2009 Sixth IFIP International Conference on Network and Parallel Computing

Modeling the Propagation Process of Topology-Aware Worms: An Innovative Logic Matrix Formulation

Xiang Fan
School of Management and Information Systems

Central Queensland University, Australia
x.fan2@cqu.edu.au

Yang Xiang
School of Management and Information Systems
Center for Intelligent and Networked Systems
Central Queensland University, Australia
y.xiang@cqu.edu.au

Abstract

This paper presents a study on modeling the propagation process of topology-aware worms. Topology-aware worms are more intelligent and adaptive to network topologies than other worms, thus are more difficult to control. Due to the complexity of the problem, no existing work has solved the problem of modeling the propagation of topology-aware worms. Our major contributions in this paper are firstly, we propose an innovative logic matrix formulation of the propagation process of topology-aware worms; and secondly, we find, from the applications of the formulation in our experiments, the impacts of two different topologies, namely the simple random graph topology and the pseudo power law topology, on a P2P worm's mean coverage rate in the P2P overlay network. The proposed innovative logic matrix formulation, which is a discrete time deterministic propagation model of topology-aware worms, can translate the propagation process of topology-aware worms into a sequence of logic matrix operations. Its effectiveness and efficiency are demonstrated by its applications in our experiments.

1. Introduction

Worms can be classified according to the techniques by which they discover new targets to infect. Scanning, which 'entails probing a set of addresses to identify vulnerable hosts' [1], is the most widely employed technique by worms. Scanning could be implemented differently, which leads to several different types such as random scanning, localized scanning [2], sequential scanning [3], routable scanning [4], selective scanning [4], importance scanning [5, 6], and topological scanning, which was employed by the Morris Internet Worm of 1988 as its target discovery technique [7]. Worms employing all other types of scanning except topological scanning among the above types do not need to have any knowledge on topology of the network they intend to propagate across. On the other hand, worms employing topological scanning must have the more or the less information on the network they intend to propagate over, or have the capability to discover that information if they do not have it in advance. Therefore, worms employing topological scanning are also called topology-aware worms.

A typical example of topology-aware worms is a worm attacking a flaw in a Peer-to-Peer (P2P) application and propagating across the P2P network by getting lists of peers from its victims and directing its subsequent attacks to those peers. This sort of topology-aware worm is called P2P worm. The Slapper worm [8] of 2003 was a typical example of P2P worms. The subsequent appearance of variations of the Slapper worm (the Slapper.B worm a.k.a. Cinik and the Slapper.C worm a.k.a. Unlock) indicates that exploit code, viruses and worms are becoming increasingly complex and sophisticated [8]. They are posing a serious challenge to network security. Due to the recent popularity of P2P systems with increasing number of users, P2P systems can be a potential vehicle for worms to achieve faster propagation across the Internet. Worm propagation on top of P2P systems could result in significant damages as illustrated by [9]. In order to find an effective and efficient countermeasure against the propagation of topology-aware worms in general, and P2P worms in particular, we must fully understand their propagation process. In this paper, we propose an innovative logic matrix formulation of the propagation process of topology-

978-0-7695-3837-2/09 $26.00 © 2009 IEEE
DOI 10.1109/NPC.2009.30

aware worms, which can be used to describe the propagation process of this type of worms.

2. Related work

Mathematical models developed to model propagation of infectious diseases have been adapted to model propagation of worms [10]. In epidemiology area, both deterministic and stochastic models exist for modeling the spreading of infectious diseases [11-14]. In network security area, both deterministic and stochastic models of worms based on their respective counterpart in epidemiology area have emerged. Deterministic models of worms could be further divided into two categories: continuous time and discrete time. Stochastic models of active worms are based on the theory of stochastic processes. All of them are discrete time in nature.

In the classical simple epidemic model [11-14], all hosts stay in one of the only two states at any time: 'susceptible' (denoted by 'S') or 'infectious' (denoted by 'I'), and thus it is also called the SI model. Staniford et al. presented a propagation model for the Code-RedI v2 worm [15], which is essentially the above classical simple epidemic model. The classical general epidemic model (Kermack-McKendrick model) [11-14] improves the classical simple epidemic model by considering removal of infectious hosts due to patching. The two-factor worm model [10] extends the classical general epidemic model by accounting for removal of susceptible hosts due to patching and considering the pairwise rate of infection as a variable rather than a constant.

The discrete time deterministic Analytical Active Worm Propagation (AAWP) model [16] takes into account the time an infectious host takes to infect other hosts, which is an important factor for the spread of active worms [17]. Since propagation of active worms is a discrete event process, this model of active worms is more accurate than its continuous time counterparts in the deterministic regime.

Rohloff and Basar presented a stochastic density-dependent Markov jump process propagation model [18] for worms employing the random scanning approach drawn from the field of epidemiology [12, 19]. Sellke et al. presented a stochastic Galton-Watson Markov branching process model [20] to characterize the propagation of worms employing the random scanning approach.

A more detailed survey on modeling the propagation process of worms can be found in [21].

The formulation proposed in this paper is a discrete time deterministic propagation model of topology-aware worms.

3. The proposed innovative formulation

At the beginning of this section, we extend definition of a matrix to allow its elements to be variables or constants of logic type; and term such kind of matrices logic matrices. Several operations of logic matrices are defined. Then, topology and state of a network are represented by its topology logic matrix and state logic matrix, respectively. Finally, an innovative logic matrix formulation of the propagation process of topology-aware worms is derived from first principle.

3.1. Logic matrices and their operations

We extend definition a matrix to allow variables or constants of logic type as its elements. The value of a variable of logic type can only be one of the only two logic constants: True (denoted by 'T') or False (denoted by 'F'). Therefore, a logic matrix could be defined as a two-dimensional array of elements 'T' and 'F' only. If a logic matrix has only one row or one column, we can also call it a logic row vector or logic column vector, respectively.

We define degree of a variable l of logic type (denoted by $\deg(l)$) as 1 when its value is 'T', and 0 when 'F'; and define degree of a logic matrix L (denoted by $\deg(L)$) as the total number of its elements whose value is 'T'. According to the above definitions, degree of a logic matrix L could be worked out by summing degree of its each element l, that is,

$$\deg(L) = \sum \deg(l). \quad (1)$$

Two logic matrices A and B can be added if and only if their dimensions are the same, that is, they all have the same number of rows and columns. The resultant $S = A + B$ is a logic matrix of the same dimension with its element s_{ij} (lies in the i-th row and the j-th column) being the results of logic OR of the corresponding elements a_{ij} and b_{ij} of the two logic matrices to be added. It can be defined mathematically as follows:

$$s_{ij} = a_{ij} \text{ OR } b_{ij}. \quad (2)$$

It could be easily derived that degree of the resultant logic matrix S cannot be less than that of both logic matrices A and B to be added; and that degree of the resultant logic matrix cannot be greater than sum of degree of each logic matrix to be added, that is,

$$\deg(A) \le \deg(S) \le \deg(A) + \deg(B); \quad (3)$$

and

$$\deg(B) \le \deg(S) \le \deg(A) + \deg(B). \quad (4)$$

A logic matrix A can be multiplied by another logic matrix B if and only if their inner dimensions are the same, that is, number of columns of the multiplicand logic matrix (the left one) is equal to number of rows of the multiplier logic matrix (the right one). Mutation law, which applies to logic matrix addition, does not apply to logic matrix multiplication. The product $P = A \times B$ is a logic matrix of the same number of rows as A and the same number of columns as B. We define value of element p_{ij} (lies in the i-th row and the j-th column) of the product to be determined by the following equation:

$$p_{ij} = \sum_{k=1}^{n} a_{ik} \text{ AND } b_{kj}, \quad (5)$$

where AND stands for logic AND operation, n denotes inner dimensions of the multiplicand and multiplier logic matrices, and $\sum_{k=1}^{n}$ represents logic OR operation of all those resultants of logic AND operations when k from 1 to n, inclusive.

Now the stage for later discussion has been set. In the next two sub-sections, we will introduce the concepts of a network's topology logic matrix and state logic matrix, respectively; and derive our innovative logic matrix formulation of the propagation process of topology-aware worms from first principle.

3.2. Logic matrix representations

According to the traditional graph theory, a computer network could be represented by a directed graph G, with its set of vertices V representing all computers connected to form the network, and its set of directed edges E representing all directed links among these computers. A directed link from computer i to computer j means computer i is able to send messages to computer j, but computer j is not able to send messages to computer i.

Topology of a computer network consisting of n computers could be represented by a n by n square matrix with its element t_{ij} (lies in the i-th row and the j-th column) indicating whether there is a directed link from computer i to computer j. Under the traditional directed graph theory, the numeric constant 1 is used to indicate there is a directed link, and 0 to indicate there is not.

We, in this paper, propose a different approach to indicating the existence or not of a directed link. The logic constant 'T' is used to indicate there is a directed link, and 'F' to indicate there is not. Therefore, topology of a computer network consisting of n computers could be represented by a n by n logic square matrix T. We term it topology logic matrix of the network.

Each row of the topology logic matrix of a computer network forms a logic row vector, which is a logic vector representation of outbound link(s) of a particular computer belonging to the network. We call this logic vector the computer's topology out-degree logic vector. Each column of the topology logic matrix of a computer network forms a logic column vector, which is a logic vector representation of inbound link(s) of a particular computer belonging to the network. We call this logic vector the computer's topology in-degree logic vector. For example, the i-th row of a topology logic matrix represents outbound link(s) of computer i, and the j-th column of a logic matrix represents inbound link(s) of computer j.

It can be easily derived that values of topology in-degree and out-degree of each computer belonging to a network equate to degrees of the computer's topology in-degree and out-degree logic vector, respectively, which can be worked out by using equation (1) given in the previous sub-section.

Next, we represent states of all n computers belonging to a network by a logic matrix (row vector) S of dimension 1 by n with its element S_{1j} (lies in the 1st row and the j-th column) indicating whether computer j has been infected by any malware and become infectious. The logic constant 'T' is used to indicate the computer has been infected and become infectious, and 'F' to indicate it has not. We term the above logic matrix (vector) the network's state logic matrix (vector).

It can be easily derived that the total number of infected and infectious computers in a network equates to degree of the network's state logic matrix (vector), which can be worked out by using equation (1) given in the previous sub-section.

3.3. The logic matrix formulation

Based on the above extensions to matrices and their operations and extensions to the matrix representation of a network in the traditional directed graph theory, we are now ready to derive our innovative logic matrix formulation of the propagation process of topology-aware worms from first principle.

The derivation of our innovative logic matrix formulation of the propagation process of topology-aware worms is based on the following assumptions. An infectious computer will send worm packet(s) to all other computers belonging to the same network to which it has a outbound link, regardless of the state (infected and infectious or not) of those computers. A healthy (not infected and not infectious) computer belonging to a network will be infected and become infectious once it receives worm packet(s) from another infectious computer belonging to the same network. An infected and infectious computer belonging to a network will remain in that state when it receives worm packet(s) from another infectious computer belonging to the same network. The propagation process from sending worm packet(s), to receiving worm packet(s), to having the recipient infected and the infected becoming infectious will be completed in the time interval TI of strictly the same length. There are a total of n computers belonging to a logical (not physical) network under consideration. Initially, there are a total of I_0 computers which are infected and infectious.

According to the above assumptions, the logical network's initial state could be represented by its initial state logic matrix (vector) S_0 of dimension 1 by n; and the total number of initially infected and infectious computers I_0 equates to degree of S_0:

$$\mathrm{I}_0 = \deg(\mathrm{S}_0). \tag{6}$$

Time interval TI later, the logical network's state could be represented by its state logic matrix (vector) S_1 of dimension 1 by n at that time; and the total number of infected and infectious computers I_1 at that time equates to degree of S_1:

$$\mathrm{I}_1 = \deg(\mathrm{S}_1). \tag{7}$$

In the same way, time interval $2 \times TI$ later, the logical network's state could be represented by its state logic matrix (vector) S_2 of dimension 1 by n at that time; and the total number of infected and infectious computers I_2 at that time equates to degree of S_2:

$$\mathrm{I}_2 = \deg(\mathrm{S}_2). \tag{8}$$

Generally, time interval $g \times TI$ later, the logical network's state could be represented by its state logic matrix (vector) S_g of dimension 1 by n at that time; and the total number of infected and infectious computers I_g at that time equates to degree of S_g:

$$\mathrm{I}_g = \deg(\mathrm{S}_g). \tag{9}$$

In the same way, time interval $(g+1) \times TI$ later, the logical network's state could be represented by its state logic matrix (vector) S_{g+1} of dimension 1 by n at that time; and the total number of infected and infectious computers I_{g+1} at that time equates to degree of S_{g+1}:

$$\mathrm{I}_{g+1} = \deg(\mathrm{S}_{g+1}). \tag{10}$$

During the above propagation process, the total number of infected and infectious computers keeps increasing prior to a certain time point. Finally, time interval $(G+1) \times TI$ later the total number of infected and infectious computers I_{G+1} at that time will be equal to I_G, which reveals the above propagation process will actually stop at time point $G \times TI$.

We notice that the logical network's state at time point $(g+1) \times TI$ represented by its state logic matrix (vector) S_{g+1} is fully determined by its state at time point $g \times TI$ represented by its state logic matrix (vector) S_g and its logical topology represented by its topology logic matrix T. We find the relationship among S_{g+1}, S_g, and T could be mathematically described as follows:

$$\mathrm{S}_{g+1} = \mathrm{S}_g + \mathrm{S}_g \times \mathrm{T}, \tag{11}$$

where $\times$ stands for logic matrix multiplication, and $+$ denotes logic matrix addition, both of which have been defined in sub-section 3.1.

In the above equation, S_g is a 1 by n logic matrix and T is a n by n square logic matrix. The resultant of $\mathrm{S}_g \times \mathrm{T}$ (denoted by S'_g) is a 1 by n logic matrix (row vector) representing all computers belonging to the network that could be infected at time point $(g+1) \times TI$ given the network's state at time point $g \times TI$ represented by its state logic matrix (row vector) S_g and its logical topology represented by its topology logic matrix T. According to definition of logic matrix multiplication given in sub-section 3.1, value of the j-th element of S'_g (denoted by $s'_{g_{1j}}$) is determined by equation (5), where i equates to 1 because both the multiplicand and the resultant logic matrix are logic row vectors, that is,

$$s'_{g_{1j}} = \sum_{k=1}^{n} s_{g_{1k}} \text{ AND } t_{kj}, \quad (12)$$

where $s_{g_{1k}}$ stands for the value of the element which lies in the k-th column of S_g, and t_{kj} denotes value of the element which lies in the k-th row and the j-th column of T.

In the above equation, t_{kj} for all k from 1 to n actually represents compute j's topology in-degree logic vector. The resultant of $s_{g_{1k}}$ AND t_{kj} will be logic 'T' if and only if both values of $s_{g_{1k}}$ and t_{kj} are logic 'T', which indicates at time point $g \times TI$ computer k is infectious and computer j has an inbound link from computer k. The logic OR operation of all those resultants of logic AND operations when k from 1 to n, inclusive, denoted by $\sum_{k=1}^{n}$ in the above equation actually says if there exists at least one value of k from 1 to n, inclusive, which makes the value of the resultant of $s_{g_{1k}}$ AND t_{kj} to be logic 'T', the value of $s'_{g_{1j}}$ will be logic 'T'. Therefore, equation (12) actually says if at time point $g \times TI$ at least one computer among those computers from which computer j has an inbound link is infectious, computer j will be infected and become infectious at time point $(g+1) \times TI$.

Then, according to the definition of logic matrix addition given in sub-section 3.1, it could easily derived that $\mathrm{S_g} + \mathrm{S_g} \times \mathrm{T}$ actually just adds all those computers that could be infected at time point $(g+1) \times TI$ represented by $\mathrm{S_g} \times \mathrm{T}$ to the network's state at time point $g \times TI$ represented by S_g. The resultant of the above logic matrix addition operation represents the network's state at time point $(g+1) \times TI$, which is represented by S_{g+1}. Hence, equation (11) gets proved.

The framework formed by Equations (9) and (11) along with the criterion proposed in this paper used to determine whether the propagation process has actually stopped is a discrete time deterministic propagation model of topology-aware worms. We call the above framework the logic matrix formulation of the propagation process of topology-aware worms. The formulation can translate the propagation process of topology-aware worms into a sequence of logic matrix operations, which are easily implemented with any matrix-friendly mathematics programs.

4. Applications of the formulation

We apply the logic matrix formulation of the propagation process of topology-aware worms proposed in this paper to investigate the impacts of two different topologies, namely the simple random graph topology and the pseudo power law topology on the coverage rate of topology-aware worms. Coverage rate (denoted by CR in this paper) of a worm is defined in this paper as a ratio in percentage of the maximum number of computers belonging to a network that could be infected and become infectious to the total number of computers n belonging to the same network. According to the criterion proposed in this paper used to determine whether the propagation process of a topology-aware worm has actually stopped, coverage rate of a topology-aware worm in a network could be worked out by using the following equation:

$$CR = \frac{\deg(S_G)}{n} \times 100\%, \quad (13)$$

where S_G represents the state logic matrix of the network at the time point when the propagation process has just stopped.

4.1. The simple random graph topology

Firstly, we apply the proposed logic matrix formulation of the propagation process of topology-aware worms to investigate the impacts of two parameters, namely the number of initially infected computers I_0 belonging to a network and the mean value of topology out-degree $E(D_{out})$ of the network, on the coverage rate CR of a particular sort of topology-aware worm called P2P worm in the network.

We program the proposed formulation with MathWorks' MATLAB, which is a matrix–friendly mathematics program. Our implementation in MATLAB assumes there are a total of $n = 10{,}000$ peers (computers) belonging to the logical P2P overlay network under consideration. Therefore, the topology of the overlay network is represented by its topology logic matrix T, which is a 10,000 by 10,000 square logic matrix; and the its initial state is represented by its initial state logic

matrix S_0 ($I_0 = \deg(S_0)$), which is a 1 by 10,000 logic matrix (row vector). We randomly select all initially infected peers (computers) from all peers belonging to the overlay network.

Mean value of topology out-degree $E(D_{out})$ of the overlay network is determined by the following equation:

$$E(D_{out}) = \frac{\sum_{i=1}^{n} \deg(T_i)}{n}, \quad (14)$$

where T_i stands for the i-th row of T, which is actually the topology out-degree logic vector of peer (computer) i belonging to the overlay network. In the experiments conducted for this sub-section, we assume each peer has the same value of topology out-degree. Peers to which each peer has outbound links are randomly selected from all peers except the peer itself belonging to the overlay network, which means we do not allow loop, that is, no peer has an outbound link to itself. Therefore, we call the topology of the overlay network in the experiments conducted for this sub-section the simple random graph topology.

We conduct our experiments with MATLAB under different combinations of values of I_0 and $E(D_{out})$.

Firstly, we fix the number of initially infected peers (computers) I_0 belonging to the overlay network to be 1, and try to find out the impact of mean value of topology out-degree $E(D_{out})$ on the coverage rate CR of P2P worms in the overlay network. We randomly select all initially infected peer(s) from all peers belonging to the overlay network. A total of 5 scenarios ($E(D_{out})$ from 1 to 5, inclusive) are investigated. Experiment for each scenario is repeated 100 times. Then, the mean value of coverage rate and coefficient of variation of coverage rate are worked out. Results from the experiments are listed in Table 1.

Table 1. The simple random graph topology (when there is only 1 initially infected peer randomly selected from all peers)

Mean Value of Topology Out-Degree	Mean Value of Coverage Rate (%)	Coefficient of Variation of Coverage Rate (%)
1	1.23	54.81
2	79.64	0.68
3	94.08	0.27
4	98.06	0.16
5	99.31	0.09

As shown by the above experimental results, mean value of topology out-degree has great impact on both mean value and coefficient of variation of coverage rate of P2P worms in the overlay network featuring the simple random graph topology. Increase in mean value of topology out-degree results in increase in mean value of coverage rate but decrease in coefficient of variation of coverage rate. When mean value of topology out-degree is increased to 3, mean value of coverage rate is increased to over 90% and its coefficient of variation becomes very small, which indicates 3 is the minimum mean value of topology out-degree which can make a P2P worm be able to infect most peers with very high certainty.

After that, we fix the number of initially infected peers (computers) I_0 belonging to the overlay network to be 10, and repeat the above experiments. Results from the experiments are listed in Table 2.

Table 2. The simple random graph topology (when there are a total of 10 / 100 initially infected peers randomly selected from all peers)

Mean Value of Topology Out-Degree	Mean Value of Coverage Rate (%)	Coefficient of Variation of Coverage Rate (%)
1	4.28 / 13.53	16.22 / 5.43
2	79.80 / 80.06	0.63 / 0.62
3	94.10 / 94.16	0.27 / 0.26
4	98.03 / 98.06	0.15 / 0.15
5	99.30 / 99.31	0.08 / 0.09

The above experimental results show similar trends to those shown by Table 1, which indicates the impact of number of initially infected peers on the coverage rate of a P2P worm in the overlay network featuring the simple random graph topology is insignificant.

4.2. The pseudo power law topology

Secondly, we apply the proposed logic matrix formulation of the propagation process of topology-aware worms to investigate the impacts of two parameters, namely the number of initially infected computers I_0 belonging to a network and the maximum value of topology out-degree $Max(D_{out})$ of the network, on the coverage rate CR of P2P worms in the network. In the experiments conducted for this sub-section, we assume only a very small number (10 in our experiments) of peers have the maximum value of topology out-degree, and all other peers have the minimum value (1 in our experiments) of topology out-degree. Although the distribution of topology out-

degree in our experiments does not strictly follow power law, it does have the most important features of power law distribution, namely peers with maximum value of topology out-degree are rare and most peers have minimum value of topology out-degree. Therefore, we call the topology of the overlay network in the experiments conducted for this sub-section the pseudo power law topology.

We conduct our experiments with MATLAB under different combinations of values of I_0 and $Max(D_{out})$.

Firstly, we fix the number of initially infected peers (computers) I_0 belonging to the overlay network to be 1, and try to find out the impact of maximum value of topology out-degree $Max(D_{out})$ on the coverage rate CR in the overlay network of topology-aware worms. We randomly select all initially infected peer(s) from all peers belonging to the overlay network. A total of 5 scenarios ($Max(D_{out}) = 100,1000,2000$) are investigated. In the experiments conducted for this sub-section, we assume each peer has either the maximum value of topology out-degree or the minimum value of topology out-degree. Peers to which each peer has outbound links are randomly selected from all peers except the peer itself belonging to the overlay network, which means we do not allow loop, that is, no peer has an outbound link to itself. Experiment for each scenario is repeated 100 times. Then, the mean value of coverage rate and coefficient of variation of coverage rate are worked out. Results from the experiments are listed in Table 3.

Table 3. When there is only 1 initially infected peer randomly selected from all peers

Maximum Value of Topology Out-Degree	Mean Value of Coverage Rate (%)	Coefficient of Variation of Coverage Rate (%)
100	3.17	200.74
1000	13.83	209.20
2000	14.54	226.10

As shown by the above experimental results, when all initially infected peers are randomly selected from all peers, maximum value of topology out-degree has a little impact on both mean value and coefficient of variation of coverage rate of P2P worms in the overlay network featuring the pseudo power law topology. Increase in maximum value of topology out-degree results in a little increase in mean value of coverage rate and a little increase in coefficient of variation of coverage rate as well, which indicates the small gain in coverage rate could be offset by the small loss in certainty. The worm is not able to infect most peers with high certainty.

After that, we fix the number of initially infected peers (computers) I_0 belonging to the overlay network to be 10, and repeat the above experiments. Results from the experiments are listed in Table 4.

Table 4. When there are a total of 10 initially infected peers randomly selected from all peers

Maximum Value of Topology Out-Degree	Mean Value of Coverage Rate (%)	Coefficient of Variation of Coverage Rate (%)
100	11.25	79.51
1000	33.06	111.27
2000	36.23	120.07

The above experimental results show similar trends (just an insignificantly higher coverage rate and an insignificantly lower coefficient of variation of coverage rate) to those shown by Table 3, which indicates, when all initially infected peers are randomly selected from all peers, the impact of number of initially infected peers on the coverage rate of a P2P worm in the overlay network featuring the pseudo power law topology is insignificant.

Finally, we randomly select all initially infected peers (computers) from only those peers with maximum topology out-degree, and repeat all the above experiments described in this sub-section. Results from the experiments are listed in Table 5 and Table 6 for $I_0 = 1$ and $I_0 = 10$, respectively.

Table 5. When there is only 1 initially infected peer randomly selected from only those peers with maximum topology out-degree

Maximum Value of Topology Out-Degree	Mean Value of Coverage Rate (%)	Coefficient of Variation of Coverage Rate (%)
100	20.74	26.65
1000	78.21	11.17
2000	95.33	0.89

Table 6. When there are a total of 10 initially infected peers randomly selected from only those peers with maximum topology out-degree

Maximum Value of Topology Out-Degree	Mean Value of Coverage Rate (%)	Coefficient of Variation of Coverage Rate (%)
100	38.50	1.53
1000	85.19	0.41
2000	95.94	0.19

As shown by the above experimental results, when all initially infected peers are randomly selected from only those peers with maximum topology out-degree, maximum value of topology out-degree has a great impact on both mean value and coefficient of variation of coverage rate of P2P worms in the overlay network featuring the pseudo power law topology. Increase in maximum value of topology out-degree results in increase in mean value of coverage rate but decrease in coefficient of variation of coverage rate. However, the impact of number of initially infected peers is insignificant. When maximum value of topology out-degree reaches 2,000, the worm is able to infect most peers with very high certainty, regardless of number of initially infected peers.

5. Conclusions and future research

This paper presents a study on modeling the propagation process of topology-aware worms. Our major contributions in this paper are firstly, we propose an innovative logic matrix formulation of the propagation process of topology-aware worms; and secondly, we find, from applications of the formulation in our experiments, the impacts of two different topologies, namely the simple random graph topology and the pseudo power law topology, on a P2P worm's mean coverage rate in the P2P overlay network.

We believe the innovative logic matrix formulation proposed in this paper, which is a discrete time deterministic propagation model of topology-aware worms described by a difference equation of logic matrix, is a highly effective and efficient tool for investigating the propagation process of topology-aware worms in general and P2P worm in particular.

In the future, we are going to incorporate removal of susceptible and/or infectious computers (peers) into the proposed discrete time deterministic propagation model of topology-aware worms, which will greatly enhance the adaptability of the framework proposed in this paper.

6. References

[1] N. Weaver, V. Paxson, S. Staniford, and R. Cunningham, "A Taxonomy of Computer Worms," in *WORM '03*, Washington D.C., USA, 2003, pp. 11-18.

[2] D. Moore, C. Shannon, and J. Brown, "Code-Red: A Case Study on the Spread and Victims of an Internet Worm," in *IMW '02*, Marseille, France, 2002, pp. 273-284.

[3] C. C. Zou, D. Towsley, and W. Gong, "On the Performance of Internet Worm Scanning Strategies," University of Massachusetts Technical Report: TR-03-CSE-07, 2003.

[4] C. C. Zou, D. Towsley, W. Gong, and S. Cai, "Routing Worm: A Fast, Selective Attack Worm Based on IP Address Information," in *PADS '05*, 2005, pp. 199-206.

[5] Z. Chen and C. Ji, "Importance-Scanning Worm Using Vulnerable-Host Distribution," in *IEEE GLOBECOM*, 2005, pp. 1779-1784.

[6] Z. Chen and C. Ji, "A Self-Learning Worm Using Importance Scanning," in *WORM '05*, Fairfax, VA, USA, 2005, pp. 22-29.

[7] E. H. Spafford, "The Internet Worm Program: An Analysis," *ACM SIGCOMM Computer Communication Review,* vol. 19, pp. 17-57, 1989.

[8] I. Arce and E. Levy, "An Analysis of the Slapper Worm," in *IEEE Security & Privacy*, 2003, pp. 82-87.

[9] W. Yu, "Analyze the Worm-Based Attack in Large Scale P2P Networks," in *The 8th IEEE International Symposium on High Assurance Systems Engineering (HASE 2004)*, 2004.

[10] C. C. Zou, W. Gong, and D. Towsley, "Code Red Worm Propagation Modeling and Analysis," in *CCS '02*, Washington D.C., USA, 2002, pp. 138-147.

[11] R. M. Anderson and R. M. May, *Infectious Diseases of Humans: Dynamics and Control*. Oxford: Oxford University Press, 1991.

[12] H. Andersson and T. Britton, *Stochastic Epidemic Models and Their Statistical Analysis*. New York: Springer-Verlag, 2000.

[13] N. T. Bailey, *The Mathematical Theory of Infectious Diseases and Its Applications*. New York: Hafner Press, 1975.

[14] J. C. Frauenthal, *Mathematical Modeling in Epidemiology*. New York: Springer-Verlag, 1980.

[15] S. Staniford, V. Paxson, and N. Weaver, "How to Own the Internet in Your Spare Time," in *Security '02*, San Francisco, CA, USA, 2002, pp. 149-167.

[16] Z. Chen, L. Gao, and K. Kwiat, "Modeling the Spread of Active Worms," in *IEEE INFOCOM*, 2003, pp. 1890-1900.

[17] Y. Wang and C. Wang, "Modeling the Effects of Timing Parameters on Virus Propagation," in *WORM '03*, Washington D.C., USA, 2003, pp. 61-66.

[18] K. Rohloff and T. Basar, "Stochastic Behavior of Random Constant Scanning Worms," in *14th ICCCN*, San Diego, CA, USA, 2005, pp. 339-344.

[19] D. J. Daley and J. Gani, *Epidemic Modelling: An Introduction*. Cambridge: Cambridge University Press, 1999.

[20] S. Sellke, N. B. Shroff, and S. Bagchi, "Modeling and Automated Containment of Worms," in *DSN '05*, 2005, pp. 528-537.

[21] Y. Xiang, X. Fan, and W. Zhu, "Propagation of Active Worms: A Survey," *International Journal of Computer Systems Science & Engineering,* vol. 24, pp. 157-172, 2009.