Three-Dimensional Based Trust Management Scheme for Virus Control in P2P Networks

Lin Cai and Roberto Rojas-Cessa,
Networking Research Laboratory
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Newark, NJ 07102, USA.
Email: {lc76, rojas}@njit.edu.

***Abstract*—Peer-to-peer (P2P) networking is widely used to exchange, contribute, or obtain files from any participating user. In these networks, worms, viruses and intruding files find an open door to the downloading host, creating a convenient environment for successful proliferation throughout the network. Trust management is a promising proactive mechanism to prevent virus dissemination. Current trust models use peer reputation for this purpose. However, when viruses have infectious properties, peer reputation may not be enough to limit their proliferation. In this paper, we show that peer reputation alone cannot bound epidemics in an infectious environment. Therefore, this paper introduces a trust management scheme that uses the combination of trust values of peers and infection values of both peers and content. Moreover, to improve the efficiency on the calculation of trust values of ratio-based normalization models, we propose a model for trust value calculation using a three-dimensional (3D) normalization to represent peer activity with high accuracy. We show that the proposed trust management scheme can bound virus proliferation to a small number of peers, without inhibiting file-downloading activity.**

***Index Terms*—Malware, P2P, peer-to-peer, trust management, virus proliferation, downloading activity.**

I. Introduction

In P2P networks, all peers provide resources, including bandwidth, storage space, and computing power. Therefore, when nodes join a P2P network, the demand for information increases and so does the total capacity of the system. The potential of these networks for information distribution is currently under the consideration for deployment of massive applications such as IPTV [1], [2], where video sources can rely on intermediate peers for further distribution of content. Furthermore, any user with Internet access and acceptable bandwidth can participate in complex distribution networks, as proven by Napster [3] and Gnutella [4] for sharing music files.

A peer user, usually interested in the content of the received information, pre-approves storing the downloaded file and, most likely, executes it. This pre-acceptance process leaves a front door for viruses to the local host. Several interesting studies about virus proliferation have been presented [15]-[16]. They consider network topology and features that describe the proliferation profile of a specific virus. Among other properties, viruses tend to have a spreading rate in function to the network density. Analysis of virus proliferation models is beyond the scope of this paper. Viruses or malware[1] have usually a disruptive objective, whether they aim to the host computer, to retrieve user information that can be illegally profitable, or to affect communication resources (e.g. denial of service). Depending on the characteristics, viruses in a host may or may not affect other stored files.

The general countermeasure in a host against viruses is the use of an anti-virus program. The successful detection by this protection software is based on the knowledge of existing hazardous files, which are identified by their properties or signatures. Therefore, a new virus can be unnoticeably hosted in a peer until the detection program is updated for its identification. Furthermore, after a virus is detected in a peer, the detection software may remove the threat. However, this information might be kept from other peers as it may be considered information of only local significance.

Trust management schemes are a promising approach to detect misbehavior and suppress malware propagation in P2P networks. In addition to use a level of trust for each peer, participants of P2P networks can distribute trust information about peers in different networks scenarios to decrease the effect of misbehaving hosts. Trust information about peers can be built through evaluation of the interaction history of peers [10], [11]. Moreover, trust-based incentive schemes can potentially discourage free-riders and selfish peers by only offering services to cooperative peers [12], [13], [14].

Recently, a few localized trust management systems have been proposed for supporting trusted collaborations and suppress malware propagation. The scheme in [14] calculates the trust value by getting votes from all peers. For a large scale P2P network, it may be complex to collect votes from the majority of the peers due to practical network constraints of each anticipated peer. The scheme in [18] is based on localized trust evaluation and in warning dissemination to prevent others from downloading a file from a suspicious peer. The scheme aims to limit the proliferation of malware under the assumption that there is no local file infection. In other words, when a malware-free peer downloads a file containing malware, other existing files in the peer are not infected. However, viruses may seek to spread themselves by piggy-backing onto other

[1] We may refer to virus or malware interchangeably in this paper as we are targeting those with similar characteristics in their proliferation.

files, or infecting them. With this viral characteristic in mind, we consider the infectious risk of files in P2P networks.

To bound virus proliferation, we propose a new trust management scheme that considers the combination of peer trust values and a warning messaging system with file reputation and peer infectious values. In addition, we proposed to a three-dimensional normalization of trust and reputation values that provides information about the transaction history between two peers with more accuracy than a ratio based scheme does. We call this 3D based trust management scheme. We show the performance of current approach in trust management under file infectious possibilities and show how file infectious underscores trust management schemes based on peer reputation. We analyze the performance of the proposed scheme using computer simulation. The results show that our scheme is efficient for virus epidemic control in P2P networks.

The remainder of this paper is organized as follows. Section II describes the proposed scheme, the terms and the parameters for evaluation of peer trust, and the operation of the proposed management scheme in a P2P network. Section III shows the performance results obtained through computer simulation. Section IV presents our conclusions.

II. Combined Reputation-based Trust Model

The considered P2P network has N peers. Each peer has a file reputation table and a peer trust table. A file reputation table holds file identifications known to have a high profile in the network, and it is used for evaluating the risk associated with a file. We assume that file identities remain unchanged for long periods of time. Although there are several alternatives to assigning this identification to files (e.g., file name or file length), this is out of the scope of this letter. The trust table stores trust and infectious values, which are used by the peer to select a downloading peer source. The trust table in peer i is denoted as $T(i)$. The trust value of peer i on peer j, is denoted as $T_v(i,j)$, where $T_v(i,j) \in [0,1]$. For example, $T_v(i,j) = 0$ means that peer i has no trust for peer j and any filed downloaded from j would expected to be infected with probability 1.0. On the other hand $T_v(i,j) = 1$ means that peer i trusts peer j and any file downloaded from j is expected to be innocuous with probability 1.0. Therefore, in the selection of the downloading source, peer j has top priority to become the downloading source. Peer i updates his trust table after downloading a file from peer j by re-evaluating its trust and infectious values about peer j according to the experienced interactions with peer j. Any peer in the system that is trusted by any other peer is called trustee and any peer that trusts a trustee is called truster. A peer has higher trust on those referred by its immediate trustee than on trustees of trustees. Both the distance of the trustee and the relationship type define the term *social distance*. In general, the trust value decreases proportional to the *social distance* between peers. The second value in the trust table is the infectious value $I_v(i,j)$, which represents the information that peer i has about the possibility of having files infected at peer j. A large infectious value means a high possibility that files are infected in peer j due to the presence of an infected file in the past. An infected download is defined as a download of a file containing a detected virus. A clean download is defined as a download of a file with no (detected) virus. When a peer downloads an infected file, other existing files in this peer can get infected with probability P_I. A peer has a virus-detection software that detects a virus with probability P_d.

The file reputation table holds the reputation value of file f_l as $F(i, f_l)$, which indicates the historical record of whether a file has been a virus carrier (or a virus itself) at peer i as reported by others. This value is calculated as a function of the number of warning messages received about f_l, and the value increases with each warning message.

A. Ratio Based Local Trust Values Normalization

In a distributed environment, peers rate each other after each transaction. For example, in Eigentrust [17], peer i may rate a download as negative, if the file downloaded is inauthentic, malicious, tampered with, or if the download is disturbed when passing trough known unreliable links. Each peer i stores the number satisfactory transactions it has had with peer j, $sat(i,j)$ and the number of unsatisfactory transactions it has had with peer j, $unsat(i,j)$. In order to aggregate local trust values, Eigentrust uses a normalized local trust value c_{ij} as $c_{ij} = \frac{S_{ij}}{\sum S_{ij}}$, where S_{ij} is defined as $S_{ij} = sat(i,j) - unsat(i,j)$ and $\sum S_{ij}$ is the number of differential transactions of peer i with all other peers it has interacted with. Through the normalization procedure, all values are bounded between 0 and 1. Previous works in P2P reputation systems [1]-[8] have all been based on similar notions of local trust values. In *dynamic trust* [18], the local trust value is calculated as $c_{ij} = \frac{sat(i,j)}{tol(i,j)}$, where $tol(i,j)$ means total number of transactions between peer i and peer j which is equal to $sat(i,j) + unsat(i,j)$. We can observe that all above trust normalized values are obtained through a ratio-based calculation.

However, there are some drawbacks of normalization approach. These c_{ij} values are relative and therefore, there is no absolute interpretation. For example, if $c_{ij} = c_{ik}$, we know that peer j has the same reputation as peer k as peer i has of peer j, but it is unknown if both are highly (or hardly) reputable, or if both of them are mediocre. The reputed peer and the mediocre peer may have the same trust value. For example, let's assume that $sat(i,j) = 1$, $total(i,j) = 1$, $sat(i,k) = 10000$, and $total(i,k) = 1000$, the we know that $c_{ij} = \frac{1}{1} = 1$, $c_{ik} = \frac{10000}{10000} = 1$ by performing the calculation according to *dynamic trust*. For peer i, the trust values of c_{ij} and c_{ik} are equal, which is unfair to peer k since the total number of transaction between peer i and peer k is ten thousand times higher than the total number of transaction between peer i and peer j.

B. Three Dimensional Value Normalization

Different with the ratio calculation, the calculation is based on the ideas of closeness, used mainly in studying the behavior of functions close to values at which they are undefined. For example, the function $y = \alpha^{-\frac{\beta}{x}}$, where $0 < \alpha < 1$ and

$\beta > 1$, is close to 1 as x increases, and α and β are two parameters controlling the approaching speed to the given value 1. Specifically, the approaching speed to the given value 1 is reduced with the increase of β.

Based on $y = \alpha^{-\frac{\beta}{x}}$, we make the y-axis as the spinning axle and x-axis as the base. After 360 degree rotation, we get a three-dimensional curve or a surface.

To determine the tri-dimensional surface, we cut the surface vertically from the top. To calculate the trust value, we make the $sat(i, j)$ and $tol(i, j)$ as the input variables x and y. We define the local trust value as:

$$c_{ij} = A(i, j) \times \alpha^{-\frac{\beta(i,j)}{\sqrt{sat(i,j)^2+tol(i,j)^2}}} \tag{1}$$

$\beta(i, j)$ in peer i is dynamically updated according to the performance of peer j. $A(i, j)$ reflects the total number of complains sent to peer i pointed by peer j, where $0 < A(i, j) < 1$. We choose to normalize the local trust values in this manner because it models the trust value aggregation fairly and it can reflect the real transition history accurately. Moreover, the boundary of the trust function is between 0 and 1. Hence, the advantages of the previous schemes obtained through trust value normalization are maximally preserved in the new scheme. Furthermore, the proposed scheme has some new features. First, the trust value is not only related to the proportion of satisfied transactions but also the total number of transaction history. Second, the model is more flexible since we can adjust different approaching speeds by adjusting α and β. For example, in the startup period, we can choose a small slope to control the increase speed of the trust value, if a peer constantly provides satisfactory transactions, a larger slope, means quicker approach speed to 1, can be given to the corresponding peer. Through the three dimension trust value management, peers have more flexibility to control the trust value calculation.

C. Management Scheme

The trust management scheme works as follows. When peer i searches for file f_l, it checks the local file's reputation in the file record. If the file's reputation value is found at the database and is above the acceptable reputation threshold, Th_R, then the peer proceeds to find the file source.

The values held by a peer are updated after different actions take place. These are described as follows.

File Search. A peer i sends a request for f_l to all trustees whose trust value is above the admissible threshold value Th_T (i.e., trustable trustees). Peer i chooses the peer that has the largest T_v and the lowest infectious value among those who have a copy of the requested file. If the file is not available from peer i's trustable trustees, the peer sends a recursive query for f_l to all trustees. In this query, the receiving trustee searches for the requested file among its own trustees. This process is performed recursively until either a fruitful search is achieved or there are no more trustees to query. After a recursive query, if peer k is introduced to i, new values are calculated: $T_v(i, k) = T_v(i, j)T(j, k)$, and $I_v(i, k) = I_v(i, j) + I_v(j, k)$, then the peer proceeds to the selection of a downloading source.

Post-download update. If the download of f_l is determined to be clean:

$$\begin{aligned} sat(i,j) =& \quad sat(i,j) + 1 \\ tol(i,j) =& \quad tol(i,j) + 1 \\ T_v(i,j) =& \quad A(i,j) \times \alpha^{-\frac{\beta(i,j)}{\sqrt{sat(i,j)^2+tol(i,j)^2}}} \end{aligned}$$

$I_v(i, j)$ remains unchanged.

If the download of the f_l is determined infected:

$$\begin{aligned} \beta(i,j) =& \quad \beta(i,j) + 1 \\ T_v(i,j) =& \quad A(i,j) \times \alpha^{-\frac{\beta(i,j)}{\sqrt{sat(i,j)^2+tol(i,j)^2}}} \\ I_v(i,j) =& \quad I_v(i,j) + 1 \\ F(i,f_l) =& \quad F(i,f_l) + 1 \end{aligned}$$

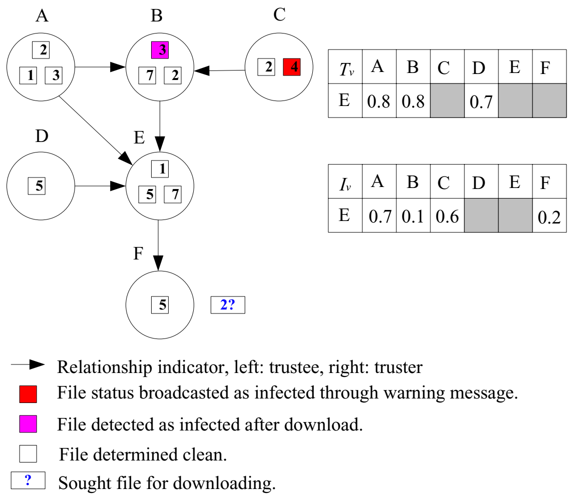
During this phase, if $T_v(i, j) < th_w$, where th_w is the threshold to trigger a warning process, peer i issues warning messages to all its trusters. In this way, peers exchange only critical information about other interacting peers. A warning message has the following format: $\{ID, v_j, f_m, \Delta, d\}$, where ID is the warning identification number, v_j is the identification of the peer that served as the source of a threatening file, f_m is the file's name, Δ indicates the decrement of the trust value at peer i, and d is the maximum number of truster hops the warning message is allowed to propagate.

Post-warning updates. After receiving a warning message from peer k about peer j, peer i updates the trust values. If $T_v(i, k) > Th_T$:

$$\begin{aligned} A(i,j) =& \quad A(i,j) \times \theta \\ T_v(i,j) =& \quad A(i,j) \times \alpha^{-\frac{\beta(i,j)}{\sqrt{sat(i,j)^2+tol(i,j)^2}}} \\ I_v(i,j) =& \quad I_v(i,j) + \frac{(d-1)}{d} \\ F(i,f_l) =& \quad F(i,f_l) + \frac{(d-1)}{d} \\ \Delta =& \quad \Delta\frac{d-1}{d}. \end{aligned}$$

θ is the rate of warning message aggregation and $0 < \theta < 1$. Because the forwarding of the warning message is bound by d, this value is also updated as $d = d-1$. If the updated $d > 1$ and $\Delta T_v(k, i) > th_w$, peer i sends a warning message to its trusters with the updated values.

Figure 1 shows a simple example of the truster-trustee relationship between Peers A to F in a P2P network. The tail of an arrow indicates the trustee peer and the head indicates the truster. This example shows the social distance between Peers A and F as d = 2. In this example, Peer F seeks File 2, available in peers A, B, and C. However, Peer F has no T_v and I_v values for those nodes because Peer F has not been a trustee of them yet. Therefore, Peer F estimates the first values of them via Peer E's intervention. Peer E has a high T_v value about Peer A (T_v(E, A)=0.8), high T_v value about Peer B (T_v(E, B)=0.8), and no T_v value about C. Furthermore, Peer E has a high I_v value about Peer A (I_v(E, A)=0.7) as Peer B has sent a warning message about Peer A after the downloaded File 3 was detected to be infected (and the T_v value has dropped enough to trigger the warning). Peer E also has a moderate I_v value about Peer B (I_v(E, B)=0.1) and a high I_v value about Peer C (I_v(E, C)=0.6) as a warning was received about File 4. After that, Peer F knows that: Peer A has an acceptable T_v value and a high I_v value, Peer B has a high T_v value and a moderate I_v value, and Peer C has low T_v and high I_v values. Peer F then decides to download File 2 from Peer B, as B has the lowest I_v value between A and B. In a different case, if F seeks File 4, it would notice that this file is considered viral, and it would desist.


A
B
C
D
E
F
2
1 3
3
7 2
2 4
5
1
5 7
5
2?
Relationship indicator, left: trustee, right: truster
File status broadcasted as infected through warning message.
File detected as infected after download.
File determined clean.
?
Sought file for downloading.

T_v	A	B	C	D	E	F
E	0.8	0.8		0.7		

I_v	A	B	C	D	E	F
E	0.7	0.1	0.6			0.2

Fig. 1. Example of relationship of peers in P2P network and files.

III. Performance Analysis

We simulated a P2P network using a mesh topology, with 100 nodes selected randomly as active peers in the mesh. In the beginning, the peers do not know each other. The trust relationship is built through the downloading interactions. Here, we consider two trust value normalization approaches in the trust model. One is ratio based (RA) trust model and another one is 3D model, the approaches are labeled by they nomenclature in the following graphs, for brevity. Here, time is considered slotted, where time slots have same duration, and a task, such as a download or trust value evaluation, occurs in a time slot. We show the robustness of the system by estimating the number of nodes infected. In the trust model, the attacker joins the system from the third time slot. The attack contains some clean files and some infected files or viruses. A peer downloads a file from an unknown peer if and only if he can not find the downloading source anywhere else. There are 150 files in the mesh network. Among them, 20% percent are popular files. To reveal the effect of the attacker to the network, there are 10 files owned by the attacker.

Figure 2 shows the performance of the RA and 3D based schemes with different local virus detection probability. This figure shows two curves with the 3D based scheme in solid lines, and the RA scheme in empty lines. From this figure, we can see that after the attacker joins the network, the performance of the RA scheme degrades quickly. After 1400 downloads, almost all peers are infected. The attacker successfully subverts the system. However, the infection is controlled in certain degree in 3D based scheme. Two-thirds of the peers are still clean after 1400 downloads. This shows that the new trust management scheme can bound the malware proliferation in the network using a reliable detection software and without considering local infection at a peer.

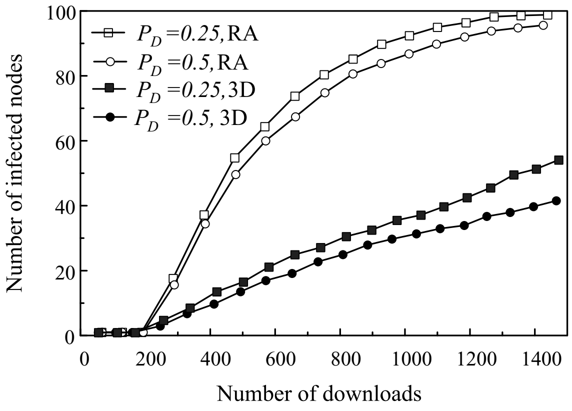


Fig. 2. Comparison of RA and 3D based schemes, under $P_D = \{0.25, 0.5\}$, with no local infection and alert delay.

Figure 3 shows the effect of internal infection $P_I = \{0, 0.25, 0.5\}$. Specifically, it shows shows the degree of proliferation of malware using the RA scheme and our proposed 3D scheme where I_v is used, under $Pd = 0.5$ with no propagation delay for distributing the alert messages. The system performance degrades quickly with the increase of P_I. These results are also shown in terms of the number of downloads. In this figure, the performance of the RA scheme decreases as the P_I increases, i.e., the chances of infecting other files in the same host by malware increases. On the other hand, with the proposed 3D scheme, the impact of the infection probability is greatly decreased. In the case of a high P_I value, or $PI = 0.5$, the number of infected peers drops from 90 nodes as in the case of the RA scheme to close to 50 peers in the 3D based scheme after 1000 downloads.

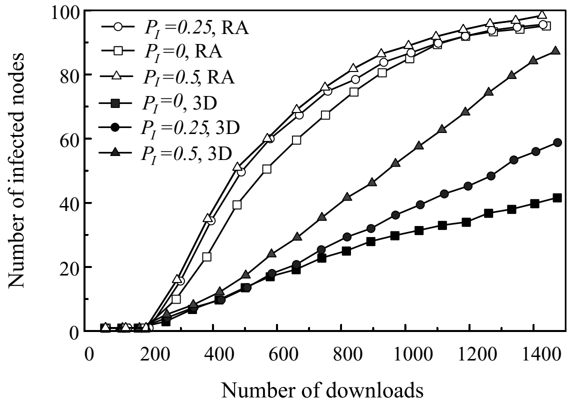


Fig. 3. Comparison of RA scheme and 3D based scheme, under $P_D = 0.5, P_I = \{0, 0.25, 0.5\}$, with no local infection and alert delay.

Figure 4 shows the evaluation of the total number of infected peers after each time slot. The curves for different P_I in Figure 4 show a similar performance to that in Figure 3. Compared with the RA scheme, the 3D based scheme not only inhibits the proliferation of malware but also bounds it.

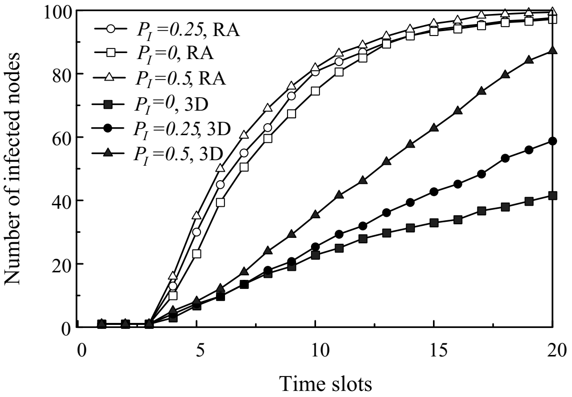


Fig. 4. Comparison of RA and 3D based scheme schemes under $P_D = 0.5, P_I = \{0, 0.25, 0.5\}$, with no local infection and alert delay.

We also evaluated the download activity of the network using the same conditions as above. Figure 5 shows the download activity of a network using the 3D based scheme, in downloads per time slot. The results show that the download activity with different P_I values has no significant changes. This means that the proposed approach does not discourage network activity.

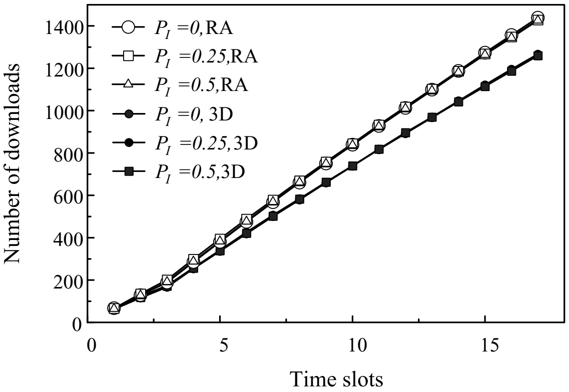


Fig. 5. Download activity of the network using the 3D based scheme.

IV. Conclusions

Trust management is a promising strategy to bound the proliferation of malware on peer-to-peer networks that can work jointly with virus detection systems. In this paper, we showed that the use of ratio based trust value updating has deficiencies in bounding the proliferation of malware. In most cases, it is highly probable that the majority of peers become infected.

Therefore, we proposed a trust management scheme that use infectious value on peers and file reputation to determine the probability of infection. To address the ambiguity that ratio based normalization used in the calculation of trust values, we proposed using a a three-dimensional (3D) normalization method. By using 3D based method and the infectious value or a peer, where the consideration of a peer having hosted an infected file, the proliferation of viruses becomes effectively bounded. By using computer simulation of a meshed peer-to-peer network we showed the improvement of this proposed approach. Furthermore, considering that trust parameters to bound proliferation have the potential of discouraging download activity in P2P networks, we studied the impact of using our proposed scheme. The simulation results showed that our approach has no impact on the download activity of the network.

References

[1] X. Xu, Y. Wang, S. P. Panwar, and K. W. Ross, "A Peer-to-Peer Video-on-Demand System using Multiple Description Coding and Server Diversity," Proc. *IEEE International Conference on Image Processing (ICIP)*, pp. 1759-1762, October 2004

[2] X. Hei, C. Liang, J. Liang, Y. Liu, and K.W. Ross, "A Measurement Study of a Large-Scale P2P IPTV System," *IEEE Trans. on Multimedia*, pp.1672-1787 , December 2007.

[3] M. Macedonian, "Distributed File Sharing: Barbarians at the Gate?" *IEEE Computer*, Vol. 33, Issue 8, pp. 99-101, Aug. 2000.

[4] Y. Wang, X. Yun, and Y. Li, "Analyzing the Characteristics of Gnutella Overlays," Proc. *IEEE IV International Conference in Information Technology*, 2007, pp. 1095-1100, 2-4 April, 2007.

[5] L. Xiong and L. Liu, "PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities," *IEEE Transactions on Knowledge and Data Engineering*, 16(7), 843-857, July 2004.

[6] P. Herrmann, "Trust-Based Procurement Support for Software Components," *Proc. 4th International Conference of Electronic Commerce Research (ICECR04)*, pp. 505-514, 2001.

[7] K. Walsh and E.G. Sirer, "Fighting Peer-to-Peer SPAM and Decoys with Object Reputation," *Proc. Third Workshop on the Economics of Peer-to-Peer Systems (P2PECON)*, pp. 138-143, Aug. 2005.

[8] G. Theodorakopoulos and B.J. Baras, "On Trust Models and Trust Evaluation Metrics for Ad Hoc Networks," *IEEE Journal on Selected Areas in Communications,* 24(2); 318-328, Feb. 2006.

[9] P. Li, Z. Wang, and X. Tan, "Characteristic Analysis of Virus Spreading in Ad Hoc Networks," *Proc. IEEE Workshop in Computationall Intelligence and Security (WCIS) 2007*, pp. 538-541, December 2007.

[10] E. Damiani, D. C. Vimercati, S. Paraboschi, P. Samarati, and F. Violante, "A Reputation-based Approach for Choosing Reliable Resources in Peer-to-Peer Networks," Proc. of *the 9th ACM conference on Computer and communications security (CCS)*, pp. 207-216, Washington, DC, November 2002.

[11] J. Shin, T. Kim, and S. Tak, "A Reputation Management Scheme Improving the Trustworthiness of P2P Networks," Proc. *IEEE International Conference on Convergence and Hybrid Information Technology (ICCHIT) 2008*, pp. 92-97, August 28-30, 2008.

[12] K. Lai, M. Feldman, I. Stoica, and J. Chuang, "Incentives for cooperation in peer-to-peer networks," Proc. *Workshop on Economics of Peer-to-Peer Systems*, pp. 997-1001, 2003.

[13] K. Ranganathan, M. Ripeanu, A. Sarin, and I. Foster "To share or not to share: An analysis of incentives to contribute in collaborative file sharing environments," *Proc. Workshop on Economics of Peer-to-Peer Systems* , pp.587-594, 2003.

[14] S. Marti and H. Garcia-Molina "Limited Reputation Sharing in P2P Systems," *Proc. of the 5th ACM Conference on Electronic commerce (EC)*, pp. 91-101, May 2004.

[15] X. Zhang, D. Saha, and H.H. Chen, "Analysis of Virus and Antivirus Spreading Dynamics," Proc. *IEEE Global Communications Conference (Globecom) 2005*, Vol. 3, pp. 1822-1826, Nov. 28-Dec 2, 2005.

[16] L-C. Chen and K.M. Carley, "The Impact of Countermeasure Propagation on the Prevalence of Computer Viruses," *IEEE Trans. on System, Man, and Cibernetics*, Vol. 34, issue 2, pp. 823-833, April 2004.

[17] S.D. Kamvar, M.T. Schlosser, and H. Garcia-Molina "The eigentrust algorithm for reputation management in p2p networks," *Proc. 12th International World Wide Web Conference*, pp. 785-791, May 2003.

[18] X. Dong, W. Yu, and Y. Pan "A Dynamic Trust Management Scheme to Mitigate Malware Proliferation in P2P network," Proc. *IEEE International Conference on Communications 2008*, pp. 1605-1609, Beijing, China, May 2008.