CETEF：A Comprehensive and Efficient Trust Evaluation Framework for Distributed Networks

Juanjuan Zhang[1,2], Jinglin Li[1], Qibo Sun[1], Ao Zhou[1]

[1] State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, China
[2] Science and Technology on Information Transmission and Dissemination in Communication,
Networks Laboratory, China
zhangjuanjuan_815@hotmail.com, {jlli, qbsun, aozhou}@bupt.edu.cn

Abstract

In order to defend the distributed networks from malicious behaviors, many trust evaluation models have been proposed by the researchers. However, most of the existing trust models do not consider the uncertainty in distributed network environment adequately. In addition, current trust models do not carry out a fine-grained analysis on the recommendations and cannot filter the malicious recommendations effectively. Therefore, current trust evaluation model cannot resist the malicious behaviors effectively. To attack these challenges, a comprehensive and efficient distributed trust evaluation framework is proposed in this paper to improve the evaluation accuracy. There are three types of trust in our trust evaluation framework: direct trust, recommendation trust and indirect trust. Firstly, the direct trust is calculated based on an improved subjective logic model. Then, the recommendation trust is calculated based on the recommendations from nodes who had interactions with both the subject node and the object node. Thirdly, upon indirect recommendations clustering, a public opinion is generated to represent the indirect trust of the node. Lastly, the three types of trust are fussed to calculate the final trust of the given node. The experimental analysis shows the effectiveness of our trust evaluation framework.

Keywords: Security, Trust evaluation, Distributed networks, Malicious behavior

1 Introduction

Network has evolved from traditional centralized computer communication platform to the ubiquitous distributed computing platform [1-2]. Distributed networks, such as P2P and wireless sensor networks, are widely applied in various scenarios. For offering efficient solutions in numerous application domains, utility of distributed networks is improving every day. Many research efforts are focused on distributed network currently.

However, distributed networks have its own drawbacks for locating in open environments. The openness nature of a distributed network makes it an ideal medium for attackers to do various vicious things. Internal attacks caused by the captured selfish or malicious nodes would greatly degrade the network performance. Recently, trust evaluation has become an essential way of dealing with these security issues.

The importance of trust evaluation [3-5] in distributed networks has been acknowledged by the researchers. Trust evaluation framework is fundamental in malicious nodes detection. Many trust models have been developed to construct trust relationships among nodes in distributed network environments [6-12]. From the literature on this topic, we find that: (1) most of the existing trust models do not consider the uncertainty in the real world network environment adequately. (2) The recommendations from the third parties are not always trusty and reliable. A node is more familiar with the nodes that it had interaction with. Thus, in order to detect the malicious nodes and resist the attacks effectively, a fine-grained analysis on the recommendations is necessary.

In order to solve the above-mentioned problems, we propose a comprehensive and efficient trust evaluation framework CETEF. The contributions of this paper are as follows:

- An improved subjective logic model is proposed to calculate the direct trust. Both the negative effect of malicious behaviors and the uncertainty factor caused by communication process are considered in direct trust calculation.
- A fine-grained analysis on the recommendations is taken. The recommendations are divided into two categories in our framework. Different from previous trust evaluation approaches, we design a simplified cluster algorithm for calculating indirect trust. We find that the proposed algorithm have a good effect against malicious behaviors in distributed network.

[*] Corresponding Author: Ao Zhou; E-mail: aozhou@bupt.edu.cn

DOI: 10.3966/160792642018121907020

- We conduct experiments to evaluate performance of the proposed trust evaluation framework CETEF. Obtained numerical results indicate that the proposed trust evaluation framework can improve the evaluation accuracy.

The rest of this paper is organized as follows: Section 2 introduces related work. Section 3 presents the overview of the trust evaluation framework. Section 4 introduces the proposed trust evaluation framework in detail. Section 5 describes the experiments, and Section 6 concludes this paper.

2 Related Work

The concept of trust evaluation is firstly introduced by M. Blaze [13] in 1996. The trust is calculated based on the supplemental security information provided by a trustable third party. However, due to the emergency of distributed network, there is an increasing requirement of decentralized and distributed trust management system. Aiming at mitigating against misbehaved node in distributed network, a significant number of trust evaluation frameworks have been proposed to counter the security threat.

Trust has been evaluated in very different ways. One characteristic-based methods have been proposed by researchers. According to properties of trust evaluation metrics, the one characteristic-based methods can be divided into two categories. The first kind of scheme employs discrete numerical values or probability values to measure the trust degree. Discrete integer numbers 0/1 can be used to evaluate the trust of the target node's behaviors. "1" denotes a positive interaction, while "0" or "-1" denotes a negative interaction [14]. Then, based on concatenation and multipath trust propagation, simple numerical calculation operators, such as minimum, maximum, and weighted average, are used to calculate trust value. The trust can be calculated by the number of positive ratings and negative ratings [15], or a continuous value falls into the range of [0, 1] [17], or the binary ratings [18]. Trust value can also be classified into four grades: worth, bad, normal and perfect [16].

The trust evaluation frameworks mentioned above are simple to descript and easy to operate. However, the one characteristic-based method is too rough to effectively depict the complex behavior in real world distributed network environment. In order to obtain a more comprehensive description of complex behaviors in real world distributed network environment, another kind of trust evaluation framework based on multi-characteristic has been proposed.

Since trust is multi-faceted even in the same context, it is still necessary to develop differentiated trust based on different aspects of nodes' behaviors. Bayesian networks can provide a flexible method to represent differentiated trust and combine different aspects of trust [19]. Another trust framework named Subjective Logic defines the representation, calculation, and combination of trust value quantitatively [20]. The concept of experience has also been introduced into the framework in paper [21] to describe and measure trust, which puts forward a kind of trust evaluation framework based on experience and probability statistics explanation.

However, those frameworks calculate the node trust based on the basic probability theory and do not take the fuzziness of trust itself into account. Subjective trust management framework considers the fuzziness of subjective trust, and constructs subjective trust management framework based on fuzzy set theory [22]. Fuzzy logic provides rules for reasoning linguistic trust metrics. However, the fuzzy logic-based method ignores the randomness in distributed network. Aiming at considering the subjective uncertainty such as randomness and fuzziness of trust value, Li [23] proposes a method to evaluate trust based on cloud model. The proposed method combines the fuzziness and uncertainty of trust.

The comparison table is shown in Table 1. Trust evaluation can distinguish between well-behaved and misbehaved nodes. Appropriate actions are taken when the misbehaved nodes have been detected. However, the uncertainty in the communication process between nodes in a real network environment has not been well considered. Apart from that, most trust evaluation frameworks simply collect the third-party trust information without distinguishing the information sources. In that case, recommendations from malicious nodes would interfere with the evaluation process. In this paper, we propose a comprehensive and efficient distributed trust evaluation framework CETEF, in which the uncertainty in a real network environment is taken into account. In addition, the mainstream opinion obtained by recommendation clustering can avoid the negative impact brought by malicious behaviors.

Table 1. Comparison table for above methods

Trust evaluation method	Related papers	Advantage	Disadvantage
One characteristic-based method	[14-18]	Simple to descript and easy to operate.	Too rough to effectively depict the complex behavior.
Multiple characteristic-based method	[19-21]	More comprehensive by considering multiple aspects of node behaviors.	Based on the probability theory and do not consider the fuzziness of trust.
Methods which take the fuzziness of trust itself into account	[22-23]	Consider the fuzziness of subjective trust.	Ignore the randomness in distributed network.

3 Preliminaries

3.1 Definitions

We think it is necessary to give a clear definition of trust before introducing our trust evaluation framework. The trust properties that are adopted in a trust evaluation model are also discussed in this section. Up to now, there is no uniform definition of trust, and several definitions are given to trust by researchers.

In this paper, trust is defined as a belief level of a node puts on another node based on the observations of historical behaviors. Trust value can be taken as a reference basis for a node to act normally.

In this paper, the node who wants to calculate the trust of another node is referred to as "trustor", and the object node to be evaluated is referred to as "trustee". The trust value falls to the range of [0, 1].

Direct trust is a kind of trust between two nodes that calculated based on their direct historical interactions.

Recommendation trust is a special type of trust relationship established based on recommendations from nodes who had interactions with both trustor and trustee. The common node is referred to as the recommendation node in this paper.

Indirect trust is a type of trust from third party. Indirect trust is calculated based on the interaction information from the nodes who only had direct interaction with the trustee.

Both recommendation trust and indirect trust are calculated based on recommendations from the third parties. However, recommendation nodes had direct interactions with both trustor and trustee while the indirect recommender only had direct interactions with trustee. As shown in Figure 1, trustor A wants to calculate the trust value of trustee B. The direct trust of A to B comes from the direct interaction experiences. The nodes in set C = $\{C_1, C_2\}$ which had direct interactions with both A and B are direct recommender. For node A, all node in set D = $\{D_1, D_2, D_3, D_4\}$, which only had direct interactions with B, are indirect recommender.

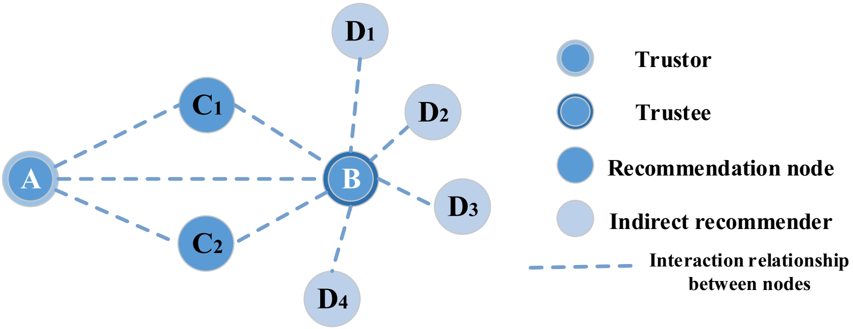

Figure 1. The network structure

3.2 System Architecture

In this section, we describe the system architecture of CETEF. In order to do a comprehensive and impartial trust evaluation, three types of trust, including direct trust, recommendation trust and indirect trust, are used to evaluate the trustworthiness of each node. Figure 2 illustrates the system architecture of our trust evaluation framework.

As shown in Figure 2, CETEF consists of five main modules: data collection module, direct trust calculation module, recommendation trust calculation module, indirect trust calculation module and final trust calculation module. The data collection module is mainly responsible for collecting information about the trustee. The trust collection module consists of the following three components: direct data collection, recommendation data collection, and indirect data collection. When a node A wants to obtain the information of another node B, the direct data collection module first checks the local record list. If the ID of B is in the local record list, the module gets all records about node B. Recommendation data collection module receives the recommendation data about node B from recommendation node, and indirect data collection module receives indirect data from indirect recommenders. The direct trust calculation module calculates the direct trust based on the communication behaviors between A and B. An improved subjective logic model is proposed to improve the evaluation accuracy. The technical detail of direct trust calculation module will be introduced in Section 4.1. Due to the malicious attacks, it is not effective to evaluate B using only direct trust. The recommendation trust calculation module deals with the behavior data received from the recommendation node. The technical detail of direct recommendation trust calculation is illustrated in Section 4.2. The indirect trust calculation module clusters indirect recommendations and obtains the indirect trust by analyzing the clustering results. The detailed technology of this module will be illustrated in section 4.3. Final trust calculation module fuses the direct trust, the recommendation trust, and the indirect trust to obtain the final trust of B. The technical detail of final trust calculation is illustrated in Section 4.4.

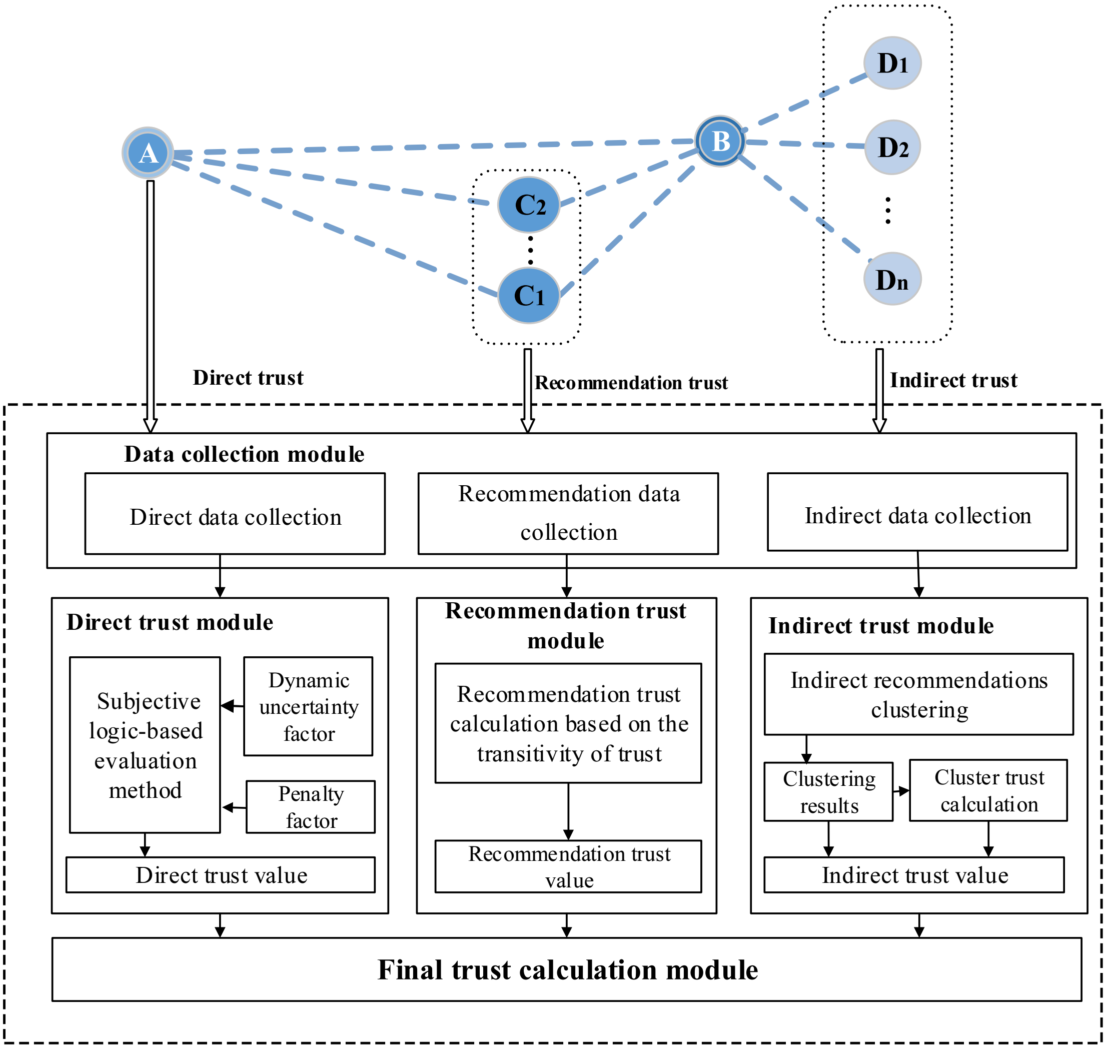
A
B
C2
C1
D1
D2
Dn
Direct trust
Recommendation trust
Indirect trust
Data collection module
Direct data collection
Recommendation data collection
Indirect data collection
Direct trust module
Subjective logic-based evaluation method
Dynamic uncertainty factor
Penalty factor
Direct trust value
Recommendation trust module
Recommendation trust calculation based on the transitivity of trust
Recommendation trust value
Indirect trust module
Indirect recommendations clustering
Clustering results
Cluster trust calculation
Indirect trust value
Final trust calculation module

Figure 2. System architecture of trust evaluation framework CETEF

4 Trust Calculation in CETEF

This section discusses the trust evaluation procedure in detail.

4.1 Direct Trust Calculation

A direct trust relationship exists if the trustor had interactions with the trustee in the past. Direct trust is calculated through observations on whether the previous interactions between the trustor and the trustee are successful. Suppose that trustor A had interactions with trustee B in the past, then the direct trust value from A to B can be calculated by reviewing the local-saved historical behavior information. In this paper, we extensively employ subjective logic model [20] to calculate direct trust. The subjective logic model uses the term "opinion" to express the trust degree of a node. In subjective logic model, opinion can be denoted by a tuple $\{b, d, u, a\}$. b denotes the trustor's belief in trustee, d denotes the trustor's disbelief in trustee, u denotes the trustor's uncertainty on trustee, and a denotes the relative atomicity. $b + d + u = 1$ and the expectation that a node acts cooperatively can be calculated as $b + a * u$.

Based on the existing subjective logic model, an improved direct trust calculation model with the consideration of the punishment intensity of negative events and the uncertainty factor is introduced in this paper. $T_{AB-direct}$ is used to denote the direct trust from trustor A to trustee B. Then, the direct trust from trustor A toward trustee B can be calculated by the following:

$$b = \frac{r}{r + ps + \mu} \quad \textbf{(1)}$$

$$d = \frac{ps}{r + ps + \mu} \quad \textbf{(2)}$$

$$u = \frac{u}{r + ps + \mu} \quad \textbf{(3)}$$

$$\mu = \omega_C \times (r + s) \quad \textbf{(4)}$$

$$T_{AB-direct} = b + au = \frac{r + a\mu}{r + ps + \mu} \quad \textbf{(5)}$$

where r and s are the number of positive and negative interactions respectively. The effect of negative interactions should be greater than the positive one. Hence we define p as the penalty factor, and its value is greater than 1. Since the distributed network is in a natural unstable and noisy environment, an uncertainty factor μ is defined to demonstrate the uncertainty in node interactions. The factor is always defined as a constant in the existing subjective logic-based method [24-25]. However, the effect degree of μ would decrease with the increasing of interaction frequency. In this paper, we define the uncertainty factor as a dynamic factor. The value of the uncertainty factor μ changes with the total interaction frequency. In Equation (4), ω_c is used to control the effect degree of uncertainty, and $\omega_c \in [0,1]$. The relative atomicity of the proposition (a) is set to 1/2 in this paper.

As can be seen from the equations, if trustee act cooperatively and honestly in the network, the number of positive interactions would increase. Therefore, $T_{AB-direct}$ also increases. Otherwise, $T_{AB-direct}$ declines rapidly and the malicious node is heavily punished.

4.2 Recommendation Trust Calculation

The node trust information from third parties are not always reliable. Hence, we need to propose an efficient strategy to filter the node trust information. As introduced in Section 2, node trust information from third parties can be divided into two parts: recommendation information from recommendation nodes and indirect information from indirect recommenders. Comparing with other nodes, the trustor is more familiar with the node who had direct interactions with him. Therefore, the two types of trust are treated differently. The recommendation trust is obtained based on the recommendations from the nodes who had direct interactions with both trustor and trustee.

As shown in Figure 1, the nodes in set C= {C_1, C_2, ..., C_i,..., C_n} had interactions with both A and B. The recommendation trust value from A to B is calculated by the following:

$$T_{AB-recommend} = \frac{\sum_{i=1}^{n} T_{AC_i-direct} \times T_{C_iB-direct}}{n} \quad \textbf{(6)}$$

where n is the number of recommendation nodes.

4.3 Indirect Trust Calculation

In indirect trust calculation, the trustor would transmit a request message to other nodes. The node receiving the request message first checks the local record list. If the ID of the trustee is in the local record list, the node would response to the request message.

As shown in Figure 1, suppose that the trustor A didn't have any interaction with trustee B in the past. The nodes in set D={D_1, D_2, …, D_i,..., D_m} only had direct interactions with B in the past. The nodes in D are indirect recommenders for A. In this condition, A can ask nodes in D for the trust information of B. Suppose indirect recommendation trust from D_i towards B is denoted by T_{D_iB}. A can get the following set:

$$T_{indirect-seq} = \left\{ T_{D_1B}, T_{D_2B}, \ldots, T_{D_iB}, \ldots, T_{D_mB} \right\}.$$

A large amount of trust values have been received from the indirect recommenders. Because there are many malicious nodes in the distributed network, some of the received trust values cannot reflect the actual behaviors of the trustee. Different from the recommendation nodes, the trustor is not familiar with the indirect recommenders and cannot judge whether the indirect recommenders are honest or not. In addition, the actual trust degree of the trustee is unknown currently. Therefore, these trust values cannot be classified through simple mathematical statistics. To address the problem, we propose a simplified cluster algorithm to resist the attack from the malicious indirect recommenders. To distinguish between malicious nodes and honest nodes, we first obtain the mainstream opinion and the minority opinion by analyzing $T_{indirect-seq}$. We try to classify $T_{indirect-seq}$ into two clusters. Firstly, we select the maximum value in $T_{indirect-seq}$ (denoted by $max\ T_{indirect-seq}$) and the minimum value in $T_{indirect-seq}$ (denoted by $min\ T_{indirect-seq}$) as the centroids of two initial clusters. Then the algorithm classify items in $T_{indirect-seq}$ into two clusters based on the Euclidean distance [26]. The detailed clustering process is illustrated in Algorithm 1.

As can be seen in the Algorithm 1, we classify $T_{indirect-seq}$ into two stable clusters if the centroids of the two clusters do not change. The trust values in $cluster_i$ denotes the mainstream opinion. Although various types of malicious behaviors may have negative impact on the network, they can only effect the minority opinions. By filtering all trust values that are not in $cluster_i$, the trust evaluation model can resist the attack of malicious and misbehaved nodes.

Algorithm 1. Simplified Trust Cluster Algorithm

```
Input: ${T_{D_1B}, T_{D_2B}, ..., T_{D_iB}, ..., T_{D_mB}}$
Output: $cluster_i$
initialize two cluster centroids
$cen_1$=max $T_{indirect-seq}$ , $cen_2$ =min $T_{indirect-seq}$
do
    assign each item in $T_{indirect-seq}$ to the nearest centroid
    recount the centroid of each cluster
until the centroids do not change
return $cluster_i$ with the larger number of elements
```

We now calculate the indirect trust based on the subjective logic model mentioned in Section 3.1.

Firstly, the trustor calculates the average value ($\overline{t_i}$) of $cluster_i$ as well as the standard deviation (σ_i) of $cluster_i$. The trust values that do not fall within the range of $(\overline{t_i}-\sigma_i, \overline{t_i}+\sigma_i)$ are filtered out since they are far from the mean value.

Then, the trustor's subjective opinion about $cluster_i$ is calculated. The trust value that is close to the mean value is considered as the belief opinion while the rest is regarded as uncertain opinion. Therefore, the belief in opinion (b_p) is the percentage of trust values that fall within the range of $(\overline{t_i}-\sigma_i, \overline{t_i}+\sigma_i)$. The relative atomicity (a) is the average trust value of $cluster_i$. The uncertainty in opinion (u_p) is the percentage of trust values which fall outside the range of $(\overline{t_i}-\sigma_i, \overline{t_i}+\sigma_i)$. Since the dishonest trust values have been filtered out, the disbelief in opinion can be set as 0. Hence, the opinion about $cluster_i$ is a 4-tuple $(b_p, 0, u_p, a)$. The trustor's subjective opinion about the cluster $cluster_i$ can be calculated by the following:

$$T_{A-cluster_i} = b_p + au_p \tag{7}$$

Finally, the indirect trust value of trustee can be calculated by the following:

$$T_{AB-indirect} = T_{A-cluster_i} \times cen_i \tag{8}$$

where cen_i denotes the centroid of $cluster_i$.

4.4 Final Trust Calculation

Based on the direct trust, the recommendation trust, and the indirect trust, the final trust is calculated as follows:

$$\begin{aligned} T_{AB-comprehensive} = \omega_i T_{AB-direct} &+ (1-\omega_i)\frac{\omega_r N_r}{\omega_r N_r + N_i} T_{AB-recommend} \\ &+ (1-\omega_i)\frac{N_i}{\omega_r N_r + N_i} T_{AB-indirect} \end{aligned} \tag{9}$$

where $\omega_i (0 < \omega_i < 1)$ is the weight of the direct trust. We will illustrate the impact of ω_i and ω_r in the experiment section. N_r denotes the number of recommendation node, N_i denotes the number of indirect recommenders.

5 Experiment Results

In order to demonstrate the advantage of our framework in trust evaluation and malicious behaviors detection, we conduct a serial of experiments. Firstly, we evaluate the performance of all frameworks on trust computation error, and detection ratio. Then, we study the performance of our framework on different simulation parameters.

5.1 Experiment Setup

We have implemented CETEF based on the software OMNeT++ [27]. We construct a distributed network which consists of 100 nodes. The nodes are divided into two sets in our simulation: honest nodes and malicious nodes. The percentage of malicious nodes is denoted by *malicious_rate*. We list some parameters used during trust evaluation in Table 2.

Table 2. Simulation parameters

Parameter	Description	Range
malicious_rate	Percentage of malicious nodes	$[0,1]$
mrate	Percentage of interactions that a malicious peer act dishonestly	$[0,1]$
interact_num	Average interactions frequency for a node	$[0,\infty)$
ω_c	Uncertainty factor in direct trust calculation	$[0,1)$
p	Penalty factor for negative behaviors in direct trust calculation	$(0,\infty)$
ω_i	Weight of the direct trust	$[0,1]$
ω_r	Weight of the recommendation trust	$(0,\infty)$

5.2 Performance Evaluation

5.2.1 Comparison of Trust Evaluation

In this section, we compare our trust framework with the trust framework EDTM proposed in [25]. The performance metrics includes the evaluation deviation of indirect recommendation trust and the evaluation deviation of final trust for the well-behaved node. As for a node in the network, the deviation of trust evaluation denotes the difference between the trust value and the actual possibility for a node to act honestly. In these experiments, we vary *malicious_rate* from 0% to 70%, with a step of 10%.

As shown in Figure 3, with the increase of malicious rate, the indirect recommendation trust value of CETEF is much more stable than that of EDTM. The indirect recommendation trust value of CETEF falls to the range of (0.75, 0.97), while that of EDTM decreases rapidly from 0.97 to less than 0.3.

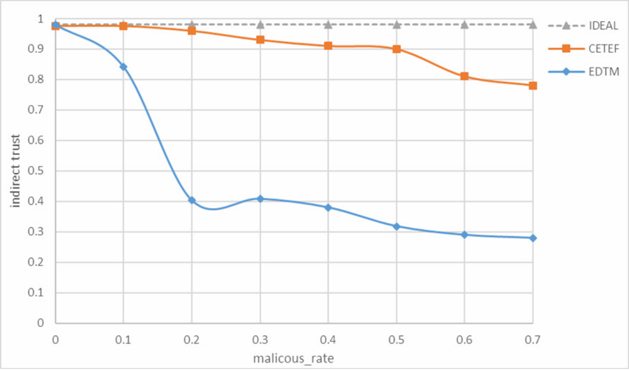

Figure 3. Indirect trust deviation

Figure 4 shows the results on final trust calculation. With the increase of malicious rate, the final trust shows the same downward trend as the indirect recommendation trust. The trust value of CETEF falls to 0.81 from the ideal value 0.97, while the trust value of EDTM declines dramatically to 0.6.

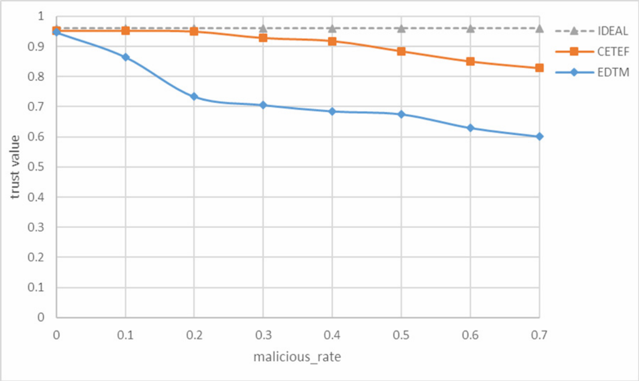

Figure 4. Final trust deviation

An honest node is always judged as an honest one by our CETEF. However, the compared framework may misjudgment it as a malicious node when the number of malicious nodes increases. In summary, the experimental results indicate the robustness of our CETEF in hostile network environment.

5.2.2 Comparison of Trust Computation Error

We take trust computation error (TEC for short) [28] as the effectiveness evaluation metric. TEC is calculated by the following:

$$TCE = \sqrt{\frac{\sum_{i\in U}\left[\tau_t(i) - p_t(i)\right]^2}{|U|}} \qquad \textbf{(10)}$$

where $|U|$ denotes the number of nodes in the network, and $\tau_t(i)$ denotes the trust value of *i-th* node at the time *t*. $P_t(i)$ denotes the expected possibility for the node *i* to act honestly, hence $P_t(i)=1$ means *i-th* node acts as an honest one at the time *t*, otherwise $P_t(i)=0$. A smaller trust computation error indicates a higher evaluation accuracy.

Figure 5 illustrates the trust computation error of EDTM and CETEF. With the increase of malicious rate, the trust computation error of EDTM increases more quickly compared to CETEF. The performance of EDTM drops almost linearly. Therefore, EDTM is less robust against the increasing of malicious behaviors. Our framework CETEF obtains better evaluation accuracy under the same condition.

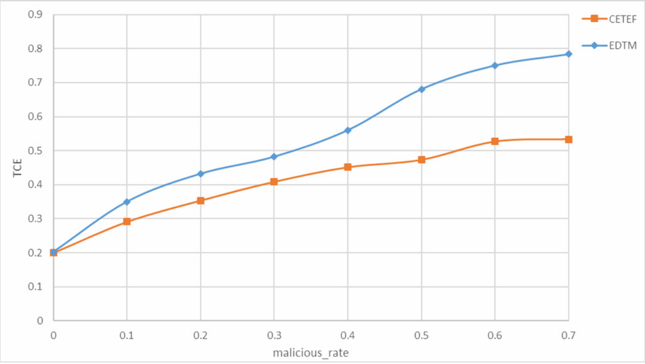

Figure 5. Trust computation error comparison

5.2.3 Comparison of Detection Ratio

In this experiment, we compare our trust evaluation framework with EDTM with respect to the malicious nodes detection ratio. *malicious_rate* varies from 0% to 70%, with a step value of 10%. In Figure 6, the ODJECT line illustrates the actual malicious rate of the network. As shown in Figure 6, the detection ratio of our framework is barely close to OBJECT, while the difference between EDTM and OBJECT is significantly larger. Compared with EDTM, our trust evaluation framework obtain better malicious nodes detection ratio in all experimental settings.

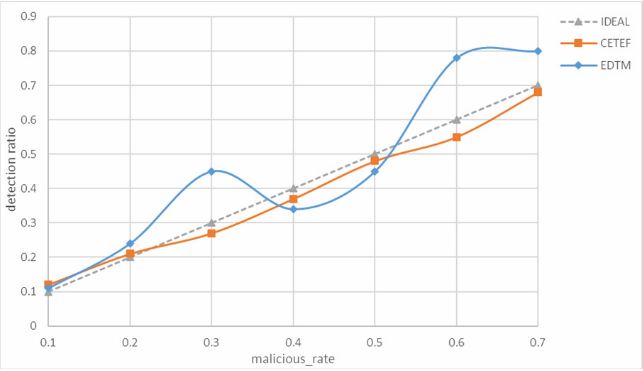

Figure 6. Malicious nodes detection ratio

5.3 Studies on Parameters

5.3.1 Impact of Penalty Factor *p*

In the direct trust calculation module, the parameter *p* denotes the punishment intensity for malicious behavior. To study the impact of *p*, we vary the value of *p* from 1 to 4 with a step value of 1. We set ω_c = 0.01, (ω_i, ω_r) = (2, 0.8) in the experiment. In order to show the impact clearly, the object value, which is calculated based on the actual possibility for a node to act dishonestly, is derived.

Hence, the direct trust value is more accurate if it is closer to the object value. Figure 7 shows the results of direct trust value when we vary the value of *p* from 1 to 4. We can see from the result that the direct trust for a node decreases with the increase of *mrate*. In addition, CETEF achieves better effect when *p* falls to the range of [2, 3].

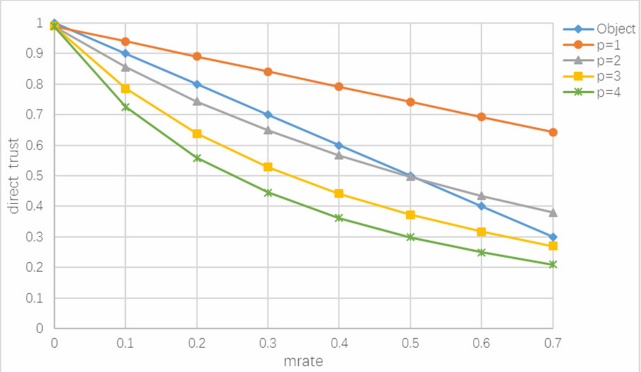

Figure 7. Impact of penalty factor *p*

5.3.2 Impact of Parameter ω_r

The parameter ω_r is used to adjust the importance of recommendation trust in final trust calculation. To study the impact of ω_r, we vary the value of ω_r and calculate the final trust value of the honest node.

As shown in Figure 8, when the percentage of malicious nodes does not exceed 40%, the trust evaluation results are almost similar under different values of ω_r. When the network environment is relatively well, the cluster algorithm in indirect trust evaluation module can effectively filter out the malicious information. However, the trust value shows a downward trend as the malicious rate continually grows. It turns out that with an increasing number of malicious nodes in the network, the higher ω_r is, the more stable the obtained trust value is. Hence, node trust information from common neighbors is more reliable than from the indirect recommenders. That’s because the increasing number of malicious behaviors badly interfere with the trust evaluation process.

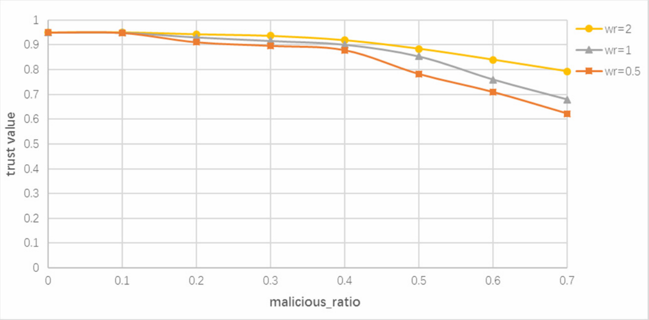

Figure 8. Impact of parameter ω_r

5.3.3 Impact of Parameter ω_i

To study the impact of ω_i, we vary the value of ω_i from 0.2 to 0.8 with a step value of 0.2. We set ω_c = 0.01, ω_r = 2, *p*=3 in the experiment. Figure 9 shows that the final trust value decreases and deviates from the actual value when the malicious rate increases from 0% to 70%. That’s because the increasing number of malicious nodes would provide incorrect information to interfere with the trust evaluation. When ω_i >0.5, the trust evaluation result becomes more accurate and the evaluation framework becomes more stable. Therefore, suitable ω_i value will bring about better evaluation accuracy. In other words, this observation indicates that the weight of first-hand information should be larger than the second-hand information.

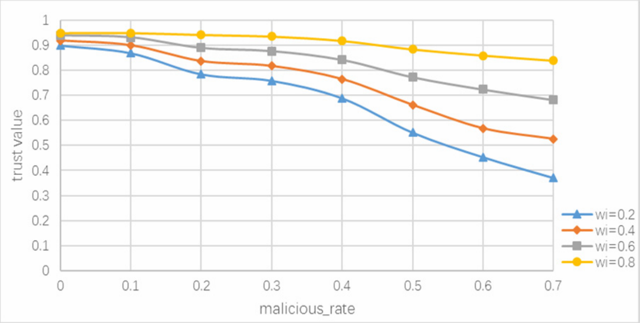

Figure 9. Impact of parameter ω_i

6 Conclusion

In this paper, an efficient trust evaluation framework CETEF is proposed to calculate the trust value of participating nodes and defend trust evaluation framework against malicious attacks. During CETEF, we discuss how to calculate the direct trust, the recommendation trust, the indirect trust, and the final trust. Simulation results demonstrate that CETEF is attack-resistant, and provides an efficient mechanism for detecting malicious nodes. Our future work includes investigating the optimal parameters selection in different environments.

Acknowledgments

The work is supported by Science and Technology on Information Transmission and Dissemination in Communication Networks Laboratory, the NSFC (61571066), and the Fundamental Research Funds for the Central Universities.

References

[1] Z. Chen, L. Li, J. Gui, Fuzzy Theory for the P2P Subject Trust Evaluation Model, *International Journal of Advancements in Computing Technology*, Vol. 4, No. 8, pp. 67-74, April, 2012.

[2] J. Dubey, V. Tokekar, Bayesian Network Based Trust Model with Time Window for Pure P2P Computing, *IEEE Global Conference on Wireless Computing and Networking*, Lonavala, India, 2014, pp. 219-223.

[3] P. Manuel, A Trust Model of Cloud Computing Based on Quality of Service, A*nnals of Operations Research*, Vol. 233, No. 1, pp. 281-292, October, 2015.

[4] X. Wang, L. Liu, J. Su, RLM: A General Model for Trust Representation and Aggregation, *IEEE Transactions on Services Computing*, Vol. 5, No. 1, pp. 131-143, January-March, 2012.

[5] F. Liu, L. Wang, L. Gao, H. Lie, H. Zhao, A Web Service Trust Evaluation Model Based on Small-world Networks, *Knowledge-Based Systems*, Vol. 57, No. 2, pp. 161-167, 2014.

[6] N. Li, S. K. Das, A Trust-based Framework for Data Forwarding in Opportunistic Networks, *Ad Hoc Networks*, Vol. 11, No. 4, pp. 1497-1509, April, 2013.

[7] W. Jiang, G. Wang, J. Wu, Generating Trusted Graphs for Trust Evaluation in Online Social Networks, *Future Generation Computer Systems*, Vol. 31, No. 2, pp. 48-58, February, 2014.

[8] G. Zhan, W. Shi, J. Deng, Design and Implementation of TARF: A Trust-aware Routing Framework for WSNs, *IEEE Transactions on Dependable and Secure Computing*, Vol. 9, No. 2, pp. 184-197, March/April, 2012.

[9] D. He, C. Chen, S. Chan, J. Bu, A V. Vasilakos, A Distributed Trust Evaluation Model and Its Application Scenarios for Medical Sensor Networks, *IEEE Transactions on Information Technology in Biomedicine*, Vol. 16, No. 6, pp. 1164-1175, November, 2012.

[10] R. Chen, F. Bao, M J. Chang, H. Cho, Dynamic Trust Management for Delay Tolerant Networks and Its Application to Secure Routing, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 25, No. 5, pp. 1200-1210, May, 2014.

[11] S. Jiang, J. Zhang, Y S. Ong, An Evolutionary Model for Constructing Robust Trust Networks, *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems*, St. Paul, MN, 2013, pp. 813-820.

[12] D. Jia, F. Zhang, S. Liu, A Robust Collaborative Filtering Recommendation Algorithm Based on Multidimensional Trust Model, *Journal of Software*, Vol. 8, No. 1, pp. 11-18, January, 2013.

[13] M. Blaze, J. Feigenbaum, J. Lacy, Decentralized Trust Management, *IEEE Symposium on Security and Privacy*, 1996, pp. 164-173.

[14] K. Aberer, Z. Despotovic, Managing Trust in a Peer-2-peer Information System, *Proceedings of the Tenth International Conference on Information and Knowledge Management*, New York, NY, 2001, pp. 310-317.

[15] P. Resnick, R. Zeckhauser, Trust among Strangers in Internet Transactions: Empirical Analysis of ebay's Reputation System, in: M. R. Baye (Ed.), *The Economics of the Internet and E-commerce*, Emerald Group, 2002, pp. 23-25.

[16] W. Yuan, W. Cheng, A Multi-granularity Trust Model for Distributed Peer-to-peer Networks, *Journal of Huazhong University of Science and Technology*, Vol. 31, No. 11, pp. 88-91, November, 2007.

[17] U. Maurer, Modelling a Public-key Infrastructure, in: E. Bertino, H. Kurth, G. Martella, E. Montolivo (Eds.), *Computer Security, 4th European Symposium on Research in Computer Security*, Springer, 1996, pp. 325-350.

[18] S. Ganeriwal, L. K. Balzano, M. B. Srivastava, Reputation-based Framework for High Integrity Sensor Networks, *ACM Transactions on Sensor Networks*, Vol. 4, No. 3, pp. 1-12, May, 2008.

[19] Y. Wang, J. Vassileva, Bayesian Network Trust Model in Peer-to-peer Networks, *International Workshop on Agents and Peer-to-Peer Computing*, Estoril, Portugal, 2005, pp. 23-34.

[20] A. Jøsang, A Subjective Metric of Authentication, *5th European Symposium on Research in Computer Security*, Louvain-Neuve, Belgium, 1998, pp. 329-344.

[21] T. Beth, M. Borcherding, B. Klein. Valuation of Trust in Open Network, *Proceedings of the European Symposium on Research in Security*, Brighton, UK, 1994, pp. 1-18.

[22] T. Wen, C. Zhong, Research of Subjective Trust Management Model Based on the Fuzzy Set Theory, *Journal of Software*, Vol. 14, No. 8, pp. 1401-1408, August, 2003.

[23] D. Li, Uncertainty in Knowledge Representation, *Engineering Science*, Vol. 2000, No. 10, pp. 74-79, October, 2000.

[24] W. Gao, G. Zhang, W. Chen, Y. Li, A Trust Model Based on Subjective Logic, *IEEE Fourth International Conference on Internet Computing for Science and Engineering*, Harbin,

China, 2009, pp. 272-276.

[25] J. Jiang, G. Han, F. Wang, et al., An Efficient Distributed Trust Model for Wireless Sensor Networks, *IEEE Transactions on Parallel and Distributed System*s, Vol. 26, No. 5, pp. 1228-1237, May, 2015.

[26] P E. Danielsson, Euclidean Distance Mapping, *Computer Graphics and image processing*, Vol. 14, No. 3, pp. 227-248, November, 1980.

[27] OMNET++Discrete Event Simulation System, https://omnetpp.org/intro.

[28] L. Xiong, L. Liu, PeerTrust: Supporting Reputation-based Trust in Peer-to-peer Communities, *IEEE Transactions on Data and Knowledge Engineering*, Vol. 16, No. 7, pp. 843-857, July, 2004.

Biographies

Juanjuan Zhang received the Master degree in computer science and technology at State Key Laboratory of Networking and Switching Technology from Beijing University of Posts and Telecommunications in 2017.

Jinglin Li is an associate professor at the Beijing University of Posts and Telecommunication, China. His current research interests include mobile Internet, internet of things, internet of vehicles and convergence network service & security technologies

Qibo Sun received his Ph.D. degree in communication and electronic system from the Beijing University of Posts and Telecommunication in 2002. He is currently an associate professor at the Beijing University of Posts and Telecommunication in China. He is a member of the China computer federation. His current research interests include services computing, internet of things, and network security.

Ao Zhou is an assistant professor at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications. She received her Ph.D. degree in computer science at Beijing University of Posts and Telecommunications of China in 2015. Her research interests include cloud computing, service reliability.