Result and Analysis: Data Sharing between Peer-to-Peer using Trust Model

Janmejay Kale
Student, PG Department, MBES
College of Engineering, Ambajogai

V. R. Chirchi
Asst.Professor,
PG Department, MBES College of
Engineering,Ambajogai

ABSTRACT

In this implemented project, using open nature of Peer to Peer systems that helps to expose the malicious activity. Building trust relationships among peers can reduce attacks of malicious peers. Peers create its own trust network in their proximity by using local information available and do not try to learn global trust information. Based on trust information it classifies the peers whether peer is trustworthy or not. In this paper used the technique called Self Organizing Trust Model (SORT) that aims to reduce malicious activity in Peer to Peer system by establishing trust relations among peers in their proximity. Trust information is evaluated based on service, trust values of each peers and it is based on past interactions. Which one peer having highest trust ratio that is computed using service and trust values of earlier interaction that peer to be selected for next interaction. This trust information helps to build a secure environment to transmit a packet. Simulation experiments on a file sharing application show that the proposed model can mitigate attacks on different malicious behavior models. In the experiments, good peers were able to form trust relationships in their proximity and isolate malicious peers.

Keywords

Peer to Peer system, Trust Management, Security, Establishing Trust Information, Past Interaction

1. INTRODUCTION

Systems work on collaboration of peers to accomplish tasks. Peer to peer system contain both type of peers like good peers and also malicious peers. We need to classify the both type of peers by creating long-term relationships among peers. Peers can provide a more secure environment by reducing risk and helps in future peer to peer interactions. However, establishing trust in an unknown peer is difficult in such a malicious environment. Furthermore, trust is a social concept and hard to measure with numerical values. Classifying peers as either trustworthy or untrustworthy is not sufficient in most cases. Metrics should have precision so peers can be ranked according to trustworthiness. Interactions and feedbacks of peers provide information to measure trust among peers. Interactions with a peer provide certain information about the peer but feedbacks might contain deceptive information. In the existing system, a central server is used to store and manage trust information, for example, ebay. The central server securely stores trust information and defines trust metrics but lot of problems could happen. Since there is no central server in most peer to peer systems, peers organize themselves to store and manage trust information about [1][2].

Management of trust information is dependent to the structure of peer to peer network. In distributed hash table (DHT)-based approaches, each peer becomes a trust holder by storing feedbacks about other peers [1],[3],[4]. In unstructured networks, each peer stored trust information about peers in its neighborhood or peers interacted in the past [2],[5],[6]. A peer sends trust queries to learn trust information of other peers. A trust query is either flooded to the network or sent to neighborhood of the query initiator. Generally, computing trust information is not global and does not reflect opinions of all peers. In this implemented system, using the technique called Self-Organizing Trust Model (SORT) that aims to reduce malicious activity in a peer to peer system by establishing trust relations among peers in their proximity. No a priori information or a trusted peer is used to leverage trust establishment. Peers do not try to collect trust information from all peers. Each peer develops its own local view of trust about the peers interacted in the past. In this way, good peers form trust groups in their proximity and can isolate malicious peers. Since peers generally tend to interact with small set of peers [7], forming trust relations in proximity of peers helps to mitigate attacks in a peer to peer system.

In SORT, peers are assumed to be strangers to each other at the beginning. A peer becomes an acquaintance of another peer after providing a service, example, uploading a file. If a peer has no acquaintance, it chooses to trust strangers. An acquaintance is always preferred over a stranger if they are equally trustworthy. Using a service of a peer is an interaction, which is evaluated based on time and bandwidth of the interaction, and satisfaction of the requester. SORT defines three trust metrics. Reputation metric is calculated based on recommendations. It is important when deciding about strangers and new acquaintance. Reputation loses its importance as experience with an acquaintance increases. Service trust and recommendation trust are primary metrics to metrics to measure trustworthiness in the service and recommendation contexts, respectively. The service trust metric is used when selecting service providers. The recommendation trust metric is important when requesting recommendations. When calculating the reputation metric, recommendations are evaluated based on the recommendation trust metric. SORT helps to reduce the malicious activity in the peer to peer network by building trust relationships among peers. It helps to form secure environment to transmit the packet and only good peers have interactions with each other. We implemented a peer to peer file sharing simulation tool and conducted experiments to understand impact of SORT in mitigating attacks. Parameters related to peer capabilities (bandwidth, number of shared files), peer behavior (online/offline periods, waiting time for sessions), and resource distribution are approximated to several empirical results [8],[9],[10]. This enabled us to make more realistic observations on evolution of trust relationships.

2. RELATED WORK

2.1 Existing System

In an existing system, establishing trust in an unknown entity is difficult in such a malicious environment. Furthermore, trust is a social concept and hard to measure with numerical values. Metrics are needed to represent trust in computational models. Classifying peers as either trustworthy or untrustworthy is not sufficient in most cases. Metrics should have precision so peers can be ranked according to trustworthiness. Interactions and feedbacks of peers provide certain information about the peer but feedbacks might contain deceptive information. This makes assessment of trustworthiness a challenge. The main problem with existing system is centralized server it is used to store and manage the information about peers. Every time peer need to ask server for which peer is to be selected for next interaction so it takes lot of time and bandwidth wastage. If server got failure then all the information about the peers could be lost.

2.1.1 Drawbacks in Existing System:

- A trust query is either flooded to the network or sent to neighborhood of the query initiator.
- Calculated trust information is not global and does not reflect opinions of all peers.
- Time consuming process.
- More bandwidth usage.

2.2 . Proposed System

To solve the problems with existing system by using the technique called Self-Organizing Trust Model that aims to reduce malicious activity in a peer to peer system by establishing trust relations among peers in their proximity. Each peer stores and manage its own database and contain trust information about other peers. Peers keeps trust information based on time, bandwidth and parameter satisfaction.

2.2.1 Advantages in Proposed System

- SORT, instead of considering a particular trust holder’s feedback as authentic.
- Improving evaluate interactions and recommendations.
- FloodRQ methods can be helpful to identify some attackers before an attack.
- Less bandwidth usage.
- Less time consuming.

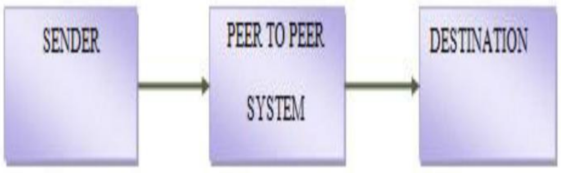


Fig.1. Level 0 Architectural

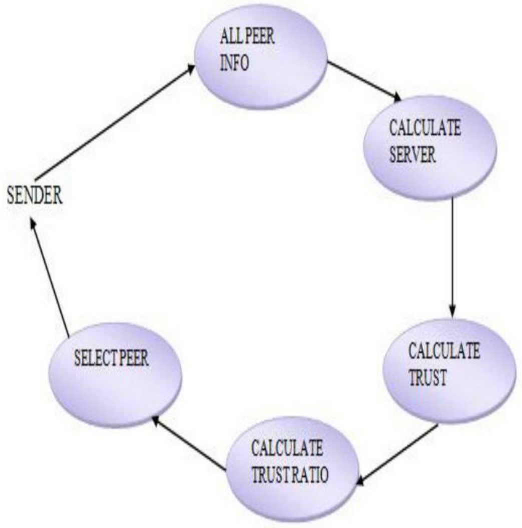


Fig.2. Level 1 Architectural

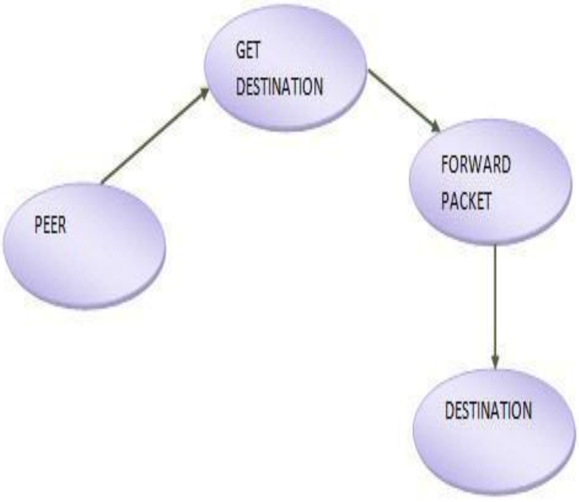


Fig.3. Level 2 Architectural Diagram

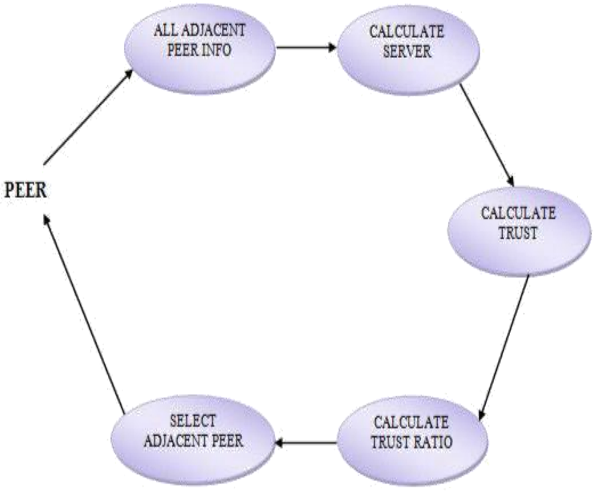


Fig.4. Level 3 Architectural Diagram.

Figs. 1, 2, 3 and 4 shows system architecture of each levels. Each level diagrams defines step by step procedure to execute the implemented work.

3. EXPERIMENTAL RESULTS

Methods There are four different cases are studied to understand effects of trust calculation methods under attack conditions.

No Trust: Trust information is not used for up loader selection. An up loader is selected according to its bandwidth. This method is the base case to understand if trust is helpful to mitigate attacks.

No Reputation Query: An up loader is selected based on trust information but peers do not request recommendations from other peers. Trust calculation is done based on SORT equations but reputation value is always zero for a peer. This method is will helps us to assess if recommendations are helpful.

Flood Reputation Query: Sort equations are used but a reputation query is flooded to the whole network. This method will help us to understand if getting more recommendations is helpful to mitigate attacks.

3.1. Attacker Models

Attackers can perform service-based and recommendation-based attacks. Uploading a virus infected or an inauthentic file is a service-based attack. Giving a misleading recommendation intentionally is a recommendation-based attack. There are two types of misleading recommendations.

- Unfairly high
- Unfairly low

A good peer uploads authentic files and gives fair recommendations. A malicious peer (attacker) performs both service and recommendation-based attacks. Four different attack behaviors are studied for malicious peers: Naive, discriminatory, hypocritical, and oscillatory behaviors. A non malicious network consists of only good peers. A malicious network contains both good and malicious peers. If malicious peers do not know about each other and perform attacks independently, they are called as individual attackers. Individual attackers may attack each other. For individual attackers, attack behaviors are as follows:

Naive: The attacker always uploads infected/inauthentic files and gives unfairly low recommendations about others.

Discriminatory: The attacker selects a group of victims and always uploads infected/inauthentic files to them. It gives unfairly low recommendations about victims.

For other peers, it behaves as a good peer.

Hypocritical: The attacker uploads infected/inauthentic files and gives unfairly low recommendations with x percent probability [3][5]. In the other times, it behaves as a good peer.

Oscillatory: The attacker builds a high reputation by being good for a long time period. Then, it behaves as a naïve attacker for a short period of time. After the malicious period, it becomes a good peer again.

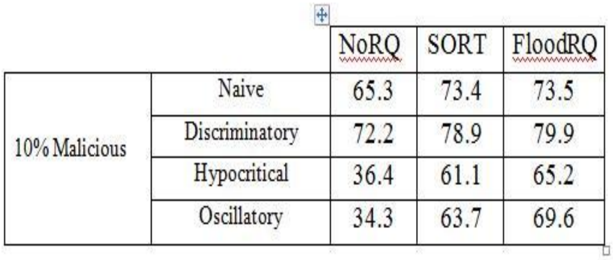
3.2 Analysis on Individual Attackers

This section explains the results of experiments on individual attackers. For each type of individual attacker, creating the network topology that is 10 percent malicious. This network topology is tested with four trust calculation methods. In the experiments, a hypocritical attacker behaves malicious in 20 percent of all interactions. A discriminatory attacker selects 10 percent of all peers as victims. An oscillatory attacker behaves good for 1000 cycles and malicious for 100 cycles.

3.3 Service-Based Attacks

Table 1 shows the percentage of Service-based attacks prevented by each trust calculation method. When a malicious peer uploads an infected/inauthentic file, it is recorded as a service-based attack. Number of attacks in No trust method is considered as the base case to understand how many attacks can happen without using trust information. Then, number of attacks observed for each trust calculation method is compared with the base case to determine the percentage of attacks prevented. In the table, NoRQ and FloodRQ denote "No reputation query" and "flood reputation query" methods, respectively. In a 10 percent malicious network, all methods can prevent more than 60 percent of attacks of naïve attackers. NoRQ method's performance is close to other methods since a good peer identifies a naïve attacker after having the first interaction. Thus, recommendations are not very helpful in the naïve case. For discriminatory attackers, the situation is similar since their naïve attacks easily reveal their identity to victims. For the hypocritical and oscillatory attackers, a good peer may not identify an attacker in the first interaction. Therefore, recommendations in SORT and FloodRQ methods can be helpful to identify some attackers before an attack happens and graphical result as shown in Figs.5 to 8.

Table 1: Percentage of Service-Based Attacks Prevented For Individual Attackers



		NoRQ	SORT	FloodRQ
10% Malicious	Naive	65.3	73.4	73.5
	Discriminatory	72.2	78.9	79.9
	Hypocritical	36.4	61.1	65.2
	Oscillatory	34.3	63.7	69.6

Recommendation-based attacks are not considered because recommendations may provide deceptive recommendation about other peers. Shows actual performances of each service. Here, it compares different level of service's result. Each transaction computes trust ratios of each peer before taking transaction. System will select bandwidth free service based on these results. Sort technique provide better results when compare to No Reputation Query. It takes less time to forward the data to destination when compare to other two services. Here assumed specific values of service and trust values is 5 and 10 respectively.

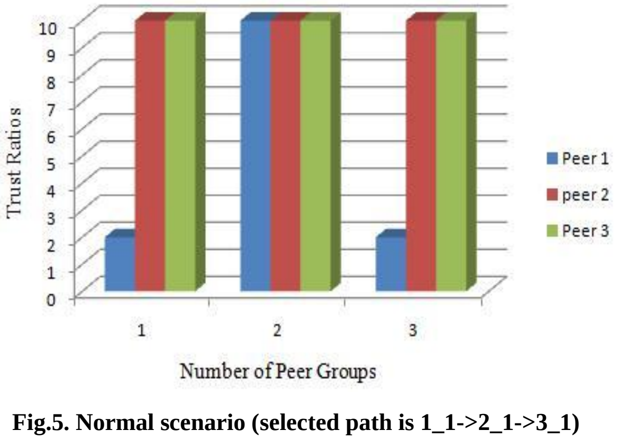


Fig.5. Normal scenario (selected path is 1_1->2_1->3_1)

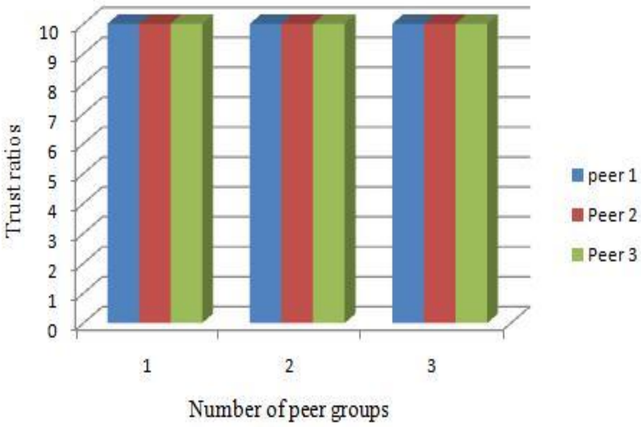


Fig.6. Congestion scenario (selected path is 1_2->2_1->3_2)

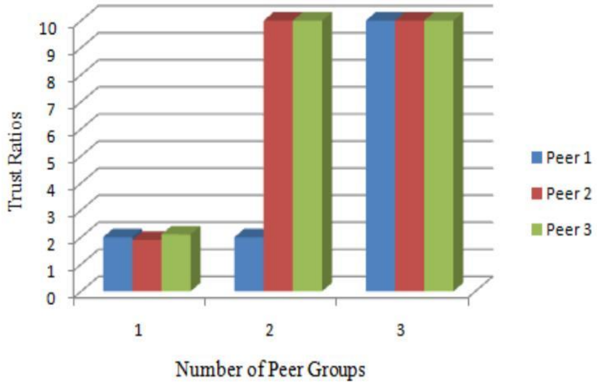


Fig.7. Bandwidth free scenario

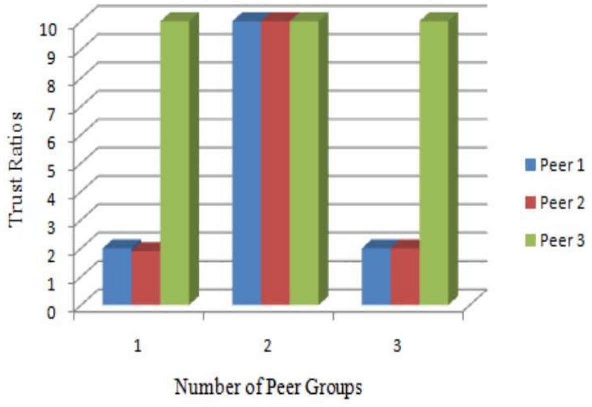


Fig.8. Next transaction. (selected path 1_3->2_1->3_3).

4. CONCLUSION

A trust model for peer to peer network is presented, in which a peer can develop a trust network in its proximity. A peer can isolate malicious peers around itself as it develops trust relationships with good peers. Two context of trust, service and recommendations contexts are defined to measure capabilities of peers in providing services and giving recommendations. Interactions and recommendations are considered with satisfaction, time and bandwidth. This implemented work provided better security for peer to peer system. This system provides better result compare to earlier methods. In future using trust information does not solve all security problems in P2P systems but can enhance security and effectiveness of systems. If interactions are modeled correctly, SORT can be adapted to various P2P applications.

5. REFERENCES

[1] A.B.Can and B.Bhargava,"SORT: Self ORganizing Trust Model for Peer-to-Peer Systems," IEEE Transactions on Dependable and Secure Computing,vol.10,no.1,pp.14-27,Jan.2013

[2] K. Abere and Z. Despotovic, "Managing Trust in a Peer to peer Information System," Proc.10th Int'l Conf.Information and Knoewledge Management (CIKM), 2001.

[3] F. Cornelli, E. Damiani, S.D.C di Vimercati,S. Paraboschi, and P.Samarati, "Choosing Reputable Servents in a Peer to peer Network," Proc. 11th World Wide Web Conf.(WWW),2002.

[4] S. Kamvar, M.Schlosser, and H. Garcia-Molina, "The (Eigentrust) Algorithm for Reputation Management in Peer to peer Network," Proc. 12th world Wide Web Conf.(WWW),2003.

[5] L.Xiong and L.Liu, "Peertrust: Supporting Reputation-Based Trust for Peer to Peer Ecommerce Communities," IEEE Trans. Knowledge and Data Eng.,Vol. 16, no. 7, pp.843-857, July 2004.

[6] A.A. Seluk, E. Uzun, and M.R. Pariente, "A Reputation-Based trust Management System for Peer to peer Network," Proc. IEEE/ACM Fourth Int'l Symp. Cluster Computing and the Grid(CCGRID), 2004.

[7] R. Zhou, K. Hwang, and M. Cai, "Gossiptrust for Fast Reputation Aggregation in Peer to peer Network," IEEE Trans. Knowledge and Data Eng., vol.20,no.pp. 1282-1295,Sept.2008.

[8] J . Kleingberg, "The Small-World Phenomenon: An Algorithmic Perspective," Proc. 32nd ACM Symp. Theory of Computing, 2000.

[9] S . Saroiu, P. Gummadi, and S. Gribble, "A Measurement Study of peer to peer File sharing Systems,"Proc.MultimediaComputingand Networking,2002.

[10] M . Ripeanu, I.Foster, and A. Iamnitchi, "mapping the Gnutella Network: Properties of Large- Scale Peer to Peer Systems and Implications for System Design," IEEE Internet Computing, vol. 6, no. 1,pp. 50-57 Jan 2002.

[11] S. Saroiu, K. Gummadi, R. Dunn, S.D.Gribble, and H.M. Levy, " An Analysis of internet Conent Delivery Systems," Proc. Fifth USENIX Symp. Operating Systems Design and Implemenation(OSDI),2002.