Modeling the Propagation of Worms in Networks: A Survey

Yini Wang, Sheng Wen, Yang Xiang, *Senior Member, IEEE,* and Wanlei Zhou, *Senior Member, IEEE,*

***Abstract*—There are the two common means for propagating worms: scanning vulnerable computers in the network and spreading through topological neighbors. Modeling the propagation of worms can help us understand how worms spread and devise effective defense strategies. However, most previous researches either focus on their proposed work or pay attention to exploring detection and defense system. Few of them gives a comprehensive analysis in modeling the propagation of worms which is helpful for developing defense mechanism against worms' spreading. This paper presents a survey and comparison of worms' propagation models according to two different spreading methods of worms. We first identify worms characteristics through their spreading behavior, and then classify various target discover techniques employed by them. Furthermore, we investigate different topologies for modeling the spreading of worms, analyze various worms' propagation models and emphasize the performance of each model. Based on the analysis of worms' spreading and the existing research, an open filed and future direction with modeling the propagation of worms is provided.**

***Index Terms*—Network security, Worms, Propagation, Modeling.**

I. Introduction

WORMS and their variants have been a persistent security threat in the Internet from the late 1980s, causing large parts of the Internet becoming temporarily inaccessible, huge amount of financial loss and social disruption especially during the past decade. For example, the Code Red worm [1] in 2001 infected at least 359,000 hosts in 24 hours and had already cost an estimated $2.6 billion in damage to networks previous to the 2001 attack [2]. The Blaster worm [3] of 2003 infected at least 100,000 Microsoft Windows systems and cost each of the 19 research universities an average of US$299,579 to recover from the worm attacks [4]. Conficker worm [5], [6] was the fifth-ranking global malicious threat observed by Symantec in 2009 and infected nearly 6.5 million computers by attacking Microsoft vulnerabilities. Stuxnet [7], first discovered in June 2010, is a highly sophisticated computer worm. Initially, it targeted Siemens industrial software and equipment. Later the same year, it damaged the Iran nuclear program which used embargoed Siemens equipment procured secretly. Therefore, worms and their variants have evolved into a weapon in the information warfare worldwide. According to the official Internet threat report of the Symantec Corporation [8], worms and resembling attacks account for 1/4 of the total threats in 2009 and nearly 1/5 of the total threats in 2010. In order to prevent worms from spreading into a large scale, researchers focus on modeling their propagation and then, on the basis of it, investigate the optimized countermeasures. Similar to the research of some nature disasters, like earthquake and tsunami, the modeling can help us understand and characterize the key properties of their spreading. In this field, it is mandatory to guarantee the accuracy of the modeling before the derived countermeasures can be considered credible. In recent years, although a variety of models and algorithms have been proposed for modeling the propagation mechanism of worms, as well as for trying to catch and stop the spread of worms, the propagation of worms is still prevailing. In order to prevent worms from propagating and to mitigate the impact of an outbreak, we need to have a detailed and quantitative understanding of how a worm spreads. Moreover, it is significant to know the advantages and the limitations of the existing worms' propagation models, which have a potentially strong impact on predicting the spreading tendency of worms and are essential for developing defense mechanism [9] against the spreading of them. However, most previous researches [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22] discuss efforts that are related to their proposed work. Some survey papers [23], [24], [25], [26] introduce the life cycle of worms and investigate the propagation of worms, but they focus on exploring detection and containment systems rather than giving a comprehensive classification and comparison of the existing models.

Therefore, we are motivated to provide a thorough analysis of the spreading procedure and highlight the performance of worms' propagation models that benefit the defense against them. Firstly, worms leverage different kinds of methods to identify vulnerable hosts and spread themselves. In order to have a deep insight into how worms propagate across the Internet, we introduce various target discovery techniques of worms. Secondly, a network topology defines how the computers within the network are arranged and connected to each other. Since it plays a significant role in determining the worms' propagation and the overall spreading scale, we analyze four typical topologies of networks that are widely used in modeling the propagation of worms. Thirdly, mathematical models are very useful to describe the dynamics and measure the speed of worms' propagation. Therefore, this paper provides a detailed study of current mathematical model that have been established and compares their effectiveness.

The rest of the paper is organized as follows. Definition, categorization and the propagation of worms are introduced

Manuscript received September 28, 2012; revised March 15, 2013 and June 30, 2013.

The authors are with the Deakin University, School of Information Technology, Melbourne, Australia (e-mail: yiniwang@au.ibm.com, {wsheng, yang, wanlei}@deakin.edu.au).

Digital Object Identifier 10.1109/SURV.2013.100913.00195

in Section 2, which set the stage for later sections. A variety of primitive and advanced target discovery techniques are provided in details in Section 3. In Section 4, typical topologies of network for modeling the worms' propagation are investigated. Worm propagation models, which are the analytical tools for describing the dynamics and measuring the propagation speed of worms, are analyzed in Section 5. Section 6 concludes this paper and points out future research direction in modeling the propagation of worms.

II. Definition, Categorization and Propagation of Worms

A. Definition of Worms

A computer worm is a program that self-propagates across a network exploiting security or policy flaws in widely-used services [27]. Worms and viruses are often placed together in the same category, however there is a technical distinction. A virus is a piece of computer code that attaches itself to a computer program, such as an executable file. The spreading of viruses is triggered when the infected program is launched by human action. A worm is similar to a virus by design and is considered to be a sub-class of viruses. It differs from a virus in that it exists as a separate entity that contains all the code needed to carry out its purposes and does not attach itself to other files or programs. Therefore, we distinguish between worms and viruses in that the former searches for new targets to transmit themselves, whereas the latter searches for files in a computer system to attach themselves to and which requires some sort of user action to abet their propagation [28].

B. Worm Categorization

A worm compromises a victim by searching through an existing vulnerable host. There are a number of techniques by which a worm can discover new hosts to exploit. According to the target-search process, we can divide worms into two categories: scan-based worms and topology-based worms.

1) Scan-based Worms: A scan-based worm (scanning worm) propagates by probing the entire IPv4 space or a set of IP addresses and directly compromises vulnerable target hosts without human interference, such as Code Red I v2 (2001), Code Red II (2001), Slammer/Sapphire (2003), Blaster (2003), Witty (2004) [29], Sasser (2004) [30] and Conficker (2009) [5], [6]. A key characteristic of a scan-based worm is that it can propagate without dependence on the topology. This means that an infectious host is able to infect an arbitrary vulnerable computer.

Scan-based worms employ various scanning strategies, such as random scanning and localized scanning, to find victims when they have no knowledge of where vulnerable hosts reside in the Internet. Random scanning selects target IP addresses randomly, whereas worms using the localized scanning strategy scan IP addresses close to their addresses with a higher probability compared to addresses that are further away.

2) Topology-based Worms: A topology-based worm, such as an email worm and a social network worm, relies on the information contained in the victim machine to locate new targets. This intelligent mechanism allows for a far more efficient propagation than scan-based worms that make a large number of wild guesses for every successful infection. Instead, they can infect on almost every attempt and thus, achieve a rapid spreading speed. Secondly, by using social engineering techniques on modern topological worms, most Internet users can possibly fail to recognize malicious codes and become infected, therefore resulting in a wide range of propagation.

A key characteristic of a topology-based worm is that it spreads through topological neighbors. For example, email worms, such as Melissa (1999) [31], Love Letter (2000) [32], [33], Sircam (2001)[34], MyDoom (2004) and Here you have (2010), infect the system immediately when a user opens a malicious email attachment and sends out worm email copies to all email addresses in the email book of the compromised receiver. For social network worms such as Koobface, the infected account will automatically send the malicious file or link to the people in the contact list of this user.

C. The Propagation of Worms

Worms have attracted widespread attention because they have the ability to travel from host to host and from network to network. Before a worm can be widely spread, it must first explore the vulnerabilities in the network by employing various target discovery techniques. Subsequently, it infects computer systems and uses infected computers to spread itself automatically (as with scan-based worms) or through human activation (as with topology-based worms).

During the propagation of worms, hosts in the network have three different states: susceptible, infectious and removed. A susceptible host is a host that is vulnerable to infection; an infectious host means one which has been infected and can infect others; a removed host is immune or dead so cannot be infected by worms again. According to whether infected hosts can become susceptible again after recovery, researchers model the propagation of worms based on three major models: *SI* models (if no infected hosts can recover), *SIS* models (if infected hosts can become susceptible again) and *SIR* models (if infected hosts can recover). On the basis of these models, researchers also presented various defense mechanisms against the propagation of worms.

Although a great deal of research has been done to prevent worms from spreading, worm attacks still pose a serious security threat to networks for the following reasons. Firstly, worms can propagate through the network very quickly by various means, such as file downloading, email, exploiting security holes in software, etc. Some worms can potentially establish themselves on all vulnerable machines in only a few seconds [10]. Secondly, the rapid advances of computer and network technologies allow modern computer worms to propagate at a speed much faster than human-mediated responses. Thirdly, in order to propagate successfully, worms are becoming more complicated and increasingly efficient. It is therefore of great importance to characterize worm attack behaviors and analyze propagation procedures, which can efficiently provide patch strategies for protecting networks from worm attacks.

III. Target Discovery Techniques of Worms

Worms employ distinct propagation strategies such as random, localized, selective and topological scanning to spread.

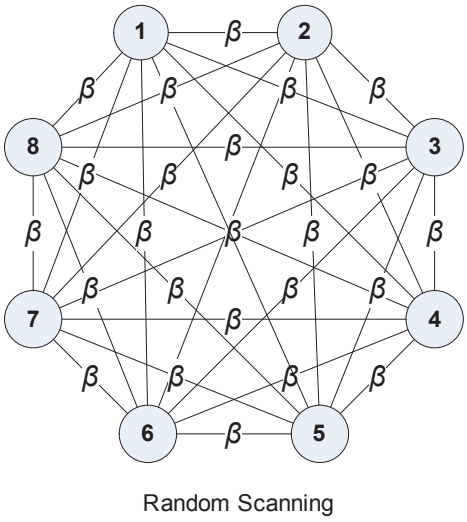


Fig. 1. Graphical representation of random scanning

In this subsection, we discuss these target discovery techniques and some of their different sub-classes.

A. Scan-based Techniques

Scanning is a very common propagation strategy due to its simplicity and is the most widely employed technique by some well-known scan-based worms such as Code Red, Code Red II, Slammer, Blaster, Sapphire, and Witty worm. Scan-based techniques probe a set of addresses to randomly identify vulnerable hosts or work through an address block using an ordered set of addresses [27].

1) Random Scanning: Random scanning selects target IP addresses randomly, which leads to a fully-connected topology with identical infection probability β for every edge (shown in Fig. 1). Several types of scanning strategies, such as uniform, hit-list, and routable scanning, are implemented on the basis of random scanning.

a) Uniform Scanning: Uniform scanning is the simplest strategy to compromise targets when a worm has no knowledge of where vulnerable hosts reside. It picks IP addresses to scan from the whole IPv4 address space with equal probability. This means a worm selects a victim from its scanning space without any preference. Thus, it needs a perfect random number generator to generate target IP addresses at random. Some famous worms, such as Code Red I v1 and v2 [1], and Slammer [35] employed this scanning approach to spread themselves. However, Code-Red I v1 used a static seed in its random number generator and thus generated identical lists of IP addresses on each infected machine. This meant the targets probed by each infected machine were either already infected or impregnable. Consequently, Code-Red I v1 spread slowly and was never able to compromise a high number of hosts. Code-Red I v2 used a random seed in its pseudo-random number generator and thus, each infected computer tried to infect a different list of randomly generated IP addresses. This minor change resulted in more than 359,000 machines being infected with Code-Red I v2 in just fourteen hours [36].

b) Hit-list Scanning: Hit-list scanning was introduced by Staniford et al. [10], which can effectively reduce the infection time at the early stage of worm propagation. A hit-list scanning worm first scans and infects all vulnerable hosts on the hit-list, then continues to spread through random scanning. The vulnerable hosts in the hit-list can be infected in a very short period because no scans are wasted on other potential victims. Hit-list scanning hence effectively accelerates the propagation of worms at the early stage. If the hit-list contains IP address of all vulnerable hosts, (called a complete hit-list), it can be used to speed the propagation of worms from beginning to end with the probability of hitting vulnerable or infected hosts equal to 100%. Flash worm [10] is one such worm. It knows the IP addresses of all vulnerable hosts in the Internet and scans from this list. When the worm infects a target, it passes half of its scanning space to the target, and then continues to scan the remaining half of its original scanning space. If no IP address is scanned more than once, then a flash worm is the fastest spreading worm in terms of its worm scanning strategy [37]. Due to bandwidth limitation, however, flash worms cannot reach their full propagation speed. Furthermore, in the real world it is very hard to know all vulnerable hosts' IP addresses. Therefore, complete hit-list scanning is difficult for attackers to implement considering the global scale of the Internet.

c) Routable Scanning: The routable scanning approach probes each IP address from within the routable address space in place of the whole IPv4 address space. Therefore, it needs to determine which IP addresses are routable. Zou et al. [38] presented a BGP routable worm as BGP routing tables contain all routable IP addresses. Through scanning the BGP routing table, the scanning address space Ω of BGP routable worms can be effectively reduced without missing any targets. Currently about 28.6% of the IPv4 address has been allocated and is routable. However, worms based on BGP prefixes have a large payload, which leads to a decrease in the propagation speed. Consequently, a Class A routing worm was presented by Zou et al. [38], which uses IPv4 Class A address allocation data. The worm only needs to scan 116 out of 256 Class A address space, which contributes 45.3% of the entire IPv4 space. Routable scanning therefore, improves the spreading speed of worms by reducing the overall scanning space.

2) Localized Scanning: Instead of selecting targets at random, worms prefer to infect IP addresses that are closer by. Localized scanning strategies choose hosts in the local address space for probing. This leads to a fully-connected topology as shown in Fig. 2, where nodes within the same group (group 1 or group 2) infect each other with the same infection probability β_1, while nodes from different groups infect each other with infection probability β_2.

a) Local Preference Scanning: Since vulnerable nodes are not uniformly distributed in the real world, a worm can spread itself quickly when it scans vulnerability dense IP areas more intensively. For this reason, the local preference scanning approach is implemented by attackers, which selects target IP addresses close to a propagation source with a higher probability than addresses farther away. Some localized scanning worms (Code Red II [39], [40], [41], [42] and Blaster worm [11]) propagate themselves with a high probability

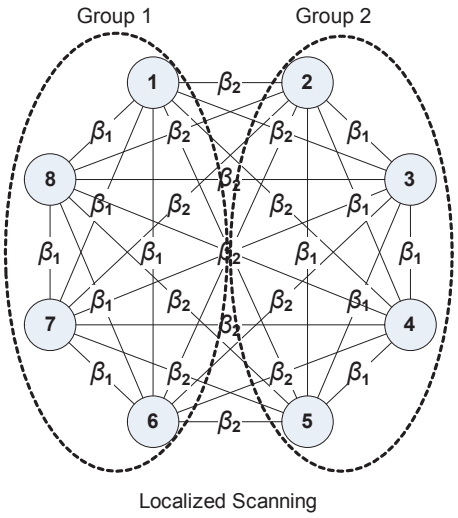


Fig. 2. Graphical representation of localized scanning

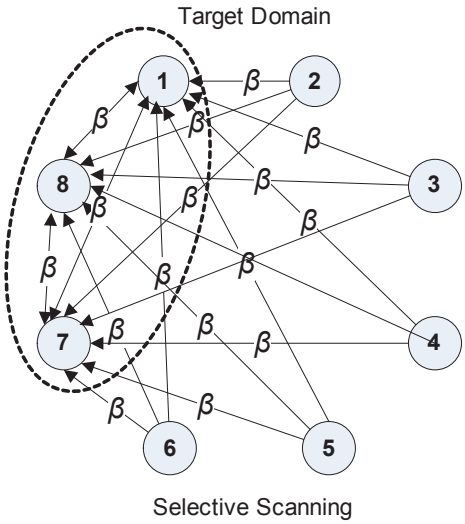


Fig. 3. Graphical representation of selective scanning

in certain IP addresses for the purpose of increasing their spreading speed. Taking Code Red II as an example, the probability of the virus propagating to the same Class A IP address is 3/8; to the same Class A and B IP address is 1/2; and to a random IP address is 1/8.

b) Local Preference Sequential Scanning: Different from random scanning, the sequential scanning approach scans IP addresses in order from a starting IP address selected by a worm [37]. Blaster [43] is a typical sequential scan worm because it chooses its starting point locally as the first address of its Class C /24 network with a probability of 0.4 and a random IP address with a probability of 0.6. In selecting the starting point of a sequence, if a close IP address is chosen with higher probability than an address far away, we use the term 'local preference sequential scanning'. According to an analysis in [37], a worm employing a local preference sequential scanning strategy is more likely to repeat the same propagation sequence, which results in wasting most of the infection power of infected hosts. Consequently, the local preference sequential scanning approach slows down the spreading speed in the propagation of worms.

c) Selective Scanning: Selective scanning is implemented by attackers when they plan to intentionally destroy a certain IP address area rather than the entire Internet, that is, the scanning space is reduced to those selected IP addresses. The selective scanning strategy can lead to an arbitrary topology as shown in Fig. 3, where node 4 scans nodes 1, 8 and 7 with infection probability β. If a worm only scans and infects vulnerable hosts in the target domain, it is referred to as *Target-only* scanning. In selective scanning, attackers care more about the spreading speed of a worm in the target domain than the scale of the infected network. According to the analysis in [37], target-only scanning can accelerate the propagation speed if vulnerable hosts are more densely distributed in the target domain.

B. Topology-based Techniques

Topology-based (or topological scanning) techniques are mainly used by worms spreading through topological neighbors. This strategy can lead to an arbitrary topology, as shown in Fig. 4, where node $N_i(i = 1, 2, ..., 8)$ scans its neighbors with a different infection probability $\beta_i(i = 1, 2, ..., 10)$. Note that the topology discussed in this section reflects the logical connection between the Internet users and their social friend. A typical example of worms that employ topology-based techniques to launch attacks are email worms. When an email user receives an email message and opens the malicious attachment, the worm program will infect the user's computer and send copies of itself to all email addresses that can be found in the recipient's machine. The addresses in the recipient's machine disclose the neighborhood relationship. Melissa [44] is a typical email worm which appeared in 1999. It looks through all Outlook address books and sends a copy of itself to the first 50 individuals when an infected file is opened for the first time. After Melissa, email worms have become annoyingly common, completed with toolkits and improved by social engineering, such as Love letter in 2000, Mydoom in 2004 and W32.Imsolk in 2010. Recently, topology-based techniques have been used by some isomorphic worms such as Bluetooth worms [12], p2p worms [13], [45], and social networks worms [46]. For example, Koobface [47] spreads primarily through social networking sites. It searches the friend list of a user and posts itself as links to videos on their friend's website. When a user is tricked into visiting the website that hosts the video, they are prompted to download a video codec or other necessary update, which is actually a copy of the worm. Users may have difficulty determining if a link was posted by a friend or the worm.

Topology-based techniques utilize the information contained in the victim's machine to locate new targets. This intelligent mechanism allows for a far more efficient propagation than scan-based techniques that make a large number of wild guesses for every successful infection. Instead, they can infect on almost every attempt and thus, achieve a rapid spreading speed. A common feature of topology-based techniques is to involve human interference in the propagation of worms. Taking email worms as an example, the worm program can

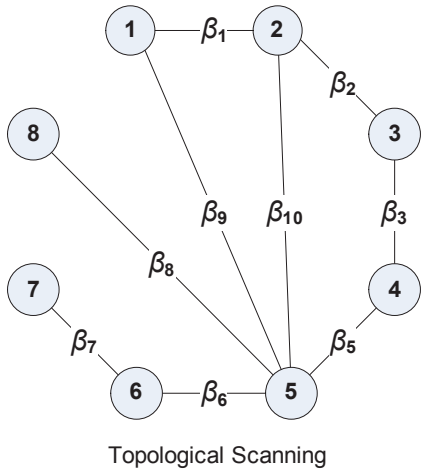


Fig. 4. Graphical representation of topological scanning

infect the user's machine and become widespread only when an email user opens the worm email attachment. Thus, whether or not a computer can be infected by malicious emails is determined by human factors including the user's personal habits of checking emails and the user's security consciousness.

IV. Topologies for Modeling the Propagation of Worms

The topology of a network plays a critical role in determining the propagation dynamics of a worm. In the research of epidemic modeling, many types of networks (for example, [11], [14], [48], [49], [50], [51], [52]) are adopted to study the effect of epidemic propagation. In this section, we will introduce four typical topologies of networks that are widely used in modeling the propagation of worms.

A. Homogenous Networks

In a homogenous network, each node has roughly the same degree. A fully-connected topology and a standard hypercubic lattice are two typical examples of homogeneous networks. The propagation of worms on homogenous networks satisfies the homogenous assumption that any infected host has an equal opportunity to infect any vulnerable host in the network. Thus, there is no topological issue in the homogenous networks. In real scenarios, most scan-based worms, such as Code Red I, Code Red II, and Slammer, exploit vulnerabilities through scanning the entire or part of IP space without any dependence on the properties of the Internet topology. Thus, homogeneous networks are more suitable for modeling the spreading of scan-based worms.

Recently, many researches [10], [37], [39], [11] studied random scanning worms on homogenous networks using differential equation models. These models assume all hosts in the network can contact each other directly and thus, their topologies are treated as fully-connected graphs. Chen et al. [11] proposed an analytical active worm propagation (AAWP) model for randomly scanning worms on the basis of homogenous networks. Yan and Eidenbenz [12] present a detailed analytical model that characterizes the propagation dynamics of Bluetooth worms. It assumes all individual devices are homogenously mixed. Zou et al. [39] proposed a two-factor worm model to characterize the propagation of the Code Red worm. This model adopts the homogeneous network, that is, they consider worms that propagate without the topology constraint.

B. Random Networks

A random network is a theoretical construct which contains links that are chosen completely at random with equal probability, such as Erdös-Renyi (ER) random network [48]. Using a random number generator, one assigns links from one node to a second node. Random links typically result in shortcuts to remote nodes, thus shortening the path length to otherwise distant nodes [53]. A random network is a non-homogenous network, which means each node may not have same node degree. When a worm propagates on the random graph network, the random-graph topology has an impact on the spreading procedure. Recent work [54], [55] provided mechanisms to specify the degree distribution when constructing random graphs and further characterize the size of the large connected component. Fan and Xiang [13] investigated the impact of worm propagation over a simple random graph topology. It assumes each host has the same out-degree. Hosts to which each host has an outbound link are randomly selected from all hosts except the host itself. Of course, the degrees of nodes in a random graph may not be all equal. Zou et al. [14] studied the email worm propagation on a random graph. The random graph network was constructed with n vertices and an average degree E[k]$\geq$2. Here, k represents the vertex degree of a node in a graph. The mean of degrees in a graph is denoted by E[k]. From the analysis of Zou's model, a random graph cannot reflect a heavy-tailed degree distribution and thus, it is not suitable for modeling topology-based worms.

C. Small-World Networks

A small-world network is a type of mathematical graph, which interpolates between a regular network and a random network. It occurs by replacing a fraction p of the links of a d dimensional lattice with new random links. In a small-world network, most nodes are not neighbors of one another, but can be reached from every other node by a small number of hops or steps. Small-world networks are highly clustered and have a small characteristic path [56]. Some researchers have observed the dynamic propagation of worms on small-world networks. G. Yan et al. [57] considered the BrightKite graph to investigate the impact of malware propagation over online social networks. Compared with the random graph, the BrightKite graph [58] has a similar average shortest path length and a smaller clustering coefficient, and thus, it closely reflects a small-world network structure. Zou et al. [14] modeled email worm propagation on a small-world network that has an average degree E[k]$>$4. It firstly constructs a regular two-dimensional grid network and then connects two randomly-chosen vertices repeatedly until the total number of edges reaches E[k]$\cdot$ n/4. From the analysis of Zou's model, a small-world network still cannot provide a heavy-tailed degree distribution and thus, is not suitable for modeling topology-based worms.

D. Power-Law Networks

Power-law networks are networks where the frequency f_d of the out-degree d is proportional to the out-degree to the power of a constant α: $f_d \propto d^{\alpha}$ [59]. The constant α is called the power-law exponent. In a power-law network, nodes with the maximum topology degree are rare and most nodes have the minimum topology degree. Recent works have shown that many real-world networks are power-law networks such as social networks [60], [51], [61], [62], [63], [64], neural networks [65], and the Web [66], [67].

Zou et al. [14] and Ebel et al. [49] investigated email groups and found that they exhibited characteristics of a power-law distribution. The simulation model proposed by Zou et al. [14] studied the dynamic propagation of an email worm over a power-law topology. Although email worms spread slower on a power-law topology than small world topology or random topology, the immunization density is more effective on a power-law topology. Fan and Xiang [13] presented a logic 0-1 matrix model and observed the propagation of worms on a pseudo power law topology. Z. Chen and C. Ji [68] constructed a spatial-temporal model and analyzed the impact of malware propagation on a BA (B*á*rab*á*si-Albert) network [67], which is a type of power-law network. W. Fan et al. [46] assumed that the node degree of Facebook users exhibits the power-law distribution and constructed the network using two models: the BA (B*á*rab*á*si-Albert) model and the GLP (Generalized Linear Preference) model.

E. Perspective of Real World Topologies

Topology properties affect the spread of topology-based worms, which can either impede or facilitate their propagation and maintenance. Existing works [14], [68], [50] show that structures and characters of the network have strong impact on the spreading speed and scale of worms. However, in this field, all existing research adopts simulation to evaluate analytical models, such as [68], [69]. In real-world scenarios, the spread of most worms (e.g. email worms) is typically impossible to track given the directed, topological manner in which they spread. Thus, researchers generally adopt simulations to evaluate proposed models. Although some worms, such as Nyxem [70] (an email worm), can automatically generate a single http request for the URL of an online statistics page when it compromised a computer, the statistics of Nyxem also cannot present a precise investigation on the spread of email worms due to the legitimate access, repeated probes and DDoS attacks to the web page [71]. Thus, most current researches mainly rely on the above four network topologies, which reflect the characters of real network essentially, to investigate the propagation procedure of worms. For example, in terms of the scan-based worms, most of them propagate through the Internet and are able to directly hit a target without human activation, thus they are more suitable for being modeled by homogenous networks.

The characters of social networks and the impacting of social structures on the propagation of worms have been intensively investigated in many works [57], [68], [60]. Adamic et al. [60] found that the network exhibits small-world behavior through studying an early online social network. Mislove et al. [51] presented a large-scale measurement study and analysis of the structure of four popular online social networks: Flickr, Orkut, YouTube and LiveJournal. Their results confirm the power-law, small-world and scale-free properties of online social networks. Yan et al. [57] studied the BrightKite network and found that the highly skewed degree distributions and highly clustered structures shown in many social networks are instrumental in spreading the malware quickly at its early stage.

The topology of an email network plays a critical role in determining the propagation dynamics of an email worm [14], [49]. Zou et al. [14] examined more than 800,000 email groups in Yahoo! and found that it is heavy-tailed distributed, which exhibits the character of power-law networks. Ebel et al. [49] studied the topology of email network that constructed from log files of the email server at Kiel University and found that it exhibits a scale-free link distribution and pronounced small-world behavior. Although the topology of social and email networks varies, we can derive the structure of topologies on the basis of previous statistical analysis in real social [51], [72], [73], [74], [75], and then, use 2K-series method [76] to generate the social topologies.

Modeling the propagation of the topology-based worms should be independent of the network topologies. It means that the models can be effective and correct in any kind of network topology and can reflect the tendency of the spreading of worms. However, the accuracy of modeling can be impacted by different network topologies [68], [71]. The essence of the inaccuracy is caused by the spreading cycles in the propagation path [71]. The spreading cycles formed in the modeling lead to considerable errors in estimating the infection probabilities. Different topologies have different number of spreading cycles. The probabilistic effect from these cycles also varies according to their network structure. Thus, one modeling method may have different performance in the accuracy when topology changes.

In order to eliminate the errors caused by the spreading cycles, previous analytical models, such as [11], [12], [68], [77], assume nodes in the network to be spatial independent. However, this assumption results in the overestimation of the number of infected users. To address this problem, Chen et al. [68] approximated the reciprocal spreading probabilities through a Markov model, which can partially avoid the over-estimation caused by the dependency between users and their neighbors. Nevertheless, it only focuses on removing 1-order cycles but not higher order cycles. It is not enough to eliminate the errors. Consequently, the model cannot effectively reduce the errors and accurately estimate the infection probabilities. To overcome the shortcoming of the Markov model, Wen et al. [71] proposed a spatial-temporal analytical model and provided a stronger approximation of spatial dependence. They found that the cycles from 1-order to 5-order have significant effect on the propagation, and thus, they need to be removed in the modeling. The model in [71] is able to eliminate the effect of the spreading cycles. Therefore, it can avoid overestimation of propagation probabilities through removing the effect of spreading cycles and effectively resolve the impact on the modeling caused by different network topologies.

V. Worm Propagation Models

In the area of network security, worms have been studied for a long time [23], [24], [25]. Early works mainly refer to the academic thought on epidemic propagation and thus, models are constructed according to the state transition of each host including *Susceptible-Infectious* (denoted by '*SI*') models [78], *Susceptible-Infectious-Susceptible* (denoted by '*SIS*') models [79], and *Susceptible-Infectious-Recovered* (denoted by '*SIR*') models [50], [80], [72]. In the *SI* framework, all hosts stay in one of only two possible discrete states at any time: susceptible or infectious, which ignores the recovery process. The difference between *SIS* models and *SIR* models depends on whether infected hosts can become susceptible again after recovery. If this is the case, we use the term *SIS* model. Otherwise, if a host cannot become susceptible again once it is cured, we use the *SIR* model, where all hosts stay in one of only three states at any time: susceptible (denoted by '*S*'), infectious (denoted by '*I*'), removed (denoted by '*R*').

Currently, many mathematical models [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [22] have been proposed for investigating the propagation of scan-based and topology-based worms on the basis of different state transition models. For the convenience of readers, we list typical worm propagation models in Fig.5 In the figure, according to the target discovery techniques of worms, the models are primarily divided into two categories. Each model is further divided into subcategories (Fig. 5). Each one of these categories is discussed in the following sections.

A. Homogenous Scan-based Model

The homogenous worm propagation model relies on the homogeneous assumption that each infectious host has an equal probability of spreading the worm to any vulnerable peer in a network. Hence, the homogenous model is based on the concept of a fully connected graph and is an unstructured worm model that ignores the network topology. It can accurately characterize the propagation of worms using scan-based techniques to discover vulnerable targets, such as Code Red [81], [82], Code Red II [39], and Slammer [35]. Scan-based worms scan the entire network and infect targets without regard to topological constraints which means that an infectious host is able to infect an arbitrary vulnerable peer. Up to now, many researchers have modeled the propagation procedure of different types of scan-based worms on the basis of the homogenous assumption. The homogenous model can be further divided into two categories: continuous time and discrete time. A continuous time model is expressed by a set of differential equations, while a discrete time model is expressed by a set of difference equations.

1) Continuous-time Model:

a) Classical Simple Epidemic Model: The Classical Simple Epidemic Model [78], [15], [83], [84], [85] is a *SI* model. In this model, the state transition of any host can only be *S*→*I*, and it is assumed a host will remain in the 'infectious' state forever once it has been infected by a worm. Denote by $I(t)$ the number of infectious hosts at time t; N the total number of susceptible hosts in the network before a worm spreads out. Thus, the number of susceptible hosts at time t is equal to $[N - I(t)]$. The classical simple epidemic model for a finite population can be represented by the differential equation below:

$$\frac{dI(t)}{dt} = \beta I(t)[N - I(t)] \tag{1}$$

where, β stands for the pair-wise rate of infection in epidemiology studies [83] . It represents a ratio of infection from infectious hosts to susceptible hosts. At the beginning, $t = 0$, $I(0)$ hosts are infectious, and in the other $[N - I(0)]$ all hosts are susceptible.

The Classical Simple Epidemic Model is the most simple and popular differential equation model. It has been used in many papers (for example, [10], [37], [39], [11]) to model random scanning worms, such as Code Red [39] and Slammer [35].

b) Uniform Scan Worm Model: If a worm (i.e. Code Red, Slammer) has no knowledge of the distribution of vulnerable hosts in the network, uniformly scanning all IP addresses is the simplest method to spread itself. Once a host is infected by a worm, it is assumed to remain in the infectious state forever. The uniform scan worm model specifies the abstract parameter β in the classical simple epidemic model based on information pertaining to the scanning rate and IP space of the network. Denote by $I(t)$ the number of infectious hosts at time t; N the total number of susceptible hosts in the network before a worm spreads out. Thus $[N - I(t)]$ is the number of susceptible hosts at time t. Suppose an average scan rate η of a uniform scan worm is the average number of scans an infected host sends out per unit of time. Denote by δ the length of a small time interval. Thus, an infected host sends out an average of $\eta\delta$ scans during a time interval δ. Suppose the worm uniformly scans the IP space that has Ω addresses, every scan then has a probability of $1/\Omega (1/\Omega \ll 1)$ to hit any one IP address in this scanning space. Therefore, on average, an infected host has probability q to hit a specific IP address in the scanning space during a small time interval δ.

$$q = 1 - (1 - 1/\Omega)^{n\delta} \approx n\delta/\Omega, \qquad 1/\Omega \ll 1 \tag{2}$$

Here, during the time interval δ, the probability that two scans sent out by an infected host will hit the same vulnerable host is negligible when δ is sufficiently small. Consequently, the number of infected hosts at time $t + \delta$ will be:

$$I(t + \delta) = I(t) + I(t) \cdot [N - I(t)]\eta\delta/\Omega \tag{3}$$

Taking $\delta \to 0$, according to the epidemic model (1), the uniform scan worm model can be represented by (4):

$$\frac{dI(t)}{dt} = \frac{\eta}{\Omega} I(t)[N - I(t)] \tag{4}$$

At time $t = 0$, $I(0)$ represents the number of initially infected hosts and $[N - I(0)]$ is the number of all susceptible hosts.

Some variants of random scanning worms (hit-list worms [10], flash worms [10], [37], and routable worms [86]) cannot be directly modeled by (4). However, through the extension of the uniform scan worm model, the propagation of these variants of worms can be accurately modeled.

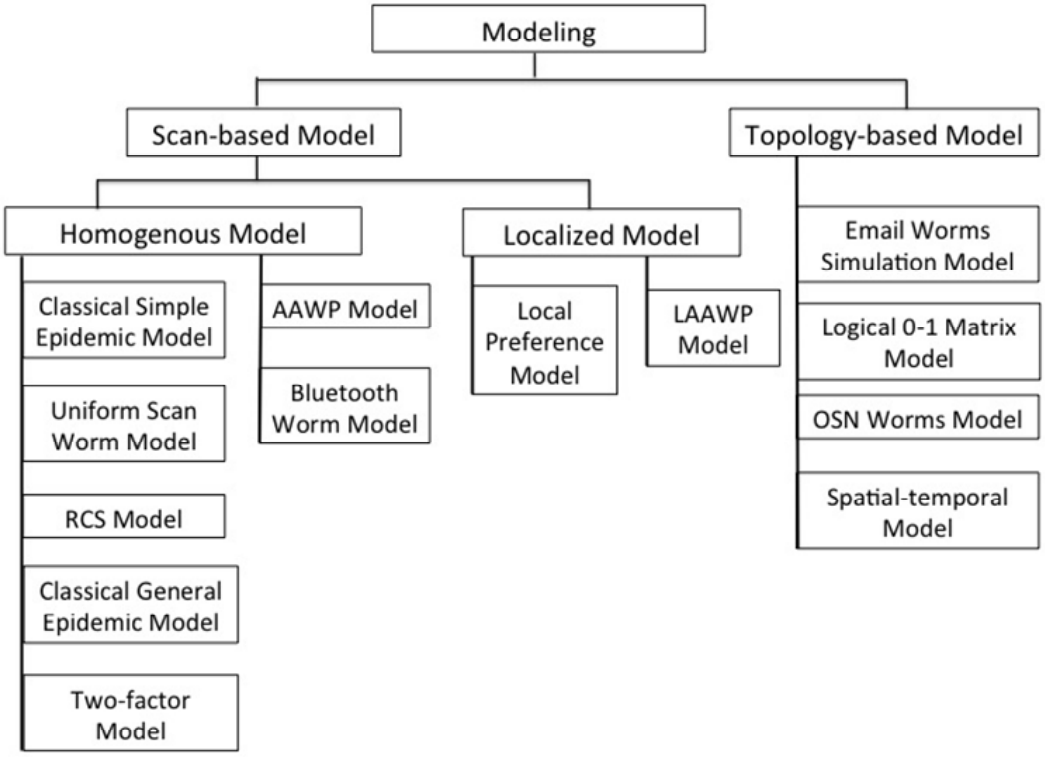


Fig. 5. Taxonomy of worm modeling

Staniford et al. [10] introduced a variant of random scanning worms, called the *hit-list worm*. It first scans and infects all vulnerable hosts on the hit-list, then randomly scans the entire Internet to infect others just like an ordinary uniform scan worm. We can assume the vulnerable hosts on the hit-list to be the initially infected hosts $I(0)$ and ignore the compromising time since they can be infected in a very short time [10]. As a result, a hit-list worm can be modeled by (4) along with a large number of initially infected hosts determined by the size of the worm's hit-list.

A flash worm is a variant of the hit-list strategy, introduced by Staniford et al. [10]. When a flash worm infects a target, it simply scans half of its scanning space as the other half has been passed to the target including the target host. Since it knows the IP addresses of all vulnerable hosts, that is, the size of scanning space $\Omega = N$, which is much smaller than the entire IPv4 address space ($\Omega = 2^{32}$), and because no IP address is scanned more than once, the flash worm could possibly infect most vulnerable hosts in the Internet in tens of seconds. For this reason, the time delay caused by the infection process of a vulnerable host cannot be ignored in modeling the spreading of flash worms. Denote by ε the time delay, which is the time interval from the time when a worm scan is sent out to the time when the vulnerable host infected by the scan begins to send out worm scans. We assume a flash worm uniformly scans the address list of all vulnerable hosts. Then, based on the uniform scan model (4), the flash worm (uniform scanning) can be modeled by (5):

$$\frac{dI(t)}{dt} = \frac{\eta}{N} I(t-\varepsilon)[N - I(t)], \qquad I(t-\varepsilon) = 0, \forall t < \varepsilon \quad (5)$$

Another variant of random scanning worms is a routable worm. Zou et al. [38] found that currently around 28.6% of IPv4 addresses are routable and thus, they presented a BGP routing worm. It uses BGP routing prefixes to reduce the worm's scanning space Ω. When a BGP routing worm uniformly scans the BGP routable space, it can be modeled by (4), where Ω equals 28.6% of all IP addresses.

Zou et al. [37] investigated and compared the propagation performance of random scanning worms and their variants (for example, Code Red, a hit-list worm, a flash worm and a BGP routable worm). Assume the number of vulnerable hosts (N) is 360 000, and worms have the same scan rate, i.e., η = 358/min. Suppose the size of a worm's hit-list is 10 000, that is, $I(0) = 10000$, while Code Red, the flash worm and the BGP routable worm have 10 initially infected hosts, that is, $I(0) = 10$. The scanning space for the BGP routable worm is 28.6% of the entire IP address space, while the Code Red worm and the hit-list worm scan all IP addresses $\Omega = 2^{32}$. For the flash worm, the scanning space $\Omega = N$. From the results of the experiment shown in Fig. 6, the flash worm is the fastest spreading worm, which finishes infection within 20 seconds, while Code Red finishes infection after around 500 minutes. At the early stage of propagation, because of a large number size of the hit-list, the hit-list worm can infect more vulnerable hosts than Code Red and the BGP routable worm. Compared with Code Red and the hit-list worm, the BGP routable worm has a smaller scanning space and thus, the infection speed of the routable worm is faster.

c) RCS Model: Staniford et al. [10] presented a RCS (Random Constant Spread) model to simulate the propagation of the Code Red I v2 worm, which is almost identical to

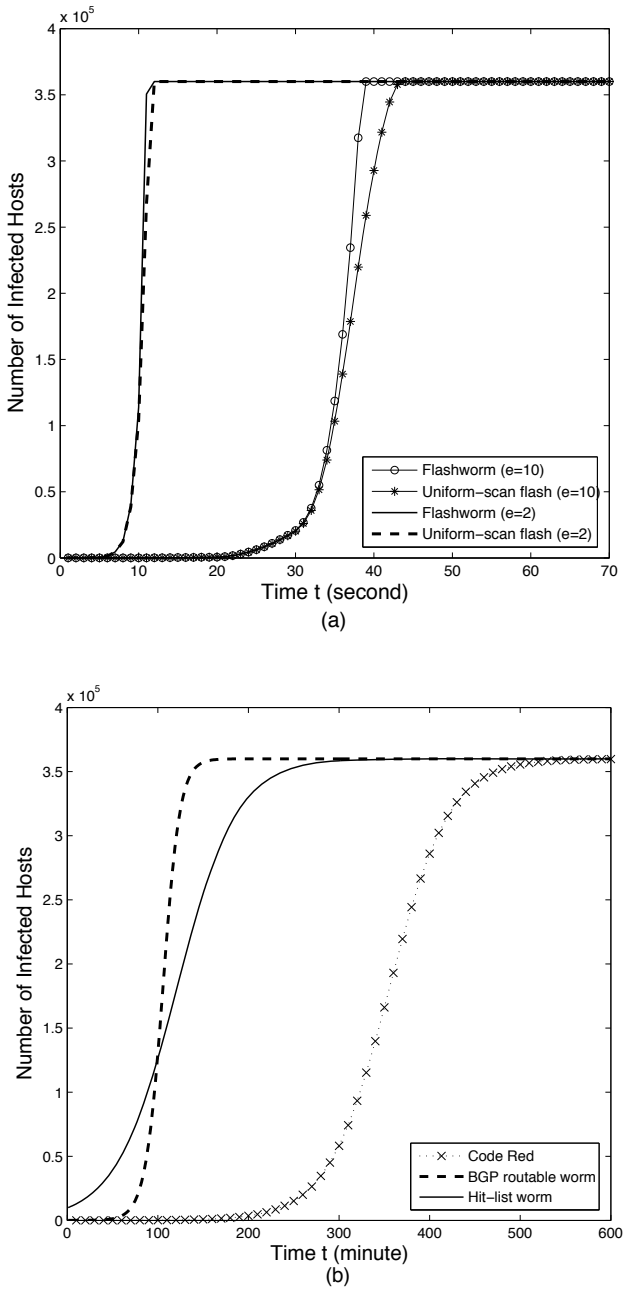


Fig. 6. Worm propagation of Code Red, BGP routable, hit-list, and flash worm

the classical simple epidemic model. Let $a(t) = I(t)/N$ be the fraction of the population that is infectious at time t. Substituting $I(t)$ in equation (1) with $a(t)$, and then deriving the differential equation (6) below, yields the equation used in [10]:

$$\frac{da(t)}{dt} = ka(t)[1 - a(t)] \tag{6}$$

with solution:

$$a(t) = \frac{e^{k(t-T)}}{1 + e^{k(t-T)}} \tag{7}$$

where, $k = \beta N$, and T is a constant of integration that fixes the time position of the incident. Differential equation (6) is a logistic equation. For early t, $a(t)$ grows exponentially, that is, the number of infectious hosts is nearly exponentially increased at the early stage of worm propagation. For large t, $a(t)$ goes to 1 (all susceptible hosts are infected).

d) Classical General Epidemic Model: Different from the classical simple epidemic model, the Kermack-McKendrick model considered the removal process of infectious hosts [78]. In the Kermack-McKendrick model, all hosts stay in one of only three states at any time: susceptible (denoted by '*S*'), infectious (denoted by '*I*'), removed (denoted by '*R*'). Once a host recovers from the disease, it will be immune to the disease and stay in the 'removed' state forever. The removed hosts can no longer be infected and they do not try to infect others. Therefore, the Kermack-McKendrick model is in the framework of a *SIR* model.

Let $I(t)$ denote the number of infectious hosts at time t and use $R(t)$ to denote the number of removed hosts from previously infectious hosts at time t. Denote β as the pair-wise rate of infection and γ as the rate of removal of infectious hosts. Then, based on the classical simple epidemic model (1), the Kermack-McKendrick model can be represented by (8):

$$\begin{aligned} \frac{dI(t)}{dt} &= \beta I(t)[N - I(t) - R(t)] - \frac{dR(t)}{dt} \\ \frac{dR(t)}{dt} &= \gamma I(t) \end{aligned} \tag{8}$$

where, N is the size of the finite population. The Kermack-McKendrick model improves the classical simple model by introducing a 'removed' state for each host which means some infectious hosts either recover or die after some time.

e) Two-factor Model: The Kermack-McKendrick model includes the removal of infectious hosts in the propagation of worms, but it ignores the fact that susceptible hosts can also be removed due to patching or filtering countermeasures. Furthermore, in the real world, the pair-wise rate of infection β decreases with the time elapsed in the spreading procedure due to the limitation of network bandwidth and Internet infrastructure, while the Kermack-McKendrick model assumes β is constant. Therefore, Zou et al. [39] introduced a two-factor model, which extends the Kermack-McKendrick model by considering human countermeasures and network congestion.

In the two-factor model, the removal process consists of two parts: removal of infectious hosts and removal of susceptible hosts. Denote $R(t)$ as the number of removed hosts from the infectious population and $Q(t)$ as the number of removed hosts from the susceptible population. $R(t)$ and $Q(t)$ involve people's security awareness against the propagation of worms. Moreover, in consideration of the slowed down worm scan rate, the pair-wise infection rate β is modeled as a function of time t, $\beta(t)$, which is determined by the impact of worm traffic on Internet infrastructure and the spreading efficiency of the worm code. Then, the two-factor model can be represented by (9):

$$\begin{aligned} \frac{dI(t)}{dt} &= \beta I(t)[N - I(t) - R(t) - Q(t)] - \frac{dR(t)}{dt} \\ \frac{dR(t)}{dt} &= \gamma I(t) \end{aligned} \tag{9}$$

where, N is the finite population size; $I(t)$ denotes the number of infectious hosts at time t; $\beta(t)$ is the pair-wise rate of infection at time t; and γ stands for the rate of removal of infectious hosts. The two-factor model improves the Kermack-McKendrick model through consideration of two major factors that affect worm propagation: human countermeasures like cleaning, patching or filtering and the slowing down of the worm infection rate.

2) Discrete-time Model:

a) AAWP Model: Chen, Gao and Kwiat [11] presented an AAWP (Analytical Active Worm Propagation) model to take into account the characteristics of random scanning worms spreading according to the homogenous assumption. It assumes that worms can simultaneously scan many machines in a fully-connected network and no hosts can be repeatedly infected. In this model, active worms scan the whole IPv4 address ($\Omega = 2^{32}$) with equal likelihood, therefore, the probability any computer is hit by one scan is $1/2^{32}$. Denote m_t as the total number of vulnerable hosts (including the infected hosts); denote n_t as the number of infected hosts at time tick $t(t \geq 0)$. At time tick $t = 0$, the number of initially vulnerable hosts m_0 is equal to N and the number of initially infected hosts n_0 is equal to h. We suppose s is the scanning rate, and the number of newly infected hosts in each time tick t is equal to $(m_i - n_i)[1-(1-1/2^{32})^{sn_t}]$. Assume that d represents the death rate and p denotes the patching rate. Then, in each time tick the number of vulnerable hosts without being infected and the number of healthy hosts will be $(d+p)n_t$. Therefore, on average in the next time tick $t+1$, the number of total infected hosts can be represented by (10):

$$n_{t+1} = n_t + (m_t - n_t)[1 - (1 - \frac{1}{2^{32}})^{sn_t}] - (d+p)n_t \quad (10)$$

In each time tick, the total number of vulnerable hosts including infected hosts is $(1-p)m_t$, and thus, at time tick t, $m_t = (1-p)^t m_0 = (1-p)^t N$. Therefore, we can derive (11) as follows:

$$n_{t+1} = (1-d-p)n_t + [(1-p)^t N - n_t][1-(1-\frac{1}{2^{32}})^{sn_t}] \quad (11)$$

where $t \geq 0$ and $n_0 = h$. Formula (11) models the propagation of random scanning worms analytically, and the iteration procedure will stop when all vulnerable hosts are infected or the number of infected hosts remains the same when worms spread.

b) Bluetooth Worm Model: G. Yan and S. Eidenbenz [12] presented a detailed analytical model that characterizes the propagation dynamics of Bluetooth worms. It captures not only the behavior of the Bluetooth protocol but also the impact of mobility patterns on the propagation of Bluetooth worms. This model assumes all individual Bluetooth devices are homogeneously mixed and advances time in a discrete fashion. Through analyzing a single infection cycle, it derives the duration of an infection cycle $T_{cycle}(t)$ and the number of new infections out of the infection cycle $\alpha(t)$. According to the pair-wise infection rate $\beta(t)$ derived from $\alpha(t)$ and new average density of infected devices at time t, this model can estimate the Bluetooth worm propagation curve. From this model, the average density of infected devices in the network at time t_{k+1} is defined by (12):

$$i(t_{k+1}) = i(t_k) \cdot \frac{\rho(t_k)}{i'(t_k) + (\rho(t_k) - i'(t_k))e^{-\alpha' \cdot \rho(t_k)/(\rho(t_k) - i'(t_k))}} \quad (12)$$

where $i'(t_k)$ is the maximum value between $i(t)$ and $1/S_{inq}(t)$ to ensure at least one infected device in the radio signal covers. $\rho(t_k)$ is the average device density at time t_k. Since the worm growth rate can change, and in order to avoid overestimating the number of new infections out of the infection cycle, it uses α' to achieve a better estimation of worm propagation, which is defined by (13):

$$\alpha' = \frac{\rho(t_k) - i(t_k)}{\rho(t_k)} \cdot \alpha(t_k) + \frac{i(t_k)}{\rho(t_k)} \cdot \alpha(t_x) \quad (13)$$

At the early phase, α' is close to αt_k and at the late state of the worm propagation, α' is close to $\alpha(t_x)$. Here, $t_x$is the latest time when an infected device starts their infection cycle after time t but before time t_{k+1}. This model predicts that the Bluetooth worm spreads quickly once the density of the infected devices reach 10 percent, although it propagates very slowly at the early stage.

B. Localized Scan-based Model

Since vulnerable nodes are not uniformly distributed, some localized scanning worms (Code Red II [39], [40], [41] and Blaster worm [11]) propagate the virus with a high probability in certain IP addresses for the purpose of increasing their spreading speed. Taking Code Red II as an example, the probability of the virus propagating to the same Class A IP address is 3/8; to the same Class A and B IP address is 1/2; and to a random IP address is 1/8. Therefore, the localized scanning worm employs a non-homogenous pattern to spread itself in the network. The localized scan-based model can be further divided into two categories: continuous time and discrete time. A continuous time model is expressed by a set of differential equations, while a discrete time model is expressed by a set of difference equations.

1) Continuous-time Model:

a) Local Preference Model: Zou et al. [37] took advantage of a continuous time model to describe the spread of localized scanning worms. In this local preference model, it is assumed that a worm has probability p of uniformly scanning IP addresses that have the same first n bits and probability $(1-p)$ of uniformly scanning other addresses. Suppose that the worm scanning space contains K networks where all IP addresses have the same first n bits and each network has $N_k(k = 1, 2, ..., K)$ initially vulnerable hosts. Denote by $I_k(t)$ the number of infected hosts in the k-th network at time t; and denote by β' and β'' the pair-wise rates of infection in local scan and remote scan, respectively. Then we have:

$$\beta' = \frac{p\eta}{2^{32-n}}, \beta'' = \frac{(1-p)\eta}{(K-1)2^{32-n}}$$
$$\frac{dI_k(t)}{dt} = [\beta' I_k(t) + \sum_{j \neq k} \beta'' I_j(t)] \cdot [N_k - I_k(t)] \quad (14)$$

where η represents the average number of scans an infected host sends out per unit of time. Since hosts are not uniformly distributed over the whole Internet, this model supposes only the first m networks $(m < K)$ have uniformly distributed vulnerable hosts, i.e., $N_1 = ... = N_m = N/m, N_{m+1} = ... = N_k = 0$. Thus, the worm propagation on each network follows (15):

$$\frac{dI_k(t)}{dt} = [\beta' + (m-1)\beta''] \cdot I_k(t)[N_k - I_k(t)], k = 1, 2, ..., m \tag{15}$$

Suppose $I_k(0) = I_1(0) > 0, k = 2, 3, ..., m$. We then have:

$$\frac{dI(t)}{dt} = [\frac{\beta' + (m-1)\beta''}{m}] \cdot I(t)[N - I(t)] \tag{16}$$

(14) describes the number of newly infected hosts at time tick t with respect to the entire Internet. This local preference model uses differential equations to reflect the propagation of localized worms that probe different IP addresses with their own preference probabilities.

2) Discrete-time Model:

a) LAAWP Model: LAAWP (Local Analytical Active Worm Propagation) model is a discrete time model extended from the AAWP model [11]. It characterizes the propagation of worms employing the localized scanning strategy to probe subnets. The worm scans a random address with a probability of p_0. For an address with the same first octet, the probability is given by p_1, while an address with the same first two octets is scanned with probability p_2. In order to simplify the model, both the death rate and patching rate are ignored in the AAWP model. This model assumes localized worms scan a subnet containing 2^{16} IP addresses instead of the whole Internet. This subnet is divided into three parts according to the first two octets. Subnet 1 is a special subnet, which has a larger hit-list size. The average number of infected hosts in subnet 1 is denoted b_1 and the average number of scans hitting subnet 1 is represented by k_1. Subnet 2 contains $2^8 - 1$ subnets which have the same first octet as subnet 1. The average number of infected hosts in subnet 2 is denoted by b_2 and the average number of scans hitting subnet 2 is represented by k_2. The other $2^{16} - 2^8$ subnets belong to subnet 3, which has b_3 infected hosts and k_3 scans on average. Therefore, the number of infected hosts in the next time tick is represented by (17):

$$b_{i+1} = b_i + (\frac{N}{2^{16}} - b_i)n_i[1 - (1 - \frac{1}{2^{16}})^{k_i}] \tag{17}$$

where i=1, 2, or 3. k_i (i=1, 2 or 3) indicates the total number of scans in different subnets coming from the local subnet, the same first octet subnets and the global subnets. The calculation of k_i (i=1, 2 or 3) is as follows:

$$\begin{aligned}
k_1 =& p_2 s b_1 + p_1 s \lfloor b_1 + (2^8 - 1)b_2 \rfloor / 2^8 \\
&+ p_0 s \lfloor b_1 + (2^8 - 1)b_2 + (2^{16} - 2^8)b_3 \rfloor / 2^{16} \\
k_2 =& p_2 s b_2 + p_1 s [b_1 + (2^8 - 1)b_2] 2^8 \\
&+ p_0 s [b_1 + (2^8 - 1)b_2 + (2^{16} - 2^8)b_3] / 2^{16} \\
k_3 =& p_2 s b_3 + p_1 s b_3 \\
&+ p_0 s [b_1 + (2^8 - 1)b_2 + (2^{16} - 2^8)b_3] / 2^{16}
\end{aligned}$$

The LAAWP model adopts deterministic approximation to reflect the spreading of worms that preferentially scans targets close to their addresses with a higher probability.

C. Topology-based Model

Both homogenous scan-based models and localized scan-based models reflect unstructured worms' propagation without regard to topological constraints. However, a topology-based model describes a structure dependent propagation of worms, which relies on the topology for the spreading of viruses such as email worms [14], p2p worms [13], [86], and social network worms [57], [46], [51]. In this subsection, we introduce some typical topology-based discrete-time models.

1) Email Worms Simulation Model: Zou et al. [14] presented a simulation model on the propagation of email worms. It considered the probability of opening an email attachment and email checking frequency, and then compared internet email worm propagation on power law topologies, small world topologies and random graph topologies. In the proposed model, the probability of each user opening a worm attachment can be treated as an infected probability and the distribution of email checking times can represent the propagation probability.

Due to the high likelihood that email users will also receive email from those they send email to, the Internet's email network is modeled as an undirected graph. According to the distribution of Yahoo! Email groups, authors believe the Internet email network conforms to a heavy-tailed distribution and model the email network topology as a power law network, which follows $F(\alpha) \propto K^{-\alpha}$. The constant α is the power law exponent that determines the degrees of nodes in the network. A larger maximum topology degree requires a larger power law exponent, and a larger expected value of topology degree demands a smaller power law exponent. This model uses $\alpha = 1.7$ to generate the power law network with the total number of hosts $|V| = 100000$ and an average degree of 8. The highest degree for this power law network is 1833 and the lowest degree is 3.

Email worms depend on email users' interaction to spread. When a user checks an email with a malicious attachment, this user may discard it or open the worm attachment without any security awareness. This user's behavior is represented by an opening probability $C \sim N(0.5, 0.3^2)$ in this model. Then, when a malicious email attachment is opened, the email worm immediately infects the user and sends out a worm email to all email addresses found on this user's computer. Thus, the email checking time is an important parameter that contributes to the propagation speed of the email worm. In this simulation model, the email checking time T follows a Gaussian distribution: $T \sim N(40, 20^2)$. This model discusses two cases under different infection assumptions: non-reinfection and reinfection. The main difference is whether a user in the infectious state can be infected again. If the victims can be infected each time they are visited by worms, it is assumed to be a reinfection scenario. Otherwise, infected users send out worm copies only once even if they open a worm attachment again. We refer to this as a non-reinfection scenario. This email simulation model only considers the propagation of reinfection email worms, which is described in Algorithm 1.

Algorithm 1
Simulation Model: The discrete-time email worm simulator

```
/* step 1: Initialize parameters */
1. initialize the number of infected nodes infectednum;
2. initialize the email checking time CheckingTime and
opening probability OpeningProb (both follow Gaussian
distribution);
3. initialize the number of worm emails: VirusNum,
NextVirusNum;
4. timetick = 1;

/* step 2: Sending worm emails*/
timetick = timetick + 1;
for i = 1 to the number of total email users do
    if (user i is not HEALTHY or timetick == 2) then
        if (user i is checking emails) then
            if (user i is DANGER) then
                user i is INFECTED;
                infectednum = infectednum + 1;
            end if
            for sendnum = 1 to the number of worm emails
do
                for link = 1 to all the links of user i do
                    if (user i opens a worm attachment) then
                        sending worm emails;
                    end if
                end for
            end for
            the number of user i's worm email is reset as 0
        end if
    end if
end for

/* step 3: Update Current Node Status */
for i = 1 to the number of total email users do
    if (the number of worm emails is not 0) then
        if (user i doesn't check the email) then
            if (user i is not INFECTED) then
                user i is DANGER;
            end if
            record the number of worm attachments user i
received newly
            reset user i's CheckingTime(i);
        end if
    else
        record the total number of worm attachments user i
received
    end if
    user i's CheckingTime − 1;
end for
Re_InfectedNum(timetick) = infectednum;
```

According to the discrete-time email worm simulator, the propagation of email worms on a power-law network under the non-reinfection and reinfection scenarios, as shown in Fig. 7, illustrate that the spreading speed in the reinfection case is faster and the number of infected hosts at the end of propagation is higher than the non-reinfection case. Based on this simulation model, Zou et al. studied the selective immunization defense against email worms. According to their analysis, in a power law topology, if the top 29% of the most-connected nodes are removed from the network, the email network will be broken into separated fragments and no worm outbreak will occur.

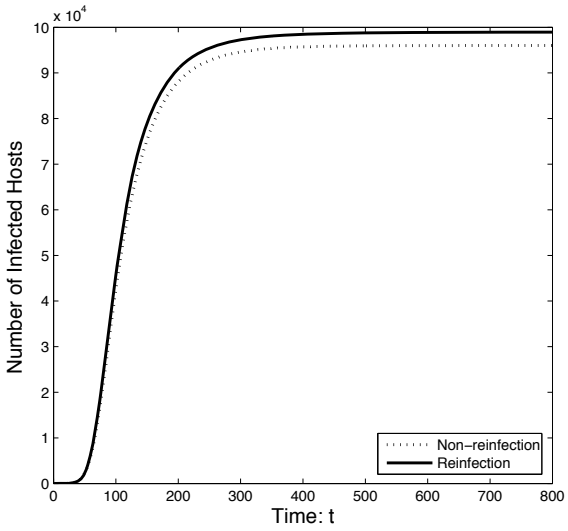


Fig. 7. Propagation on a power-law network: reinfection vs. non-reinfection

2) Logic 0-1 Matrix Model: Fan and Xiang [13] used a logic matrix approach to model the spreading of P2P worms. They presented two different topologies: a simple random graph topology and a pseudo power law topology. The research studied their impacts on a P2P worm's attack performance and analyzed related quarantine strategies for these two topologies. This model uses a logic matrix (denoted by matrix T) to represent the topology of a P2P overlay network. It adopts two constants of logic type (True or 1, False or 0) as the value of matrix variables. The logic constant 'T' indicates the existence of a directed link between two nodes in the network, and the logic constant 'F' is used to indicate there is no directed link. The i-th row of a topology logic matrix represents all outbound links of node i; and the j-th column of the topology logic matrix represents all inbound links of node j. This 0-1 matrix stands for the propagation ability of nodes, i.e. whether they can allow the virus to spread or not.

This logic 0-1 matrix model is a discrete-time deterministic propagation model of P2P worms under three different distributions: infectious state (denoted by logic vector *S*), vulnerability status (denoted by logic vector *V*) and quarantine status (denoted by logic vector *Q*). Where the logic vector S_g represents the current state g of the logical P2P overlay network and the logic vector S_{g+1} represents the next state of the logical P2P overlay network, we have:

$$S_{g+1} = S_g + S_g^{new} \tag{18}$$

Here, 1-entries in the vector S_g^{new} represent the transition to infectious at state $g+1$. S_g^{new} varies in consideration of different distributions of *S*, *V*, and *Q*. If all nodes are vulnerable to the worm and no nodes are quarantined, then

we have (19):

$$S_{g+1} = S_g + S_g T \tag{19}$$

If all nodes are not vulnerable to the worm and no nodes are quarantined, then we have (20):

$$S_{g+1} = S_g + S_g T V \tag{20}$$

If all nodes are vulnerable and some nodes are quarantined, then we have (21):

$$S_{g+1} = S_g + S_g T \bar{Q} \tag{21}$$

where $\bar{Q}$ stands for the distribution of those unquarantined nodes.

This logic 0-1 matrix model translates the propagation processes of P2P worms into a sequence of logic matrix operations. According to the analysis of this model, authors discovered the relation between out-degree, vulnerability and coverage rate in power law topologies and simple random graph topologies respectively, and then proposed quarantine strategies against P2P worms.

3) OSN (Online Social Networks) Worms Model: Fan and Yeung [46] proposed two virus propagation models based on the application network of Facebook, which is the most popular among social network service providers. The difference between email worms and Facebook worms, as the authors highlight, is that people only check if there are any new emails and then log out, while people spend more time on Facebook. In Facebook, two users' accounts appear in each other's friends list if they have confirmed their status to be friends. Thus, the topology of this network is treated as an undirected graph and is constructed by a power-law distribution in the models.

Facebook application platform based model: since Facebook provides an application platform that can be utilized by attackers to publish malicious applications, one of the worm propagation models is based on the Facebook application platform. Users of Facebook can install applications to their accounts through this platform. If a user added a malicious application, their account is infected and an invited message is sent to all their friends to persuade them to install the same application, which leads to the spreading of the worm application. The probability of installing one application for user i is:

$$P_{user}(i,t) = \frac{AppS_i(t)^{\rho} + init_{user}}{\sum_{j=1}^{N_{user}} (AppS_j(t)^{\rho} + init_{user})} \tag{22}$$

where $Apps_i(t)$ is the number of applications that user i has installed at time step t. The parameter ρ reflects the effect of preferential installation. $init_{user}$ is used to show the initial probability $P_{user}(i,t)$ of a user who does not install any application. Since there are many new installations every day, the probability of one application selected by user i from the application list is:

$$P_{app}(k,t) = \frac{Install_k(t) + init_{user}}{\sum_{j=1}^{N_{app}} (Install_j(t) + init_{user})} \tag{23}$$

where $init_{app}$ defines the initial probability $P_{app}(k,t)$ of an application without any installation. When a malicious application is installed, invitation messages are sent to all the friends of this infected user. Assuming each user has received c invitations at time step t. Then the probability the user is infected is:

$$P_{virus} = \frac{\alpha}{(1 - \frac{Install_{N_{app}}(t)}{N_{user}} \cdot \frac{APPS_i(t)}{N_{app}})^c} \tag{24}$$

where σ is the percentage of users who accepted the invitations. The infected number $I(t)$ is changed when a malicious application is installed.

Sending messages based model: this model investigates the propagation of worms through the sending of messages to friends, which is similar to email worm propagation. When users of Facebook receive malicious emails and click them, these users are infected and worm email copies are sent to their friends. At each time tick, a user can log-in to Facebook with a log-in time $T_{login}(i)$, which follows an exponential distribution. The mean value of $T_{login}(i)$ follows a Gaussian distribution $N(\mu_{Tl}(t), \sigma_{Tl}^2)$. The online time that users spend on Facebook is $T_{online}(i)$, which follows a Gaussian distribution $N(\mu_{To}(t), \sigma_{To}^2)$. All of the online users may open the malicious email with a probability of P_{click}, which follows a Gaussian distribution $N(\mu_p(t), \sigma_p^2)$. The worm propagates until no more new users are infected in the online social network.

4) Spatial-temporal Model: In the work of Chen and Ji [68], a spatial-temporal random process was used to describe the statistical dependence of malware propagation in arbitrary topologies. This spatial-temporal model is a stochastic discrete time model that reflects the temporal dependence and the spatial dependence in the propagation of malware. The temporal dependence means that the status of node i (infected or susceptible) at time $t+1$ depends on the status of node i at time t and the status of its neighbors at time t. The temporal dependence of node i can be shown as (25) and (26):

$$P(X_i(t+1) = 0 \mid X_i(t) = 0) = \delta_i \tag{25}$$

$$P(X_i(t+1) = 1 \mid X_i(t) = 0, X_{N_i}(t) = x_{N_i}(t)) = \beta_i(t) \tag{26}$$

where $X_i(t)$ denotes the status of a network node i at time t (t represents discrete time): if node i is infected at time, $X_i(t) = 1$; if node i is susceptible at time t, $X_i(t) = 0$. $X_{Ni}(t)$ is used to denote the status of all neighbors of node i at time t and the vector $x_{Ni}(t)$ is the realization of $X_{Ni}(t)$. If node i is susceptible at time t, it can be compromised by any of its infected neighbors and become infected at the next time step $t+1$ with a birth rate $\beta_i(t)$. Otherwise, node i is infected and has a death rate δ_i to recover at the next time step $t+1$. The transition probabilities characterize the temporal evolution due to infection and recovery.

Denoting by $R_i(t)$, the probability that node i recovers from infected to susceptible status at time $t+1$, is:

$$R_i(t) = P(X_i(t+1) = 0, X_i(t) = 1) = \delta_i P(X_i(t) = 1) \tag{27}$$

If node i is susceptible at time t, the probability that node i remains susceptible at the next time step can be defined as:

$$\begin{aligned} S_i(t) &= P(X_i(t+1)=0|X_i(t)=0) \\ &= \sum_{x_{N_i}(t)} [P(X_{N_i}(t)=x_{N_i}(t)|X_i(t)=0)(1-\beta_i(t))] \end{aligned} \tag{28}$$

where a joint probability $P(X_{Ni}(t) = x_{Ni}(t) \mid X_i(t) = 0)$ representing the status of all neighbors of node i at time t characterizes the spatial dependence according to the network topology and the interaction between nodes. Based on (27) and (28), the probability that node i is infected at time $t+1$ can be represented by (29).

$$P(X_i(t+1)=1) = 1 - R_i(t) - S_i(t)P(X_i(t)=0) \tag{29}$$

Formula (24) reflects an iteration process of malware propagation according to the status of a node at time t and the status of all neighbors of this node i at time t, which characterizes the spatial and temporal statistical dependencies. Consequently, the expected number of infected nodes at time t, $n(t)$, can be computed:

$$n(t) = E\lfloor \sum_{i=1}^{M} X_i(t) \rfloor = \sum_{i=1}^{M} P(X_i(t)=1) \tag{30}$$

Though (24) can be used to study the behavior of malware propagation, the cost of computing $S_i(t)$ is large especially when a node has a great number of neighbors. Therefore, authors presented two models to simplify the challenge posed by the spatial dependence: the Independent Model and the Markov Model.

The *Independent Model* assumes that the status of all nodes at time t is spatially independent. This means no propagation cycles are formed when worms propagate via some intermediate nodes because the infected probability of a node is not influenced by its neighbors. Thus, the independent model neglects the spatial dependence. However, the status of a node at a given time is related to its status at the last time tick and thus, it still remains temporally dependent. The state evolution of node i in the independent model can be represented by (31):

$$P(X_i(t+1)=1) = 1 - R_i(t) - S_i^{ind}(t)P(X_i(t)=0) \tag{31}$$

where

$$S_i^{ind}(t) = \prod_{j \in N_i} \lfloor 1 - \beta_{ji} P(X_j(t)=1) \rfloor$$

The *Markov Model* assumes that the status of a node is related to its neighbors, but its neighbors cannot be influenced by each other at the same time. This assumption can result in propagation cycles via a single intermediate node, however this can be solved with conditional independence in the network space. If the status of node i's neighbors at the same time step is spatially independent give the status of node i, then the state evolution of a node in the Markov model can be represented by (32):

$$P(X_i(t+1)=1) = 1 - R_i(t) - S_i^{mar}(t)P(X_i(t)=0) \tag{32}$$

where

$$S_i^{mar}(t) = \prod_{j \in N_i} \lfloor 1 - \beta_{ji} P(X_j(t)=1|X_i(t)=0) \rfloor$$

D. Comparison of Worm Propagation Models

A comparison of the various mathematical models of worms discussed above is summarized in Table I. The classical simple epidemic model is the most widely used model for investigating the propagation of scan-based worms using a continuous-time differential equation. Some previous works, such as the uniform scan worm model and the RCS model, are derived from the classical simple epidemic model, which assumes two states for all hosts: susceptible and infectious, and will stay in the infectious state forever when a host is infected. However, these models are not suitable for cases where the infected and infectious nodes are patched or removed. Consequently, the classical general epidemic model (Kermack-McKendrick model) has been proposed to extend simple epidemic models by introducing a removal process of infectious peers. Continued improvements [39], [87] on modeling worm propagation have considered immunization defense. Zou et al. [39] proposed a two-factor worm model, which developed the general epidemic model by taking into account both the effect of human countermeasures and decreases in the infection rate.

The above models adopt a continuous-time differential equation to observe and predict worm spreading in the network. As scanning IP addresses or logical neighbors is usually performed in discrete time [88], a host cannot infect other hosts before it is infected completely. Thus, strictly speaking, the propagation of worms is a discrete event process. A continuous-time model can possibly result in a different spreading speed and infected scale because a host begins devoting itself to infecting other hosts even though only a "small part" of it is infected. Consequently, modeling worm propagation at each discrete time tick is more accurate than using continuous time. The AAWP model, the LAAWP model and the Bluetooth worm model are constructed according to a discrete event process. The AAWP model characterizes the spread of active worms that employ random scanning. LAAWP is extended from the AAWP model and takes into account the characteristics of local subnet scanning worms spreading. The Bluetooth worm model analyzes the propagation dynamics of Bluetooth worms. It captures not only the behavior of the Bluetooth protocol but also the impact of mobility patterns on the propagation of Bluetooth worms.

All of the above models including continuous-time and discrete-time rely on the homogenous mixing assumption that any infected host has equal opportunity to infect any vulnerable host in the network. However, worms that use a localized scanning strategy, such as Code Red II, require non-homogenous consideration of population locality [10]. Consequently, the local preference model assumes a local preference scanning worm has probability p to uniformly scan addresses which share its first n bits in the network and probability $(1-p)$ to uniformly scan other addresses. Besides, Zou et al. [14] analyzed the propagation of email worms and pointed out that models based on the homogenous mixing assumption overestimate the propagation speed of an epidemic

TABLE I
A COMPARISON OF WORM PROPAGATION MODELS

Worm Propagation Models	Network Topology	Graphical Representation of Topology	Modeling Method	Propagation Process	Model Type	Infection Type
Classical Simple Epidemic Model	H	UG	A	C	SI	Not considered
Uniform Scan Worm Model	H	UG	A	C	SI	Not considered
RCS Model	H	UG	A	C	SI	Not considered
Classical General Epidemic Model	H	UG	A	C	SIR	Not considered
Two-factor Model	H	UG	A	C	SIR	Not considered
AAWP Model	H	UG	A	D	SIR	Non-reinfection
Bluetooth Worm Model	H	UG	A	D	SI	Not considered
Local Preference Model	Non-H	UG	A	C	SI	Not considered
LAAWP Model	Non-H	UG	A	D	SIR	Non-reinfection
Email Worms Simulation Model	R/SW/PL	UG	S	D	SI	Reinfection
Logic 0-1 Matrix Model	R/PL	DG	A	D	SIR	Non-reinfection
OSN Worms Model	PL	UG	S	D	SI	Non-reinfection
Spatial-temporal Model	H/PL	DG	A	D	SIS	Non-reinfection

H: homogenous mixing; R: random network; SW: small-world network; PL: power-law network;
UG: undirected graph; DG: directed graph;
C: continuous-time event; D: discrete-time event;
A: analytical; S: simulation;
SI: susceptible-infected model; SIR: susceptible-infected-recovered model; SIS: susceptible-infected-susceptible model

in a topological network, especially in the early stages when a small number of nodes are infected and clustered with each other. In order to avoid overestimation, the researchers provide a discrete-time simulation model and mainly study the email worm propagation over a power-law topology. This simulation model can more accurately simulate the propagation of email worms than previous homogenous mixing differential equation models. However, this model describes the email worm propagation tendency instead of modeling the dynamic spreading procedure between each pair of nodes. Secondly, they discussed the lower bound for the non-reinfection case, but their model is not capable of accurately eliminating the errors caused by reinfection. Moreover, some assumptions are not realistic. For example, the authors believe that just one malicious email copy will be sent to recipients even if an infected user checks multiple emails containing worms. In reality, a malicious copy is sent whenever the infected user opens a re-infection worm email.

This logic 0-1 matrix model employs a logic matrix to represent links between each pair of hosts and models the spreading of peer-to-peer worms over a pseudo power-law topology. This model can examine deep inside the propagation procedure among nodes in the network. The model cannot avoid propagation cycles formed among intermediate nodes although it does not allow peers to have outbound links to themselves. These propagation cycles lead to the overestimation in the scale of the infected network. Besides this, their logic matrix is weak regarding an email resembling network because the weight of each link is a probability value ranging from zero to one instead of constant zero or one. The model does not consider the propagation probability and infected probability of each node, which has significant impacts on the infection procedure.

Social networks have become attractive targets for worms. Fan and Yeung [46] proposed the OSN worm model to characterize the behavior of a worm spreading on the application network of Facebook. However, these two models assume a user starts infecting others at every moment once the user is infected. In practice however, infected users spread worms only as they periodically accept invitations and install malicious applications or check newly received messages and open malicious links. As a result, they have neglected a realistic temporal delay process. Furthermore, the second model simulates the scenario of non-reinfection worm propagation, however non-reinfection worms mainly appear in the early worm cases and are not appropriate for modeling modern email worms that spread over social networks.

The above models assume computer users behave independently, that is, the status of all hosts at the same time step is spatially independent. In real scenarios, however, the propagation of topology-based worms needs human activation and thus the spreading procedure is spatial and temporally dependent. Chen et al. [68] used a spatial-temporal random process to describe the statistical dependence of worm propagation in arbitrary topologies. Although this model can outperform the previous models through capturing temporal dependence and detailed topology information, there are also some weak assumptions made. Firstly, this model adopts a SIS model, even though infected users are not likely to be infected again after they clean their computers by patching vulnerabilities or updating anti-virus software. Secondly, their model assumes that an infected computer cannot be reinfected. However, recent email worms often reinfect users, and are far more aggressive in spreading throughout the network. Thirdly, the authors ignore an important consideration regarding human behavior; the email checking time, which has been shown to greatly affect the propagation of email worms.

VI. DISCUSSION

A. Limitations

In order to eradicate topology-based worms, as well as to control and limit the impact of their outbreak, previous works [13], [14], [57] presented certain strategies to immunize a group of users in the network to prevent topology-based worms from propagating to a large scale. However, how to choose the appropriate size and membership of this subset

to constrain topological worm spreading remains a difficult question. A common view for the preferable positions of defense is at the highly-connected users [13], [14] or those with most active neighbors [57]. Indeed, popular users in a scale-free network and their intuitively short paths to other nodes in a strongly clustered small world [49], [51] greatly facilitate the propagation of an infection over the whole network, particularly at their early stage. However, counter-intuitively, recent research [69] suggested that this viewpoint may not be always the truth. Further discussion on this problem will be presented in our future work.

To the best of our knowledge, all existing works use simulation to evaluate their effectiveness. In general, people may question the accuracy of the proposed models since there is no evidence from the real world. Therefore, the first challenge in this field is to collect available real data to support the validation process. Moreover, the propagation of worms is actually affected by many factors such as time zone [89], human involvement [57]. However, seldom previous works have examined the impact of those parameters. Thus, another challenge in this field is to comprehensively analyze the overall impact on the propagation procedure of worms. Currently, social networks become more and more popular. In the meantime, an increasing number of social network worms appear and become severe threats to the Internet. In the future, modeling worms' propagation on social networks will draw more attention in our research.

Furthermore, mobile communication and cloud computing technologies boosted in recent years. Besides, virtual environment has also been widely applied in both research and industries. The worms can take advantage of these platforms and spread widely, such as mobile computing worms [12], cloud computing worms [90], worms spreading in virtual machine [91]. In order to understand the propagation properties of these worms, current researchers have extended previous models to present their propagation dynamics. The key point of this work is to capture the specific spreading characteristics so that the proposed models can accurately disclose the propagation procedure of these worms. This is one of our future works.

B. Lessons Learned

Computer worm modeling is crucial for understanding the dynamic impact of worm attacks. It provides a comprehensive approach to help researchers study the fundamental spreading patterns that characterize a worm outbreak. On the basis of it, people can then predict their potential damages and develop effective countermeasures. The modeling consists of building either a simulation model [14], [57], [46], [73], [92] or an analytical model [68], [71]. Simulation is an effective technique that is used to understand and study the tendency of worm propagation. Researchers can derive the probability of either state for each node by averaging many runs of simulation, but simulation models cannot quantify the reasons why initial parameters result in such probabilities, and further disclose the essence. For example, Zou et al. [14] relied on simulation modeling rather than on mathematical analysis. Their paper demonstrates a fairly comprehensive analysis on the impact of various parameters, different topologies and selective percolation. However, this model describes the email worm propagation tendency instead of modeling the dynamic spreading procedure between each pair of nodes. Thus, it poorly estimates the spreading speed of email worms. In addition, the works [14], [57], [46], [73] rely on simulations to model the propagation of social network worms. Their simulation models avoid the problem of "homogeneous mixing" assumption but cannot provide analytical study on the propagation. On the contrary, the analytical method can give us an insight into the impact of each worm or network parameter on the propagation of the worm. An accurate analytical model allows researchers to comprehensively study how a worm propagates under various conditions. For instance, Wen et al. [69] adopted an analytical method to locate the most suitable positions for slowing down the worm propagation. Based on the proposed analytical model, they investigate deeply on locating the best positions for thwarting the propagation of topology-based worms. It helps efficiently suppress the infected scale of the network and decrease the spreading speed of topological worms. The results of this paper support the fact that the most popular nodes may not be the most important nodes to prevent the propagation of worms. However, which group of nodes are the most important nodes is still hard to be answered from their work. Moreover, The works [45], [77], [93] used analytical models and focused on finding threshold conditions for fast extinction of worms.

VII. Conclusion

Worms and their variants are widely believed to be one of the most serious challenges in network security research. Although in recent years propagation mechanisms used by worms have evolved with the proliferation of data transmission, instant messages and other communication technologies, scan-based techniques and topology-based techniques are still the two main means for the spreading of worms. Modeling the propagation of worms can help us understand how worms spread and enable us to devise effective defense strategies. Therefore, a variety of models have been proposed for modeling the propagation mechanism. This survey firstly introduced the target discovery techniques for scan-based worms and topology-based worms respectively, illustrating their scanning methods with graphical representations. Secondly, it analyzed the characteristics of four common topologies for modeling worm propagation. Finally, this survey has described some typical mathematical models of worms that are the analytical tools for investigating dynamics and measuring the propagation of worms. We compared these modes and discussed the pros and cons of each model. An ideal worm propagation model can reflect accurate spreading tendency as time elapsed. However, topology-based worms, such as social network worms, rely on the topology of social networks, which may result in a problem of spatial dependence in the propagation procedure. This means that compromised users will infect their neighbors but the probabilities for those compromised users being infected may be due to their neighbors having been infected before and then spreading the worm to these compromised users. This results in redundant computation of infection probabilities. In order to simplify this problem, some research has assumed the status of all nodes at each

time tick to be spatially independent. However, it is a weak approximation to the spreading dynamics. Therefore, it is worthwhile to discover what the spatial dependence is and how to approximate it so that we can eliminate the redundancy, describe the real spreading probability and provide an accurate propagation trend.

REFERENCES

[1] D. Moore, C. Shannon *et al.*, "Code-red: a case study on the spread and victims of an internet worm," in *Proc. 2nd ACM SIGCOMM Workshop on Internet measurment*. ACM, 2002, pp. 273–284. [Online]. Available: http://dl.acm.org/citation.cfm?id=637244

[2] H. Berghel, "The code red worm," *Commun. of the ACM*, vol. 44, no. 12, pp. 15–19, 2001.

[3] H. V. Poor, "An introduction to signal detection and estimation," *New York, Springer-Verlag, 1988, 559 p.*, vol. 1, 1988.

[4] A. L. Foster, "Colleges brace for the next worm," *The Chronicle of Higher Education*, vol. 50, no. 28, p. A29, 2004.

[5] V. Weafer. (2010) Downadup/conficker and april fools day: One year later. [Online]. Available: http://www.symantec.com/connect/blogs/downadupconficker-and-april-fool-s-day-one-year-later

[6] Symantec. (2008) W32.downadup (win32/conficker). [Online]. Available: http://www.symantec.com/security_response/writeup.jsp?docid=2008-112203-2408-99

[7] ——. (2010) W32.stuxnet, 2010. [Online]. Available: http://www.symantec.com/security_response/writeup.jsp?docid=2010-071400-3123-99

[8] M. Fossi, G. Egan, K. Haley, E. Johnson, T. Mack, T. Adams, J. Blackbird, M. K. Low, D. Mazurek, D. McKinney *et al.*, "Symantec internet security threat report trends for 2010," *Volume*, vol. 16, p. 20, 2011.

[9] J. Hu, X. Yu, D. Qiu, and H.-H. Chen, "A simple and efficient hidden markov model scheme for host-based anomaly intrusion detection," *IEEE Network*, vol. 23, no. 1, pp. 42–47, 2009. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4804323

[10] S. Staniford, V. Paxson, N. Weaver *et al.*, "How to own the internet in your spare time." in *USENIX Security Symp.*, 2002, pp. 149–167.

[11] Z. Chen, L. Gao, and K. Kwiat, "Modeling the spread of active worms," in *INFOCOM 2003. 22nd Annu. Joint Conf. IEEE Comput. Commun. IEEE Societies*, vol. 3. IEEE, 2003, pp. 1890–1900. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1209211

[12] G. Yan and S. Eidenbenz, "Modeling propagation dynamics of bluetooth worms (extended version)," *IEEE Trans. Mobile Computing*, vol. 8, no. 3, pp. 353–368, 2009. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4624266

[13] X. Fan and Y. Xiang, "Modeling the propagation of peer-to-peer worms," *Future generation computer systems*, vol. 26, no. 8, pp. 1433–1443, 2010. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X10000737

[14] C. C. Zou, D. Towsley, and W. Gong, "Modeling and simulation study of the propagation and defense of internet e-mail worms," *IEEE Trans. Dependable and Secure Computing*, vol. 4, no. 2, pp. 105–118, 2007. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4198176

[15] H. Andersson and T. Britton, *Stochastic epidemic models and their statistical analysis*. Springer New York, 2000, vol. 4.

[16] Z. Chen and C. Ji, "Importance-scanning worm using vulnerable-host distribution," in *IEEE Global Telecommunications Conf., 2005. GLOBECOM'05.*, vol. 3. IEEE, 2005, pp. 6–pp. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1577955

[17] ——, "Optimal worm-scanning method using vulnerable-host distributions," *Int. J. Secur. Netw.*, vol. 2, no. 1/2, pp. 71–80, 2007.

[18] Z. Chen and C. Chen, "A closed-form expression for static worm-scanning strategies," in *IEEE Int. Conf. Commun., 2008. ICC'08.* . IEEE, 2008, pp. 1573–1577. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4533340

[19] Z. Chen, C. Chen, and C. Ji, "Understanding localized-scanning worms," in *IEEE Int. Performance, Computing, and Commun. Conf., 2007. IPCCC 2007.* . IEEE, 2007, pp. 186–193. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4197930

[20] Q. Wang, Z. Chen, C. Chen, and N. Pissinou, "On the robustness of the botnet topology formed by worm infection," in *2010 IEEE Global Telecommunications Conf. (GLOBECOM 2010)*. IEEE, 2010, pp. 1–6. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5684002

[21] W. Yu, X. Wang, P. Calyam, D. Xuan, and W. Zhao, "On detecting camouflaging worm," in *Computer Security Applications Conf., 2006. ACSAC'06. 22nd Annu.*. IEEE, 2006, pp. 235–244. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4041170

[22] ——, "Modeling and detection of camouflaging worm," *IEEE Trans. Dependable and Secure Computing*, vol. 8, no. 3, pp. 377–390, 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5444887

[23] Y. Xiang, X. Fan, and W. Zhu, "Propagation of active worms: a survey," *Int. J. Comput. Systems Science & Engineering*, vol. 24, no. 3, pp. 157–172, 2009.

[24] X. Fan and Y. Xiang, "Defending against the propagation of active worms," *The Journal of Supercomputing*, vol. 51, no. 2, pp. 167–200, 2010. [Online]. Available: http://link.springer.com/article/10.1007/s11227-009-0283-8

[25] T. Yong, L. Jiaqing, X. Bin, and W. Guiyi, "Concept, characteristics and defending mechanism of worms," *IEICE TRANS. Information and Systems*, vol. 92, no. 5, pp. 799–809, 2009. [Online]. Available: http://search.ieice.org/bin/summary.php?id=e92-d_5_799

[26] P. Li, M. Salour, and X. Su, "A survey of internet worm detection and containment," *IEEE Commun. Surveys & Tutorials*, vol. 10, no. 1, pp. 20–35, 2008. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4483668

[27] N. Weaver, V. Paxson, S. Staniford, and R. Cunningham, "A taxonomy of computer worms," in *Proc. 2003 ACM workshop on Rapid malcode*. ACM, 2003, pp. 11–18. [Online]. Available: http://dl.acm.org/citation.cfm?id=948190

[28] E. Levy, "Worm propagation and generic attacks," *IEEE Security & Privacy*, vol. 3, no. 2, pp. 63–65, 2005. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1423964

[29] C. Shannon and D. Moore, "The spread of the witty worm," *IEEE Security & Privacy*, vol. 2, no. 4, pp. 46–50, 2004. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1324598

[30] Symantec. (2004) W32.sasser.worm. [Online]. Available: http://www.symantec.com/security_response/writeup.jsp?docid=2004-050116-1831-99.

[31] D. M. Kienzle and M. C. Elder, "Recent worms: a survey and trends," in *Proc. 2003 ACM workshop on Rapid malcode*. ACM, 2003, pp. 1–10. [Online]. Available: http://dl.acm.org/citation.cfm?id=948189

[32] F-Secure. (2011) Love letter virus. [Online]. Available: http://www.f-secure.com/v-descs/love.shtml.

[33] Iloveyou virus lessons learned report. Army Forces Command. [Online]. Available: http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA415104&Location=U2&doc=GetTRDoc.pdf

[34] Symantec. (2001) W32.sircam.worm. [Online]. Available: http://www.symantec.com/security_response/writeup.jsp?docid=2001-071720-1640-99&tabid=2

[35] D. Moore, V. Paxson, S. Savage, C. Shannon, S. Staniford, and N. Weaver, "Inside the slammer worm," *IEEE Security & Privacy*, vol. 1, no. 4, pp. 33–39, 2003. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1219056

[36] (2001) Advisories and alerts: .ida code red worm. eEye Digital Security. [Online]. Available: http://www.eeye.com/html/Research/Advisories/AL20010717.html.

[37] C. C. Zou, D. Towsley, and W. Gong, "On the performance of internet worm scanning strategies," *Performance Evaluation*, vol. 63, no. 7, pp. 700–723, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0166531605001112

[38] 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 Workshop on Principles of Advanced and Distributed Simulation*. IEEE Computer Society, 2005, pp. 199–206. [Online]. Available: http://dl.acm.org/citation.cfm?id=1070175

[39] C. C. Zou, W. Gong, and D. Towsley, "Code red worm propagation modeling and analysis," in *Proc. 9th ACM conference on Computer and communications security*. ACM, 2002, pp. 138–147. [Online]. Available: http://dl.acm.org/citation.cfm?id=586130

[40] K. R. Rohloff and T. Basar, "Stochastic behavior of random constant scanning worms," in *Comput. Commun. Netw., 2005. ICCCN 2005. Proc. 14th Int. Conf.*. IEEE, 2005, pp. 339–344. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1523881

[41] S. H. Sellke, N. B. Shroff, and S. Bagchi, "Modeling and automated containment of worms," *IEEE Trans. Dependable and Secure Computing*, vol. 5, no. 2, pp. 71–86, 2008. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4358715

[42] Y. Wang, S. Wen, S. Cesare, W. Zhou, and Y. Xiang, "The microcosmic model of worm propagation," *The Computer Journal*,

vol. 54, no. 10, pp. 1700–1720, 2011. [Online]. Available: http://comjnl.oxfordjournals.org/content/54/10/1700.short

[43] (2001) blaster worm analysis. eEye Digital Security. [Online]. Available: http://www.eeye.com/html/Research/Advisories/AL20030811.html

[44] N. Weaver, "A brief history of the worm," *Security Focus Online*, vol. 26, 2001.

[45] R. W. Thommes and M. Coates, "Epidemiological modelling of peer-to-peer viruses and pollution." in *INFOCOM*, vol. 6, 2006, pp. 1–12.

[46] W. Fan and K. Yeung, "Online social networksparadise of computer viruses," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 2, pp. 189–197, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378437110008344

[47] W32.koobface. [Online]. Available: http://www.symantec.com/security_response/writeup.jsp?docid=2008-080315-0217-99.

[48] P. Erdos and A. Rényi, "On the evolution of random graphs," *Publ. Math. Inst. Hungar. Acad. Sci*, vol. 5, pp. 17–61, 1960.

[49] H. Ebel, L.-I. Mielsch, and S. Bornholdt, "Scale-free topology of e-mail networks," *arXiv preprint cond-mat/0201476*, 2002. [Online]. Available: http://arxiv.org/abs/cond-mat/0201476

[50] M. Boguná, R. Pastor-Satorras, and A. Vespignani, "Epidemic spreading in complex networks with degree correlations," *Statistical Mechanics of Complex Networks–Lectures Notes in Physics*, vol. 625, 2003.

[51] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proc. 7th ACM SIGCOMM conf. Internet measurement*. ACM, 2007, pp. 29–42. [Online]. Available: http://dl.acm.org/citation.cfm?id=1298311

[52] A. Vazquez, B. Racz, A. Lukacs, and A.-L. Barabasi, "Impact of non-poissonian activity patterns on spreading processes," *Physical review letters*, vol. 98, no. 15, p. 158702, 2007. [Online]. Available: http://prl.aps.org/abstract/PRL/v98/i15/e158702

[53] J. Kleinberg and R. Rubinfeld, "Short paths in expander graphs," in *Foundations of Computer Science, 1996. Proc., 37th Annu. Symp.*. IEEE, 1996, pp. 86–95. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=548467

[54] M. Molloy and B. Reed, "A critical point for random graphs with a given degree sequence," *Random structures & algorithms*, vol. 6, no. 2-3, pp. 161–180, 1995. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1002/rsa.3240060204/full

[55] ——, "The size of the giant component of a random graph with a given degree sequence," *Combinatorics probability and computing*, vol. 7, no. 3, pp. 295–305, 1998. [Online]. Available: http://journals.cambridge.org/production/action/cjoGetFulltext?fulltextid=46636

[56] D. J. Watts and S. H. Strogatz, "Collective dynamics of small-worldnetworks," *nature*, vol. 393, no. 6684, pp. 440–442, 1998. [Online]. Available: http://www.nature.com/nature/journal/v393/n6684/abs/393440a0.html

[57] G. Yan, G. Chen, S. Eidenbenz, and N. Li, "Malware propagation in online social networks: nature, dynamics, and defense implications," in *Proc. 6th ACM Symposium on Information, Computer and Communications Security*. ACM, 2011, pp. 196–206. [Online]. Available: http://dl.acm.org/citation.cfm?id=1966939

[58] N. Li and G. Chen, "Analysis of a location-based social network," in *Computational Science and Engineering, 2009. CSE'09. Int. Conf.*, vol. 4. IEEE, 2009, pp. 263–270. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5284112

[59] M. Jovanović, F. Annexstein, and K. Berman, "Modeling peer-to-peer network topologies through small-world models and power laws," in *IX Telecommunications Forum, TELFOR*, 2001, pp. 1–4.

[60] L. Adamic, O. Buyukkokten, and E. Adar, "A social network caught in the web," *First Monday*, vol. 8, no. 6, 2003. [Online]. Available: http://ojs-prod-lib.cc.uic.edu/ojs/index.php/fm/article/view/1057

[61] R. Kumar, J. Novak, and A. Tomkins, "Structure and evolution of online social networks," in *Link Mining: Models, Algorithms, and Applications*. Springer, 2010, pp. 337–357. [Online]. Available: http://link.springer.com/chapter/10.1007/978-1-4419-6515-8_13

[62] Y.-Y. Ahn, S. Han, H. Kwak, S. Moon, and H. Jeong, "Analysis of topological characteristics of huge online social networking services," in *Proc. 16th int. conf. on World Wide Web*. ACM, 2007, pp. 835–844. [Online]. Available: http://dl.acm.org/citation.cfm?id=1242685

[63] Y. Li, W. Chen, Y. Wang, and Z.-L. Zhang, "Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships," in *Proc. 6th ACM int. conf. on Web search and data mining*. ACM, 2013, pp. 657–666. [Online]. Available: http://dl.acm.org/citation.cfm?id=2433478

[64] Y. Li, B. Q. Zhao, and J. Lui, "On modeling product advertisement in large-scale online social networks," *IEEE/ACM Trans. Netw. (TON)*, vol. 20, no. 5, pp. 1412–1425, 2012. [Online]. Available: http://dl.acm.org/citation.cfm?id=2428703

[65] V. Braitenberg and A. Schüz, *Anatomy of the cortex: Statistics and geometry*. Springer-Verlag Publishing, 1991.

[66] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins, "Trawling the web for emerging cyber-communities," *Computer networks*, vol. 31, no. 11, pp. 1481–1493, 1999. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128699000407

[67] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *science*, vol. 286, no. 5439, pp. 509–512, 1999. [Online]. Available: http://www.sciencemag.org/content/286/5439/509.short

[68] Z. Chen and C. Ji, "Spatial-temporal modeling of malware propagation in networks," *IEEE Trans. Neural Netw.*, vol. 16, no. 5, pp. 1291–1303, 2005. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1510727

[69] S. Wen, W. Zhou, Y. Wang, W. Zhou, and Y. Xiang, "Locating defense positions for thwarting the propagation of topological worms," *IEEE Commun. Lett.*, vol. 16, no. 4, pp. 560–563, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6168149

[70] D. Moore and C. Shannon, "Caida: The nyxem email virus: analysis and inferences," 2004.

[71] S. Wen, W. Zhou, J. Zhang, Y. Xiang, and W. Jia, "Modeling propagation dynamics of social network worms," *IEEE Trans. Parallel Distrib. Syst.*, 2012. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6287505

[72] Y. Moreno, R. Pastor-Satorras, and A. Vespignani, "Epidemic outbreaks in complex heterogeneous networks," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 26, no. 4, pp. 521–529, 2002. [Online]. Available: http://link.springer.com/article/10.1140/epjb/e20020122

[73] C. Gao, J. Liu, and N. Zhong, "Network immunization and virus propagation in email networks: experimental evaluation and analysis," *Knowledge and information systems*, vol. 27, no. 2, pp. 253–279, 2011. [Online]. Available: http://link.springer.com/article/10.1007/s10115-010-0321-0

[74] B. Rozenberg, E. Gudes, and Y. Elovici, "Sisr–a new model for epidemic spreading of electronic threats," in *Information Security*. Springer, 2009, pp. 242–249. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-04474-8_20

[75] Y. Wang, S. Wen, S. Cesare, W. Zhou, and Y. Xiang, "Eliminating errors in worm propagation models," *IEEE Commun. Lett.*, vol. 15, no. 9, pp. 1022–1024, 2011. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5958564

[76] P. Mahadevan, D. Krioukov, K. Fall, and A. Vahdat, "Systematic topology analysis and generation using degree correlations," *ACM SIGCOMM Comput. Commun. Review*, vol. 36, no. 4, pp. 135–146, 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1159930

[77] D. Chakrabarti, J. Leskovec, C. Faloutsos, S. Madden, C. Guestrin, and M. Faloutsos, "Information survival threshold in sensor and p2p networks," in *INFOCOM 2007. 26th IEEE Int. Conf. Comput. Commun.*. IEEE, 2007, pp. 1316–1324. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4215738

[78] M. Roberts and J. Heesterbeek, "Mathematical models in epidemiology," *Mathematical models*, 2003.

[79] R. Pastor-Satorras and A. Vespignani, "Epidemic spreading in scale-free networks," *Physical review letters*, vol. 86, no. 14, p. 3200, 2001. [Online]. Available: http://prl.aps.org/abstract/PRL/v86/i14/p3200_1

[80] Y. Moreno, J. B. Gómez, and A. F. Pacheco, "Epidemic incidence in correlated complex networks," *Physical Review E*, vol. 68, no. 3, p. 035103, 2003. [Online]. Available: http://pre.aps.org/abstract/PRE/v68/i3/e035103

[81] D. Moore, "Caida analysis of code-red," *The Cooperate Association for Internet Data Analysis*, 2001.

[82] D. Moore and C. Shannon, "The spread of the code-red worm (crv2)," 2001.

[83] D. J. Daley, J. Gani, and J. J. M. Gani, *Epidemic modelling: an introduction*. Cambridge University Press, 2001, vol. 15.

[84] R. M. Anderson, R. M. May, and B. Anderson, *Infectious diseases of humans: dynamics and control*. Wiley Online Library, 1992, vol. 28.

[85] N. T. Bailey *et al.*, *The mathematical theory of infectious diseases and its applications*. Charles Griffin & Company Ltd, 5a Crendon Street, High Wycombe, Bucks HP13 6LE., 1975.

[86] J. Wu, S. Vangala, L. Gao, and K. A. Kwiat, "An effective architecture and algorithm for detecting worms with various scan." in *NDSS*, 2004.

[87] C. Wang, J. C. Knight, and M. C. Elder, "On computer viral infection and the effect of immunization,"

in *Computer Security Applications, 2000. ACSAC'00. 16th Annu. Conf.*. IEEE, 2000, pp. 246–256. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=898879

[88] S. Xing and B.-P. Paris, "Measuring the size of the internet via importance sampling," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 6, pp. 922–933, 2003. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1217278

[89] D. Dagon, C. C. Zou, and W. Lee, "Modeling botnet propagation using time zones." in *NDSS*, vol. 6, 2006, pp. 2–13.

[90] S. Biedermann and S. Katzenbeisser, "Detecting computer worms in the cloud," in *Open Problems in Network Security*. Springer, 2012, pp. 43–54. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-642-27585-2_4

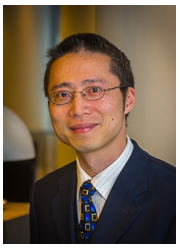
[91] T. Garfinkel and M. Rosenblum, "When virtual is harder than real: Security challenges in virtual machine based computing environments." in *HotOS*, 2005.

[92] C. Gao and J. Liu, "Modeling and restraining mobile virus propagation," *IEEE Trans. Mob. Compu*, 2013. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6138859

[93] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos, "Epidemic spreading in real networks: An eigenvalue viewpoint," in *Reliable Distributed Systems, 2003. Proc. 22nd Int. Symp.*. IEEE, 2003, pp. 25–34. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1238052

Dr. Yini Wang is a postdoctoral researcher at IBM Australia Research Lab. At IBM, her focus is on disaster management, smarter finical life and social network. She received her PhD in Computer Science from Deakin University, Australia in 2012. Her research interests include network and system security, distributed systems and networking.

Sheng Wen graduated with a degree in computer science from Central South University of China in 2012. He is currently working toward the Ph.D. degree at the school of information technology, Deakin University, Melbourne, Australia, under the supervision of Prof. Wanlei Zhou and Prof. Yang Xiang. His focus is on modelling of virus spread, information dissemination and defence strategies of the Internet threats.



Professor Yang Xiang received his PhD in Computer Science from Deakin University, Australia. He is currently a full professor at School of Information Technology, Deakin University. He is the Director of the Network Security and Computing Lab (NSCLab). His research interests include network and system security, distributed systems, and networking. In particular, he is currently leading his team developing active defense systems against large-scale distributed network attacks. He is the Chief Investigator of several projects in network and system security, funded by the Australian Research Council (ARC). He has published more than 130 research papers in many international journals and conferences, such as IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, IEEE Transactions on Information Security and Forensics, and IEEE Journal on Selected Areas in Communications. Two of his papers were selected as the featured articles in the April 2009 and the July 2013 issues of IEEE Transactions on Parallel and Distributed Systems. He has published two books, Software Similarity and Classification (Springer) and Dynamic and Advanced Data Mining for Progressing Technological Development (IGI-Global). He has served as the Program/General Chair for many international conferences such as ICA3PP 12/11, IEEE/IFIP EUC 11, IEEE TrustCom 13/11, IEEE HPCC 10/09, IEEE ICPADS 08, NSS 11/10/09/08/07. He has been the PC member for more than 60 international conferences in distributed systems, networking, and security. He serves as the Associate Editor of IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, Security and Communication Networks (Wiley), and the Editor of Journal of Network and Computer Applications. He is the Coordinator, Asia for IEEE Computer Society Technical Committee on Distributed Processing (TCDP). He is a Senior Member of the IEEE.

Professor Wanlei Zhou received the B.Eng and M.Eng degrees from Harbin Institute of Technology, Harbin, China in 1982 and 1984, respectively; the PhD degree from The Australian National University, Canberra, in 1991; and the DSc degree from Deakin University, Victoria, Australia, in 2002. He is currently the Chair Professor of Information Technology and the Head of School of Information Technology, Faculty of Science and Technology, Deakin University, Melbourne, Australia. Before joining Deakin University, Professor Zhou worked in a number of organisations including University of Electronic Science and Technology of China in Chengdu, China, Apollo/HP in Massachusetts, USA, National University of Singapore in Singapore, and Monash University in Melbourne, Australia. His research interests include network security, distributed and parallel systems, bioinformatics, mobile computing, and e-learning. He is a senior member of the IEEE. He has published more than 200 papers in refereed international journals and refereed international conferences proceedings. Professor Zhou was the General Chair / Program Committee Chair / Co-Chair of a number of international conferences, including ICA3PP, ICWL, PRDC, NSS, ICPAD, ICEUC, HPCC, etc., and has been invited to deliver keynote address in a number of international conferences, including TrustCom, ICWL, CIT, ISPA, etc.