Secure and Distributed P2P Reputation Management

Jianli Hu[1]
Institute of Networks & Information Security, School of Computer,
National University of Defense Technology, Changsha, China
Email: lxman82@gmail.com

Quanyuan Wu, Bin Zhou
Institute of Networks & Information Security, School of Computer,
National University of Defense Technology, Changsha, China
Email: {quanyuan, binzhou}@nudt.edu.cn

***Abstract*—The open, sharing and anonymous nature of peer-to-peer (P2P) network has offered opportunities and threats for the development of distributed computing technology. One feasible way to minimize threats is to establish the reputation-based global trust model. Most of current trust mechanisms are not only unable to restrain effectively such malicious behavior as collusive attacks, but also take no consideration for the security of the trust management. As for these problems, this paper proposes a secure and effective reputation based distributed P2P global trust management model (DSRM), and presents its corresponding distributed storage mechanism of reputation information, and security protection protocol. Theoretical analysis and simulation experiments show that, DSRM has advantages in combating various malicious behaviors such as ordinary malicious behaviors and collusions, and suppressing the sybil attackers and trust information tamper peers in transmission over the current global trust management models, and demonstrates more robustness and effectiveness.**

***Index Terms*—P2P, trust, reputation, security, storage**

I. INTRODUCTION

In recent years, pee-to-peer (P2P) computing has achieved its popularity in many distributed applications, including file-sharing, digital content delivery, and P2P Grid computing [1]. However, peer anonymity and autonomy make P2P networks quite vulnerable to attacks by selfish and malicious peers. Previous work [2-5] shows that we can utilize the trust theory in social networks to construct reputation-based trust models, to suppress effectively these malicious behaviors.

Most of the existing reputation-based trust models compute the trusted rank of one peer based on its transaction histories with others, and it is very likely that the peer with the highest trust value is looked on as the service provider. To a certain degree, this approach has some effects on the simple malicious behavior patterns, but shows little effect in dealing with the complex attacks and disturbance activities on reputation systems, such as collusions. Besides, most of current researches concentrate on the design and implementation of the trust system, and pay less attention to the security problem confronted by its reputation management. In fact, security of reputation management is the key element assuring the normal running of the trust management system (TMS), and is as important as any other element of the reputation management. Thus, it is necessary to discuss and analyze the security mechanism of the TMS in this paper.

With these research problems in mind, we propose a reputation based distributed P2P trust model integrated with the security mechanism for the reputation information management (DSRM), for P2P networks, and give the corresponding Terrace-based [2] distributed storage scheme and the security protocol for protecting the distributed reputation information of the TMS. Simulation experiments exhibit DSRM not only can restrain behaviors of the simple malicious peers and collusive peers, but also can counter many attacks concerning security to TMS, such as sybil attacks and reputation information tamper attacks in transmission, etc. The remaining parts of the paper are organized as follows: Section II reviews the related work. Section III formally introduces our trust model DSRM without security mechanism. Section IV provides its distributed storage mechanism and the security requirements of the reputation management. Section V describes the related security protocol for assuring the security of the reputation management. Section VI simulates and discusses DSRM. Finally, we conclude the paper.

II. LITERATURE REVIEW

We can classify the trust model into two categories, the

1. Corresponding author, Jianli Hu (Email: lxman82@gmail.com)

trust model which relies on the 3rd party and that independent of the 3rd party. The representative of the former is the PKI-based trust model [6]. In this system, one or several power peers take charge of a set of trusted peers, and these power peers can issue certificates to newly-joined trusted peers. The certificate is used as the warrant for consuming the network resource. However, this kind of system is usually center-dependent, which is not in accordance with the nature of P2P networks, and has the risk of single points of failure. The trust model independent of the 3rd party can be categorized into two kinds, including the micro-payment based model [7] and the social trust network based model. The former model takes the virtual money as the means of service exchanges, needing some monitoring system to trace each small transaction process, and is not feasible in actual engineering applications.

We can sort the model based on the social trust network into the local trust model and the global trust model. In the local trust model, a peer determines the trustworthiness of one peer by converging the feedbacks through retrieving some limited other neighbor peers, and its direct interaction experiences with this peer. However, this system gets feedbacks only through broadcasting locally, and these feedbacks are local and partial [2]. On the contrary, the global trust model gets the unique global reputation information of one peer through the iterative computation of satisfactory ratings between neighbor participants. Usually, the global trust value of an arbitrary peer is determined by two parts: the local reputation ratings from the peers, which have ever transacted with this peer, and these peers' global trust values. The reputation management in EigenTrust [4] and PeerTrust [5] largely pay attention to the efficient and reliable requirements of reputation storage and access, on the basis of DHT topology, but neglect the security demand for the reputation management.

Besides the studies for the reputation based trust model, some researchers have studied the security mechanism of the reputation information management related tightly with the trust model. Reference [8] provides an enhanced security transmission protocol for trust management. It uses random numbers to suppress the *replay attack*, and utilizes digital signature algorithm to assure the integrity and consistency of reputation information. However, its public key is transmitted in unsecure channel. What's more, its response packets are transmitted back by the reverse sequence, and any peer in its path can launch the *man-in-the-middle attack*.

In Reference [9], each peer only stores its direct interactive experience, used for other peers' lookup. Simultaneously, To prevent the information from being tampered with deliberately, it uses such security mechanism as encryption and digital signature scheme to consist a deletion and tamper-proof information chain, the head of which is decided by the peer itself. Additionally, Reference [10] puts forward its security message transmission protocol in *Web of Trust* based distributed reputation system OpenPrivacy.

Based on the analysis for the current trust management mechanisms, Reference [11] argues that it is a required feature for the anonymity to prevent the archive peers from being attacked by malicious the peers in trust management. Therefore, this literature proposes a reputation information management mechanism with anonymity nature TrustMe, based on public-key cryptography. Its basic idea is like this: it makes use of its own security infrastructure to assure the anonymity characteristic in accessing the reputation information, and protect the reputation information management mechanism, by introducing a boostrapping server (BS).

Based on the security mechanism in TrustMe, Reference [12] offers a DHT-based distributed trust management mechanism RepMan, in which a security assuring protocol based on BS and public-key encryption scheme is used to assure the efficiency in accessing trust information , and the security and reliability for the TMS. However, there are some weaknesses as follow:

(1) Like TrustMe, RepMan also takes BS as the trust management infrastructure, which is equivalent to the Certification Authority (CA). It is center-dependent, which can achieve less scalability, and has the risk of single points of failure, just like PKI.

(2) The encryption scheme and the identity authentication is too complicated, and it is hard to cope with the overhead tradeoff between the TMS and the security protocol, since it will increase greatly the extra running cost and overhead for the TMS. Thereby, we deem that it is more instructive in theoretic references than in real engineering applications.

With respect to the sybil attack problem, Reference [2] gives an authentication method $Cent_{IP\text{-}ID}$, which matches the peer's IP address with its identifier to find its real identity. This method assumes that peer u and peer v know the IP address of each other (supposed IP_u and IP_v) in advance, but the identity verification process is implemented between peer u and peer v's archive peer D_v (we will give the definition of archive peer in Section IV). However, in terms of the anonymity feature of Terrace storage mechanism, peer u has no idea of peer D_v' IP address IP_{Dv}. Moreover, as a normal user peer, peer D_v probably has not ever interacted with peer u, so it also may have no knowledge of peer u's IP address IP_u. Therefore, the true effect of $Cent_{IP\text{-}ID}$ in reality needs further verifications.

III. Reputation Based Trust Model

Firstly, the definitions of the satisfactory degree evaluation and the direct trust value are given, and then we define the global trust value (GTV).

Definition 1 Reputation ratings. After transacting with each other, one peer i (the service consumer) will submit its ratings of satisfactory degree to the other peer j (the service provider), which can be defined as the following map function $f(i,j)$:

$$f(i,j)=\begin{cases}1, & satisfactory\\ -1, & unsatisfactory\end{cases} \quad (1)$$

in which, each time peer i consumes the service from peer j normally, it may rate the transaction as positive (1), otherwise, as negative (-1).

Therefore, during the time fraction t (t is decided by the concrete application. For example, six months), the reputation ratings peer i puts to peer j can be defined as $s_{ij} = \sum f(i, j)$, which can be expressed by another formula:

$$S_{ij} = Sat_{ij} - Unsat_{ij} \tag{2}$$

where, Sat_{ij} and $Unsat_{ij}$ represent the numbers of satisfactory and unsatisfactory transactions peer i has had with peer j, respectively.

Definition 2 Direct trust value. In order to aggregate local trust values and describe the real local trust value more precisely, it is necessary to normalize them in some manner, ensuring that all values will be between 0 and 1. Thus, we can define the direct trust value peer i assigns to peer j as follows:

$$R_{ij} = \begin{cases} \dfrac{\max(S_{ij}, 0)}{\sum_j \max(S_{ij}, 0)}, & \sum_j \max(S_{ij}, 0) \neq 0 \\ \varepsilon, & \sum_j \max(S_{ij}, 0) = 0 \end{cases} \tag{3}$$

where, ε denotes the lowest trust threshold.

Definition 3 In the network N, the GTV of an arbitrary peer i (denoted T_i) is defined in (4).

$$T_i = \sum_{j \in K} R_{ji} * T_j \tag{4}$$

where, K denotes the peer set which consists of peers, who have ever interacted with peer i, and offered feedbacks to it.

Assuming the GTV vector is $T = [T_1, T_2, \cdots, T_n]^T$, then the matrix form of (4) is as follows:

$$T = R^T * T \tag{5}$$

in which, R is the direct trust value given in Definition 2. The iterative convergence feature of (5) determines whether we can get the computation results of the GTV vector T. In fact, we can utilize the *Jacobi* iterative approach to prove the fact that we can get the needed iterative results from (5). Here, we omit the proof details, and the interested readers can refer to related literatures.

IV. Reputation Information Management Security Requirements

As for the reputation information management, firstly, we provide the distributed storage mechanism of the reputation information. Secondly, we analyze the security requirements of the reputation information management, and finally we present a security protocol to protect the reputation information management.

A. Distributed Storage Mechanism of Reputation Information

Based on the Terrace topology proposed in Reference [2], we construct a DSRM-oriented distributed storage mechanism of the reputation information. Terrace, a DHT-based structured topology, can be used as the underlying infrastructure of the trust management, providing necessary trust assurances to the above structured or unstructured P2P applications. Terrace takes advantage of the approach of the uni-hash function, meaning a peer participates in this topology with a random logical address, which can assure the security of the anonymous storage of the reputation information.

In detail, as to DSRM, the relevant information for reputation computation of a peer is stored into its corresponding logical peer. Through a even hash function (for example, SHA-1) *HDT*, We can map the identifier of peer i to the logical address (supposed d) of a certain peer in Terrace tree, and the process is $d=HTD(ID_i)$. The peer in Terrace tree, whose address is d, is named archive peer. Each archive peer, whose logical address is d, at least includes such a data structure as Table I.

In Table I, ID_i denotes peer i's identifier; $ID_{j1}, \cdots, ID_{ji}$ denote the identifier sequence of feedback peers; $Sat_{j1}, \cdots, Sat_{jt}$ and $Usat_{j1}, \cdots, Unsat_{jt}$ represent the number sequences of satisfactory transactions and unsatisfactory transactions reported by the peers who have ever transacted with peer i in a certain time, respectively; $T_{j1}^{(k)}, \cdots, T_{jT}^{(k)}$ denote the current GTV sequence; $T_i^{(k+1)}$ denote the latest GTV of peer i calculated by peer d. KU_i denotes the public key of peer i.

B. Running Process and Security Requirements of Trust Management Model

The design of the security mechanism of the distributed trust management is tightly related with the running process of the corresponding TMS. Thus, before discussing the security requirements of the trust management of the TMS, we analyze the running process of the TMS when it is deployed into P2P networks, and its classic process is demonstrated as shown in Fig. 1.

At first, we show the notations used in this section: we use symbol i, j and k to represent the service requesting peer, the service response peer and the archive peer of the service response peer, respectively.

Step1. peer i searches the needed service via the specified inquiry mechanism (*a*), and receives the corresponding the response of the available service response peer j (*b*);

Step2. peer i sends the inquiry request of the trust value to peer j's archive peer k (*c*), and receive peer k's feedbacks (*d*);

Step3. In light of the service choosing policy (for example, choose the peer with the highest trust value as the service provider), peer i chooses peer j as the service provider, and send back the confirmation message (*e*). Thus, it begins to consume the service (for example, download some files) (*f*);

Step4. After the transaction, based on the satisfactory degree to the service, peer i submits the reputation ratings to peer k (*g*).

In the above process, the operations related to the reputation information storage, access and transmission include process *c*, *d* and *g*. In process *c*, peer i sends the inquiry request of peer j's GTV to peer k, in process *d*, process *d* returns the GTV of peer j to peer i, and in process *g*, peer i submits the reputation ratings to peer k. Therefore, the real operations about the reputation information only include process *d* and process *f*. Concretely, the security risks existing in the above global trust model are as follow:

(1) Impersonation peer problem

Peers in P2P networks are strange to each other. To reach the aim of successful transaction, one peer should

TABLE I. THE STRUCTURE OF DOCUMENTARY POINT OF PEER *i*

ID_i		KU_i	$T_i^{(k+1)}$
ID_{j1}	Sat_{k1i}	$Unsat_{k1i}$	$T_{j1}^{(k)}$
ID_{j2}	Sat_{k2i}	$Unsat_{k2i}$	$T_{j2}^{(k)}$
⋮	⋮	⋮	⋮
ID_{jt}	Sat_{kti}	$Unsat_{kti}$	$T_{jt}^{(k)}$

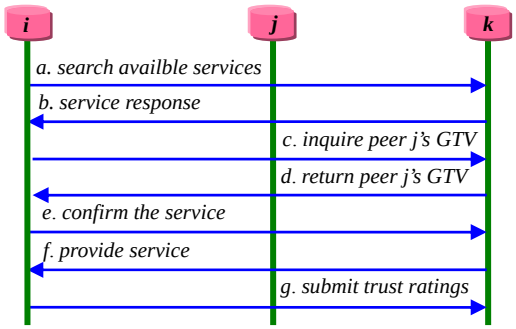

Figure 1. The running process of TMS

be able to recognize and validate the other's real identity, prevent impersonation peers and unauthorized peers accessing. In the trust management, the peer with lower GTV can impersonate the identity of the peer with higher GTV to endanger the TMS: (1) The sybil attacker imposes negative effect on some peers' GTV by deceiving these peers' archive peers, and (2) provides malicious services to others as the identity of the norm peer.

(2) Reputation information tamper problem in transmission

In terms of the above analysis, we know that, in the running process of the TMS, the reputation information (GTV or the reputation ratings) will transmit between peers, which needs the support of the underlying network infrastructure. The reputation information is possibly intercepted and tapered with by the malicious 3rd party without any security mechanism, which will destroy the integrity of the reputation information, and even compromise the availability and effectiveness of the TMS itself. Thus, we should integrate some security mechanism to protect the transmitted reputation information.

V. SECURITY PROTOCOL

The goal to design the security protocol is to provide security assurances for the reputation information access and transmission in the TMS. To get rid of the center-dependent feature of symmetric key scheme, this paper also introduces public-key cryptography as the security infrastructure. Assuming each peer in the P2P network has a public and private key pair (KU_i, KR_i), in which, KR_i is kept secretly by peer *i*, and KU_i is issued publicly over the network, and all other peers can get it by accessing peer *i*'s archive peer.

Before describing the security protocol of the trust management, we give the meanings of the symbols used in this section, as shown in Table II.

A. Sybil Attacker Preventing Mechanism

We capitalize on the public key scheme as the basis of the identity authentication. Since we have not introduced the trusted 3rd party as the verification infrastructure, the authentication mechanism becomes relatively complex. According to our above assumption that each peer's private key is stored into the peer itself, and the public key is stored into its archive peer. Therefore, we can take advantage of this assumption, to establish a challenge-response protocol, whose processes are illustrated in Fig. 2.

Step1. *u*: $KU_{Dv}(ID_u, R1_u) \xrightarrow{Terrace} D_v$. Peer *u* uses peer *v*'s archive peer D_v's public key (the archive peer in Terrace topology is also regarded as the ordinary user peer in the P2P network) to encrypt the information composed of its own identifier ID_u and a random number $R1_u$, and sends the resulting message to peer D_v. This step is ready for the real authentication process in the following steps.

Step2. D_v: $KU_u(K_s, R1_u, R1_v) \xrightarrow{IP} u$. On receiving the message, peer D_v decrypts the above message, and gets ID_u and $R1_u$. However, it cannot confirm the sending peer's real identity. Therefore, it encrypts the information composed of $R1_u$ sent by peer *u*, $R1_v$ generated by peer D_v itself, and a session key K_s with its own public key, and responds peer *u* with it.

When receiving this message, peer *u* decrypts it and obtains $R1_u$, $R1_v$ and K_s. Thus, peer D_v's identity is confirmed in that only peer D_v can decrypt the first message with his own private key, which is only possessed by peer D_v, and get $R1_u$. Additionally, since peer u sends $R1_u$ to only for a little while, it can know that the second message sent by peer D_v is a newly generated message, instead of a replay message. Thus, this mechanism can effectively suppress the *replay attack*. Conversely, if peer u is unable to gain $R1_u$ from the message sent by peer D_v, then we can confirm the fact that the identity of peer has been impersonated.

At the same time, it sends back a message composed of $R1_v$ encrypted with K_s to peer D_v to authenticate the real identity of peer D_v.

Step3. *u*: $K_s(R1_v) \xrightarrow{Terrace} D_v$. On receiving this message, peer D_v decrypts it and gets $R1_v$, so the identity of peer *u* is confirmed in that only peer u can decrypt the second encrypted message sent by peer D_v with its own private key, and get $R1_v$ and K_s.

TABLE II. THE DESCRIPTIONS OF SYMBOLS

Symbols	Descriptions
KR_i	peer i's private key
KU_i	peer i's public key
EP_k	to encrypt with public key k
DP_k	to decrypt with public key k
SIG_i	to sign with peer k's private key
H	to process with the hash function
TS_i	time stamp generated by peer i
ID_i	peer i's identifier
D_i	peer i's archive peer
IP_i	peer i's IP address
TInformation	reputation information (GTV or reputation ratings)
$R1_i(R2_i)$	the random number generated by peer i

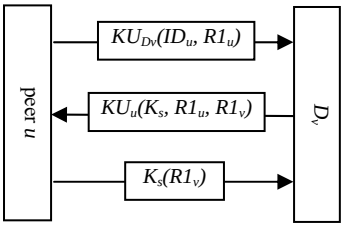

Figure 2. Public key based identity authentication protocol in the trust management

We can impose severe punishment mechanisms to the sybil peer: set its trust value to the lowest threshold, or drop all connections with it, and banish it out from the P2P network.

B. Tamper-Proof Mechanism in Reputation Information Transmission

As analyzed in Section IV, to obtain the GTV of the response peer, the request peer has to send inquiry request to the archive peer of the response peer, and the archive peer returns the needed information. After the transaction, the request peer needs to submit the reputation ratings to the archive peer. All the reputation information interacting processes need to be implemented via the reputation information transmission in the channels among peers. Furthermore, the reputation information is transmitted with the format of plain text, which is easily intercepted and tampered with by other malicious peers. In order to prevent this situation, we present a scheme composed of the public-key system based digital signature and the message digest algorithm, to assure the integrity of reputation information in transmission. We assume there are two peers, such as peer *i* and peer *j*, and they are the sender and the receiver, respectively. We can use RSA or EIGamal as the digital signature algorithm, and MD5 or SHA-1 as the message digest algorithm. In detail, this mechanism can be described as follow:

(1) *i:* $SIG_i(TS_i, H(TInformaiton))$*+ TInformation* $\xrightarrow{IP}$ *j*

Step1. Peer *i* copes with the trust information *TInformation* with SHA-1, and gets the corresponding message digest information *H(TInformaiton)* (*a*);

Step2. On obtaining the message digest information, peer *i* generates current time stamp TS_i. After that, it signs the information composed of the message digest information and TS_i, with its private key $SIG_i(TS_i, H(TInformaiton))$ (*b*);

Step3. Combining the information signed in Step2 and the trust information in the format of plain text to the mixed information $SIG_i(TS_i, H(TInformaiton))$*+ TInformation*, and sending it peer *j* (*c*);

(2) *j:* $DP_{KUi}(SIG_i(TS_i, H(TInformaiton)))$

Step4. After receiving the information sent in Step3, peer j separates the information, and gets the information $SIG_i(TS_i, H(TInformaiton))$ signed in Step2 and the plain trust information *TInformation*. After that, peer *j* obtains peer *i*'s public key by inquiring peer *i*'s archive peer D_i, decrypts the signed message digest $DP_{KUi}(SIG_i(TS_i, H(TInformaiton)))$, and get the primitive message digest *H(TInformaiton)* (*d*);

Step5. Peer *j* processes the plain trust information gained in Step4 with the same hash function as used in Step1, and gets the resulting message digest *H(TInformaiton)* (*e*);

Step6. Finally, peer *j* compares the message digest obtained in Step4 with that in Step5. If the result is unequal, peer *j* can conclude that the trust information has been tampered with or destroyed in transmission, and the transmitted information is useless. Thus, peer *j* can require peer *i* to re-submit the needed trust information. Otherwise, peer *j* can regard the received trust information as the useful information (*f*).

In the above process, the processes from Step1 to Step3 are executed in the sender *i* (*see* Fig. 3), while the processes from Step4 to Step6 are implemented in the receiver *j* (*see* Fig. 4). In addition, we can see that the above security mechanism has not implemented the confidential function, and the reason is that it is unnecessary to hide the content of the trust information in transmission, which is open to all peers in the P2P network. Moreover, the proposed security mechanism not only has the integrity verification function of the trust information, but also has the identity authentication and non-repudiation functions.

VI. SYSTEM PERFORMANCE ANALYSIS

We apply the file sharing application as the simulation case. The simulation setup is as follows: The community consists of 1000 peers. We allocate 10000 files into each peer with even random probability distribution, and the detailed simulation setting is shown in Table III. In simulation, assuming that all the files can be located successfully, that each file is possessed by at least one normal peer, and that the newly joined peer has a probability of 10% to be chosen as the service provider. Here, we simulate 100 query cycles, and each peer can execute transactions for 100 times.

To compare, we simulate EigenTrust trust model at the same time. In addition, we also simulate the P2P network without deploying any trust system, in which each peer chooses randomly the service provider to download the needed resource every time (denoted *NoTrust*). The evaluation standard is the successful transaction rate (STR), which is described as the percentage of the number of successful transactions with respect to the total transaction number. The index intuitively reflects the applying effect of the trust model. The hardware platform of simulation consists of CPU for AMD Athlon™ 64 X2 Dual 1.9GHZ, and the memory of 1GMB, and the simulation software is developed in Java.

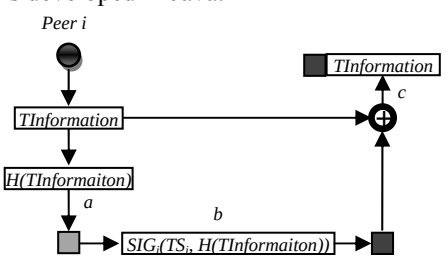

Figure 3. The signature process of peer *i*

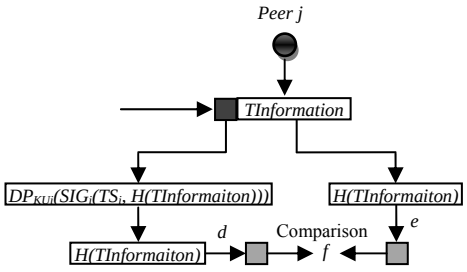

Figure 4. The verification process of peer *j*

TABLE III. SIMULATION PARAMETERS SETTINGS

N	# of the total number of peers in community	1000
N_f	# of the total number of files	10000
P_{res}	% of the probability in response to query requests	1
Nd	# of the degree of the normal peer	3
Md	# of the degree of the malicious peer	6
ε	# of the lowest trust threshold	0.05
TTL	# of the forwarding depth of query requests	4

A. Behavior Pattern Definition

In order to evaluate the effectiveness of DSRM to suppress the attacks of malicious peers, we design several malicious behavior patterns as follow:

- The simply malicious peer, being the basic type of malicious peer, only provides malicious uploading service, which is named SMS for short.
- The collusive malicious peers are organized into a group, which give exaggerated ratings and highly trusted services to members within the group, and denigrated ratings and malicious services to members outside the group. We named this type of peer as CM.

The sybil attacker impersonates the peer with higher GTV, denigrates this peer, and offer dishonest services to others, which can be represented as SA.

B. SMS Simulation and Discussion

This experiment is mainly used to test the effectiveness of DSRM confronted with different scale of SMS peers. For comparison, we also make simulations for EigenTrust and *NoTrust* under the same environments. As demonstrated in Fig. 5, as there is no malicious peer in the system, their initial STRs all can reach as high as 98%. With the percentage of SMS peers increasing, the curve for *NoTrust* drops fastest, and when its percentage gets to 50%, its STR drops only to about 20%. As to the other two models, the STR for DSRM decreases a little, while that for EigenTrust declines sharply. As the percentage of SMS peers goes up to 50%, the STR for EigenTrust is no more than 50%, but the same index for DSRM is about 70%. The results exhibit that it is effective for our DSRM to suppress malicious behaviors of SMS peers.

C. CM Simulation and Discussion

CM peers are familiar with each other in their group, and they may collaborate with each other to boost up their own ratings. Concretely, they may rate the peers in their collusion group very high and rate outsiders very low. This type of malicious peers will produce more threats to the trust model itself. From Fig. 6, we can see that CM peers can easily obtain a higher trust value with the increase of CM peers. Due to lack of effective punishment mechanisms for CM peers in EigenTrust, the STR for EigenTrust decreases greatly, while DSRM can effectively cope with the malicious behaviors of CM peers, and keep the STR in a higher level. As shown in Fig. 6, as the percentage of CM peers increase to 50%, the STR for DSRM still can reach as high as 68%. However, the counterpoint for EigenTrust is only 35%. The above simulation results prove that DSRM can show more effectiveness and robustness against the malicious behaviors of CM peers.

D. SA Simulation And Discussion

To test the effect of the security protocol in DSRM, we simulate DSRM without integrating any security mechanism (supposed GPTM) and EigenTrust simultaneously under the same experimental circumstances. When receiving the inquiry requests from other peers, the SA peer always disguises the peer with higher GTV to respond these requests, and tries to entice these peers to submit unsatisfactory ratings to the real peer, by providing unreliable services to these peers, resulting in lots of unsatisfactory ratings in the trust system. Based on the knowledge, we suppose SA peers always respond the inquiry requests with the frequency of 1.5-2.5 times higher than the norm trusted peers, and launch the sybil attacks. We observe varying tendency of STR with the scale of SA peers changing, and the simulation results are shown in Fig. 7.

As shown in Fig. 7, with the number of SA peers increasing, the STRs for GPTM and for EigenTrust both decrease to a certain extent. When the percentage of SA peers reaches 30%, the STRs for GPTM and for EigenTrust are 74% and 55%, respectively. However, when the percentage of SA peers reaches as high as 50%, the corresponding STRs are 66% and 43%, respectively. This is because no sybil attack preventing mechanisms are integrated into EigenTrust, leading to the obviously negative effect on the STR from SA peers. Moreover, there is similar situation existing in GPTM, which shows the fact that SA peers will significantly compromise DSRM, if no security mechanisms are deployed into it. However, as for DSRM, since we apply the security defense mechanism to SA peers, the sybil attack problem can be tackled perfectly. Thus, as demonstrated in Fig. 7, except the unconspicuous service failure due to the interior unreliable behaviors in DSRM, SA peers cannot do any harm to it.

E. Efficiency of the Security Protocol in DSRM

In order to observe that, when the security protocol is deployed into DSRM, how much it has influence the whole trust management system without any security mechanism, we assume 30% of peers in the P2P network are SMS peers. Under this condition, we do two simulation experiments: one is executed under the circumstances of DSRM, and the other is carried out under the

circumstances of GPTM. The simulation results are shown in Fig. 8.

Fig. 8 compares the varying tendency of STR over time, from which, we can see these two types of TMSs can effectively recognized SMS peers, though there is little differences for the recognizing effect. After around 280ms, GPTM can absolutely distinguish the "good peers" from the SMS peers, leading to the stable STR in Fig. 8, while DSRM needs more time (roughly 390ms) to achieve the same level. The reason is that once the security protocol integrated into DSRM is initiated, the running processes of the identity authentication mechanism and the reputation information tamper proof mechanism in transmission both need to consume some time. The time gap is about 110ms, which is a very little gap for the real engineering applications. The results illuminate that the time overhead of the security protocol proposed in this paper is very little. Therefore, this security protocol shows better efficiency, and is feasible to be applied to the real engineering environments.

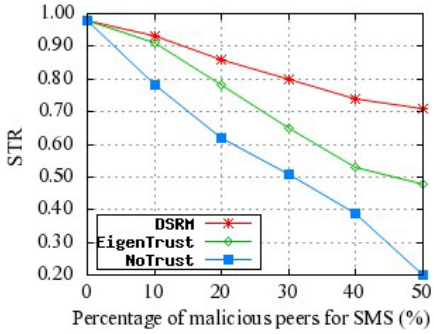

Figure 5. The varying tendency of STR with the percentage of SMS peers

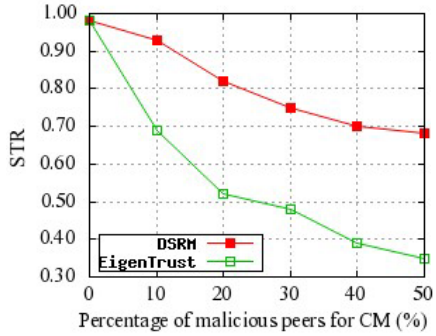

Figure 6. The varying tendency of STR with the percentage of CM peers

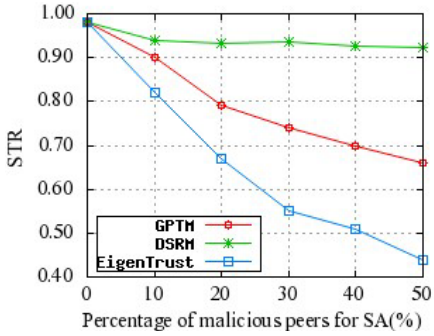

Figure 7. The varying tendency of STR with the percentage of SA peers

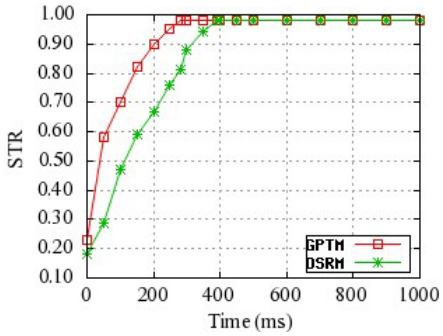

Figure 8. The varying tendency comparison of STR over time between DSRM and GPTM

VII. CONCLUSIONS

The authenticity of the reputation information is the basis of assuring the normal running of TMS. After analyzing the security risks existing in the current TMS, this paper proposes a reputation based secure and distributed trust management model, and gives the corresponding reputation information storage mechanism and security defense protocol. The security protocol is used to cope with the sybil attack problem and the reputation information tamper problem in transmission. Analysis and simulation results show, in contrast with the current global trust models, that the proposed model is more robust and effective on attacks from various malicious peers, including peers with malicious behaviors and peers with security threats, and shows more improvements in the security feature of the trust management. Additionally, the security protocol is feasible to be deployed into the real engineering applications associated with the trust management.

ACKNOWLEDGMENT

The authors acknowledge the useful comments from the anonymous reviewers. This research was supported by the National Grand Fundamental Research Program (973 Program) of China under Grand No. 2005CB321800, the National High Technology Research and Development Program (863 Program) of China under Grant No. 2007AA010301, the National Science Foundation for Distinguished Young Scholars of China under Grant No. 60625203, and the National Natural Science Foundation of China under Grant No. 69903011.

REFERENCES

[1] Zhang Q, Sun Y, Liu Z, Zhang X, Wen XZ. "Design of a distributed P2P-based grid content management architecture". In: Ilow J, ed. Proc. of *the 3rd Communication Networks and Services Research Conf*. New York: IEEE Press, 2005. 339−344.

[2] Dou Wen, Wang Huaimin, Jia Yan , et al. "A recommendation-based peer-to-peer trust model". *Journal of Software*, 2004, 15(4): 571-583 (in Chinese).

[3] Wang Y, Vassileva J. "Bayesian network trust model in peer-to-peer networks". In: Moro G, ed. *Proc. of the 2nd Int'l Workshop on Agents and Peer-to-Peer Computing*. Berlin: Springer-Verlag, 2004. 23−34.

[4] Kamwar S. D, Schlosser M. T, Hector Garcia-Molina.

"The EigenTrust algorithm for reputation management in P2P networks". In: *Proceedings of the 12th International Conference on World Wide Web*, Budapest, Hungary, 2003, 640-651.

[5] Xiong L, Liu L. "PeerTrust: Supporting reputation-based trust in peer-to-peer communities". *IEEE Transactions on Data and Knowledge Engineering*, Special Issue on Peer-to-Peer Based Data Management, 2004, 16(7) : 843-857.

[6] J.Altman,PKI Security for JXTA Overlay Networks. Sun Microsystem, Palo Alto, Tech Rept: TR-I2-03-06,2003

[7] Golle P, Leyton-Brown K, Mironov I. "Incentives for sharing in peer-to-peer networks". In: Wellman MP, Shoham Y, eds. *Proc. of the 3rd ACM Conf. on Electronic Commerce*. New York: ACM Press, 2001. 264-267.

[8] Siekab, Kshemkal YANIAD, Singhal M. "On the security of polling p rotocols in Peer-to-Peer systems". *Proceedings of the Fourth International Conference on Peer2to2Peer Computing (P2Pp04)*. Washington: IEEE Computer Society, 2004: 241-249.

[9] Dewan P, Dasgupta P. "Securing reputation data in peer-to-peer networks". *International Conference on Parallel and Distributed Computing and Systems (PDCS 2004)* [2007-04-11].

[10] Labalme F., Burton K., "Enhancing the Internet with Reputations". *An OpenPrivacy White Paper*, Mar. 2001.

[11] Singh A., Liu L. "TrustMe: Anonymous Management of Trust Relationships in Decentralized P2P Systems", in *Proceedings of the third IEEE International Conference on P2P Computing*, Linköping, Sweden. Sept, 2003. p142-149.

[12] Tang Yang-Bin, Wang Huai-Min, et al. "Clique-Based Reputation System in Self-Organizing Internet-Based Virtual Computing Environment". *Journal of Software*, 2007, 18(8): 1968-1986(in Chinese).

Jianli Hu was born in Wuhan, Hubei, China, in January, 9th, 1976. He received his B. E. degree in computer science in 1999, from the Mechanical Engineering College, Shijiazhuang, China. In 2003, he received his M. E. degree in military commend from Military Command Institute, Zhangjiakou, China, and in 2006, he received his Ph.D. degree in computer science from the Mechanical Engineering College, Shijiazhuang, China.

He has attended many project research, many of which were in part supported by the National Grand Fundamental Research Program (973 Program) of China and National High Technology Research and Development Program (863 Program) of China. He currently concentrates on P2P and trust research in the Computer School of the National University of Defense Technology, Changsha, Hunan, China, as a postdoctoral researcher. He has published many articles in domestic and overseas core journals or academic conferences, three of which are including: Jianli Hu et al. "FCTrust: a robust and efficient feedback credibility-based distributed trust model for p2p networks". In: Proceedings of *the 2008 International Symposium on Trusted Computing*. Zhangjiajie, China, November 18-21, 2008. Jianli Hu et al. "Distributed and effective reputation mechanism in p2p systems". In: Proceedings of *the 2008 International Conference on Computer Science and Software Engineering*. Wuhan, China, December 12-14, 2008. Jianli Hu et al. "RBTrust: a recommendation belief based distributed trust management model for p2p networks". In: Proceedings of *the 2008 International Workshop on Massive Network Storage Systems and Technologies*. Dalian, China, September, 25-17, 2008. His current research interests include mobile agent computing, P2P computing, grid computing, electronic commerce, and network security.

Dr. Hu is a member of CCF. He was awarded the Mechanical Engineering College Special Research Prize in Applied Science in 2006 as the highest standing graduate in the faculty of Applied Science.

Bin Zhou is an associate professor in the School of Computer of the National University of Defense Technology (NUDT), Changsha, Hunan, China. He received his BS, MS and Ph.D. degrees in computer science from NUDT, in 1994, in 1997, and in 2000, respectively. His interests are in distributed computing technology and data mining technology.

Quanyuan Wu, as a doctoral supervisor, is a professor in computer science in the School of Computer of the National University of Defense Technology. His research interests are in artificial intelligence, distributed computing, middleware application.