A Novel Trust Evaluation Model for Mobile P2P Networks

Xu Wu

Department of Computer Science, Xi'an Institute of Posts and Telecommunications,
Xi'an, 710121, China
xrdz2006@163.com

Abstract. Trust is one of key factors which influence the development of mobile P2P networks. However, current trust evaluation models are not applicable to mobile P2P networks properly due to some of its characteristics such as heterogeneous nature of the peers, limited-range as well as unreliability of wireless links. In the paper main factors that influence the trust in mobile P2P networks are identified. Based on the analyzed result, we propose a novel trust evaluation model, which helps the networks to operate normally with high probability. Our model does not employ cryptographic approaches or certification mechanisms, so it is light enough to fit well with mobile P2P networks without great overheads. In addition, it also effectively solves the trust problem when no prior interaction history exists, an issue that has not been addressed in many models. The proposed model is shown to be robust in the presence of attackers through simulation.

Keywords: model; trust; P2P networks; mobility

1 Introduction

A wireless mobile network is a cooperative network where each node requires to collaborate with each other to forward packets from a source to a destination. It is obvious that mobile P2P systems are different from the wired ones, since each object is able to move around and each has a limited radio range. Compared to a fixed peer-to-peer system, the mobile network environment is more distributed, with wider participants. Traditional security techniques cannot be applied directly to the mobile P2P networks due to the limitations of the wireless medium, expensive bandwidth, and the limitations of the mobile devices [1]. Therefore, computation-intensive techniques like public-key cryptography are not expected to be used in mobile P2P networks. Such a distinction is also beyond the ability of the conventional key management scheme because we cannot guarantee the secrecy of each peer's private key, In addition, mobile devices are susceptible to a variety of attacks for example, eavesdropping, denial of services, wormhole, and Sybil attack. Even a few malicious peers can easily spread deceitful data and make the networks be in confusion without great efforts. Therefore, some smart trust management schemes are needed to identify trustworthiness of mobile peers in order to distinguish between malicious peers and innocuous peers, and to strengthen reliable peers and weaken suspicious peers.

However mobile P2P networks pose some unique challenges, many trust evaluation models [2-6] are not applicable to mobile P2P networks properly. In the paper characteristics of mobile P2P networks are discussed, and main factors that influence the trust in mobile P2P networks are analyzed. Based on the analyzed result, we propose a novel trust evaluation model for resilient mobile P2P networks, which helps the networks to operate normally with high probability.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 presents the proposed trust model. Section 4 contains experimental study. Finally, we conclude this paper in Section 5.

2 Related Work

EigenTrust [2] model is designed for the reputation management of P2P systems. The global reputation of peer *i* is marked by the local trust values assigned to peer *i* by other peers, which reflects the experience of other peers with it. The core of the model is that a special normalization process where the trust rating held by a peer is normalized to have their sum equal to 1. The shortcoming is that the normalization could cause the loss of important trust information. Runfang Zhou and Kai Hwang [3] proposed a power-law distribution in user feedbacks and a computational model, i.e., PowerTrust, to leverage the power-law feedback characteristics. The paper used a trust overlay network (TON) to model the trust relationships among peers. PowerTrust can greatly improve global reputation accuracy and aggregation speed, but it can not avoid the communication overhead in global trust computation.

A new trust model based on recommendation evidence is proposed for P2P Networks by Tian Chun Qi et al [4]. The proposed model has advantages in modeling dynamic trust relationship and aggregating recommendation information. It filters out noisy recommendation information. Thomas Repantis and Vana Kalogeraki [5] propose a decentralized trust management middleware for ad-hoc, peer-to-peer networks, based on reputation. In the work, the middleware's protocols take advantage of the unstructured nature of the network to render malicious behavior, and the reputation information of each peer is stored in its neighbors and piggy-backed on its replies.

Recently, there are many approaches studying trust management of wireless networks. The significant efforts done so far are to manage trust with the help of Certificate Authority (CA) or Key Distribution Center (KDC). A CA/KDC is responsible for setting up the foremost trust relationships among all the nodes by distributing keys or certificates [6]. However, this strategy suffers from difficulty on collecting t certificates efficiently. In the distributed CA scheme [7], Kong et al. mentioned that the trust between a to-be-member node and t member nodes in its neighborhood can be established by out-of-bound physical proofs, such as human perception or biometrics. However, we can find that this method is far from practical.

3 Trust Evaluation Model

Our trust model has two types of trust: direct trust and recommendation trust. Direct trust is the trust of a peer on another based on the direct interacting experience and is used to evaluate trustworthiness when a peer has enough interacting experience with another peer. On the other hand, recommendation trust is used when a peer has little interacting experience with another one. Recommendation trust is the trust of a peer on another one based on direct trust and other peers' recommendation. In the section we firstly introduce five trust factors which influence the trust in such a mobile environment. We then present the details about how to evaluate the trustworthiness of peers by these trust factors.

3.1 The Trust Factors

Communication of P2P application: This factor contains communication ratio information. When a peer finds a certain event, if its neighbor peers also find the same event and broadcast the results of event, communication ratio values for those neighbor peers go up. If they do not communicate, communication ratio values for those peers go down. This factor represents the level of selfishness and normality of a peer. If a peer does not participate in communication in the networks continuously for its battery saving or some other Roubles, its trust value will be degraded.

$$V_i = \frac{vs_i - vf_i}{vs_i + vf} \tag{1}$$

V_i: communication value of peer *i,* where $1 \le i \le k$.

vs_i: communicating success count of node *i*

vf_i: communicating failure count of node *i*

Communicating results: This factor represents the result information of finding malicious events. This factor consists of communicating data and communicating time for the events. The information of this factor is used to check a consistency of each mobile peer and to find malicious peers in the networks. The inconsistency check result affects the value of consistency factor, C_i. When peer *j* checks the inconsistency of peer *i's* communicating results, if the results are out of relatively standard bound of node *j ,* node *j* estimates the results to be inconsistent or deceitful data. Such an estimation for peer *i,* affects the value of the consistency factor, C_i .

$$R_i = < cd_i, ct_i > \tag{2}$$

R_i: communicating result value of peer *i,* where $1 \le i \le k$

cd_i: communicating data of peer *i*

ct_i: communicating time of peer *i*

Consistency: This factor represents a level of consistency of a peer. Based on this factor, we can identify malicious peers, and filter out their communicating data in the networks.

$$C_i = \frac{cc_i - ic_i}{cc_i + ic_i} \tag{3}$$

C_i : Consistency value of peer *i,* where $1 \le i \le k$

cc_i : consistent communicating count of peer *i*

ic_i : inconsistent communicating count of peer *i*

Power: This factor represents remained lifetime of a mobile peer. As we compute trust values in consideration of this factor, we can reduce additional processes which would be necessary to handle some power- managing policies. In addition, some peers which have high trust values are likely to process more jobs than the other peers which have low trust values. In that case, the higher trust value a peer has, the earlier the peer meets its end. According to the adoption of this power factor, we can prevent such a biased power exhaustion.

$$-1 \le P_i \le 1 \tag{4}$$

P_i : Power value of peer *i,* where $1 \le i \le k$

Size of interactions: Size has different meanings in different mobile P2P environments. For example, in a mobile P2P file sharing network, the size of interaction expresses the file size shared in each interaction. Size of interactions is an important factor that should be considered in the trust model. For peers without any interacting history, most previous trust models often define a default level of trust. But if it is set too low, it would make it more difficult for a peer to show trustworthiness through its actions. If it is set very high, there may be a need to limit the possibility for peers to "start over" by re-registration after misbehaving. In our trust model, the introduction of the size of interactions effectively solves the trust problem of peers without any interacting history.

3.2 The Computational Model

Consider the situation where peer *i* wants to interact with peer *j* in order to accomplish a certain task. There are two ways in which to calculate trust value: direct and recommendation.

Direct trust is denoted as $D(T_i(j), S)$, where $T_i(j)$ is the direct trust value that peer *i* calculates for peer *j*. S expresses peer *j*'s level of size of interaction which is granted by peer *i*. The level of size of interaction satisfies the following rules.

(1) The lowest level is given to a new peer that doesn't have any interaction history.

(2) A certain level is updated if the number of successful interactions reaches the predefined number in the level. The predefined number is decided by the peer itself. The lower the current level is, the more the number of successful interactions it needs.

(3) The predefined successful interaction number in a certain level is increased if interactions fail due to malicious activities.

Direct trust computation involves an assignment of weights to the trust factors. We define W_i as a weight which represents importance of a particular factor from 0, unimportant, to +1, most important. The weight is dynamic and dependent on the application. If $P_j \neq -1$, Direct trust value that peer *i* calculates for peer *j* is computed by the following equation:

$$T_i(j) = \frac{W_1 C_j + W_2 V_j + W_3 P_j}{\sum_{j=1}^{3} W_j} + pen(m)\frac{1}{1+e^{-n}} \tag{5}$$

where $0 < W_j \le 1$. In case of $P_j = -1$, we just assign -1 to $T_i(j)$ and exclude the node from the networks because it totally cannot work in the networks. Because each mobile peer uses histograms for the accumulative trust evaluation, some malicious or compromised peers that broadcast inconsistent or deceitful data continuously can be found and classified in this trust computing process. $pen(m)$ denotes the punishment function. $\frac{1}{1+e^{-n}}$ is the acceleration factor, where n denotes the number of fail. It can make trust value to drop fast when the interaction is failed.

When two peers have little interaction experience, other peers' recommendation is needed for trust establishment. Recommendation trust is the trust of a peer on another one based on direct trust and other peers' recommendation. Let we assume that peer *j* requests an interaction with peer *i* and the size of the interaction is Q. First, peer *i* computes peer *j*'s direct trust denoted as $D(T_i(j), S)$.

(1) If $Q \le S$ and $T_i(j)$ reaches a certain value (which is set by peer *i*), peer *i* considers peer *j* to be trustworthy. It will then decide to interact with peer *j*.

(2) If $Q \le S$ but $T_i(j)$ fails to reach a certain value, peer *i* chooses to join a group based on its interest. Then it checks its own group and location with GPS and floods a HELLO message which containing a packet <GroupID, Position> to announce itself to other peers by using Echo protocol [8], then requests requests all other members of the group to cast a vote for peer *j* from the perspective of trust in the level of Q. For any new peer without any interaction history, its trust value would be 0 and would be granted the lowest level of the size of interaction. Without requesting, it will be permitted to interact at the lowest level.

(3) If $Q \le S$ but $T_i(j)$ fails to reach a certain value, peer *i* immediately refuses to interact with peer *j*.

(4) If $Q \le S$ and $T_i(j)$ reaches a very high value, peer *i* chooses to join a group based on its interest and then requests requests all other members of the group to cast a vote for peer *j* from the perspective of trust in the level of Q.

Second, after the other peers receive the poll request message, they will decide whether to cast the vote based on the following formula. Let *e* denotes a voting peer, then

$$DT_e(j) = \sum_{m=1}^{N(j)} \left(\frac{W_1 C_j + W_2 V_j + W_3 P_j}{\sum_{j=1}^{3} W_j} + pen(m)\frac{1}{1+e^{-n}} \right) \tag{6}$$

where $DT_e(j)$ is the poll value of *e* in *j*. $N(j)$ denotes the total number of interactions *e* has conducted with *j* at level Q.

Lastly, peer *i* gathers up all poll information of peer *j* from the repliers and gets peer *j*'s recommendation trust by this equation:

$$T = \frac{\sum_{i=1}^{N(w)} R(w) \times p}{N(w)} \tag{7}$$

where $N(w)$ denotes the total number of votes and $R(w)$ denotes peer w's vote accuracy factor which is in the range of (0, 1). p is related to $DT_w(j)$ such that if $DT_w(j) > 0$, $p = 1$, else $p = 0$.

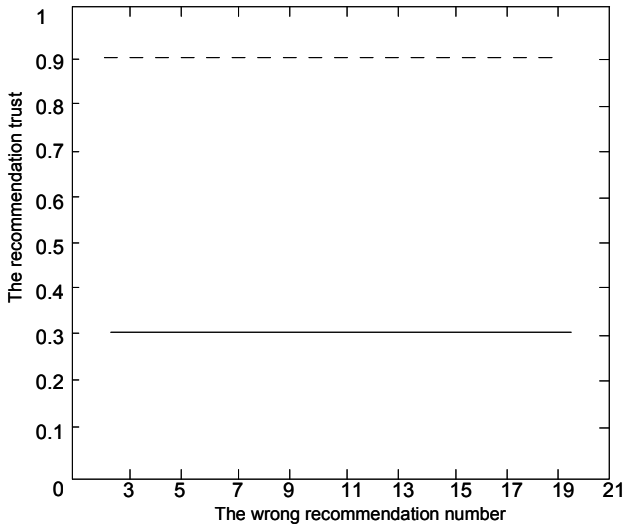
4 Experimental Study

Experiments have been carried out to study the effectiveness and the benefits of our proposed model. In a real environment, there may exist some vicious attacks including malicious recommendations or cheating in the accumulation of trust in small-size interactions. In addition, it should solve the trust problem when there is no interaction history or little trust value.

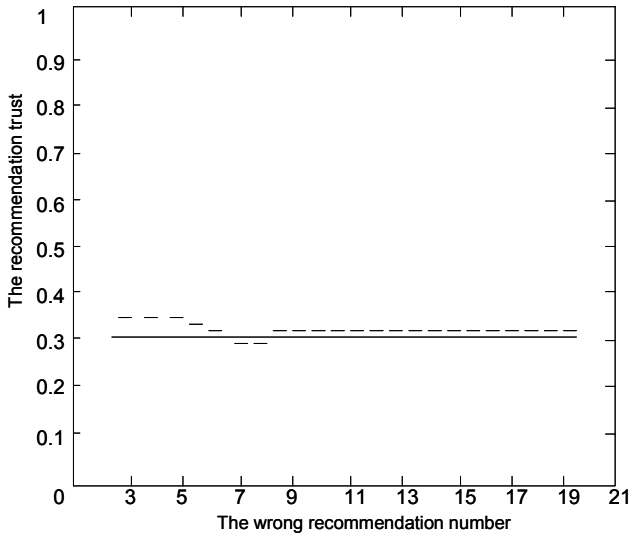
Table 1. Default parameters in simulation experiments

Number of Peers	300
Communicating range (m)	70
Simulation area (m^2)	500×500
Number of malicious Peers	0% - 70% of all peers
Risk attitude	averse, neutral, seeking
Communication protocol	802.11
Life time (s)	[50,100]
Maximum speed (m/s)	20

The simulation environment is set up as follows: we create 300 peers that will perform interacting in a mobile p2p resource sharing system. 300 mobile peers are uniformly distributed at the area whose size is $500m \times 500m$. Communicating range of a mobile device is *70m.* The simulated experiments were run on a dual-processor Dell server and the operation system installed on this machine is Linux with kernel 2.6.9. To make our simulation as close to the real mobile p2p systems where peers often go offline, we simulate the offline peers by assigning every peer a random lifetime (or Time-To-Live) within the step range [50, 100]. After reaching the lifetime, the peer will not respond to any service request, and won't be counted in the statistics either. After one more step, the peer comes alive again with a new life time randomly chosen from the range [50, 100]. In this analysis, we assume that all mobile peers have a same amount of battery power and participate in communication positively regardless of their roles. Each peer acts as both client and server to share its resources with other peers, and communicates with each other via IEEE 802.11. The default parameters in simulation experiments are showed in the table 1.



(a)



(b)

Fig. 1. Trust evaluation results

In the first experiment we evaluate the trust evaluation model in terms of its efficiency of excluding malicious recommendations in the network. We implement and simulate a file sharing system. The environments of the system are as follows. 300 mobile peers are uniformly distributed at the area whose size is $500m \times 500m$. Communicating range of a mobile device is 70m. In this analysis, we assume that all mobile peers have a same amount of battery power and participate in communication positively regardless of their roles. So, we consider only a consistency evaluation factor. Fig. 1 shows the simulation result in which the broken line denotes the recommendation trust value Tm that includes malicious peers' recommendations and

the solid line denotes the real recommendation trust value Tr that doesn't include any malicious recommendations. In this simulation, a same malicious recommendation event occurs every 10 seconds. As we can see Fig. 1 (a), normal recommendation trust value is 0.3, but a malicious recommendation result is 0.9 by few malicious peer which broadcasts three times as high as a normal recommendation result. This indicates the vulnerability of a system without a trust evaluation scheme. Fig. 1(b) shows the process of filtering inconsistent data of a malicious node which acts inconsistently after certain seconds with a proposed trust evaluation scheme. We can see that Tm fluctuates around Tr but the scale of the fluctuation is very small. The earlier the system detects a malicious node, the lower the malicious recommendations of it can affect the aggregated result.

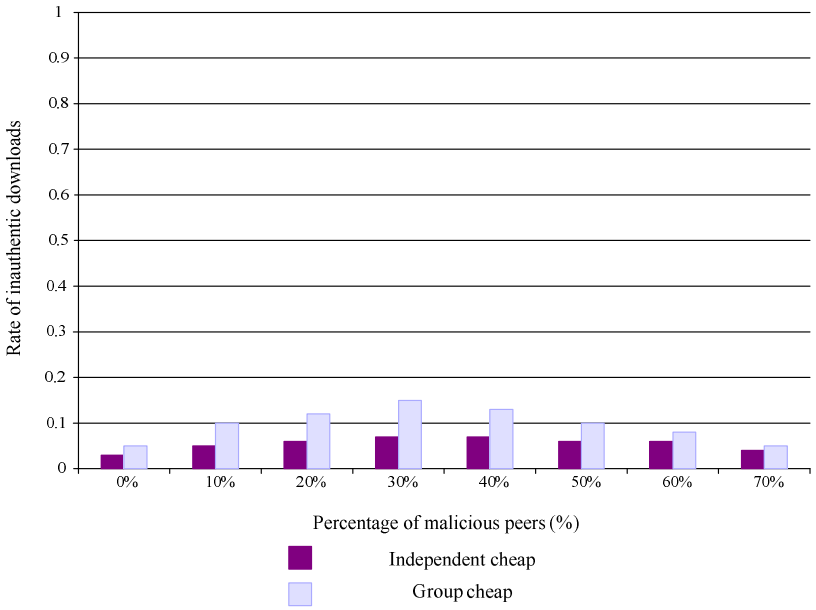


Fig. 2. Simulation results of peers under independent cheat and group cheat

In the second experiment, we assess the performance of our mechanism under two attack models: independent cheat and group cheat. Our experiment also points out that the trust model is also sensitive to the group cheat. In the experiment, we add a number of malicious peers to the network such that malicious peers make up between 0% and 70% of all peers in the network. Fig. 2.shows what is happening. In this figure, we compare the independent cheap and group cheat. Under independent cheat, the malicious peers firstly accumulate trust values through small interactions, gaining a relatively high trust. After trusted by most adjacent peers, the peer takes advantage of its high trust value to attack another peer, which means to always provide an inauthentic file to another peer when selected as download source. Group cheat is that there is a group in which the peer of the group provides an authentic file to each other and provides an inauthentic file to the peer outside the group. The rate of inauthentic downloads under independent cheat or group cheat increases at the beginning, then starts to drop when the number of malicious peers reaches to 30%-40% of all peers in the network. The reason is that the trust computing mechanism used in our experiments punishes this behavior by lower the trust values quickly. Since malicious

peers found by the mechanism will lose choice selected as download sources. As a result, the rate of inauthentic downloads will drop. However, due to the good rating coming from the cheating group, the rate of inauthentic downloads under group cheat drops more slowly than the one under independent peer. Yet one thing remains assured: the rate under group cheat is still dropping and will drop to 5%. Even if no malicious peers are present in the system, downloads are evaluated as inauthentic in 3%-5% of all cases – this accounts for mistakes users make when creating and sharing a file, e.g., by providing the wrong meta-data or creating and sharing an unreadable file.

5 Conclusion and Future Work

The realization of trust mechanism in mobile p2p networks is quite different due to some characteristics of mobile environment, which indicates the trust between participants can not be set up simply on the traditional trust mechanism. In the paper we proposed a novel trust evaluation model for mobile P2P networks. The main factors that influence the trust in mobile P2P networks are identified. Our model does not employ cryptographic approaches or certification mechanisms, so it is light enough to fit well with mobile P2P networks without great overheads. To the best of our knowledge, our approach is one of the incipient researches on trust evaluation model for mobile P2P networks that can detect malicious and compromised mobile peers. In addition, the proposed model effectively solves the trust problem of peers without any interacting history. We expect that our trust evaluation model can help to make resilient mobile P2P networks. In the near future, we would like to test our trust into more real mobile P2P systems and analyze the system performances.

References

1. Takeshita, K., Sasabe, M., Nakano, H.: Mobile P2P Networks for Highly Dynamic Environments. In: 6th IEEE International Conference on Pervasive Computing and Communications, Hong Kong (2008) 453-457
2. Kamvar, S.D., Schlosser, M.T., Molina, H.G.: The EigenTrust Algorithm for Reputation Management in P2P Networks. In: 12th International Conference on Word Wide Web, Budapest, Bulgaria (2003) 640-651
3. Zhou, R., Hwang, K.: PowerTrust: A Robust and Scalable Reputation System for Trusted P2P Computing. IEEE Transactions on Parallel and Distributed Systems, Vol.18, No.5 (2007)
4. Tian, C.Q., Zou, S.H., Wang, W.D., Cheng, S.D.: A New Trust Model Based on Recommendation Evidence for P2P Networks. Chinese Journal of Computers, Vol.31, No.2 (2008) 271-281
5. Thomas, R., Vana, K.: Decentralized trust management for ad-hoc peer-to-peer networks. In: 4th international workshop on Middleware for Pervasive and Ad-Hoc Computing, Melbourne, Australia (2006)
6. Zhou, L., Haas, Z.J.: Securing ad hoc networks. IEEE Special Issue on Network Security, Vol.13, No.6 (1999) 24–30

7. Kong, J., Zerfos, P., Luo, H., Lu, S., Zhang, L.: Providing robust and ubiquitous security support for mobile ad-hoc networks. In 9th International Conference on Network Protocol (2001) 25-260
8. Sastry, N., Shankar, U., Wagner, D.: Secure verification of Locotion Cloims. In 2nd ACM worhhop on Wireless security, New York (2003) 1-10