RECOMMENDATION, TRUST AND SERVICE BASED SELF TOPOLOGY CONSTRUCTION

PKKumaresan, B Sundaramurthy, P.Praveen Kumar, S Karthik, T Geetha

Abstract— **Open nature of peer-to-peer systems exposes them to malicious activity. Building trust relationships among peers can mitigate attacks of malicious peers. The present 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 a recommendation are considered while evaluating recommendations. In the Enhancement work the malicious node will be identified and reported to the monitor. The monitor will verify with the neighbor confirmation and evict the malicious node and also that will give the alert about malicious node to the entire network node.**

Index Terms— **Peer-to-peer systems, trust management, reputation, security.**

I. 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 Self-ORganizing Trust model (SORT) 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 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, 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 aninteraction, 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. A peer may be a good service provider but a bad recommender or vice versa. Thus, SORT 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. SORT defines three trust metrics. Reputation metric is calculated based on recommendations. It is important when deciding about

Manuscript received Nov 05, 2014
PKKumaresan, Prof CSE, VMKV Engg College, Salem, India
B Sundaramurthy, Asso Prof CSE, VMKV Engg College, Salem, India
P.Praveen Kumar, Asst Prof IT, VMKV Engg College, Salem, India
S Karthik, Asso Prof IT, VMKV Engg College, Salem, India
T Geetha, Asst Prof CSE, VMKV Engg College, Salem, India

strangers and new acquaintances. Reputation loses its importance as experience with an acquaintance increases. Service trust and recommendation trust are primary 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.

Here implemented a P2P 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 (file sizes, popularity of files) are approximated to several empirical results . This enabled us to make more realistic observations on evolution of trust relationships. We studied 16 types of malicious peer behaviors, which perform both service and recommendation-based attacks. SORT mitigated service-based attacks in all cases. Recommendation-based attacks were contained except when malicious peers are in large numbers, e.g., 50 percent of all peers. Experiments on SORT show that good peers can defend themselves against malicious peers without having global trust information. SORT's trust 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.

II. Service Trust Metric (STIJ)

When evaluating an acquaintance's trustworthiness in the service context, a peer first calculates competence and integrity belief values using the information in its service history. Competence belief represents how well an acquaintance satisfied the needs of past inter actions . Let cbij denote the competence belief of pi about pj in the service context. Average behavior in the past interactions is a measure of the competence belief. When evaluating competence, interactions should be considered in proportion to their weight and recentness. Then, pi calculates cbij follows

$$cb_{ij} = \frac{1}{\beta_{cb}} \sum_{k=1}^{shij} (s_{ij}^{k} \cdot w_{ij}^{k} \cdot f_{ij}^{k})$$

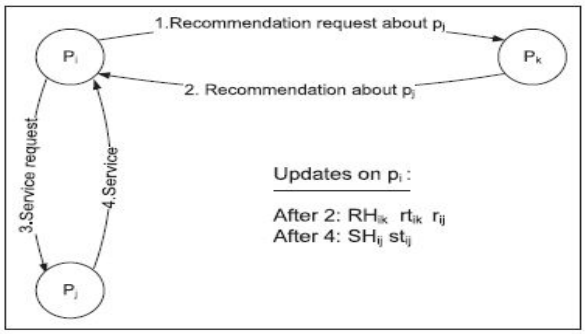

Fig. 1. Operations when receiving a recommendation and having an interaction.

TABLE 1
Notations on the Trust Metrics

Notation	Description
s_{ij}^{k}	p_i's satisfaction about k^{th} interaction with p_j
w_{ij}^{k}	weight of p_i's k^{th} interaction with p_j
f_{ij}^{k}	fading effect of p_i's k^{th} interaction with p_j
r_{ij}	p_i's reputation value about p_j
st_{ij}	p_i's service trust value about p_j
rt_{ik}	p_i's recommendation trust about p_k
sh_{ij}	size of p_i's service history with p_j

Algorithm 1. GETRECOMMENDATIONS(p_j)

$\mu_{rt} \Leftarrow \frac{1}{|A_i|} \sum_{p_k \in A_i} rt_{ik}$
$\sigma_{rt} \Leftarrow \frac{1}{|A_i|} \sqrt{\sum_{p_k \in A_i} (rt_{ik} - \mu_{rt})^2}$
$th_{high} \Leftarrow 1$
$th_{low} \Leftarrow \mu_{rt} + \sigma_{rt}$
$rset \Leftarrow \emptyset$
while $\mu_{rt} - \sigma_{rt} \leq th_{low}$ and $|rset| < \eta_{max}$ **do**
  for all $p_k \in A_i$ **do**
    if $th_{low} \leq rt_{ik} \leq th_{high}$ **then**
      $rec \Leftarrow$ RequestRecommendation(p_k,p_j)
      $rset \Leftarrow rset \cup \{rec\}$
    end if
  end for
  $th_{high} \Leftarrow th_{low}$
  $th_{low} \Leftarrow th_{low} - \sigma_{rt}/2$
end while
return $rset$

III. Reputation Metric (RIJ)

The reputation metric measures a stranger's trustworthiness based on recommendations. In the following two sections, we assume that pj is a stranger to pi and pk is an acquaintance of pi. If pi wants to calculate rij value, it starts a reputation query to collect recommendations from its acquaintances. Algorithm 1 shows how pi selects trustworthy acquaintances and requests their recommendations. Let _max denote the maximum number of recommendations that can be collected in a reputation query and jSj denote the size of a set S. In the algorithm, pi sets a high threshold for recommendation trust values and requests recommendations from highly trusted acquaintances first. Then, it decreases the threshold and repeats the same operations .To prevent excessive network traffic, the algorithm stops when _max recommendations are collected or the threshold drops under value.

IV. Selecting Service Providers

When pi searches for a particular service, it gets a list of service providers. Considering a file sharing application, pi may download a file from either one or multiple uploaders. With multiple uploaders, checking integrity is a problem since

any file part downloaded from an uploader might be inauthentic. Some complex methods utilizing Merkel hashes, secure hashes, and cryptography can be used to do online integrity checking with multiple uploaders. Since this issue is beyond the scope of this paper, the next sections assume one uploader scenario.

Service provider selection is done based on service trust metric, service history size, without them I could not have achieved this height and the almighty for showering all blessings on us.competence belief, and integrity belief values. When pi wants to download a file, it selects an uploader with the highest service trust value. If service trust values are equal, the peer with a larger service history size (sh) is selected to prioritize the one with more direct experience. If these values are equal, the one with a larger cb _ ib=2 value is chosen. If cb _ ib=2 values are equal, the one with larger competence belief value is selected. If these values are equal, upload bandwidths are compared. If the tie cannot be broken, one of the equal peers is randomly selected. pi might select a stranger due to its high reputation. For example, if pm is a stranger, pi sets stim ¼ rim according to (6). If pm is more trustworthy than all acquaintances, pi selects pm as the service provider.

ACKNOWLEDGEMENT

First of all I would like to thank God and my Parents,

- I would like next to thank our college Chairman Dr.N.JAYARAJ,MJFD.Lit.,Ph.D.,D.HONS and who all are the patrons for us in all our proceedings.
- I heartly thank ourPrincipal Dr.K.P.PADMANABAN B.E., M.E., Ph.D., for his constant encouragement and for providing all facilities, which helped me to complete this project.
- I would like to extend my sincere thanks to our Head of Department Prof.LAKSHMIPALANIYAPPAN B.E., M.S., M.S (By Research) (CSE)., for her incessant support throughout the project.
- I wish to express my deep sense of thanks to my project guide Mrs. M.REGINA BEGAM M.E., Asst Professor, Computer Science and Engineering for their fervent support and encouragement which helped me for the completion of this project.
- I sincerely thank all the teaching and non-teaching staff of Computer Science and Engineering department for giving creative ideas, which helped me to finish the project successfully at the right time.

REFERENCES

[1] 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.

[2] B. Yu, M.P. Singh, and K. Sycara, "Developing Trust in Large- Scale Peer-to-Peer Systems," Proc. IEEE First Symp. Multi-Agent Security and Survivability, 2004

[3] D. Talia and P. Trunfio, "Toward a Synergy between P2P and Grids," IEEE Internet Computing, July/Aug. 2003.

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

[5] 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

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

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

[8] S. Song, K. Hwang, R. Zhou, and Y.-K. Kwok, "Trusted P2P Transactions with Fuzzy Reputation Aggregation," IEEE Internet Computing, vol. 9, no. 6, pp. 24-34, Nov.-Dec. 2005