 CMC, vol.59, no.2, pp.397-404, 2019

A Hierarchical Trust Model for Peer-to-Peer Networks

Nehal Al-Otaiby[1], Heba Kurdi[1, *] and Shiroq Al-Megren[1]

Abstract: Trust has become an increasingly important issue given society's growing reliance on electronic transactions. Peer-to-peer (P2P) networks are among the main electronic transaction environments affected by trust issues due to the freedom and anonymity of peers (users) and the inherent openness of these networks. A malicious peer can easily join a P2P network and abuse its peers and resources, resulting in a large-scale failure that might shut down the entire network. Therefore, a plethora of researchers have proposed trust management systems to mitigate the impact of the problem. However, due to the problem's scale and complexity, more research is necessary. The algorithm proposed here, HierarchTrust, attempts to create a more reliable environment in which the selection of a peer provider of a file or other resource is based on several trust values represented in hierarchical form. The values at the top of the hierarchical form are more trusted than those at the lower end of the hierarchy. Trust, in HierarchTrust, is generally calculated based on the standard deviation. Evaluation via simulation showed that HierarchTrust produced a better success rate than the well-established EigenTrust algorithm.

Keywords: Peer-to-peer network, trust management, reputation, malicious peers.

1 Introduction

Peer-to-peer (P2P) networks are comprised of a set of nodes that can directly communicate with each other without the need for a centralised server. Peers (users) in these networks are characterised by their anonymity, and anonymous peers can join or leave the network at any time. This makes P2P networks an ideal environment for malicious peers, who may provide harmful resources or act in a malicious manner. This, of course, puts the security of the network at risk and is considered to be one of the main challenges of P2P systems [Androutsellis-Theotokis and Spinellus (2004)]. Therefore, the challenge of establishing trust relationships between peers to ensure reliable file/service sharing and safe e-transactions is receiving increasing attention [Yang, Qin, Wang et al. (2010)]. It is important to build trust relationships between peers to encourage sharing of resources [Bhise and Kamble (2016)]. As a result of this increased interest, many researchers have focused on developing trust and reputation systems for various networks, e.g. [Kamvar, Schlosser and Garcia-Molina (2003); Kurdi (2015); Xiong and Liu (2017); Zhang, Zheng, Liu et al. (2011); Kurdi, Alshayban, Altoaimy et al. (2018); Bursell (2005); Mondal and Kitsuregawa (2006); Shala, Wacht, Trick et al. (2017); Zhao and Li (2013), Xie, Yuan, Zhou et al. (2018)].

[1] College of Computer and Information Science, King Saud University, Riyadh 11642, Saudi Arabia.

[*] Corresponding Author: Heba Kurdi. Email: hkurdi@ksu.edu.sa.

EigenTrust is one of the most popular reputation algorithms used in P2P networks; this algorithm uses an eigenvector in its calculation of a trust value [Kamvar, Schlosser and Garcia-Molina (2003)]. HonestPeer extends the original EigenTrust algorithm to overcome the issue of peers congregating around pre-trusted peers [Kurdi (2015)]. HonestPeer selects the most reputable (i.e., honest) peers dynamically based on the quality of the files each peer provides. Another trust-based algorithm, PeerTrust, calculates trust based on several factors such as feedback, number of transactions, credibility and community context factors [Xiong and Liu (2004)]. Alternatively, algorithms such as GroupTrust organise peers into different groups based on their evaluation of the same or similar services [Zhang, Zheng, Liu et al. (2011)]. In these instances, the evaluation of the trustworthiness of peers outside the group is based on their local and global reputations, which are aggregated from other peers. TrustFeer is another trust management system proposed for P2P federated clouds [Kurdi, Alshayban, Altoaimy et al. (2018)]. The algorithm relies on subjective logic opinions that are based on the reputation of peers and on service level agreements (SLAs) when evaluating the trustworthiness of peers.

As shown above, the literature contains many trust models. However, none consider a trust hierarchy built from a peer's direct experience, recommendations, reputation and feedback. This paper proposes HierarchTrust, a novel trust model that relies on the standard deviation in its trust calculations. HierarchTrust determines trust based on values assigned to various factors, including trust, reputation, feedback and recommendations, which are checked in order of their reliability as measures of trust. At the top of the trust hierarchy is direct experience, followed by recommendations and feedback, and finally reputation. There is more confidence in the elements at the top of the hierarchical form, and confidence in the trust ranking decreases as we move away from the top of the hierarchy.

The remainder of this paper is organised as follows. Section 2 describes the trust model in detail. Section 3 explains the methodology that has been followed in the experiments, while Section 4 provides the result and a discussion of the experiments. Finally, Section 5 presents the conclusions.

2 System design

In a P2P network, a file requester sends file requests to its neighbouring peers. After receiving the requested file, the peer gives a rating to the provider peer based on the authenticity of the file received. Trust value, reputation, feedback and recommendations are computationally derived from the positive and negative ratings given to each peer by their peers. The proposed model then determines the trustworthiness of the file providers in the P2P network. The model applies the standard deviation in its computation of trust in all the trust models used in HierarchTrust. Fig. 1 illustrates the abstract architecture of the HierarchTrust algorithm.

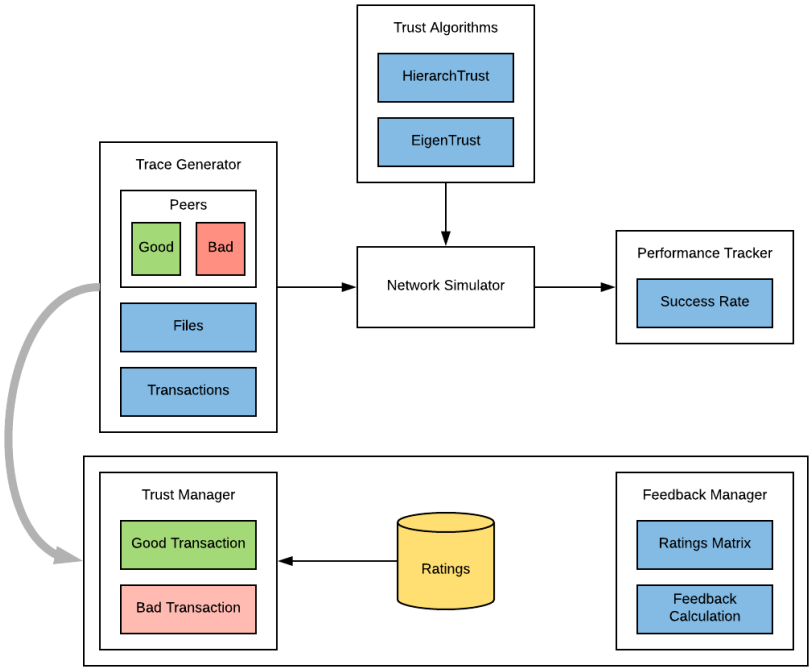

Figure 1: Abstract system model

The basic components of the proposed system include transactions, files and provider peers. The transactions component maintains all the transactions in the P2P network. The file component stores the files in the network and is utilised frequently by the transactions component and by peers. Each provider peer in the network houses the following components:

I. Trust manager: The trust manager is responsible for calculating trust after each transaction. It consists of two main components: good transactions and bad transactions, where the calculation of trust depends based on the file received. The trust manger receives trust ratings from the ratings database.

II. Ratings database: This database maintains a record of the ratings for each transaction.

III. Feedback manager: The feedback manager is comprised of two components: a rating matrix component, which records received ratings in a matrix, and a feedback calculation component, which calculates the level of trust in the feedback based on the ratings matrix.

HierarchTrust is based on four trust models, which increases the possibility of choosing a trusted peer from among all the peers in a P2P network. The trust computation model is comprised of the following elements:

I. A trust value computed based on a peer's direct experience.

II. A trust value computed based on recommendations from a peer's friends.

III. A trust value computed from the feedback ratings received from other peers who have transacted with the peer in the past about another peer in the network.

IV. A trust value computed based on the reputation ratings provided by each peer after each transaction.

The trust metric is computed as in Eq. (1):

$$Trust = TV + REC + FB + REP \tag{1}$$

where *TV* is the trust value, *REC* is the recommendation received from the peer's friends, *FB* is the feedback rating and *REP* is the reputation of the provider.

The trust value (*TV*) is computed after each transaction using the standard deviation based on the positive and negative ratings received. After a good or a bad transaction, respectively, the trust value is calculated using Eq. (2) or Eq. (3):

$$TV = \frac{NT}{1 + (NT - OT)} \tag{2}$$

or

$$TV = OT - (0.15 \times OT) \tag{3}$$

where *NT* is the new trust value, which is computed as $NT = 1 - Standard\ deviation$, and *OT* is the old trust value.

The recommendation *REC* is computed at each transaction when the *TV* is unavailable. The *REC* is computed as follows:

$$REC = R_1 \times R_2 \times R_3 \tag{4}$$

where R_1 is the first recommender and R_2 is the best friend of the first recommender if the first recommender does not have enough experience with the provider and so on.

The feedback (*FB*) is calculated for each transaction if neither *TV* nor *REC* are available. *FB* relies on the positive and negative ratings received for each transaction. Finally, if *TV*, *REC* and *FB* are unavailable, the reputation (*REP*) is computed for each transaction.

3 Methodology

The proposed trust management system is used to assess the trustworthiness of peers in P2P networks. We hypothesise that utilising the proposed algorithm, HierarchTrust, will increase the success rate of good peers. To prove this hypothesis, the open source P2P trust simulator QTM was used to simulate a file-sharing P2P network [QTM]. This is similar to the approach taken in previous studies [Kurdi, Alshayban, Altoaimy et al. (2018); Bursell (2005); Mondal and Kitsuregawa (2006); Shala, Wacht, Trick et al. (2017); Zhao and Li (2013); Lu, Wang, Xie et al. (2016)]. The simulations were conducted on a system with an Intel Core i7 processor, 1.80 GHz speed, and 8 GB RAM running Windows 10 Home (68 bit). Software tools include NetBeans IDE, Visual C++ 2010 Express and JGRASP version 1.8.8_23.

The performance of the proposed HierarchTrust algorithm is assessed comparatively against the well-established EigenTrust algorithm [Kamvar, Schlosser and Garcia-Molina (2003)]. Success rate was utilised as a performance measure and was computed as follows:

$$Success\ rate = \frac{Number\ of\ authentic\ files\ received\ by\ good\ peers}{Numbers\ of\ transactions\ established\ by\ good\ peers} \quad (5)$$

The number of malicious peers was controlled in the evaluation to provide a representative sample of a P2P environment. The number of transactions was stabilised at 400 transactions, and the number of malicious peers was varied as 10, 20, 30, 40, 50, 60 and 70 peers. Two models of attack were simulated: pure attacks and feedback-based attacks utilising both naïve and collective strategies. The total number of peers was set at 300. The network also maintained a constant number of files, in this case, 2000 files.

4 Results and discussion

The following results represent a practical application of the previously described methodology. The experimental setup considers two different types of attacks, pure attacks and feedback-based attacks, utilising naïve and collective strategies, as shown in Figs. 2, 3, 4, and 5, respectively.

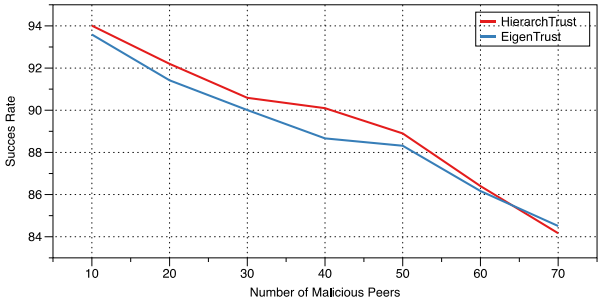

Figure 2: The success rate of good peers with a varying number of pure attacks from malicious peers and a naïve strategy

Figs. 2 and 3 illustrate the success rate of good peers under both the HierarchTrust algorithm and the EigenTrust algorithm as the number of malicious peers is increased from 10 to 70. The attacks shown in these figures are pure attacks that utilise either naïve or collective strategies. HierarchTrust displays a better success rate than EigenTrust, even as the number of malicious peers is increased. Similar to EigenTrust, the success rate of good peers with HierarchTrust decreases as the number of malicious peers increases.

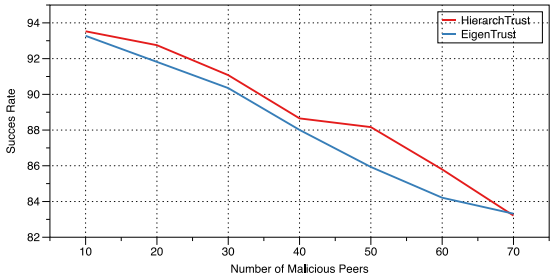

Figure 3: The success rate of good peers with a varying number of pure attacks from malicious peers and a collective strategy

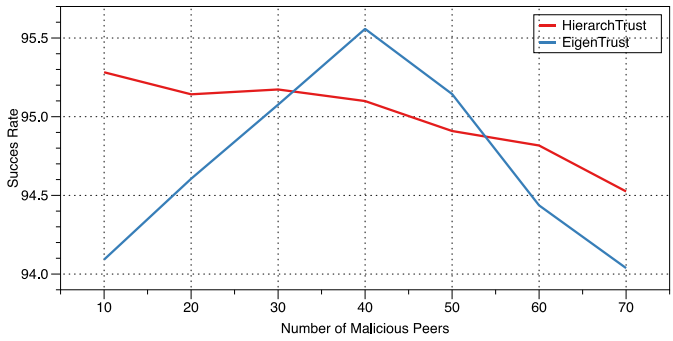

Figure 4: The success rate of good peers with a varying number of feedback attacks from malicious peers and a naïve strategy

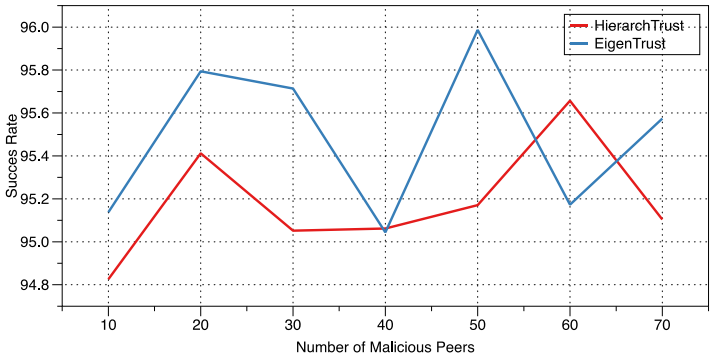

Figure 5: The success rate of good peers with a varying number of feedback attacks from malicious peers and a collective strategy

Figs. 4 and 5 demonstrate the success rate of good peers under feedback-based attacks from a varying number of naïve and collective malicious peers using the proposed algorithm and the EigenTrust algorithm. The figures clearly show the stability of HierarchTrust compared to the well-established EigenTrust approach. In fact, under the naïve strategy (see Fig. 4), the proposed algorithm outperforms EigenTrust for more than half of the malicious attacks shown. However, when a collective attack strategy was used, EigenTrust outperforms HierarchTrust (see Fig. 5).

5 Conclusion

P2P networks are beneficial as a method for resource sharing, but the inherent openness of these systems makes them an ideal space in which malicious peers can prosper. Therefore, numerous trust and reputation models have been proposed to tackle this problem. This paper proposes HierarchTrust, a trust management system that computes trust based on standard deviation using various trust values (trust, reputation, feedback and recommendations), which are computationally derived from positive and negative ratings given to each peer by their peers. The performance of HierarchTrust was comparatively assessed in a simulated environment against EigenTrust to determine the success rate of

good peers as the number of malicious peers was varied. The results show the superiority of the proposed algorithm under pure attacks in comparison to EigenTrust. HierarchTrust also proved stable under feedback-based attacks utilising naïve strategies, but EigenTrust proved stronger against collective strategies. This limitation lends itself to future work. Several other types of threat will also be considered in the future.

Acknowledgement: The research was supported by a grant from the research Center of the Center for Female Scientific and Medical Colleges.

References

Androutsellis-Theotokis, S.; Spinellis, D. (2004): A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys*, vol. 36, no. 4, pp. 335-371.

Bhise, A. M.; Kamble, S. D. (2016): Detection and mitigation of sybil attack in peer-to-peer network. *International Journal of Computer Network and Information Security*, vol. 8, no. 9, pp. 56-63.

Bursell, M. (2005): Security and trust in P2P systems. *Peer-to-Peer Computing: the Evolution of a Disruptive Technology*, pp. 145-165.

Kamvar, S. D.; Schlosser, M. T.; Garcia-Molina, H. (2003): The eigentrust algorithm for reputation management in P2P networks. *Proceedings of the 12th international conference on World Wide Web*, pp. 640-651.

Kurdi, H.; Alshayban, B.; Altoaimy, L.; Alsalamah, S. (2018): Trustyfeer: a subjective logic trust model for smart city peer-to-peer federated clouds. *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1-13.

Kurdi, H. (2015): Honestpeer: an enhanced eigentrust algorithm for reputation management in P2P systems. *Journal of King Saud University-Computer and Information Sciences*, vol. 27, no. 3, pp. 315-322.

Lu, K.; Wang, J.; Xie, L.; Zhen, Q.; Li, M. (2016): An eigentrust-based hybrid trust model in P2P file sharing networks. *Procedia Computer Science*, vol. 94, no. 2016, pp. 366-371.

Mondal, A.; Kitsuregawa, M. (2006): Privacy, security and trust in P2P environments: a perspective. *Proceedings of the 17th International Workshop on Database and Expert Systems Applications*, pp. 682-686.

QTM (2018): P2P trust simulator. https://rtg.cis.upenn.edu/qtm/p2psim.php3.

Shala, B.; Wacht, P.; Trick, U.; Lehmann, A.; Ghita, B.; Shiaeles, S. (2017): Ensuring trustworthiness for p2p-based m2m applications. *Proceedings of the Internet Technologies and Applications*, pp. 58-63.

Xie, X.; Yuan, T.; Zhou, X.; Cheng, X. (2018): Research on trust model in container-based cloud service. *Computers, Materials & Continua*, vol. 56, no. 2, pp. 273-283.

Xiong, L.; Liu, L. (2004): Peertrust: supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843-857.

Yang, L.; Qin, Z.; Wang, C.; Liu, Y.; Feng, C. (2010): A P2P reputation model based on ant colony algorithm. *Proceedings of the International Conference on Communications,*

Circuits and Systems, pp. 236-240.

Zhang, Y.; Zheng, H.; Liu, Y.; Li, K.; Qu, W. (2011): A grouptrust model based on service similarity evaluation in P2P networks. *International Journal of Intelligent Systems*, vol. 26, no. 1, pp. 47-62.

Zhao, H.; Li, X. (2013): Vectortrust: trust vector aggregation scheme for trust management in peer-to-peer networks. *Journal of Supercomputing*, vol. 64, no. 3, pp. 805-829.