov model, and the simulation results on a connected ER random graph with 10,000 nodes, $\overline{k} = 10$, and $\delta = 0.1$. When compared with the simulation results, the Epidemiological model over-predicts the final size of infection when $\beta \geq 0.02$, whereas the AAWP model and the independent model slightly over-predict it. The results of the Markov model and the simulation overlap for $0.001 \leq \beta \leq 1$. Therefore, the Markov model is the most accurate one among all these models.

Figure 34(b) gives another comparison of the UCN model, the independent model, the Markov model, and the simulation results on a BA network with 10,000 nodes, $\overline{k} = 4$, and $\beta = 0.1$. The UCN model over-predicts the final size of infection, whereas the independent model slightly over-predicts it. The results of the Markov model and



(a) An ER random graph with 10,000 nodes, $\overline{k} = 10$, and $\delta = 0.1$.

(b) A BA network with 10,000 nodes, $\overline{k} = 4$, and $\beta = 0.1$.

Figure 34: Performance comparisons in estimating the final size of infection.

the simulation overlap for $0.001 \leq \delta \leq 1$. Therefore, both the independent model and the Markov model are shown to be good estimators of the final size of infection, and the Markov model is more accurate than the independent model.

7.7 Summary

In this chapter, we have presented a spatial-temporal model to study the dynamic spreading of worms that employ different scanning methods. Making use of this model, we have studied the impact of the underlying topology on worm propagation. We show that the detailed topology information and the spatial dependence are key factors in modeling the spread of worms. The independent model incorporates the detailed topology information and thus outperforms the previous models. Our Markov model incorporates both the detailed topology information and the simple spatial dependence, and thus achieves a greater accuracy than the independent model, especially when both the birth rate and the average nodal degree are small. Moreover, when the graph is dense, each node fluctuates independently about its mean value, and thus both models perform well. These results are validated through analysis and extensive simulations on large networks using real and synthesized topologies.

The class of models we have investigated are biased, i.e., with a reduced complexity. Hence, the accuracy of such models is important. The relative entropy is used as a performance measure and shows that a performance gap still exists between the Markov model and the reality. Formulations are needed to incorporate the more spatial dependence into the model. Furthermore, as both models are motivated by the spirit of the mean-field approximation in machine learning, a formal treatment of the mean-field approximation to include the temporal dependence will be studied in our future work. As part of the ongoing work, we also plan to estimate the parameters of worm propagation (e.g., the birth rate and the death rate) and use our proposed models to study the countermeasures for controlling the spread of worms. Our modeling approach may also help to understand a wide range of information propagation behaviors in Internet, such as BGP update streams and file sharing in peer-to-peer applications.

CHAPTER VIII

CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

8.1 Research Contributions

In this thesis, the research on characterizing worm attack behaviors, analyzing Internet vulnerabilities, and developing effective countermeasures has been conducted. Research contributions have been made in the following areas:

1. Designing an optimal worm-scanning method.
2. Analyzing two sub-optimal worm-scanning methods.
3. Evaluating the vulnerability of the Internet.
4. Modeling the spread of topological-scanning worms.

8.1.1 Designing an Optimal Worm-Scanning Method

Most Internet worms use random scanning. The distribution of vulnerable hosts on the Internet, however, is highly non-uniform over the IP-address space. This implies that random scanning wastes many scans on invulnerable addresses and more virulent scanning schemes may take advantage of the non-uniformity of a vulnerable-host distribution. Questions then arise as to how attackers may exploit such information and how virulent the resulting worm may be. These issues provide "worst-case scenarios" for defenders and "best-case scenarios" for attackers when the vulnerable-host distribution is available.

In Chapter 3, a new worm-scanning method, called *importance scanning*, is designed. Important scanning results from importance sampling in statistics and scans the IP-address space according to an empirical distribution of vulnerable hosts. An

analytical model is developed to relate the infection rate of worms with the importance-scanning strategies. Based on parameters chosen from Witty and Code Red worms, the experimental results show that an importance-scanning worm can spread much faster than either a random-scanning worm or a routing worm. In addition, a game-theoretical approach suggests that the best strategy for defenders is to scatter applications uniformly in the entire IP-address space.

8.1.2 Analyzing Two Sub-Optimal Worm-Scanning Methods

The use of side information by an attacker can help a worm speed up the propagation. This philosophy has been the basis for advanced worm-scanning mechanisms such as hitlist scanning, routable scanning, and importance scanning. Some of these scanning methods use information on vulnerable hosts. Such information, however, may not be easy to collect before a worm is released. Questions then arise whether and how a worm can self-learn or exploit such information while propagating; and how virulent the resulting worms may be. As an optimal scanning strategy is difficult to implement, two practical sub-optimal scanning methods are investigated.

In Chapter 4, a self-learning worm using the static importance scanning and the botnet structure is designed and studied. The self-learning worm is demonstrated analytically and empirically to have the ability to accurately estimate the underlying vulnerable-host distribution in /8 subnets when only 500 infected hosts are observed. Experimental results based on parameters chosen from Code Red v2 and Witty worms show that a self-learning worm can spread much faster than a random-scanning worm, a permutation-scanning worm, and a Class-A routing worm. Furthermore, some guidelines for detecting and defending against such self-learning worms are also discussed.

In Chapter 5, localized scanning, a simple yet effective technique used by attackers to search for vulnerable hosts, is studied. Localized scanning trades off between the

local and the global search of vulnerable hosts and has been used by Code Red II and Nimda worms. First, the relationships between vulnerable-host distributions and the spread of localized-scanning worms are characterized through mathematical modeling and analysis. Then, an optimal localized-scanning strategy that provides an upper bound on the spreading speed of localized-scanning self-propagating codes is designed. Furthermore, three variants of localized scanning are constructed. Specifically, the feedback localized scanning and the ping-pong localized scanning adapt the scanning methods based on the feedback from the probed host, and thus spread faster than the original localized scanning and meanwhile have a smaller variance.

8.1.3 Evaluating the Vulnerability of the Internet

In Chapter 6, three aspects are investigated jointly: (a) a *network vulnerability* as the non-uniform vulnerable-host distribution, (b) *threats*, i.e., intelligent worms that exploit such a vulnerability, and (c) *defense*, i.e., challenges for fighting the threats. First, five data sets are studied, and consistent clustered vulnerable-host distributions are observed. Then, a new metric, referred to as the *non-uniformity factor*, is presented. The non-uniformity factor quantifies the unevenness of a vulnerable-host distribution. This metric is essentially the Renyi information entropy and better characterizes the non-uniformity of a distribution than the Shannon entropy. Next, the infection rate and the propagation speed of network-aware worms are measured analytically and empirically. A representative network-aware worm is shown to increase the spreading speed by exactly or nearly a non-uniformity factor when compared to a random-scanning worm at the early stage of worm propagation. This implies that when a worm exploits an uneven vulnerable-host distribution as a network-wide vulnerability, the Internet can be infected much more rapidly. Furthermore, the effectiveness of defense strategies on the spread of network-aware worms is analyzed.

Experimental results demonstrate that counteracting network-aware worms is a significant challenge for the strategies that include host-based defense and IPv6.

8.1.4 Modeling the Spread of Topological-Scanning Worms

Topology information is a fundamental element that enables topological-scanning worms, such as the Morris worm. The spread of topological-scanning worms, however, is especially hard to model. The difficulty lies in characterizing the impact of topologies and the interactions among nodes in both space and time.

In Chapter 7, the spread of topological worms is modeled. Our model is motivated by probabilistic graphs, which have been widely investigated in machine learning. First, a graphical representation is used to abstract the propagation of worms that employ different scanning methods. Then, a spatial-temporal random process is presented to describe the statistical dependence of worm propagation in arbitrary topologies. As the spatial dependence is particularly difficult to characterize, the problem becomes how to use simple (i.e., biased) models to approximate the spatially dependent process. In particular, the independent model and the Markov model are proposed as simple approximations. Both theoretical analysis and extensive simulations on large networks using both real measurements and synthesized topologies are conducted to test the performance of the proposed models. Our results show that the independent model can capture the temporal dependence and the detailed topology information and thus outperforms the previous models, whereas the Markov model incorporates a certain spatial dependence and thus achieves a greater accuracy in characterizing both the transient and equilibrium behaviors of worm propagation.

8.2 Future Research Directions

- **Worm Defense System Design and Analysis:** An effective yet practical worm defense system is important and urgent. Since some areas of the Internet are responsible for a disproportionate number of vulnerable hosts, the problem is

how to adequately secure these areas. A collaboration system among firewalls will be developed, exploiting the vulnerable-host distribution. The following important questions will be answered: How can firewalls cooperate with each other to block worm traffic effectively? How do firewalls treat traffic differently based on where the traffic is generated? How can the system defeat the malicious firewall(s)?

- **Worm Tomography:** Internet security and resilience require methods to detect and estimate worm behaviors. Most worms use random scanning to recruit new bots, which can be observed by *Darknet* that is defined as a globally routable address space in which no active services or servers reside. The term, *worm tomography*, is used to describe the process of inferring the characteristics of worms from Darknet observations. The primary difficulty in using this approach lies in the growth trend of background noise [51].

- **Malicious Sources Analysis:** It is important that defenders identify Internet areas that are responsible for a significant portion of attacks. We have obtained 402-day traces from DShield [84]. We attempt to discover how the malicious sources distribute across the Internet and how they change with time.

- **Game Theory Between Attackers and Defenders:** The evolution of the Internet has created a real arms race between attackers and defenders. Defenders attempt to understand the skills that attackers use and develop systems against them. On the other hand, attackers endeavor to learn the weakness of the systems that defenders build and design new methods to overcome or evade these systems. Both attackers and defenders can eventually learn about the opponent's strategies and design the optimal tactics. Therefore, this interaction naturally leads to a game theory framework between attackers and defenders. A simple game theory approach has been applied to the optimal worm-scanning

attack and defense described in this thesis. The more complex interaction in the game, however, should be considered in the future research.

REFERENCES

[1] H. Andersson and T. Britton, "Stochastic epidemic models and their statistical analysis," in *Lecture Notes in Statistics*. New York: Springer-Verlag, 2000, vol. 151.

[2] A. Barabási and R. Albert, "Emergence of scaling in random networks," *Science 286*, 1999, pp. 509-512.

[3] P. Barford, R. Nowak, R. Willett, and V. Yegneswaran, "Toward a model for sources of Internet background radiation," in *Proc. of the Passive and Active Measurement Conference (PAM'06)*, Mar. 2006.

[4] S. M. Bellovin, B. Cheswick, and A. Keromytis, "Worm propagation strategies in an IPv6 Internet," *;login:*, vol. 31, no. 1, Feb. 2006, pp. 70-76.

[5] P. J. Bickel and K. A. Doksum, *Mathematical Statistics: Basic Ideas and Selected Topics, Vol I (2nd Edition)*. New Jersey: Prentice-Hall, 2001.

[6] M. Boguñá, R. Pastor-Satorras, and A. Vespignani, "Epidemic spreading in complex networks with degree correlations," in *Statistical Mechanics of Complex Networks*, R. Pastor-Satorras et al., Eds., 2003, pp. 127-147.

[7] D. Brumley, L. Liu, P. Poosankam, and D. Song, "Design space and analysis of worm defense strategies," in *ACM Symposium on InformAtion, Computer and Communications Security (ASIACCS)*, Mar. 2006.

[8] Z. Chen, L. Gao, and K. Kwiat, "Modeling the spread of active worms," in *Proc. of INFOCOM'03*, vol. 3, San Francisco, CA, Apr. 2003, pp. 1890-1900.

[9] Z. Chen, L. Gao, and C. Ji, "On effectiveness of defense systems against active worms," Tech. Rep., 2003.

[10] Z. Chen and C. Ji, "Optimal worm-scanning method using vulnerable-host distributions," *to appear in the International Journal of Security and Networks: Special Issue on Computer and Network Security*, 2007.

[11] Z. Chen and C. Ji, "Measuring network-aware worm spreading ability," in *Proc. of INFOCOM'07*, Anchorage, AK, May 2007.

[12] Z. Chen, C. Chen, and C. Ji, "Understanding localized-scanning worms," in *Proc. of IPCCC'07*, New Orleans, LA, April 2007.

[13] Z. Chen and C. Ji, "Intelligent worms: searching for preys," *Mathematics Awareness Month: Mathematics and Internet Security Theme Essays*, 2006.

[14] Z. Chen, "Worm propagation models," *Mathematics Awareness Month: Mathematics and Internet Security Theme Essays*, 2006.

[15] Z. Chen and C. Ji, "Spatial-temporal modeling of malware propagation in networks," *IEEE Transactions on Neural Networks: Special Issue on Adaptive Learning Systems in Communication Networks*, vol. 16, no. 5, Sept. 2005, pp. 1291-1303.

[16] Z. Chen and C. Ji, "A self-learning worm using importance scanning," in *Proc. ACM/CCS Workshop on Rapid Malcode (WORM'05)*, Fairfax, VA, Nov. 2005, pp. 22-29.

[17] Z. Chen and C. Ji, "Importance-scanning worm using vulnerable-host distribution," in *Proc. 48th Ann. IEEE Global Telecommunications Conference (GLOBECOM'05)*, St. Louis, MO, Nov. 2005.

[18] E. Cooke, F. Jahanian, and D. McPherson, "The zombie roundup: understanding, detecting, and disrupting botnets," in *Proc. of Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI'05)*, Boston, MA, 2005.

[19] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. IT-13, no. 1, Jan. 1967, pp. 21-27.

[20] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[21] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, 2001.

[22] H. Ebel, L. Mielsch, and S. Bornholdt, "Scale-free topology of e-mail networks," in *Phys. Rev. E 66*, 2002, 035103(R).

[23] P. Erdös and A. Rényi, "On the evolution of random graphs," in *Publ. Math. Inst. Hung. Acad. Sci.*, Vol. 5, 1960, pp. 17-61.

[24] J. D. Esary, F. Proschan, and D. W. Walkup, "Association of random variables, with applications," in *Ann. Math. Statist.*, vol. 38, no. 5, Oct. 1967, pp. 1466-1474.

[25] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the Internet topology," in *Proc. of ACM SIGCOMM'99*, 1999, pp. 251-262.

[26] H. Feng, A. Kamra, V. Misra, and A. D. Keromytis, "The effect of DNS delays on worm propagation in an IPv6 Internet," in *Proc. of INFOCOM'05*, vol. 4, Miami, FL, Mar. 2005, pp. 2405-2414.

[27] A. Ganesh, L. Massoulie, and D. Towsley, "The effect of network topology on the spread of epidemics," in *Proc. of INFOCOM'05*, vol. 2, Miami, FL, Mar. 2005, pp. 1455-1466.

[28] M. Garetto, W. Gong, and D. Towsley, "Modeling malware spreading dynamics," in *Proc. of INFOCOM'03*, vol. 3, San Francisco, CA, Apr. 2003, pp.1869-1879.

[29] G. Gu, Z. Chen, P. Porras, and W. Lee, "Misleading and defeating importance-scanning malware propagation," submitted for publication, 2007.

[30] G. Gu, M. Sharif, X. Qin, D. Dagon, W. Lee, and G. Riley, "Worm detection, early warning and response based on local victim information," in *Proc. 20th Ann. Computer Security Applications Conf. (ACSAC'04)*, Tucson, AZ, Dec. 2004.

[31] P. Heidelberger, "Fast simulation of rare events in queueing and reliability models," *ACM Transactions on Modeling and Computer Simulation*, vol.5, no.1, Jan. 1995, pp. 43-85.

[32] J. O. Kephart and S. R. White, "Directed-graph epidemiological models of computer viruses," in *Proc. of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy*, 1991, pp. 343-359.

[33] M. Kodialam and T. V. Lakshman, "Detecting network intrusions via sampling: a game theoretic approach," in *Proc. of IEEE INFOCOM'03*, San Francisco, CA, Apr., 2003, pp. 1880-1889.

[34] E. Kohler, J. Li, V. Paxson, and S. Shenker, "Observed structure of addresses in IP traffic," in *ACM SIGCOMM Internet Measurement Workshop*, Marseille, France, Nov. 2002.

[35] S. L. Lauritzen, *Graphical Models*. London, U.K.: Oxford Univ. Press, 1996.

[36] T. M. Liggett, *Interacting Particle Systems*. New York: Springer-Verlag, 1985.

[37] M. Locasto, J. Parekh, S. Stolfo, A. Keromytis, T. Malkin, and V. Misra, "Collaborative distributed intrusion detection," Dept. Computer Science, Columbia Univ., Tech. Rep. CUCS-012-04, 2004.

[38] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: universal topology generation from a user's perspective," Tech. Rep. (User Manual) BU-CS-TR-2001-003, Apr. 2001.

[39] D. Moore, C. Shannon, and J. Brown, "Code-Red: a case study on the spread and victims of an Internet worm," in *ACM SIGCOMM/USENIX Internet Measurement Workshop*, Marseille, France, Nov. 2002.

[40] D. Moore, C. Shannon, G. Voelker, and S. Savage, "Internet quarantine: requirements for containing self-propagating code," in *Proc. of INFOCOM'03*, vol. 3, San Francisco, CA, Apr., 2003, pp. 1901-1910.

[41] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver, "Inside the Slammer worm," *IEEE Security and Privacy*, vol. 1 no. 4, July 2003, pp. 33-39.

[42] R. Narasimha, Z. Chen, and C. Ji, "Modeling malware propagation on networks: spatial dependence and its significance," submitted to *IEEE Transactions on Secure and Dependable Computing*, 2006.

[43] M. Opper and D. Saad, Eds., *Advanced Mean Field Methods, Theory and Practice*. Cambridge, MA: MIT Press, Feb. 2001.

[44] H. Orman, "The Morris worm: a fifteen-year perspective," *IEEE Security and Privacy Magazine*, vol. 1, Sept./Oct., 2003, pp. 35- 43.

[45] G. Owen, *Game Theory*. New York: Academic Press, 2001.

[46] Y. Pryadkin, R. Lindell, J. Bannister, and R. Govindan, "An empirical evaluation of IP address space occupancy," Tech. Rep. ISI-TR-2004-598, USC/Information Sciences Institute, Nov. 2004.

[47] R. Puri, "Bots & botnet: an overview," in *SANS Institute'03*, 2003.

[48] M. A. Rajab, F. Monrose, and A. Terzis, "Fast and evasive attacks: highlighting the challenges ahead," in *Proc. of the 9th International Symposium on Recent Advances in Intrusion Detection (RAID'06)*, Hamburg, Germany, Sept. 2006.

[49] M. A. Rajab, F. Monrose, and A. Terzis, "On the effectiveness of distributed worm monitoring," in *Proc. of the 14th USENIX Security Symposium (Security'05)*, Baltimore, MD, Aug. 2005, pp. 225-237.

[50] A. Renyi, *Probability Theory*. North-Holland, Amsterdam, 1970.

[51] D. W. Richardson, S. D. Gribble, and E. D. Lazowska, "The limits of global scanning worm detectors in the presence of background noise," in *Proc. ACM/CCS Workshop on Rapid Malcode (WORM'05)*, Fairfax, VA, Nov. 2005, pp. 60-70.

[52] S. Ross, *Simulation (3rd edition)*. Elsevier, 2002.

[53] S. Ross, *Introduction to Probability Models (7th edition)*. New York: Academic Press, 2000.

[54] S. Saroiu, P. K. Gummadi, and S. D. Gribble, "A measurement study of peer-to-peer file sharing systems," in *Proc. of Multimedia Computing and Networking (MMCN'02)*, San Jose, CA, Jan. 2002.

[55] S. E. Schechter, J. Jung, W. Stockwell, and C. McLain, "Inoculating SSH against address harvesting," in *Proc. 13th Ann. Network and Distributed System Security Symposium (NDSS'06)*, San Diego, CA, Feb. 2006.

[56] C. Shannon and D. Moore, "The spread of the Witty worm," *IEEE Security and Privacy*, vol. 2 No 4, Jul-Aug 2004, pp. 46-50.

[57] P.J. Smith, M. Shafi, and H. Gao, "Quick simulation: a review of importance sampling techniques in communications systems," *IEEE Jour. Selected Areas Commun.*, vol.15, May 1997, pp.597-613.

[58] D. Song, R. Malan, and R. Stone, "A snapshot of global Internet worm activity," Tech. Rep., 2001, http://www.arbor.net/downloads/research39/snapshot_worm_activity.pdf (March/2007 accessed).

[59] S. Staniford, V. Paxson, and N. Weaver, "How to 0wn the Internet in your spare time," in *Proc. of the 11th USENIX Security Symposium (Security'02)*, San Francisco, CA, Aug. 2002.

[60] S. Staniford, D. Moore, V. Paxson, and N. Weaver, "The top speed of flash worms," in *Proc. ACM/CCS Workshop on Rapid Malcode (WORM'04)*, Washington DC, Oct. 2004, pp. 33-42.

[61] P. Szor, *The Art of Computer Virus Research and Defense*. Symantec Press, 2005.

[62] R. Thomas, *The Team Cymru Bogon List v3.0 30 JUN 2005*, [Online]. Available: http://www.cymru.com/Documents/bogon-list.html (March/2007 accessed).

[63] J. Twycross and M. M. Williamson, "Implementing and testing a virus throttle," in *Proc. of the 12th USENIX Security Symposium (Security'03)*, Washington, DC, Aug. 2003, pp. 285-294.

[64] A. Vázquez, R. Pastor-Satorras, and A. Vespignani, "Large-scale topological and dynamical properties of the Internet," *Phys. Rev. E 65*, 2002, 066130.

[65] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," Dept. Statist., Univ. California, Berkeley, Tech. Rep. 649, 2003.

[66] C. Wang, J.C. Knight, and M. Elder, "On viral propagation and the effect of immunization," in *Proc. 16th ACM Annual Computer Applications Conference*, New Orleans, LA, Dec., 2000, pp. 343-354.

[67] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos, "Epidemic spreading: an eigenvalue viewpoint," in *Proc. 2003 Symposium of Reliable and Distributed Systems*, Florence, Italy, Oct. 2003.

[68] N. Weaver, "Warhol worms, the potential for very fast Internet plagues" [Online]. Available: http://www.iwar.org.uk/comsec/resources/worms/warhol-worm.htm (March/2007 accessed).

[69] E. W. Weisstein, "Chebyshev sum inequality," *From MathWorld–A Wolfram Web Resource*, http://mathworld.wolfram.com/ChebyshevSumInequality.html (March/2007 accessed).

[70] J. Wu, S. Vangala, L. Gao, and K. Kwiat, "An effective architecture and algorithm for detecting worms with various scan techniques," in *Proc. 11th Ann. Network and Distributed System Security Symposium (NDSS'04)*, San Diego, CA, Feb. 2004.

[71] Y. Xie, V. Sekar, D. A. Maltz, M. K. Reiter, and H. Zhang, "Worm origin identification using random moonwalks," in *Proc. of the IEEE Symposium on Security and Privacy (Oakland'05)*,Oakland, CA, May 2005.

[72] S. Xing and B.-P. Paris, "Measuring the size of the Internet via importance sampling," *IEEE journal on selected areas in communications*, vol. 21, no. 6, Aug. 2003, pp. 922-933.

[73] S. Xing and B.-P. Paris, "Mapping the growth of the Internet," in *Proc. of IEEE 2003 International Conference on Computer Communications and Networks*, Dallas, TX, Oct. 2003, pp 199-204.

[74] J. S. Yedidia, "An idiosyncratic journey beyond mean field theory," *Advanced Mean Field Methods, Theory and Practice*, Feb. 2001, pp. 21-36.

[75] V. Yegneswaran, P. Barford, and D. Plonka, "On the design and utility of Internet sinks for network abuse monitoring," in *Proc. of Symposium on Recent Advances in Intrusion Detection (RAID'04)*, 2004.

[76] C. C. Zou, "Witty worm propagation modeling," http://www.cs.ucf.edu/~czou/research/wittyModel.html (March/2007 accessed).

[77] C. C. Zou, L. Gao, W. Gong, and D. Towsley, "Monitoring and early warning for Internet worms," in *10th ACM Conference on Computer and Communication Security (CCS'03)*, Washington DC, Oct. 2003.

[78] C. C. Zou, W. Gong, and D. Towsley, "Code Red worm propagation modeling and analysis," in *Proc. 9th ACM Conf. on Computer and Communication Security (CCS'02)*, Washington DC, Nov. 2002, pp. 138-147.

[79] C. C. Zou, D. Towsley, W. Gong, and S. Cai, "Routing worm: a fast, selective attack worm based on IP address information," in *Proc. 19th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation (PADS'05)*, Monterey, CA, June 2005.

[80] C. C. Zou, D. Towsley, and W. Gong, "Email worm modeling and defense," in *Proc. 13th International Conf. on Computer Communications and Networks (ICCCN'04)*, Chicago, IL, Oct. 2004, pp. 409-414.

[81] C. C. Zou, D. Towsley, and W. Gong, "On the performance of Internet worm scanning strategies," *Elsevier Journal of Performance Evaluation*, vol. 63, no. 7, July 2006, pp. 700-723.

[82] CERT Coordination Center, "'Code Red II:' another worm exploiting buffer overflow in IIS indexing service DLL," CERT Incident Note IN-2001-09 [Online]. Available: http://www.cert.org/incident_notes/IN-2001-09.html (March/2007 accessed).

[83] CERT Coordination Center, CERT Advisory CA-2001-26 Nimda Worm [Online]. Available: http://www.cert.org/advisories/CA-2001-26.html (March/2007 accessed).

[84] Distributed Intrusion Detection System (DShield) [Online]. Available: http://www.dshield.org/ (March/2007 accessed).

[85] eEye Digital Security, "ANALYSIS: Blaster worm," [Online]. Available: http://research.eeye.com/html/advisories/published/AL20030811.html (March/2007 accessed).

[86] F-Secure Virus Descriptions : Slapper [Online]. Available: http://www.f-secure.com/v-descs/slapper.shtml (March/2007 accessed).

[87] Internet Protocol V4 Address Space [Online]. Available: http://www.iana.org/assignments/ipv4-address-space (March/2007 accessed).

[88] Malware Website [Online]. Available: http://www.malware.com/ (March/2007 accessed).

[89] NLANR Routing Raw Data [Online]. Available: http://moat.nlanr.net/Routing/rawdata (March/2007 accessed).

[90] PanetMath.org, "Rearrangement inequality," [Online]. Available: http://planetmath.org/encyclopedia/RearrangementInequality.html (March/2007 accessed).

[91] University of Oregon Route Views Project [Online]. Available: http://routeviews.org/ (March/2007 accessed).

[92] The CAIDA Dataset on the Witty Worm - March 19-24, 2004, Colleen Shannon and David Moore, http://www.caida.org/data/passive/witty_worm_dataset.xml (March/2007 accessed). Support for the Witty Worm Dataset and the UCSD Network Telescope are provided by Cisco Systems, Limelight Networks, the US Department of Homeland Security, the National Science Foundation, and CAIDA, DARPA, Digital Envoy, and CAIDA Members.

VITA

Zesheng Chen was born in Quanzhou, Fujian, P. R. China in September 1975. He received the Bachelor of Engineering degree and the Master of Engineering degree in Electronic Engineering from Shanghai Jiao Tong University, Shanghai, China, in 1998 and 2001, respectively. He attended the doctoral program in the School of Electrical and Computer Engineering at the Georgia Institute of Technology, Atlanta, GA, from August 2002 to May 2007. At the same time, he was a graduate research assistant in the Communication Networks and Machine Learning Laboratory at the Georgia Institute of Technology. He received the Master of Science degree and the Doctor of Philosophy degree in May 2005 and May 2007, respectively, in Electrical and Computer Engineering from the Georgia Institute of Technology.