Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 94 (2016) 366 – 371

Procedia Computer Science

International Workshop on Big Data Security and Trust Computing (BDSTC 2016)

An EigenTrust-based Hybrid Trust Model in P2P File Sharing Networks

Kun Lu[a], Junlong Wang[a], Ling Xie[a], Qilong Zhen[a], Mingchu Li[a,*]

[a]*School of Software Technology, Dalian University of Technology, Dalian, 116621, China*

Abstract

Reputation systems have been proposed to distinguish malicious peers and ensure the quality of services in P2P networks. However, only relying on global reputation scores to assess the reliability of file providers can weaken the subjective opinions of file requesters, which is vulnerable to some malicious attacks, e.g., camouflage attack. To address this issue, in this paper, we propose a novel hybrid trust model, namely HTM, based on the EigenTrust reputation management system. In HTM, a file requester utilizes a two-phase reference method to take both the global reputation and the direct trust of a file provider into consideration when deciding whether to download from it or not. In particular, in the first phase, a requester selects a provider from the responders based on the roulette wheel selection algorithm; while in the second phase, the requester considers the direct trust on the selected provider with our proposed direct trust evaluation approach. The extended simulation results show that our mechanism can identify and isolate the malicious peers both under individual attack and camouflage attack effectively, thus improving the performance of P2P file sharing networks.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Hybrid Trust; File sharing system; Reputation system.

1. Introduction

To resist malicious attacks and ensure the reliable service delivery in P2P networks, reputation systems are indispensable. EigenTrust[1], PeerTrust[2], GenTrust[3], and SecuredTrust[4] are some of reputation systems that provide defence against malicious attacks. A reputation system calculates the reputation score of a peer by aggregating the ratings from all the valuators who have interacted with this peer[5]. By referring to the global reputation scores, a peer can make preferable decisions when requesting services. EigenTrust[1] is one of the most popular reputation systems that is used practically in Gnutella. By providing the global reputation scores to file requesters when they select file owners, EigenTrust can identify and isolate malicious peers effectively.

However, in real-world scenarios, direct experience is also very important in judging the characteristics of others: people tend to ask for the recommended trust of another one when they are not familiar with each other, but they may

* Corresponding author. Tel.: +86-134-7869-1084 ; fax: +86-0411-6227-4467.
E-mail address: mingchul@dlut.edu.cn

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.08.056

trust their own judgements more than the recommended information after they know each other[6]. In fact, EigenTrust has a poor performance in the case of malicious peers camouflaging themselves as being trustworthy by uploading 40% authentic files, which can result in the inauthentic downloading rate increases to 28% in the network[1]. In such situations, inauthentic files can be downloaded easily, which reflects the fact that the reputation scores of peers are not accurate enough[7]. Although a peer can continue to download from other peers until it obtains the desired file, such multiple downloads can result in transmitting large amounts of junk or malicious data. It wastes bandwidth and decreases the efficiency of the application system.

Based on EigenTrust, we propose a two-phase hybrid trust model for P2P file sharing networks in this paper. The aim of our work is to provide a better mechanism to validate the trustworthiness of a peer so that the judgement upon file owners is more precise, thus reducing the impact of malicious attacks. The basic idea is that the requester in an interaction should consider both the global reputation and the direct trust of the file owner. Simulation results show that our proposed model is effective in isolating malicious peers under individual attacks and camouflage attacks.

The rest of the paper is organized as follows: Section 2 illustrates the proposed hybrid trust model. Section 3 demonstrates our simulations and analysis. Finally, we conclude our paper in section 4.

2. Hybrid Trust System

In this section, we first propose a direct trust evaluation method, and then give an indirect trust evaluation based the iterative approach in EigenTrust[1].

2.1. Direct Trust Evaluation

Beth[8] proposed a simple method to calculate direct trust value, which is shown in Eq. (1):

$$dt = 1 - \alpha^p \tag{1}$$

where p represents the positive number of evaluations that peer i to peer j and α is a system parameter between 0 and 1.

It is apparent that the trust value in Eq. (1) is not less than $(1 - \alpha)$, i.e., the trust is dependent on the value of α. Here, we give a variation of Eq. (1) and propose our direct trust assessing method in Eq. (2):

$$dt_{ij} = \begin{cases} (1 - 0.5^d) \cdot sat_{ij}/s\ , & if \quad d \geq 0; \\ 0 \qquad\qquad\quad , & otherwise. \end{cases} \tag{2}$$

where $d = sat_{ij} - unsat_{ij}$, and $s = sat_{ij} + unsat_{ij}$. And $sat_{ij}, unsat_{ij}$ represent the satisfactory and unsatisfactory transaction times between peer i and peer j, respectively.

From Eq. (2), we can know that dt_{ij} increases from 0 to 0.5 right off when a peer download a satisfied file from another peer for the first time. However, as the times of satisfactory transactions increase, the growth rate of direct trust value slows down. In Eq. (2), sat_{ij}/s is a simple regulatory factor which distinguishes different interaction histories. For example, assume 6 interactions in total, 4 are satisfactory, and 2 are unsatisfactory. Then, the result is 1/2 if the factor is considered, but it is 3/4 without the regulatory factor.

2.2. Indirect Trust Evaluation

The normalized direct trust of peer i to peer j (noted as c_{ij}) is denoted in Eq. (3):

$$c_{ij} = \frac{\max(dt_{ij}, 0)}{\sum_j \max(dt_{ij}, 0)} \tag{3}$$

When $\sum_j \max(dt_{ij}, 0) = 0$, then $c_{ij} = p_i$, and $p_i = 1/|P|$ if $i \in P$ (P is the pre-trusted peers set), otherwise $p_i = 0$.

Peer i may ask its friends' opinions about the target peer k. Define the indirect trust of peer i to peer k as t_{ik}, then the indirect trust calculation in EigenTrust is shown in Eq. (4):

$$t_{ik} = \sum_j c_{ij} c_{jk} \tag{4}$$

Let C be the matrix $[c_{ij}]$, and $\vec{t}$ be the vector that contains t_{ik}; then $\vec{t} = C^T \vec{c}_i$. Peer i also can ask the friends of its friends ($\vec{t} = (C^T)^2 \vec{c}_i$). After continuous iterations ($\vec{t} = (C^T)^n \vec{c}_i$), i will be able to know the trust distribution of the entire network. If n is large enough, the trust vector $\vec{t}_i$ will converge to the same vector $\vec{t}$ for all peers. In other words, $\vec{t}$ can be regarded as the vector of reputation scores of all the peers. Further, EigenTrust uses Eq. (5) to restrain malicious attacks:

$$\vec{t}^{(k+1)} = (1 - a) C^T \vec{t}^k + a\vec{p} \tag{5}$$

where a is a constant that is between 0 and 1 and $\vec{p}$ is the vector that contains p_i.

2.3. Phase One

In the first phase, the reputation scores help peers to single out a potentially reliable file provider. That is to say, after sending a file query, the requester chooses a server from the file query responders based on their reputation scores.

Now, we explain how to select the a provider. An alternative way is always selecting the peer with the highest reputation score from the response list. The global reputation score of a specified peer is same for every other peer, so the most reputable peer who is online currently will be always selected. Hence, the upload pressure of this peer would be high. To solve this problem, we use the roulette wheel selection (RWS) algorithm[9] here. The idea is that the probability of a server being selected should be proportional to its reputation score. Assume there are N peers in the network, each characterized by its reputation $r_i \geq 0 \quad (i = 1, 2, \ldots, N)$. The selection probability of the ith peer is:

$$sp_i = \frac{r_i}{\sum_{j=1}^{N} r_j}, \quad (j = 1, 2, \ldots, N). \tag{6}$$

However, the RWS algorithm is not fair to good peers who have a reputation score of 0 (e.g., the new comers) because that they have no chances to be selected as servers to upload files. To address this issue, a file requester can select from non-reputable providers with a probability of 10% and select a provider whose reputation is greater than 0 using RWS with a probability of 90%.

2.4. Phase Two

In the second phase, a file requester makes a decision of whether to download from the potential provider that selected in phase one or not. And this process is based on the direct trust level of the requester on the provider. First, assume that every peer has a direct trust threshold in another peer. In realistic scenarios, the thresholds of different peers may be distinct. For the sake of simplicity, we further assume that the thresholds of all the peers are same in this paper. Then the second phase of HTM can be described as follows: The requester, R, considers whether it has had previous interactions with the selected peer, S. If they had no interaction history, then R chooses to download the file from the S directly. Otherwise, R will reference to the direct trust on S. If the direct trust value is higher than $R's$ threshold, then it chooses to establish a connection and begin to download the file from S, otherwise it will delete S from the response list and reselect another file provider as phase one shows. The requester repeats the above process until it has selected a satisfactory provider or there are no providers available any more.

3. Experiments and analysis

This section presents the simulation and analysis. Here, we consider two attack models: individual attack and camouflage attack. Malicious peers in individual attack are isolated malicious peers, which means that the malicious peers do not know each other, and they attack the system by always uploading inauthentic file and giving negative evaluations to good downloads. Malicious peers in camouflage attack are aware of each other and always give positive evaluations each other. Moreover, they camouflage themselves by uploading some good files to enhance their credibility, thus being selected as a server and uploading inauthentic files more often.

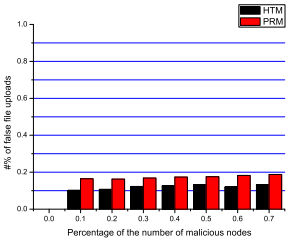

Fig. 1. Fraction of inauthentic files uploaded.

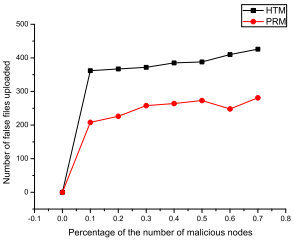

Fig. 2. Number of inauthentic files uploaded.

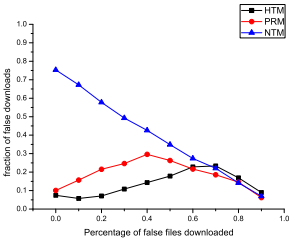

Fig. 3. Fraction of inauthentic files downloaded.

We conduct several simulation experiments and analyse the two attack models qualitatively. We use C language version of the QTM: P2P Trust Simulator framework [10] developed by University of Pennsylvania to deploy our experiments.

3.1. Set up

We simulate a small-scale P2P network containing less than 100 peers. In particular, the number of peers in the two attack models are 63 and 73, respectively, which is the same as EigenTrust. The initial details of the experimental configuration are listed in Table 1, and GP, MP and PTP stand for good peers, malicious peers and pre-trusted peers, respectively.

Table 1. Parameter Configuration in Simulations.

Notation	Definition
Parameter	Value
No. of PTP	3
No. of GP	60
No. of neighbors of MP	10
No. of neighbors of PTP	10
File number	20
Files at GP	3
File at MP	20
Top % of queries GP and PTP response to	5%
Top % of queries MP response to	20%
File provider selection algorithm	RWS
Probability of peer i with $r_i = 0$ being selected	10%
Probability of MP upload good files	0%
Direct trust threshold of every peer	0.5

3.2. Individual attack

In individual attack, good peers upload reliable files and give subjective feedbacks. And individual malicious peers always upload inauthentic files and give inauthentic feedback. We conduct eight group of experiments. In each group, there are different percentages of malicious peers and good peers. The percentage of malicious peers ranges from 0 to 70%, increasing by 10% each time from the first group to the eighth group.

Fig. 1 compares the percentage of inauthentic files uploads and total uploads under the pure EigenTrust reputation mechanism (PRM) and our hybrid trust mechanism (HTM). The statistical indicator is the percentage of uploaded inauthentic files in the network. The ordinate axis shows the malicious proportion in the network and the horizontal axis is the different percentages of malicious peers. The results are averaged over 10 runs.

Fig. 2 is similar to Fig. 1, but the statistical indicator is the number of inauthentic files that are uploaded under different proportion of malicious peers participating in the network.

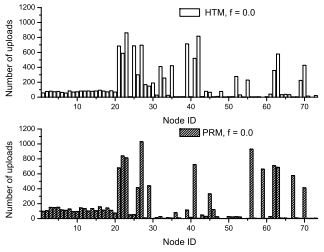

Fig. 4. Uploads distribution, $f = 0.0$.

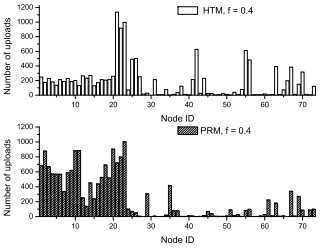

Fig. 5. Uploads distribution, $f = 0.4$.

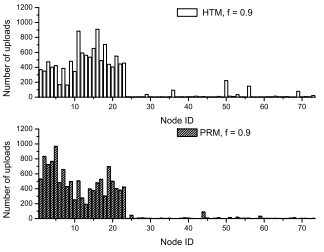

Fig. 6. Uploads distribution, $f = 0.9$.

Fig. 1 and Fig. 2 show that HTM performanced better than PRM. This benefits from the advantages of HTM, i.e., when selecting a server, the requester considers not only the reputation score but also the direct trust of the file owner. For the target peer that had an interaction history with the requester, a reference to direct trust determines the identity of the destination peer more accurately, which can reduce the probability of downloading files from a malicious peer. Therefore, HTM's accuracy in predicting the credibility of a particular download source is better than that of PRM.

3.3. Camouflage attack

In camouflage attack, the behaviours of good peers are similar to those in individual attack: they offer good files and give subjective feedback. However, malicious peers know each other and give positive feedbacks among them no matter whether the downloaded files are good or inauthentic. And malicious peers try to camouflage themselves by uploading a good file with a probability of f. In this scenario, there are 73 peers in the network, including 53 good peers, 3 pre-trusted peers and 20 malicious peers. We choose the situation of non-trust mechanism (NTM) as the baseline. The statistical indicator is the download percentages of inauthentic files. Fig. 3 shows the results.

As shown in Fig. 3, in NTM, the percentage of inauthentic files downloaded decreased gradually as the malicious peer uploaded more good files. It is obvious that since the more malicious peers upload good files, the less inauthentic good peers download inauthentic files. In cases that involve trust mechanism, the reputation scores of malicious peers increase as they provide more good files, thus resulting in malicious peers with a higher probability of being selected. This in turn provides them with more opportunities to upload inauthentic files. But uploading too many good files involves much cost for malicious peers, which is unfavourable from the perspective of the attackers. When they provide more good files, the majority of total files in the network are good. Therefore, the inauthentic downloads will decrease. Both of HTM and PRM accord with the above analysis, while our THM performs significantly better than PRM when the percentage of good files malicious peers provide is smaller than 0.6. This is mainly due to the fact that HTM improves the accuracy of the judgement with respect to the direct trust over file providers. Requesters are more likely to download from good peers who have relatively high direct trust relationships with them.

Next, we focus on three specific cases in which the probability of uploading good files by malicious peers are 0, 0.4, and 0.9. We give three group of results showing the distribution of uploads at every peer in HTM and PRM, respectively (numbers 1-20 stand for malicious peers, numbers 21-23 for pre-trusted peers, and the rest are the good peers).

From Fig. 4, we can see that the overall uploads of malicious peers are less than that of PRM. According to the second phase of our HTM, even if a malicious peer respond to a request, the requester will not easily select it to download a file. In contrast, there is a great probability that the requester continues to choose from other response peers, which can reduce the probability of downloading files from malicious peers. In this way, the proposed HTM reduces the pollution of network traffic by reducing the transmit of inauthentic files. In other words, our mechanism identifies malicious peers more effectively in this case.

The results in Fig. 5 show that the average number of uploads by malicious peers in PRM and HTM are around 520 and 200, respectively, which means HTM can effectively reduce camouflage attack. As for the reason, we give the qualitative analysis here. Since malicious peers upload many good files to the network, the reputation scores of them are greater than 0. Therefore, they are selected as servers easily in PRM, but the requesters do not know they are malicious peers. Once they are selected, they will upload a inauthentic file with a high probability, which is harmful

to the network. While in our HTM, a selected peer may be discarded due to the poor direct trust on it. This case also shows HTM can isolate the malicious peers better than PRM.

Fig. 6 shows the upload numbers caused by malicious peers both in PRM and HTM are very high and have almost the same total upload amount. This is mainly because of the fairly high probability of the malicious peer providing good files. In this case, the two-phase reference in our HTM does not have a significant influence. However, in this case, since the malicious peers always upload a lot of good files, we have reasons to regard them as good peers and there is no need to isolate them again.

4. Conclusion

In this paper we propose a hybrid trust mechanism (HTM) for P2P file sharing networks. HTM is a two-phase method to consider both of the direct trust and the global reputation of a peer. After selecting a potential file provider according to the roulette wheel selection algorithm, the file requester further considers the direct trust of the opponent, which can effectively restrain two malicious attacks in the system. Because of the increased precision of evaluating peers, HTM can increase the reliability and efficiency of P2P applications to a certain degree. The simulation results and our analysis show that the proposed HTM can reduce the downloading rate of inauthentic files effectively, thereby reducing the propagation of inauthentic files, and increasing the availability of the system.

Acknowledgements

The paper is supported by Nature Science Foundation of China under grant numbers: 61272173 and 61572095. And this paper is supported by National Undergraduate Training Programs for Innovation and Entrepreneurship under grant number: 2016101410330.

References

1. S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, “The eigentrust algorithm for reputation management in p2p networks,” in *Proceedings of the 12th International Conference on World Wide Web*, 2003.
2. L. Xiong, and L. Liu, “Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843-857, 2004.
3. U. E. Tahta, S. Sen, and A. B. Can, “GenTrust: A genetic trust management model for peer-to-peer systems,” *Applied Soft Computing,* vol. 34, pp. 693-704, 2015.
4. A. Das, and M. M. Islam, “SecuredTrust: a dynamic trust computation model for secured communication in multiagent systems,” *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 2, pp. 261-274, 2012.
5. J. Feng *et al.*, “RepHi: A novel attack against P2P reputation systems,” in *Proceedings of INFOCOM WKSHPS*, 2011.
6. A. Jøsang, R. Ismail, and C. Boyd, “A survey of trust and reputation systems for online service provision,” *Decision support systems*, vol. 43, no. 2, pp. 618-644, 2007.
7. X. Fan *et al.*, “EigenTrustp++: Attack resilient trust management,” in *Proceedings of CollaborateCom*, 2012.
8. T. Beth, M. Borcherding, and B. Klein, “Valuation of trust in open networks,” in *Computer Security - ESORICS 94*, Springer Berlin, Heidelberg, 1994, pp 1-18.
9. A. Lipowski, and D. Lipowska, “Roulette-wheel selection via stochastic acceptance,” *Physica A: Statistical Mechanics and its Applications,* vol. 391, no. 6, pp. 2193-2196, 2012.
10. QTM: P2P Trust Simulator: https://rtg.cis.upenn.edu/qtm/p2psim.php3.