ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 1, January 2014

A Trust Relation Protocol in Peer-to-Peer Network

A.Vijay Vasanth[1], M.Sreenivasan[2], Shibin Babu[3], B.Sathish[4]

Senior Assistant Professor, Christ College of Engineering and Technology, Puducherry[1]

UG Student, Christ College of Engineering and Technology, Puducherry[2]

UG Student, Christ College of Engineering and Technology, Puducherry[3]

UG Student, Christ College of Engineering and Technology, Puducherry[4]

Abstract: Open nature of peer-to-peer systems exposes them to malicious activity. Building trust relationships among peers can mitigate attacks of malicious peers. This paper presents distributed algorithms that enable a peer to reason about trustworthiness of other peers based on past interactions and recommendations. Peers create their own trust network in their proximity by using local information available and do not try to learn global trust information. Two contexts of trust, service, and recommendation contexts are defined to measure trustworthiness in providing services and giving recommendations. Interactions and recommendations are evaluated based on importance, recentness, and peer satisfaction parameters. Additionally, recommender's trustworthiness and confidence about recommendation are considered while evaluating recommendations. Simulation experiments on a file sharing application show that the proposed model can mitigate attacks on 16 different malicious behavior models. In the experiments, good peers were able to form trust relationships in their proximity and isolate malicious peers.

Index Terms- P2P, TRP, DHT, DFD, Bubble Chart

1. INTRODUCTION

PEER-TO-PEER (P2P) systems rely on collaboration of peers to accomplish tasks. Ease of performing malicious activity is a threat for security of P2P systems. Creating long-term trust relationships among peers can provide a more secure environment by reducing risk and uncertainty in future P2P interactions. However, 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 information to measure trust among peers. Interactions with a peer provide certain information about the peer but feedbacks might contain deceptive information. This makes assessment of trustworthiness a challenge. In the presence of an authority, a central server is a preferred way to store and manage trust information, e.g., eBay. The central server securely stores trust information and defines trust metrics. Since there is no central server in most P2P systems, peers organize themselves to store and manage trust information about each other. Management of trust information is dependent to the structure of P2P network. In distributed hash table (DHT)-based approaches, each peer becomes a trust holder by storing feedbacks about other peers. Global trust information stored by trust holders can be accessed through DHT efficiently. In unstructured networks, each peer stores trust information about peers in its neighborhood or peers interacted in the past. 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, calculated trust information is not global and does not reflect opinions of all peers. We propose a Trust Relation Protocol (TRP) that aims to decrease malicious activity in a P2P 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 dynamic trust groups in their proximity and can isolate malicious peers. Since peers generally tend to interact with a small set of peers, forming trust relations in proximity of peers helps to mitigate attacks in a P2P system. In TRP, peers are assumed to be strangers to each other at the beginning. A peer becomes an acquaintance of another peer after providing a service, e.g., 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 weight (importance) and recentness of the interaction, and satisfaction of the requester. An acquaintance's feedback about a peer, recommendation, is evaluated based on recommender's trustworthiness. It contains the recommender's own experience about the peer, information collected from the recommender's acquaintances, and the recommender's level of confidence in the recommendation.

If the level of confidence is low, the recommendation has a low value in evaluation and affects less the trustworthiness of the recommender.

2. EXISTING SYSTEM

In the existing system of an authority, a central server is a preferred way to store and manage trust information, e.g., eBay. The central server securely stores trust information and defines trust metrics. Since there is no central server in most P2P systems, peers organize themselves to store and manage trust information about each other. Management of trust information is dependent to the structure of P2P network. In distributed hash table (DHT) - based approaches, each peer becomes a trust holder by storing feedbacks about other peers. Global trust information stored by trust holders can be accessed through DHT efficiently. In unstructured networks, each peer stores trust information about peers in its neighborhood or peers interacted in the past. 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.

2.1 Drawbacks

➢ Calculated trust information is not global and does not reflect opinions of all peers.
➢ 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.
➢ Trust models on P2P systems have extra challenges comparing to e-commerce platforms. Malicious peers have more attack opportunities in P2P trust models due to lack of a central authority
➢ Five common attacks in P2P trust models: self-promoting, white-washing, slandering, orchestrated, and denial of service attacks.

3. PROPOSED SYSTEM

In the proposed system, we introduce a Trust Relation Protocol (TRP) that aims to decrease malicious activity in a P2P 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 dynamic trust groups in their proximity and can isolate malicious peers. Since peers generally tend to interact with a small set of peers forming trust relations in proximity of peers helps to mitigate attacks in a P2P system.

3.1 Advantages

✓ Recommendation-based attacks were contained except when malicious peers are in large numbers, e.g., 50 percent of all peers.
✓ Experiments on TRP show that good peers can defend themselves against malicious peers metrics let a peer assess trustworthiness of other peers based on local information.
✓ Service and recommendation contexts enable better measurement of trustworthiness in providing services and giving recommendations.

4. INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

➢ What data should be given as input?
➢ How the data should be arranged or coded?
➢ The dialog to guide the operating personnel in providing input.
➢ Methods for preparing input validations and steps to follow when error occur.

4.1 Objectives

1.Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

5. OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

5.1 Objectives

❖ Convey information about past activities, current status or projections of the Future.
❖ Signal important events, opportunities, problems, or warnings.
❖ Trigger an action.
❖ Confirm an action.

6. MODULES

- Peer Creation
- Upload Process
- Interaction Process
- Recommendation Model

6.1 Peer Creation

In this module, we create three peers. In TRP, peers are assumed to be strangers to each other at the beginning. A peer becomes an acquaintance of another peer after providing a service, e.g., 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. We implemented a P2P file sharing simulation tool and conducted experiments to understand impact of TRP

6.2 Upload Process

In this module, we design each peer can upload file and its updated to all the peers. The details of each file with their file name, up-loader name with their IP address are stored continuously. So the peer which needs the file can download it.

6.3 Interaction Process

In this module, we create the interaction process between each peers. The peer which wants the file cannot download it without requesting permission from the uploaded. The peer will request to the uploader with the full details, such as filename etc. The request will be received to the uploader and then its processes. If the uploader sends the file, then only the peer can download it. With the uploader permission, the peer cannot download it. In this way the peer interaction process module takes place

6.4 Recommendation Model

In this module, the recommendation is made to the other peers regarding the service or uploader. A peer may be a good service provider but a bad recommender or vice versa. Thus, TRP considers providing services and giving recommendations as different tasks and defines two contexts of trust: service and recommendation contexts. Information about past interactions and recommendations are stored in separate histories to assess competence and integrity of acquaintances in these contexts.

Using a service of a peer is an interaction, which is evaluated based on weight (importance) and recentness of the interaction, and satisfaction of the requester. An acquaintance's feedback about a peer, recommendation, is evaluated based on recommender's trustworthiness. It contains the recommender's own experience about the peer, information collected from the recommender's acquaintances, and the recommender's level of confidence in the recommendation.

7. SYSTEM DESIGN

7.1 System Architecture

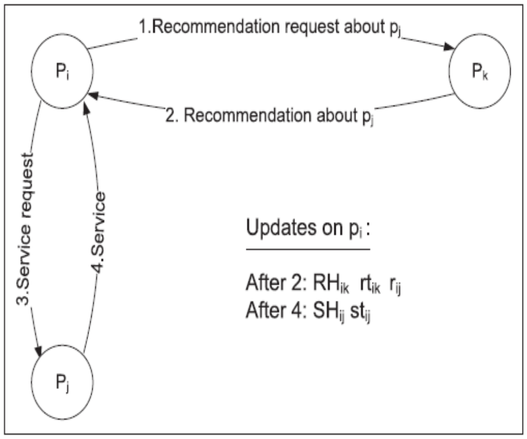
1.Recommendation request about p_j
P_i
P_k
2. Recommendation about p_j
3.Service request
4.Service
Updates on p_i :
After 2: RH_{ik} rt_{ik} r_{ij}
After 4: SH_{ij} st_{ij}
P_j

Fig 1 System Architecture

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

7.2 Block Diagram

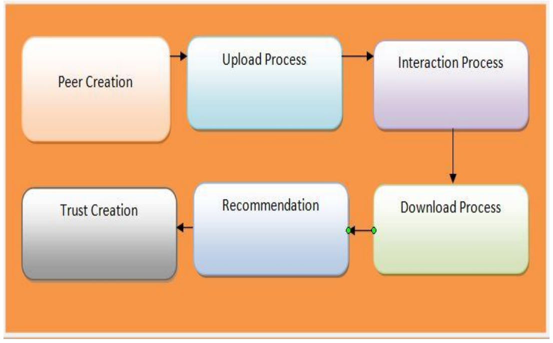

Fig 2 Block Diagram

7.3 Data Flow Diagram

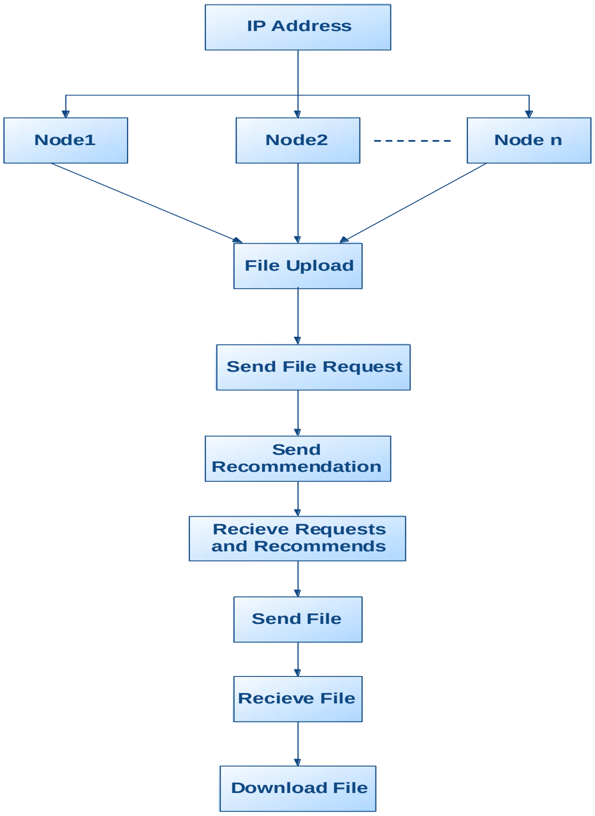

Fig 3 Data Flow Diagram

1. The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

3. DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.

4. DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.

CONCLUSION

A trust model for P2P networks 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 recommendation contexts are defined to measure capabilities of peers in providing services and giving recommendations. Interactions and recommendations are considered with satisfaction, weight, and fading effect parameters. A recommendation contains the recommender’s own experience, information from its acquaintances, and level of confidence in the recommendation. These parameters provided us a better assessment of trustworthiness. Individual, collaborative, and pseudonym changing attackers are studied in the experiments. Damage of collaboration and pseudospoofing is dependent to attack behavior. Although recommendations are important in hypocritical and oscillatory attackers, pseudospoofers, and collaborators, they are less useful in naive and discriminatory attackers. TRP mitigated both service and recommendation-based attacks in most experiments. However, in extremely malicious environments such as a 50 percent malicious network, collaborators can continue to disseminate large amount of misleading recommendations. Another issue about TRP is maintaining trust all over the network. If a peer changes its point of attachment to the network, it might lose a part of its trust network. These issues might be studied as a future work to extend the trust model. 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, TRP can be adapted to various P2P applications, e.g., CPU sharing, storage networks, and P2P gaming. Defining application specific context of trust and related metrics can help to assess trustworthiness in various tasks.

REFERENCES

[1] K. Aberer and Z. Despotovic, “Managing Trust in a Peer-2-Peer Information System,” Proc. 10th Int’l Conf. Information and Knowledge Management (CIKM), 2001.

[2] F. Cornelli, E. Damiani, S.D.C. di Vimercati, S. Paraboschi, and P. Samarati, “Choosing Reputable Servents in a P2P Network,” Proc. 11th World Wide Web Conf. (WWW), 2002.

[3] S. Kamvar, M. Schlosser, and H. Garcia-Molina, "The (Eigentrust) Algorithm for Reputation Management in P2P Networks," Proc. 12th World Wide Web Conf. (WWW), 2003.

[4] 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.

[5] A.A. Selcuk, E. Uzun, and M.R. Pariente, "A Reputation-Based Trust Management System for P2P Networks," Proc. IEEE/ACM Fourth Int'l Symp. Cluster Computing and the Grid (CCGRID), 2004.

[6] R. Zhou, K. Hwang, and M. Cai, "Gossiptrust for Fast Reputation Aggregation in Peer-to-Peer Networks," IEEE Trans. Knowledge and Data Eng., vol. 20, no. 9, pp. 1282-1295, Sept. 2008.

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

[8] S. Saroiu, P. Gummadi, and S. Gribble, "A Measurement Study of Peer-to-Peer File Sharing Systems," Proc. Multimedia Computing and Networking, 2002.

[9] 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.

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

[11] S. Marsh, "Formalising Trust as a Computational Concept," PhD thesis, Dept. of Math. and Computer Science, Univ. of Stirling, 1994.

[12] A. Abdul-Rahman and S. Hailes, "Supporting Trust in Virtual Communities," Proc. 33rd Hawaii Int'l Conf. System Sciences (HICSS), 2000.

[13] B. Yu and M. Singh, "A Social Mechanism of Reputation Management in Electronic Communities," Proc. Cooperative Information Agents (CIA), 2000.

[14] L. Mui, M. Mohtashemi, and A. Halberstadt, "A Computational Model of Trust and Reputation for E-Businesses," Proc. 35th Hawaii Int'l Conf. System Sciences (HICSS), 2002.

[15] A. Jøsang, E. Gray, and M. Kinateder, "Analysing Topologies of Transitive Trust," Proc. First Int'l Workshop Formal Aspects in Security and Trust (FAST), 2003.

[16] E. Terzi, Y. Zhong, B. Bhargava, Pankaj, and S. Madria, "An Algorithm for Building User-Role Profiles in a Trust Environment," Proc. Fourth Int'l Conf. Data Warehousing and Knowledge Discovery (DaWaK), vol. 2454, 2002.

[17] Y. Zhong, "Formalization of Dynamic Trust and Uncertain Evidence for User Authorization," PhD thesis, Dept. of Computer Science, Purdue Univ., 2004.

[18] A. Jøsang, R. Ismail, and C. Boyd, "A Survey of Trust and Reputation Systems for Online Service Provision," Decision Support Systems, vol. 43, no. 2, pp. 618-644, 2007.

[19] H. Yu, M. Kaminsky, P.B. Gibbons, and A. Flaxman, "Sybilguard: Defending against Sybil Attacks via Social Networks," ACM SIGCOMM Computer Comm. Rev., vol. 36, no. 4, pp. 267-278, 2006.

[20] N. Tran, B. Min, J. Li, and L. Subramanian, "Sybil-Resilient Online Content Voting," Proc. Sixth USENIX Symp. Networked Systems Design and Implementation (NSDI), 2009.