Modeling Malware Propagation in Gnutella Type Peer-to-Peer Networks

Krishna Ramachandran, and Biplab Sikdar
Department of Electrical, Computer and Systems Engineering
Rensselaer Polytechnic Institute
Troy, New York 12180 USA
{ramak sikdab}@rpi.edu

Abstract

A key emerging and popular communication paradigm, primarily employed for information dissemination, is peer-to-peer (P2P) networking. In this paper, we model the spread of malware in decentralized, Gnutella type of peer-to-peer networks. Our study reveals that the existing bound on the spectral radius governing the possibility of an epidemic outbreak needs to be revised in the context of a P2P network. We formulate an analytical model that emulates the mechanics of a decentralized Gnutella type of peer network and study the spread of malware on such networks. We show analytically, that a framework which does not incorporate the behavioral characteristics of peers ends up over estimating the epidemic threshold metric, $\mathcal{R}_0$. This in turn results in false positives, an undesirable feature.We also characterize the conditions under which the network may reach a malware free equilibrium and validate our theoretical results with numerical simulations.

1. Introduction

Peer to peer networks provide a paradigm shift from the traditional client server model of most networking applications by allowing all users to act as both clients and servers. The primary use of such networks so far, has been to swap media files within a local network or over the Internet as a whole [2, 3, 4, 5]. These networks have grown in their popularity in the recent past and the fraction of network traffic originating from these networks has consistently increased. The growing popularity and high penetration of P2P clients such as KaZaa, Gnutella and BitTorrent have provided virus writers with a potent means of compromising hosts on a large scale.

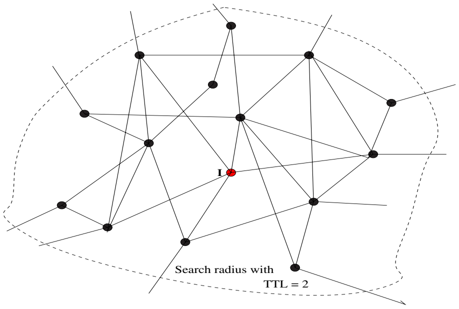


Figure 1. Infection radius when $TTL = 2$.

The use of P2P networks as a vehicle to spread malware enjoys some important advantages over worms that spread by scanning for vulnerable hosts. This is primarily due to the methodology employed by the peers to search for content. For instance, in decentralized P2P architectures such as Gnutella, where search is done by flooding the network, a peer forwards the query to it's immediate neighbors and the process is repeated until a specified threshold TTL is reached. Here TTL is the threshold representing the number of overlay links that a search query travels. A sample scenario of the above description is depicted in Fig. (1), wherein the malicious host, labeled **I**, can potentially infect all peers that are within a distance of $TTL = 2$ hops from it. A relevant example here is the $Mandragore$ worm [16], that affected Gnutella users. Having infected a host in the network, the worm cloaks itself for other Gnutella users, leading them also to believe that it is actually an MP3 music file or an image file. Every time a Gnutella user searches for media files in the infected computer, the virus always appears as an answer to the request. The design of the search technique has the following implications: first, the worms can spread much faster, since they do not have to probe for susceptible

1-4244-0054-6/06/$20.00 ©2006 IEEE

hosts and second, the rate of failed connections is less. Thus, rapid proliferation of malware can pose a serious security threat to the functioning of P2P networks.

Understanding the factors affecting the malware spread can help facilitate network designs that are resilient to such attacks, thereby ensuring proper protection of the networking infrastructure. In this paper, we address this issue and develop an analytic framework for modeling the spread of malware in a peer-to-peer environment while accounting for the architectural, topological and user related factors.

Having motivated our work, we proceed to explore various facets of the problem. The rest of the paper is organized as follows: Section 2 differentiates the work presented in this paper from existing literature; in Section 3, we lay the grounds for our modeling work and present the analytic framework in Section 4. We analyze the model in detail and Section 5 and present the numerical results validating our theory in Section 6. Finally, Section 7 presents the concluding remarks.

2. Relationship to prior work

In this section, we provide a brief overview of modeling literature in P2P networks, not necessarily in the realm of malware spread, and differentiate the current work from existing ones. Though the initial thrust in P2P research was measurement oriented, recent works, [13, 12, 15], have proposed analytical models for the temporal evolution of information in the network. In [13], a branching process approximation characterizing the file transfer was presented, while in [12], a stochastic fluid model for BitTorrent-like networks is formulated and the steady state properties of the system are analyzed. A limitation of the above works is that they are specialized to Bit-torrent like networks and the framework cannot be extended to analyze P2P networks such as Gnutella or KaZaa. Although, the authors in [15] do not model the offline/online transition, their framework is more representative of a Bit-torrent network than existing ones. Again, the model's applicability is limited and cannot be extended to a Gnutella like network.

The issue of worms in peer-to-peer networks is addressed in [8] wherein the authors perform a simulation study of the dangers posed by P2P worms and proceed to outline possible mitigation mechanisms. Modeling studies addressing malware spread in P2P networks appear in [17, 18], wherein the authors formulate a deterministic model having it's basis in the field of epidemiology. In formulating the equations for the various classes of peers, the authors assume that a vulnerable peer can be infected by any of the infected ones in the network. This assumption is certainly not true since the likely candidates for an infected peers are limited to those present TTL hops away from it and not the entire P2P network. Incorporating this detail in the model is imperative since it figures in the expression for the *basic reproduction number*, a metric that determines the presence/absence of an epidemic. Another important omission is the incorporation of user behavior in the analytic framework. Typically, users in a P2P network alternate between two states: the on state, where they are connected to other peers and partake in network activities such as query forwarding/response, query initiation etc. and the off state wherein they are disconnected from the network.

In the current work, we formulate a comprehensive model for malware spread in Gnutella type P2P networks that addresses the above shortcomings. We develop the model in two stages: first, we quantify the average number of peers within TTL hops from any given peer and in the second stage incorporate the neighborhood information into the final model for malware spread. While determining the average number of peers that are within k hops away is not feasible for arbitrary networks, the fact that the degree distribution of peers in Gnutella follows a power law distribution [4], makes the task realizable for such networks. In the next section, we report our simulation result that questions the validity of the bound on the spectral radius of the P2P adjacency matrix, that is widely accepted to hold true in the presence/absence of a large scale infection. This finding, further substantiates the need to incorporate the limited view of a peer in a P2P network into the analytic model.

3. Virus propagation in P2P graphs

Hypercubes have often been chosen as a graph model for P2P networks and in [9, 10], the authors derive a limiting condition for a virus/worm to be prevalent in the network without incorporating the underlying communication framework. Specifically, the threshold condition is derived to be: $\frac{\beta}{\delta} < \frac{1}{\rho(A)}$, where β denotes the rate at which the infection spreads, $1/\delta$, the average lifetime of an infectious node and $\rho(A)$ represents the spectral radius of the original network adjacency matrix. As we shall demonstrate, this can result in an erroneous estimation of an epidemic presence since the authors do not consider the fact that once a peer is infected, any susceptible peer within a TTL hop radius becomes a likely candidate for a virus attack.

In order to arrive at the threshold estimate for the virus spread, one needs to look at the spectral radius of the *modified adjacency matrix*, $\mathcal{M}$. Specifically, this is a graph constructed from the original adjacency matrix, wherein an edge exists between two peers as long as there are within TTL hops from each other. Mathematically, this is computed as follows: $A_{eff} = A + A^2 + \ldots + A^{TTL}$, where, A represents the P2P adjacency matrix, and

$$\mathcal{M}(i,j) = \begin{cases} 1 & \text{if } A_{eff}(i,j) > 0 \text{ and } i \neq j \\ 0 & \text{otherwise} \end{cases} \qquad (1)$$

Claim: Virus spread in a P2P network reaches endemic proportions if $\beta/\delta < 1/\rho(\mathcal{M})$.

The proof of the claim is exactly along the lines of that presented in [10], and due to want of space, we only provide the outline here. The only difference in our proof stems from the definition of a *neighbor* in a peer-to-peer network. Since the authors in [10] derive the expression from a structural point of view, they assume that a peer can be infected only by those one hop away in the overlay graph. This is clearly not true, since the communication paradigm is such that a node within TTL overlay hops is visible to a peer when querying for content. Substantiating our analysis with simulations carried out lend credence to our claim. We simulated a simple SIR epidemic on a 10000 node power law graph for both scenarios; one where the neighbors are limited to nodes directly connected in the graph ($TTL = 1$) and the second where the communication paradigm of P2P networks is incorporated. That is, a peer at given time is in one of the three states: vulnerable to a virus attack (S), infected with the virus (I) or virus free and immune to further attacks (R). A power law graph was chosen since it is representative of a Gnutella type P2P network [4] and the initial number of infective hosts was set at 50. The hop threshold for search queries, TTL, in the communication graph was set at 3, i.e., each search query travels 3 overlay hops before being discarded. The spectral radius of the structural overlay graph was 15.3497 while that of the communication graph 1361.9. The other parameters, β and δ were chosen such that $\frac{\beta}{\delta} < \frac{1}{\rho(A)}$. Specifically, $\delta = 1$ and $\beta = 0.0551$. It was observed that, even with the inequality holding true, and taking into account the communication neighborhood, the malware infects the entire network. This scenario is depicted in Figure 2. The curve where the number of infecthives decreases corresponds to the case where the communication pattern is not considered and one might falsely conclude that an epidemic does not exist. Although user behavior such as offline-online transition are not accounted for, the derivation of the threshold with all nodes online is useful as an upper bound when determining either the presence or absence of an epidemic.

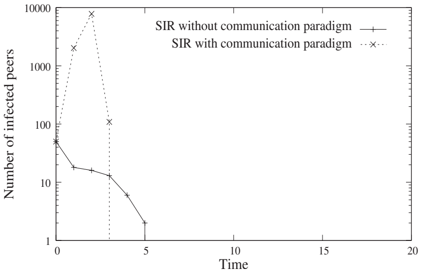


Figure 2. Virus propagation on a 10000 node sample P2P graph

4. P2P Model

In this section, we present our analytic framework for modeling the spread of information, in our case in the form of malware, in peer-to-peer networks. While the framework we develop is robust and is applicable across varied architectures such as Bit-torrent networks, we confine ourselves to the analysis of Gnutella like networks. We first describe the search process and the likelihood of file transfer and then present the model for the spread of files based on a compartmental model.

4.1. Search Mechanism

The transfer of information in a P2P network is initiated with a search request for it. There exist several search mechanisms, popular among which are flooding and the random walk. In this section, we derive an expression for the search neighborhood, z_{av}, under the assumption that the search mechanism employed is $flooding$, as is the case in Gnutella networks. In this scenario, a peer searching for a file forwards a query to all it's neighbors. A peer receiving such a request first responds affirmatively if in possession of the file and then checks the hop count of the query. If this value is greater than zero, it forwards the query outwards to it's neighbors, else, the query is discarded.

Measurement studies have shown that Gnutella networks follow a power-law degree distribution [4]. We now use the generating function approach as in [11] to quantify the number of peers reachable while searching for the file. Define the generating function for the vertex degree probability distribution as: $G_0(x) = \sum_{k=0}^{\infty} p_k x^k$, where p_k is the probability that a randomly chosen vertex on the graph has degree k. Since, the Gnutella network has a power law degree distribution, this probability is given by: $p_k = Pr(N = k) = Ck^{-\tau}$, where C and τ are constants. We first proceed to quantify the distribution of the degree of the vertex that one arrives at by following a randomly chosen edge. This information is then used to arrive at a recursive definition for the k-hop neighbors of a node in the network. Note, that when an edge is chosen at random, it is more likely that it leads to a node with a higher degree. The generating function for the probability distribution of reaching a k degree node by traversing a randomly chosen edge can then be obtained as: $\frac{\sum_k kp_k x^k}{\sum_k kp_k} = \frac{xG_0'(x)}{G_0'(1)}$. Now, the distribution of the outgoing edges from the vertex chosen has one power of x lesser than the above expression and thus can be expressed as: $G_1(x) = \frac{G_0'(x)}{G_0'(1)}$. In a similar fashion, the generating function for the number of two-hop neighbors is: $\sum_k p_k [G_1(x)]^k = G_0(G_1(x))$. Thus the recursive formulation for the distribution of the m^{th} nearest neighbors is given by $G_0(G_1(\ldots G_1(x)\ldots))$, with $m-1$ iterations of the function G_1 acting on itself. We define the above recursive convolution, yielding the generating function for the m^{th} hop neighbors as

$$G^{(m)}(x) = \begin{cases} G_0(x) & \text{for } m = 1 \\ G^{(m-1)}(G_1(x)) & \text{for } m \geq 2 \end{cases} \tag{2}$$

Differentiating the generating function and substituting $x = 1$ yields the average number of m^{th} hop neighbors. For example, the average number of one and two hop neighbors of a peer are given by $z_1 = G_0'(1) = \sum_k kp_k$ and $z_2 = G_0''(1)$ respectively. Equivalently, simple algebraic manipulation enables us to express the average number of m^{th} hop neighbors, z_m as: $z_m = \left[\frac{z_2}{z_1}\right]^{m-1} z_1$. Thus, since the search neighborhood of a peer extends up to TTL hops, we have the expression for the average neighborhood size as: $z_{av} = \sum_{i=1}^{TTL} z_i$.

4.2. Compartmental Model

We formulate our model for the P2P network as a compartmental model, with the peers divided into compartments, each signifying it's state at that time instant, and with assumptions about the nature and time rate of transfer from one compartment to another. The network is partitioned into four broad classes:

P_S	Number of peers wishing to download a particular file
P_E	Number of peers currently in the process of downloading the file
P_I	Number of peers with a copy of the file
P_R	Number of peers who either have deleted the file or are no longer interested in the file

$1/\lambda_{on}, 1/\lambda_{off}$	average peer on and off time durations
$1/\lambda$	rate at which a peer generates queries
$1/\mu$	average download time for a particular file
r_1	rate at which peers terminate ongoing downloads
r_2	rate at which peers renew interest in downloading a file after having deleted it
$1/\delta$	average time for which a peer stores a file

Table 1. Notation and P2P model parameters

Further, each class has two components; one comprising of peers of that class that are currently online, while the second represents the offline peers. For instance, $P_{I_{on}}$ denotes the peers hosting the file which are currently online and $P_{I_{off}}$, the offline peers with a copy of the file. Our formulation is based on the principle of mass action, wherein the behavior of each class is approximated by the mean number in the class at that instant of time. By employing the mean-field approach to characterize each compartment, we assume that the constituent compartments are well represented by their respective average numbers and that this representation is a differentiable funtion of time. The large size of the P2P network renders these assumptions reasonable. Another assumption is that the size of the P2P network does not vary over the time period during which the spread of information is modeled.

The dynamics of the spread of information can then be represented in terms of the constituent classes by the following deterministic system of equations

$$\begin{aligned}
\frac{dP_{S_{on}}}{dt} &= -\lambda z_{av} P_{S_{on}} P_{I_{on}} / N_P + r_1 P_{E_{on}} + r_2 P_{R_{on}} \\
&\quad -\lambda_{off} P_{S_{on}} + \lambda_{on} P_{S_{off}} && (3) \\
\frac{dP_{E_{on}}}{dt} &= \lambda z_{av} P_{S_{on}} P_{I_{on}} / N_P - r_1 P_{E_{on}} - \mu P_{E_{on}} \\
&\quad -\lambda_{off} P_{E_{on}} + \lambda_{on} P_{E_{off}} && (4) \\
\frac{dP_{I_{on}}}{dt} &= \mu P_{E_{on}} - \delta P_{I_{on}} - \lambda_{off} P_{I_{on}} + \lambda_{on} P_{I_{off}} && (5) \\
\frac{dP_{R_{on}}}{dt} &= \delta P_{I_{on}} - r_2 P_{R_{on}} - \lambda_{off} P_{R_{on}} + \lambda_{on} P_{R_{off}} && (6) \\
\frac{dP_{S_{off}}}{dt} &= \lambda_{off} P_{S_{on}} - \lambda_{on} P_{S_{off}} && (7) \\
\frac{dP_{E_{off}}}{dt} &= \lambda_{off} P_{E_{on}} - \lambda_{on} P_{E_{off}} && (8) \\
\frac{dP_{I_{off}}}{dt} &= \lambda_{off} P_{I_{on}} - \lambda_{on} P_{I_{off}} && (9) \\
\frac{dP_{R_{off}}}{dt} &= \lambda_{off} P_{R_{on}} - \lambda_{on} P_{R_{off}} && (10)
\end{aligned}$$

Note that we have strived to arrive at a generic formulation of the problem encompassing all possible scenarios. Different flavors of the model can be obtained by appropriately choosing the parameter values. For instance, $\mu = \infty, P_{E_{off}}(t) = 0 \ \forall t$ results in an SIR epidemic model. Other variants of the problem can be similarly derived. Also, the offline rates for the various classes have been kept same in order to reduce the number of variable and ease of analysis an. Different rates for each class can easily be accommodated in the model.

We now elaborate on the rationale behind the constituent equations of the model above. A transition out of class $P_{S_{on}}$ occurs if either a peers goes offline or initiates a search query that is successful. The former occurs at a rate λ_{off} while the latter is contingent on the size of the search neighborhood and number of peers in the neighborhood that are currently online and hosting the file. If requests for a particular file are generated at rate λ, the average number of queries generated per unit time is given by $\lambda P_{S_{on}}$. Further, each request on an average reaches z_{av} peers of which a fraction $P_{I_{on}}/N_P$ have the file being searched for. Here, N_P represents the total number of peers in the network, both on-line and off-line. The mean number of replicas present in the neighborhood of a peer is then $z_{av}P_{I_{on}}/N_P$. Thus, the rate at which the transition from $P_{S_{on}}$ into $P_{E_{on}}$ occurs is given by $\lambda P_{S_{on}} z_{av} P_{I_{on}}/N_P$. The peers per unit time exiting class $P_{S_{on}}$ total $(\lambda_{off} + \lambda z_{av} P_{I_{on}}/N_P)P_{S_{on}}$ and those entering number $r_1 P_{E_{on}} + r_2 P_{R_{on}} + \lambda_{on} P_{S_{off}}$. Combining the two gives the rate of change of membership of class $P_{S_{on}}$ as given in Equation (3). Equations characterizing the rates of change for the remaining compartmental classes can be derived in a similar fashion. It must be noted that the transition rates among the various compartments are assumed to be known entities.

5. Model Analysis

In this section, we analyze the model presented in the previous section, in totality and specific illustrative cases, and obtain the neccessary conditions for the global stability of the malware free equilibrium.

5.1. Malware Free Equilibrium

We now proceed with the derivation of the *basic reproduction number*, $\mathcal{R}_0$, a metric that governs the global stability of the malware free equilibrium (henceforth termed MFE). Here, $\mathcal{R}_0$ quantifies the number of vulnerable peers whose security is compromised by an infected host during it's lifetime. It is an established result in epidemiology [1], that $\mathcal{R}_0 < 1$ ensures that the epidemic dies out fast and does not attain an endemic state. Stability information of the MFE is important since this guarantees that the system continues to be malware free even if newly infected peers are introduced.

We follow the methodology presented in [6, 7], where "next generation matrices" have been proposed to derive the basic reproduction number. In this method, the flow of individuals between the states are written in the form of two vectors $\mathcal{F}$ and $\mathcal{V}$ which describe the inflow of new infected individuals and all other flows in the system, respectively. These vectors are then differentiated with respect to the state variables, evaluated at the disease (malware) free equilibrium, and only the part corresponding to the infected classes are then kept to form the matrices F and V, i.e.,

$$F = \left[\frac{\partial \mathcal{F}_i}{\partial x_j}(x_0)\right] \quad \text{and} \quad V = \left[\frac{\partial \mathcal{V}_i}{\partial x_j}(x_0)\right] \quad \text{with } 1 \leq i, j \leq m \tag{11}$$

where $\mathcal{F}_i$ and $\mathcal{V}_i$ are the i^{th} entries of $\mathcal{F}$ and $\mathcal{V}$, x_i is the i^{th} system state variable with $\dot{x}_i = \mathcal{F}_i(x) - \mathcal{V}_i(x)$, (x_0) is the disease free equilibrium and m is the number of infectious states. In our model, we have $m = 4$ corresponding to $P_{E_{on}}$, $P_{E_{off}}$, $P_{I_{on}}$ and $P_{I_{off}}$. Ordering the infectious states accordingly, from Equations (3)-(10) we have

$$\mathcal{F} = \begin{bmatrix} \lambda z_{av} P_{S_{on}} P_{I_{on}}/N_P \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{12}$$

and

$$\mathcal{V} = \begin{bmatrix} r_1 P_{E_{on}} + \mu P_{E_{on}} + \lambda_{off} P_{E_{on}} - \lambda_{on} P_{E_{off}} \\ \lambda_{on} P_{E_{off}} - \lambda_{off} P_{E_{on}} \\ \delta P_{I_{on}} + \lambda_{off} P_{I_{on}} - \lambda_{on} P_{I_{off}} - \mu P_{E_{on}} \\ \lambda_{on} P_{I_{off}} - \lambda_{off} P_{I_{on}} \end{bmatrix} \tag{13}$$

Now, at the malware free equilibrium (MFE), we have:

- $\frac{dP_{S_{on}}}{dt} = \frac{dP_{S_{off}}}{dt} = \frac{dP_{E_{on}}}{dt} = \frac{dP_{E_{off}}}{dt} = \frac{dP_{I_{on}}}{dt} = \frac{dP_{I_{off}}}{dt} = \frac{dP_{R_{on}}}{dt} = \frac{dP_{R_{off}}}{dt} = 0$
- $P_{I_{on}} = P_{I_{off}} = P_{E_{on}} = P_{E_{off}} = 0$

Substituting the above values in Eqns. (3) and (7), we get: $r_2 P_{R_{on}} = 0 \quad \Rightarrow \quad P_{R_{on}} = 0$. Again, using this result in Eqn (10) yields $P_{R_{off}} = 0$. Note that the total number of peers, given by $N_P = P_{S_{on}} + P_{S_{off}} + P_{I_{on}} +$

$P_{I_{off}}+P_{E_{on}}+P_{E_{off}}+P_{R_{on}}+P_{R_{off}}$, is a constant. Thus, at the MFE we have $N_P = P_{S_{on}} + P_{S_{off}}$, and using the relation from Eqn. (7), the peer distribution evaluates to the vector: $\{\hat{P}_{S_{on}}, \hat{P}_{S_{off}}, 0, 0, 0, 0, 0, 0\}$, where

$$\hat{P}_{S_{on}} = \frac{\lambda_{on} N_P}{\lambda_{on}+\lambda_{off}} \quad \hat{P}_{S_{off}} = \frac{\lambda_{off} N_P}{\lambda_{on}+\lambda_{off}}$$

Differentiating $\mathcal{F}$ and $\mathcal{V}$ with respect to $E_1, E_2, \cdots, E_{\mathcal{P}}, I_1, I_2, \cdots, I_{\mathcal{P}}$ and evaluating at the malware free equilibrium $\{\hat{P}_{S_{on}}, \hat{P}_{S_{off}}, 0, 0, 0, 0, 0, 0\}$, we have

$$F = \begin{bmatrix} \mathbf{0} & G \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad V = \begin{bmatrix} A & \mathbf{0} \\ -C & B \end{bmatrix}$$

with $\mathbf{0}$ representing a 2×2 zero matrix and

$$G = \begin{bmatrix} \frac{\lambda z_{av} \lambda_{on}}{(\lambda_{on}+\lambda_{off})} & 0 \\ 0 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} r_1 + \mu + \lambda_{off} & 0 \\ 0 & \lambda_{on} \end{bmatrix} - \tilde{M}$$

$$B = \begin{bmatrix} \delta + \lambda_{off} & 0 \\ 0 & \lambda_{on} \end{bmatrix} - \tilde{M}$$

$$C = \begin{bmatrix} \mu & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad \tilde{M} = \begin{bmatrix} 0 & \lambda_{on} \\ \lambda_{off} & 0 \end{bmatrix}$$

The basic reproduction number, $\mathcal{R}_0$, is then the largest absolute eigen value (spectral radius), of the matrix: FV^{-1}. That is: $\mathcal{R}_0 = \rho(FV^{-1})$, where $\rho()$ denotes the spectral radius. Using elementary matrix algebra and rearranging the terms, it can be easily verified that the product FV^{-1} can be broken down into $GB^{-1}CA^{-1}$, with the constituent matrices as enumerated above. Thus,

$$\mathcal{R}_0 = \rho(GB^{-1}CA^{-1}) \tag{14}$$

5.2. Illustrative Example

The dependency of the basic reproduction ratio on the model parameters is not immediately seen from Eqn. (14). The nature of the equation makes is difficult to decide if increasing or decreasing the value of a parameter affects $\mathcal{R}_0$ without actually simulating the model. Specifically, the intuition behind the need for modeling user behavior such as online-offline transitions is not obvious. We now illustrate, with the aid of a simple example, the impact of user behavior on the basic reproduction ratio. Peers going offline help to check the proliferation rate of the malware since the virus now has a smaller pool of vulnerable hosts whose security can be compromised. Thus P2P networks have an inherent quarantine mechanism built into their design and this feature can be exploited to curb the rate of infection. As we shall further demonstrate, a model without a provision for incorporating the user behavior often ends up overestimating $\mathcal{R}_0$, resulting in unnecessary and false alarms. In the simplified model we assume that peers do not spend time in the exposed state and that only the susceptible peers go offline. This essentially reduces to a SIR epidemic, the equations for which are

$$\begin{aligned} \frac{dP_{S_{on}}}{dt} &= -\lambda z_{av} P_{S_{on}} P_I / N_P + r_2 P_R \\ &\quad -\lambda_{off} P_{S_{on}} + \lambda_{on} P_{S_{off}} && (15) \\ \frac{dP_I}{dt} &= \lambda z_{av} P_{S_{on}} P_I / N_P - \delta P_I && (16) \\ \frac{dP_R}{dt} &= \delta P_I - r_2 P_R && (17) \\ \frac{dP_{S_{off}}}{dt} &= \lambda_{off} P_{S_{on}} - \lambda_{on} P_{S_{off}} && (18) \end{aligned}$$

Using the methodology described above, the basic reproduction number can be calculated as:

$$\mathcal{R}_0 = \frac{\lambda z_{av} \lambda_{on}}{\delta(\lambda_{on} + \lambda_{off})} \tag{19}$$

Now, consider the basic reproduction number (say $\mathcal{R}'_0$) for a model without the offline behavior, i.e., a peer is always on and in one of the following three states: susceptible, infected or immune. It can be seen that in this case: $\mathcal{R}'_0 = \frac{\lambda z_{av}}{\delta}$. Thus, we get $\frac{\mathcal{R}'_0}{\mathcal{R}_0} = \frac{(\lambda_{on}+\lambda_{off})}{\lambda_{on}}$. Indeed, if one assumes that a peer strictly alternates between on-line and off-line behavior, the probability that a peer is on-line at any given time can be derived as: $p_{on} = \frac{\lambda_{on}}{(\lambda_{on}+\lambda_{off})}$. Thus, if we assume $p_{on} = 0.5$, then a model not incorporating peer behavior ends up overestimating the epidemic threshold metric by a factor of two.

6. Results

In this section, we demonstrate the essence of our analysis presented thus far through numerical simulations. We first present our numerical results for the simple SIR epidemic described in the latter part of the previous section. The reason behind this is that the qualitative behavior of the model in Eqns. (3) - (10) is similar to that presented in Eqns. (15) - (18). Further, since the simplified model has a closed form expression for $\mathcal{R}_0$, it is easy to see it's dependence on various model parameters and this relationship can be extended to the more detailed model. The experiments were carried out using parameters emulating a 20000 node power-law graph with $\tau = 3.4$. The initial number of infectives was set at 50. From Eqn. (19), we see that $\mathcal{R}_0$ is directly proportional to λ_{on}. The essence of this equation is that, nodes staying on-line for long periods as compared to their off-line durations result in a higher intensity of malware presence in the network. Numerical simulations concurred with the above observation and are presented in Figure 3(a). A similar trend was observed for the detailed model as shown in Figure 3(b). The curve at the bottom corresponds to $\lambda_{on} = 0.1$ and the intensity of the epidemic increases monotonically with an increase in λ_{on}.

Again, Figure 4(a) substantiates our analytical result that requires the basic reproduction number to be greater than 1 for an epidemic to prevail. We see that if $\mathcal{R}_0 < 1$, the number of infected peers drops down to zero, else it reaches endemic proportions.

Finally, our argument in Section 5.2 for incorporating the user offline-online behavior in the system model is validated graphically in Figure 4(b). A system model without provision for the user behavior calculates $\mathcal{R}_0$ to be 1.0714 and predicts the presence of an epidemic while in reality, the true value happens to be 0.1531. In other words, a false alarm of an epidemic is generated. This is further confirmed by the numerical simulation which shows that the malware indeed dies out.

7. Conclusion

In the current work, we motivated the need to understand the dynamics of malware spread, especially in the context of interacting heterogeneous environments such as peer-to-peer networks. The need for an analytic framework incorporating user characteristics (e.g. off-line to on-line transitional behavior) and communication patterns (e.g. the average neighborhood size) was put forth by quantifying their influence on the basic reproduction ratio. It was proved analytically that a model that does not incorporate the above features runs the risk of grossly overestimating $\mathcal{R}_0$ and thereby falsely reporting the presence of an epidemic. Further, we show that the bound on the spectral radius for the spread of malware needs to take into account the underlying communication pattern, especially in a P2P kind of setting so as arrive at an accurate estimate.

8. Acknowledgements

This work was supported in part by NSF under grant 0347623.

References

[1] O. Diekmann, J. A. P. Heesterbeek, “Mathematical Epidemiology of Infectious Diseases: Model Builind, Analysis and Interpretation,” Wiley, 1999

[2] “Napster Protocol Specification,” March 12 2001, http://opennap.sourceforge.net/napster.txt

[3] Kazaa. http://www.kazaa.com

[4] Clip2 Company, Gnutella. http://www.clip2.com/gnutella.html

[5] B. Cohen, “Incentives Build Trust in BitTorrent,” May 2003, http://bitconjurer.org/BitTorrent/bittorrentecon.pdf

[6] P. van den Driessche and J. Watmough, “Reproduction numbers and sub-threshold endemic equilibria for compartmental models of disease transmission,” *Mathematical Biosciences,* vol. 180, pp. 29-48, 2002.

[7] J. Arnio, J. Davis, D. Hartley, R. Jordan, J. Miller and P. van den Driessche, “A multi-species epidemic model with spatial dynamics,” *Mathematical Medicine and Biology,* March 2005.

[8] L. Zhou, L. Zhang, F. McSherry, N. Immorlica, M. Costa and S. Chien, “A First Look at Peer-to-Peer Worms: Threats and Defences,” *4th International Workshop on Peer-To-Peer systems,* Ithaca, New York, February 2005.

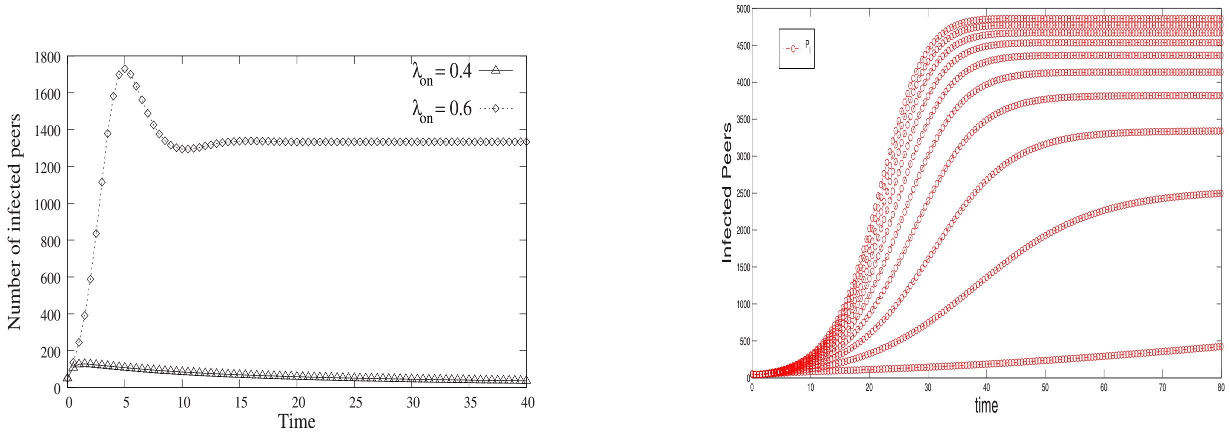
[9] A. J. Ganesh, L. Massoulie and D. Towsley, “The Effect of Network Topology on the Spread of Epidemics,” *Proceedings of IEEE INFOCOM,* Miami, USA, March 2005

[10] Y. Wang, D. Chakrabarti, C. Wang and C. Faloutsos, “Epidemic spreading in real networks: An eigenvalue viewpoint,” *SRDS 2003,* (pages 25-34), Florence, Italy

[11] M.E.J Newman, S.H. Strogatz, and D.J. Watts, “Random graphs with arbitrary degree distribution and their applications,” *Physical Review E,* vol. 64, no. 026118, 2001

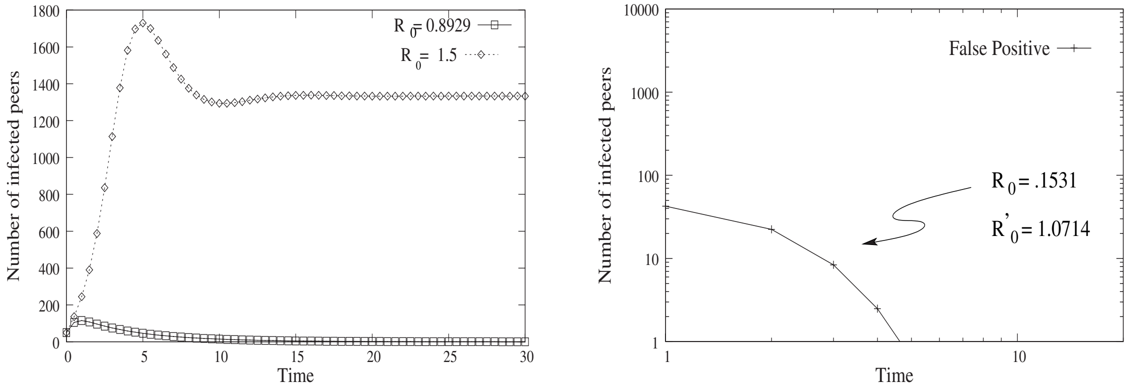
[12] D. Qiu and R. Srikant, “Modeling and performance analysis of BitTorrent-like peer-to-peer networks,” *Proceedings of ACM SIGCOMM,* Portland, OR, August 2004.

[13] X. Yang and G. de Veciana, “Service capacity in peer-to-peer networks,” *Proceedings of IEEE INFOCOM,* pp. 1-11, Hong Kong, China, March 2004.



(a) Simulation results for the system Eqn. (15) - (18) (b) Simulation results for the system Eqn. (3) - (10) ($.1 \leq \lambda_{on} \leq 1.0$)

Figure 3. Influence of online duration on infection intensity



(a) Simulation results for the system Eqn. (15) - (18) (b) False Positive: $\mathcal{R}_0' = 1.0714$ and $\mathcal{R}_0 = 0.1531$

Figure 4. Impact of $\mathcal{R}_0$ on epidemic existence

[14] M. Costa, J. Crowcroft, M. Castro and A. Rowstron, "Can we contain Internet worms ?," *HotNets-III: Third Workshop on Hot Topics in Networks,* San Diego, USA, 2003

[15] J. Mundinger and R. R. Weber, "Efficient File Dissemination using Peer-to-Peer Technology," *Technical Report, Statistical Laboratory Research Reports 2004-01,* 2004.

[16] `http://www.infoworld.com/articles/hn/xml/01/02/27/010227hnp2pvirus.html?0227alert`

[17] R.W. Thommes and M.J. Coates, "Epidemiological Models of Peer-to-Peer Viruses and Pollution," *Technical Report, Department of Electrical and Computer Engineering, McGill University,* June, 2005.

[18] R.W. Thommes and M.J. Coates, "Modeling Virus Propagation in Peer-to-Peer Networks," *Technical Report, Department of Electrical and Computer Engineering, McGill University,* June, 2005.