A Trust Evaluation Algorithm in P2P Network based on Trust Model

Zhenling Wang

Department of Computer Science and Technology, Dezhou University
Shandong Province, China 253023
wangzling@163.com
doi:10.4156/jdcta.vol5.issue8.31

Abstract
The lack of stable node identity is the central problem in peer-to-peer network security. Based on the general characteristics of trust between social networking and P2P network, a trust model and an approach of evaluation trust in P2P network are proposed. An algorithm for computing the trust values is put forward based on the relationship among hosts, nodes and pseudonyms. The analyses prove that the approach can ensure anonymity of host and defend P2P network against the Pseudo spoofing and ID stealth attacks.

Keywords: Trust Evaluation, P2P, Trust Model

1. Introduction

In P2P system, nodes are identified by pseudonym that communication nodes can not complete the authentication without stable identities [1]. The traditional access control technology can not be directly applied to the P2P environment. Eventually, P2P users face a variety of risk more serious than the tradition application, spam, computer virus, ID stealth, Pseudo spoofing [2].

In recent years, the most research for trust evaluation is based on reputation model [3]. Wang and Vassileva propose a trust model based on Bayesian Network for P2P system [4]. Its reputation is built on the basis of recommendations. Considering the reputation and risk, Adams brings forward a node centered access control system for trusted distributed system [5]. The system stimulates mutual cooperation and resource sharing in mobile collaboration environment. EigenTrust proposed by Kamvar is a global trust model, which computers the local trust value according the history transactions. Global trust value is calculated by iterative trust to achieve probability spread of trust for each node [6]. Based on EigenTrust, [7] solves the problem about the existence of global trust value.

These models evaluate the node trust in same network; do not cover how to evaluate the trust value when the node joins another P2P system. In real P2P system, each node may own many pseudonyms and could join different P2P system with different pseudonyms. Trust is context sensitive; a trust object may obtain different trust value in different circumstance.

The structure of node-centered P2P network and the real human-centered social network have a high degree of similarity. A person often plays in social circle in different roles (multiple identities), and in different social circle tends to be different credibility. Social networks constrain people everyday behavior through trust between people and the incentive mechanism, such trust relationship is built on interpersonal interaction has occurred, is an assessment about the trust of the social object depending the experience.

2. Design Considerations

The relationship among pseudonyms, nodes and hosts in the P2P environment is shown in Fig 1. N1 and N2 are different P2P networks. Host p joins N1 and interacts with node a, q through the pseudonym pl. Host q joins N1, N2 simultaneously, interacts with node a, b, p with different pseudonym q1, q3 and q3 in N1, interacts with node d through pseudonym q2 in N2, interacts with node c through pseudonym q2 and q3 in N2.
In Fig 1, node q represents different peer identified by different pseudonym. The credibility of the node is closely related to the pseudonym. The credibility of the host p could be measured by behaviors related to pseudonym p1, while the credibility of the host q relates with pseudonym q1, q2 and q3. Most trust evaluation mechanism [4-7] is essentially the trust assessment for pseudonym. For interactions between p and q, they calculate the credibility of q3 only from the perspective of p1, or get recommended information from other nodes interacted with q3 (e.g. b1), without comprehensive consideration the impact of behavior of q1, q2 on the credibility of host q. Anonymity of P2P system has led to the problem because p cannot get any information about relations among q1, q2 and q3 (all belong to host q).

Based on the above, we propose a distributed host-based trust assessment mechanism: Pseudonym trust is assessed only by the behavior of the corresponding pseudonym (e.g. q3); the node trust is assessed by the behavior of all the pseudonym used by a host in the same network (e.g. q1 and q3); host trust is assessed by the behavior of all the pseudonym of the host which may be used in different P2P system (e.g. q1, q2 and q3). For interaction between pi and q3, pseudonym pi as an evaluator calculates pseudonym trust, node trust and host of q in turn, eventually uses host trust of q as the credibility of the peer q3 to avoid the security risks brought by directly using q3’s pseudonym trust as credibility of the peer q3.

3. P2P TRUST MODEL BASED HOST

In this section, we show the approach to establish the stable identity with ring signature scheme [9] for the P2P host, and the comprehensive consideration of various factors on the credibility quantification. Let P is a set of nodes in a P2P network, H is a set of Trusted Platform built by TPM.

3.1. Stable identifier

Signer of DAA anonymous authentication uses pseudonyms as identities. Assumed a trusted computing platform 1 holds two pseudonyms in a P2P networks \(N^p_1 = \xi_1^p\) and \(N^p_2 = \xi_2^p\), fp is secret value associated with the TPM. \(\xi_1, \xi_2\) can be randomly selected by the Signer.

We put forward a stable identifier generation method. The approach has no restrictions on the choice of node pseudonym.

Let L = \(y_1, y_2, \ldots, y_n\) be public key list for an-node P2P system. The public and private key pair of node can be generated in the DAA Join phase, and be submitted to the Issuer to sign. Issuer ensures the public and private keys are submitted by legitimate TPM, and each node only submits one public private key pair. After verifying the key pair, Issuer publishes the public key list for ring signature. The complexity of ring signature is related to the size of the ring L. We assume that
each node independently chooses the size and member of the ring according their privacy protection policy.

Assumed P2P nodes p, q E P have completed the DAA protocol, node p (Signer) send data ring signature \( \sigma_M \) to q (Verifier) along with the data M. Ring signature \( \sigma_M \) has the following form:

\[
\sigma_M = (c_1, s_1, \ldots, s_n, y)
\]

Suppose p has two different pseudonym \( N^p_1 \) and \( N^p_2 \); and send data M1, M2 with different pseudonyms. Respectively, nodes q can receive signatures \( \sigma_{M_1} = (c_1, s_1, \ldots, s_n, y) \) and \( \sigma_{M_2} = (c', s'_1, \ldots, s'_n, y') \) signed by these pseudonyms. Firstly q verifies signatures, and then checks the link ability[12] between these signatures. If they are linked, this means that they are signed by the same TMP, otherwise, they are signed by different TPM platform.

This process establishes the relevance of P2P nodes by use of ring signature scheme. This relevance can be an implicit stable identity, because the verifiers only know two or more pseudonyms created by the same TPM, but not sure which one TPM, ensuring the anonymity of P2P.

3.2. Storage and measurement of trust

The result of each interaction can affect the credibility of the node. We assume that every interaction between P2P nodes will have a mutual evaluation, according to the interaction result (success or failure).

For an interaction between nodes p and q, let \( S_{Trans}^{p,q} \) be the evaluation value of the q voted by p,

\[
S_{Trans}^{p,q} = \begin{cases} 
1 & \text{success} \\
-1 & \text{failure}
\end{cases}
\]  

(2)

\( \text{sat}(p, q) \) is the total times of positive evaluation, \( \text{unsat}(p, q) \) is the total times of negative evaluation, \( S_{p,q} \) is the overall evaluation,

\[
S_{p,q} = \text{sat}(p,q) - \text{unsat}(p,q)
\]

(3)

Evaluation value reflects the views on transaction result from both sides. Each node uses the existing evaluation to calculate the credibility (trust) of the other nodes, and determines the trust relationship. Because of the large number of P2P nodes, so consider using a distributed storage structure [11] to store assessment information.

In this paper, after the completion of interaction, the node p makes the public part of ring signature of q as the keywords of Chord, mapping to the Chord node for storing the assessment information. So, all evaluation information about q will be stored at the same Chord node. Besides the evaluation value, \( \text{sat}(p, q) \) and \( \text{unsat}(p, q) \) should be recorded to calculate the total interact times. In practice, the evaluation value is given by a pseudonym on behalf of a P2P node, in order to establish stable identifier, the pseudonyms and ring signature should also be stored.

3.3. Global trust relationship

Generally, trust relationship is classified into direct trust and indirect trust; the former exists in interaction occurred between entities directly and the later generates through the recommendation [13]. Let \( D^{p,q}_Direct \) be the direct trust for Node p on q, denoted as \( T(p, q) \) in following,
A Trust Evaluation Algorithm in P2P Network based on Trust Model
Zhenling Wang
International Journal of Digital Content Technology and its Applications. Volume 5, Number 8, August 2011

\[ T_{Direct}^{p,q} = \frac{S_{p,q}}{[sat(p,q) + unsat(p,q)]} \quad (4) \]

Obviously, if the number of negative rating is more than positive number, \( T_{Direct}^{p,q} \) is negative. For those nodes there are no direct interaction, that is \( \text{sat}(p, q) + \text{unsat}(p, q) = 0 \),

\[ T_{Direct}^{p,q} = \begin{cases} S_{p,q} / [\text{sat}(p,q) + \text{unsat}(p,q)] & \text{ever interaction} \\ E_{p,q} & \text{no interaction} \end{cases} \quad (5) \]

Where \(-1 \leq E_{p,q} \leq 1\) and \( E_{p,q} \neq E_{q,p} \), \( E_{p,q} \) represent the expectation of direct trust on \( q \) from perspective \( p \), its value depends on the average estimate of credibility of any node in current interaction environment.

In addition to the direct trust, indirect trust should also be counted. Indirect trust comes from recommendation information provided by the neighboring nodes from \( p \) (e.g. node \( k \)). At the meantime, the trust of node \( k \) should also be considered. Let \( T_{indirect}^{p,q} \) be the indirect trust from \( p \) to \( q \),

\[ T_{indirect}^{p,q} = \sum_{k \in P, k \neq p, q} T_{Direct}^{p,q} \times T(k, q) \quad (6) \]

So, the trust from \( p \) to \( q \) could be quantitative as \( T_{p,q} \).

\[ T_{p,q} = \alpha \times T_{Direct}^{p,q} + (1 - \alpha) \times \sum_{k \in P, k \neq p, q} T_{Direct}^{p,k} \times T(k, q) \quad (7) \]

\( \alpha \) adjusts the direct trust and indirect trust in the assessment of the status of the trust.

3.4. Host trust assessment

A P2P node could use a pseudonym to precede credit transactions, but use another to cheat. So all behaviors of every pseudonym should be counted for node trust assessment, and the result of trust assessment should reflect the positive (negative) effects impacted by effects/malicious behavior.

In specific applications, the trust value is assessed by a pseudonym of evaluators to establish the trust relationship with other entities. In order to facilitate the presentation, we do not separate the pseudonyms of an evaluator in following. In fact, the implementation of assessment in our trust model is independent to every pseudonym of an evaluator node indeed; each pseudonym of the evaluator will get the same assessment result.

4. Pseudonym trust

Definition of pseudonym trust: credibility of a pseudonym of a node in a P2P network. Let \( T_{pseud}^{p,q_j} \), be trust value for pseudonym \( q_j \) of \( q \) from node \( p \),

\[ T_{pseud}^{p,q_j} = \alpha \times T_{Direct}^{p,q_j} + (1 - \alpha) \times T_{indirect}^{p,q_j} \quad (8) \]

\[ = \alpha \times T_{Direct}^{p,q_j} + (1 - \alpha) \times \sum_{k \in P, k \neq p, q_j} T_{Direct}^{p,k} \times T(k, q_j) \]
In (8), $T_{\text{Direct}}^{p,q_j}$ not only includes the history estimate information between current pseudonym of the evaluator (e.g. $p_i$) and $q_j$, but also includes history estimate information between other pseudonym of $p$ and $q_j$. Algorithm 1 is the calculation approach for pseudonym trust.

Algorithm 1: $h_q$ is the storage node for the evaluation information about $q_j$. $(x_i, q_j, \text{sat}(X_i, q_j), \text{unsat}(x_i, q_j), \sigma_i)$ is a record of evaluation information stored in $h_q$. $X_i$ is the evaluator side pseudonym, $q_j$ is the evaluated side pseudonym of $q_j$. $\sigma_i$ is that ring signature of $x_i$. $p_i$ is the current evaluator pseudonym of node $p$. $p_{xi}$ is a pseudonym of node ever interacted with the $x_i$. $F$ is a map from pseudonym to the host stable identity,

$$T_{\text{Direct}}^{p,q_j} = T_{\text{Direct}}^{p,q_j} = E(p_i)$$

$$\text{sat}(p, q_j) = 0$$

$$\text{unsat}(p, q_j) = 0$$

for $(x_i, q_j, \text{sat}(x_i, q_j), \text{unsat}(x_i, q_j), \sigma_i)$ in $h_q$ do

if $x_i = p_i$ or $F(x_i) = F(p_i)$

$$\text{sat}(p, q_j) = \sum \text{sat}(x_i, q_j)$$

$$\text{unsat}(p, q_j) = \sum \text{unsat}(x_i, q_j)$$

endif
endfor

computing

$$T_{\text{Direct}}^{p,q_j} = \frac{\text{sat}(p, q_j) - \text{unsat}(p, q_j)}{\text{sat}(p, q_j) + \text{unsat}(p, q_j)}$$

computing

$$T_{\text{Pseudo}}^{p,q_j} = \alpha \times T_{\text{Direct}}^{p,q_j} + (1 - \alpha)T_{\text{Indirect}}^{p,q_j}$$

Pseudonym trust evaluation do not cover the situation in which peer assessed has more than one pseudonym. So the evaluation of pseudonym trust has certain one-sidedness.

5. Node trust

Definition of Node trust: credibility of all pseudonym of a node in the same P2P network. A node trust reflects trust of all pseudonyms in a P2P network. Let $T_{\text{Peer}}^{p,q_j}$ be the node trust from node $p$ to $q$, $N_q$ be the set of pseudonym of node $q$ in a P2P network,

$$T_{\text{Peer}}^{p,q_j} = \sum_{q_j \in N_q} R_{Pseudo}^{q_j} T_{\text{Pseudo}}^{p,q_j}$$

(9)

Where

$$R_{Pseudo}^{q_j} = \frac{\sum_{k \in p \land k \in N_q} (\text{sat}(k, q_j) + \text{unsat}(k, q_j))}{\sum_{q_j \in N_q} \sum_{k \in p \land k \in N_q} (\text{sat}(k, q_j) + \text{unsat}(k, q_j))}$$
is the weight factors, its value is the ratio of the interaction times pseudonym \( q_j \) participated and the interaction times of all pseudonym of \( q \) in net \( N \). The situation \( k \) and \( q_j \) both belongs to node \( q \) has been excluded from \( R^{q_j} \). In other words, the assessment information provided by a pseudonym of \( q \) do not count for the node trust of \( q \), preventing the Pseudo spoofing attach spurned from \( q \).

6. Host trust

Definition of host trust: credibility of platform in each P2P network. Let \( T^U,V_{Host} \) be host trust from \( U \) to \( V \), \( u \) and \( v \) is the corresponding node of \( U \) and \( V \) in each P2P system. \( V_P \) is set of \( v \), \( N_y \) is the set of node \( v \) in the P2P network. \( V_j \) is a pseudonym of \( v \).

\[
T^U,V_{Host} = \sum_{v \in V^P} R^v \cdot T^U,V_{Peer}
\]

Where,

\[
R^v = \sum_{k \in N^v', k \in P} \sum_{v \in V^P} \frac{(sat(k,v_j) + unsat(k,v_j))}{(sat(k,v_j) + unsat(k,v_j))}
\]

\( R^v \) is the node trust weight factors, the value is the ratio of the interaction times node \( v \) participated and the interaction times of all pseudonym of \( V \). Similar with node trust assessment [14], the situation in which \( k \) and \( V_j \) both belong node \( v \) has been excluded from \( R^v \).

7. Analysis

7.1. Algorithm complexity

In our node-centric P2P trust model[15], trust relationships are built according to the evaluation of information received by each P2P node independently. The storage of credibility information and computation are distributed, there is no performance bottleneck.

Space complexity of Evaluation information storage is related to the number of hosts, unrelated to distributed architecture used to store Evaluation information. Assumed there are \( m \) hosts spreading in \( k \) P2P networks (\( k \ll m \)) in the Evaluation environment. Each host has \( n \) different pseudonyms on average, and each pseudonym interacts with other pseudonyms. The number of records of evaluation information need to be stored is \( 2nC^2_{m} \), which are stored in \( m \) Chord nodes averagely, each node stores \( n \) (\( m-1 \)) records. Generally, \( n \) equal \( k \) approximately (that is, each host in a P2P network, only own a pseudonym), so the number of records stored in each Chord node is about \( O(m) \).

In pseudonym trust computing algorithm, for the direct trust calculation, records in the node stored pseudonym information about \( q \) need be traversal queried, the time complexity is \( O(m) \). For indirect trust calculation, the time complexity is related with the pseudonyms that have directly interact with pseudonyms evaluated (e.g. \( q_j \)). According to the relevancy between host trust and pseudonym trust, host trust, the time complexity of the direct trust is \( O(kn^3) \). Because \( n \approx k, k \ll m \), so the time complexity of host trust computation is \( O(m^3) \).
7.2. Security analysis

The stable identity established for pseudonyms of P2P nodes is implicit. P2P user still use pseudonyms to interact with each other, but pseudonyms are mapped to the unique identity related TPM through linkable signature mechanism. In our trust model, any of the pseudonym behavior can affect the trust assessment of the TPM, which can effectively resist the Pseudo spoofing attacks. Moreover, because the pseudonym is inherently associated with the TPM, the trust evaluation mechanism could distinguish the same pseudonym created by different TPM platform, and evaluate them as different targets, effectively resist the ID stealth attack.

8. Conclusion

This paper presents a P2P trust model based on trust computing: Building the implicit stable identity for TPM (Trusted Platform Module) platform with ring signature technology, we establish the host trust evaluation mechanism in P2P environments reference to social network. According the relationship among pseudonym, node and host, bring forward the iterative relationships of trust evaluation and the algorithm for calculation.

9. References