# A Collusion-Resistant Trust Management Scheme for Distributed P2P Network

**Byong-lae Ha and Gi-hwan Cho**

**Abstract** Current trust management employed by the existing peer-to-peer networks is faced with various threats from malicious nodes. If some nodes are contaminated, the network quality is getting to be down. So it eventually makes worse the confidence of among the users. This paper deals with a trust management to protect the attack of the malicious nodes in the distributed P2P network. Especially, it aims to keep safe against the collusive attack. We try to improve the accuracy of node trust to effectively protect this attack.

**Keywords** Trust · Trust management · Collusive cheating · P2P

## 1 Introduction

Peer-to-Peer (P2P) network is known as a useful means with its extensibility feature. In P2P network, the peers can freely join and leave the system and the group membership is very dynamic. Due to its openness and lack of validation, a P2P system is vulnerable to such a kind of attack where some peers maliciously poison the system with corrupted data or harmful services [1]. For example, over 50 % of audio/video files in the Kazza are polluted, and in other P2P file sharing networks are very vulnerable to the worm virus, named VBS. Gnutella [2]. In such

B. Ha
Division of Electronics and Information, Chonbuk University, Jeonju, South Korea
e-mail: blha@jbnu.ac.kr

G. Cho (✉)
Division of Computer Engineering, Chonbuk University, Jeonju, South Korea
e-mail: ghcho@jbnu.ac.kr

Y.-H. Han et al. (eds.), *Ubiquitous Information Technologies and Applications*,
Lecture Notes in Electrical Engineering 214, DOI: 10.1007/978-94-007-5857-5_15,

cases, the user could not trust a resource received from P2P network. Even, the network reliability will be getting to be down. To defense these malicious behaviors, it is therefore very important to ensure the authenticity of shared resources. A set of trust management scheme were introduced to the P2P systems as a solution to promote a healthy collaboration relationship among participants.

The trust management basically evaluates the reliability of users based on how trustily a user acted in past. The evaluation results share other users. So it provides the user with reliability in the whole network. However, the trust management includes the various security threats. If a malicious user passes a dishonest opinion to other users, then it indirectly effects on the reliability of other users. Especially, a collusive cheating which attempts by a group of attackers will be very serious. Most existing researches were focused on the trust evaluation which is strong against only the attacks from a malicious user.

To combat with bad mouthing and collusive cheating, this paper exploits a trust management method. Particularly, we focus on a design of robust and efficient trust management in P2P networks. We construct a mathematic model of referral using credibility, and then adopt it to aggregate the referrals. The proposed management utilized a time decay function to reflect much the recent trust than that of the past. It utilizes credibility as well as similarity among the users. Finally, we discuss such problems on collusion and behaviors of malicious peers and also address the solutions to these problems.

## 2 Related Work

EigenTrust [3] is the most well-known algorithm to obtain trust for the nodes. In EigenTrust, a unique global trust value is assigned to each peer. The authors proposed a distributed iterative algorithm to calculate and update a global trust value at each node. The trust value is then used by the node to isolate malicious users and reward the peers with good reputation. When it selects a transaction counterpart, this utilizes trust value of peers to protect to take part in a malicious peer.

SFTrust [4] distinguishes the trust value for providing services from providing feedbacks. It also designed and implemented a framework to store, compute and update trust values. In short, this type of approach trust evaluation is mainly derived from the direct transactions without factoring in the quality of the evaluation, the quantity of the transactions and the time of the transactions. Xiong et al. [5] provides the distributed scheme PeerTrust that covers multiple trust factors, such as recommendation, the transaction number of the provider, the credibility of the feedback sources, transaction context, and community context. It makes use of the similarity to achieve the trust value. The feedback from those peers with higher credibility would be weighted more than those with lower.

Wang et al. [6] proposed another social-network based reputation ranking algorithm. It is capable of inferring reputation ranks more accurately when the

system is under front-peers attack. $R^2$Trust [7] proposes a robust and efficient reputation mechanism in P2P systems and also studies possible attacks to reputation mechanisms in P2P systems.

## 3 The Proposed Scheme

In this section, we will present a novel trust management that is strong against the collusive cheating. We are interested in applying the proposed method in file-sharing type of service over the distributed P2P network.

### *3.1 Overview*

To understand the proposed trust management scheme, we make a picture of a whole process. First, a consumer searches all available suppliers for the product. For each candidate, the trust value is calculated from direct trust value and indirect trust value. Direct trust value is obtained from the consumers' experience directly with the provider. Indirect trust value is formalized by the feedback of other consumers. Then, the node with the highest trust value is selected to carry out the transaction. Last, the customer files the feedback.

Let $T_{ij}$ denote the trust value from node $i$ to node $j$, the trust assessment can be formed as follows.

$$T_{ij} = \alpha \times DT_{ij} + (1 - \alpha) \times IdT_{ij} \tag{1}$$

where, $DT_{ij}$ is the direct trust being evaluated for the node $j$ based on the experience of node $i$. $IdT_{ij}$ means the indirect trust obtained from the adjacent nodes. $\alpha$ is the confidence factor where $\alpha$ means how the node $i$ can be convinced about the direct trust value by itself. If the node $i$ knows enough about the node $j$, the portion occupied by the direct trust would be increased by using the confidence factor. That has $0 < \alpha \leq 1$ range.

### *3.2 Direct Trust*

When the node $i$ directly communicates with the node $j$, the node can calculate the direct trust for node $j$ by its own experiences. Let the node $i$ has been communicated $k$th times with the node $j$, then the satisfaction for each communication is depicted as the Eq. (2). By using each of the satisfactions, the direct trust for the node $j$ can be formed like the Eq. (3).

$$R_{ij}^{k} = If\ satisfactory\ 1,\ else\ 0 \tag{2}$$

$$DT_{ij} = \sum_{k=1}^{n} f(x) \times R_{ij}^{k} \Big/ \sum_{k=1}^{n} f(x) \tag{3}$$

The satisfaction value of each communication is obtained by mapping the time decay function *f(x)* into the function $f(x) = \lambda^{n-k}$. Here, $n$ means the number of communications, and it has the range as $0.5 < \lambda < 1$ and $1 \leq k$. By making use of the time decay function, it is possible to effectively maintain the direct trust which is more reflected the recent communication results than that of the past communication results.

### 3.3 Indirect Trust

In the distributed P2P environment, it is impossible to calculate the direct trust for all nodes participated in network. In addition, there exists a possibility that the direct trust is forged by the malicious users. Therefore, it is imperative to use the other node‘s opinions.

Some papers [4, 5] suggested an idea to calculate the indirect trust by using a similarity. But the algorithms consider all similarity value with the same weight. As a result, the similarity computed from a common set of 50 nodes has no difference from that from a common set of only 5 nodes [2].

Differently with the previous works, we tried to enhance the indirect trust based on the credibility of the node providing an indirect trust and the similarity between two nodes. Let the node $i$ has been communicated $k$th times with the node $m$. The indirect trust can be depicted as the following expression.

$$IdT_{ij} = \sum_{k=1}^{n} CR_{im} \times DT_{mj}^{New} \Big/ \sum_{k=1}^{n} CR_{im} \tag{4}$$

$DT_{mj}^{New}$ is calculated by reflecting the communication frequency with the node $j$. This can be formulated as the Eq. (5). Thus, $n$ indicates the communication times between the node $m$ and the node $j$. $\beta$ is a scaling factor to keep the direct trust, and it has the range as $0.5 < \beta < 1$.

$$DT_{mj}^{New} = DT_{mj} \times \beta^{1/n} \tag{5}$$

$CR_{im}$ stands for a credibility evaluated by the node $i$ for the node $m$. The credibility can be obtained with the average opinion of the adjacent nodes. Let C*(j)* is a set of the nodes providing the indirect trust to the node $i$ for the node $j$ and |C*(j)*| is the number of nodes in the set. The relative difference is as expression (6).

$$Diff_{im} = \sum_{m \in C(j)} |IdT_{ij} - DT_{mj}^{New}| \Big/ |C(j)| \tag{6}$$

Along with the relative difference which evaluated by the node $m$ for the node $j$, the standard deviation of the nodes belonged to $S(j)$ is defined by $STD_j$. The average opinion value called $RTD_{im}$, between $i$ and $m$, can be obtained as follows Eq. (7).

$$RTD_{im} = Diff_{im}/STD_j \tag{7}$$

For the $RTD_{im}$, the node $i$ determines a baseline for the credibility of the node $m$. If $RTD$ is smaller than 1 or equal to 1, an incentive is given to the nodes in order to increase their credibility because they have similar value with the whole value. However, if $RTD$ is over than 1, a penalty is given to the nodes in order to decrease their credibility since the delivered values cannot be believed. The node $i$ can find the difference for the node $j$.

And also, the credibility can be obtained from the similarity. If the node $m$ provides an indirect trust, it determines how much the indirect trust is applied based on the credibility. This credibility can be calculated by making use of the similarity. When the node $i$ uses the indirect trust offered from the other nodes, the similarity stands how much the offered trust values are similar. The similarity utilizes the Pearson Correlation Coefficient [8]. Let $C_{im}$ is a set of nodes which the node $i$ and the node $m$ has been communicated. The similarity can be formed as the expression (8).

$$Sim_{im} = \frac{\sum_{j \in C_{im}} (DT_{ij} - \overline{DT_i})(DT_{mj} - \overline{DT_m})}{\sqrt{\sum_{j \in C_{im}} (DT_{ij} - \overline{DT_i})^2 \sum_{j \in C_{im}} (DT_{mj} - \overline{DT_m})^2}} \tag{8}$$

where, $\overline{DT_i}$ indicates an average value of the direct trust which holds by the node $i$. The result of $Sim_{im}$ has the range of $[-1, 1]$. In order to take the boundary of the similarity through [0, 1] range, we make the function $f(y)$ which mapped into the $f(y) = (y + 1)/2$. The similarity is utilized to measure the credibility about the node $m$.

Generally, a collusive attack tries to forming the group which consists of a set of malicious nodes. The group usually brings a bad mouthing attack which gives a high value inside the group while gives a low value to the node outside a group. The nodes taken part in this attack has different the indirect trust that of other nodes. Therefore, it is enough to be protected if the node credibility is controlled according to the value of $RTD$. The credibility can be obtained with utilizing $RTD_{im}$ and $f(y)$ value mapped by the $Sim_{im}$. $CR_{im}$ is defined by Eq. (9).

$$
\begin{aligned}
&CR_{im}^{k+1} = CR_{im}^{k} + f(y)(1 - CR_{im}^{k})(1 - RTD_{im}) \\
&0 \le RTD_{im} \le 1,\ 0.5 < f(y) \le 0.99,\ k > 1 \\
&CR_{im}^{k+1} = CR_{im}^{k} - {}^{1}/_{f(y)}(CR_{im}^{k})(1 - {}^{1}/_{RTD_{im}}) \\
&0 \le RTD_{im} \le 1,\ 0.1 < f(y) \le 0.5,\ k > 1 \\
&CR_{im}^{k+1} = 0.1,\ CR_{im}^{k} \le 0.1,\ k > 1 \\
&CR_{im}^{0} = 0.5,\ k = 0
\end{aligned}
\tag{9}
$$

where, $k$ means the nodes belonged to $S(j)$ provide $k$th indirect trust, then a node continuously revises their credibility based on the indirect trust values. In case of attempting a collusive attack, the value of the group inside and outside will be different. This difference can be effectually found out because of reflecting the value of many nodes through the indirect trust enough. Therefore, the trust management can effectively handle with these attacks, particularly, it can be strong against the collusive cheating attack.

## 4 Experiment Evaluation

In order to evaluate our proposed model, the proposed scheme has been compared with EigenTrust [3] and one without any trust method. Our evaluation has been conducted whether the pre-trusted nodes exist or not in the EigenTrust. Based on the file sharing, the experiment has been done for the attack scenarios. According to given attack scenarios, a user behavior was defined for the evaluation. For this purpose, by using QTM simulator [9] which is the trust management evaluation model, the user behavior has been defined and eventually used to evaluate the proposed scheme.

### *4.1 Experiment Environment*

QTM simulator defines the user behavior model by two parameters. That is, the clean-up means the probability that a user removes a dead file from the library, and the honesty is the probability that a user gives an honest opinion to the others. In the experiment, we define two kinds of nodes. The good users may always arrange the library and provide the valid files to the others. In this meaning, the clean-up can show a quality of service of a user. According to attacks, malicious nodes can be divided into two categories. The simulator makes use of the types of user, as shown in Table 1. The experiment makes use of the parameters as shown in the Table 2.

**Table 1** The user model initial parameter

| User type | Cleanup (%) | Honesty (%) | Source |
|---|---|---|---|
| Good | 90–100 | 100 | Best |
| Purely malicious | 0–10 | 0 | Worst |
| Malicious provider | 0–10 | 100 | Random |

**Table 2** Simulation parameter

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| Number of users | 100 | $\alpha$ | 0.6 |
| Number of transactions | 200 | $\lambda$ | 0.5 |
| Number of files | 10000 | $\beta$ | 0.8 |
| Zipf of coefficient | 0.4 | Pre-trust(EigenTrust) | 0, 2 |

The file is distributed to the users, and the communication is taken place between two users randomly chosen. We adapt the validity measurement proposed in [10] to evaluate the experimental results. This can be described as follow.

$$Success\,Rate\,of\,Transmission = \frac{\#\,of\,valid\,file\,received\,good\,users}{\#\,of\,transactions\,attempted\,by\,good\,users} \quad (10)$$

## 4.2 Evaluation

For all experimental results, x-axis shows the percentage of the attackers and y-axis means the evaluation metric. If the metric is close to 1, the success rate is higher.

As a first evaluation scenario, the case of that the malicious providers are given in the network as defined in the Table 1 is measured. As shown in Fig. 1a, even though the number of malicious nodes is increased, the evaluation results are nearly the same between the proposed one and EigenTrust. Surely, none of trust management means shows very low success rate. These results come from that two methods can distinguish the malicious users, because even if the malicious providers provide the malicious files, they provide always an honest opinion to the users.

The second scenario is measured in the case of that the purely malicious users are given in the network as defined in the Table 1. Because purely malicious users may submit a bad opinion, the evaluation result is shown in Fig. 1b. As shown in the result, the success rate of the proposed scheme is relatively higher than the other, which is differently that of the first scenario.

As a third evaluation scenario, we tried to measure the case that there are purely malicious users and the collusive cheating attacks which the malicious users form a small group to attack cooperatively. This scenario causes a serious threat in the trust management. As shown in Fig. 2, if the pre-trusted nodes are not given,

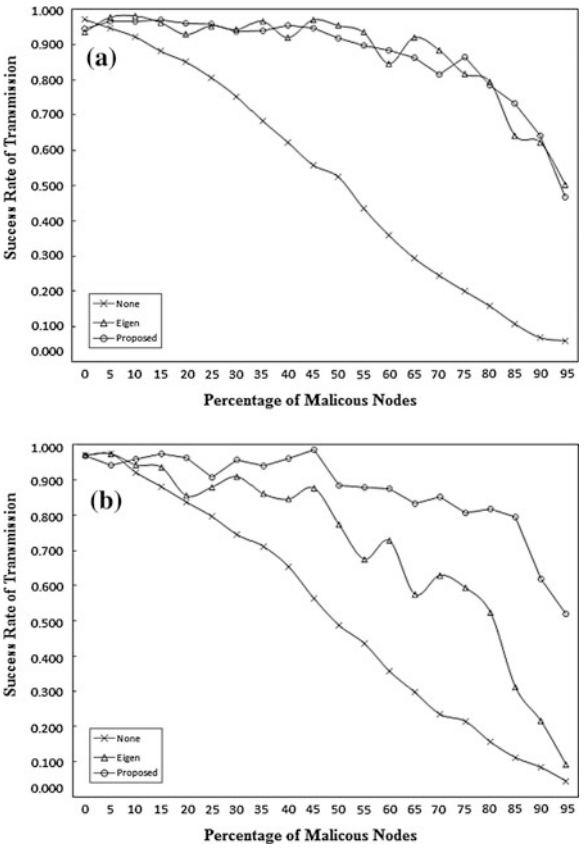


**Fig. 1** Success rate with varying (**a**) the malicious providers, **b** the purely malicious providers

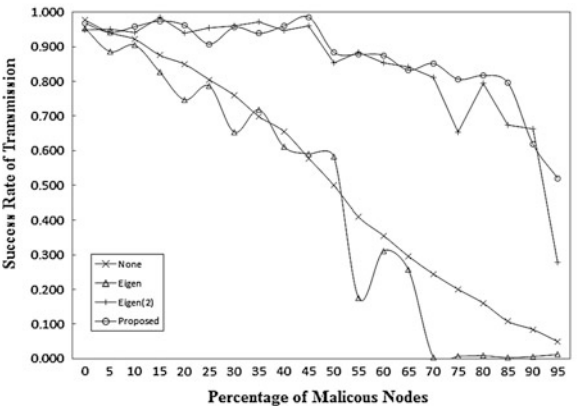


**Fig. 2** Success rate with the purely malicious providers and the collusive cheating attacks, with two pre-trusted nodes

EigenTrust is worse than the case which doesn't use any trust method. This comes from that EigenTrust overrides the real opinion with the bad opinion from malicious users. In differently, let see the case that two pre-trusted nodes are given in EigenTrust. With giving two pre-trusted nodes, the success rate is getting to be higher than that of no pre-trusted node. So, it means that the pre-trust node influences the effect on trust mechanism. Consequently, the proposed scheme shows higher success rate than EigenTrust with the pre-trust nodes. The proposed scheme can manage different kinds of malicious scenarios, especially the collusive attack, without assuming pre-trusted nodes.

## 5 Conclusion

In this paper, we proposed a trust evaluation scheme for effectually managing with the difference attacks in distributed P2P network. Particularly, the proposed method is strong against collusive cheating. It makes use of the time decay function in order to more reflect the recent reliability. In addition, the similarity of the user's assessments is utilized in order to reflect the credibility from the adjacent nodes. According to the experimental results, the proposed scheme effectively manages with the various attacks. In most cases of attack scenarios, its success rate is higher than that of EigenTrust.

Our research on trust management is going to go along several directions. First, this paper made use of two types of user model, so we are going to apply our scheme into the other user modes. Second, we are investigating different threat models of P2P networks and exploring mechanisms to make proposed trust management more robust against malicious behaviors.

## References

1. Liu, Y.H.: A two-hop solution to solving topology mismatch. IEEE Trans. Parallel Distrib. Syst. **19**(11), 1591–1600 (2008)
2. Wang, M., et al.: An Adaptive and Robust Reputation Mechanism for P2P Network, Professional on IEEE ICC, pp. 1–5 (2010)
3. Kamvar, S.D., et al., The EigenTrust Algorithm for Reputation Management in P2P Networks, Professional on WWW, pp. 640–651 (2003)
4. Zhang Y.C, et al., SFTrust: A Double Trust Metric based Trust Model in Unstructured P2P System, Professional on IPDPS, pp. 1–7 (2009)
5. Xiong, L., Liu, L.: PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities, IEEE transactions on knowledge and data. Engineering **16**(7), 843–857 (2004)
6. Wang, Y.F., Nakao, A.: Poisonedwater: An improved approach for accurate reputation ranking in P2P networks. Future Gen. Comp. Syst. **26**(8), 1317–1326 (2010)
7. Tian, C., Yangc, B.: R2Trust, a reputation and risk based trust management framework for large-scale, fully decentralized overlay networks. Future Gen. Comp. Syst. **27**(8), 1135–1141 (2011)

8. Pearson Correlation Coefficient. http://en.wikipedia.org/wiki/Pearson product moment correlation coefficient
9. QTM: Quantitative Trust Management. http://rtg.cis.upenn.edu/qtmpaper
10. West, AG., et al., An evaluation framework for reputation management systems, Book chapter for trust modeling and management in digital environments: From social concept to system development, pp. 282–308 (2009)