International Journal of Research

International Journal of Research (IJR)

e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015

Available at http://internationaljournalofresearch.org

Peer to Peer system Security Establishing based on Trust Management Past Interaction Information

Pushpagiri Sowjanya[1]& T.Rajesh[2]

[1]M-Tech, Dept. of CSE,G.Narayanamma institute of technology & science (for women)

Mail Id: - sowjanyap1990@gmail.com

[2]Asst Professor, Dept. of CSE,G.Narayanamma institute of technology & science (for women)

Mail Id: - rajesht531@gmail.com

Abstract

In this implemented project, utilizing open nature of Peer to Peer systems that avails to expose the maleficent activity. Building trust relationships among peers can reduce attacks of malevolent peers. Peers engender its own trust network in their proximity by utilizing local information available and do not endeavor to learn ecumenical trust information. Predicated on trust information it relegates the peers whether peer is trustworthy or not. In this paper utilized the technique called Self Organizing Trust Model (SORT) that aims to reduce maleficent activity in Peer to Peer system by establishing trust cognations among peers in their proximity. Trust information is evaluated predicated on accommodation, trust values of each peers and it is predicated on past interactions. Which one peer having highest trust ratio that is computed utilizing accommodation and trust values of earlier interaction that peer to be culled for next interaction. This trust information avails 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 malevolent comportment models. In the experiments, good peers were able to compose trust relationships in their proximity and isolate malignant peers.

Keywords: Peer to Peer system; Trust Management; Security; Establishing Trust Information; Past Interaction

1. Introduction

Peer to peer (P2P) systems merges sizably voluminous number of computers that enters or leave network frequently. In peer to peer systems individual machine can communicate with each other's and apportion resources without dealing the central coordinator. Building long term trust relationships provides more secure environment which reduces risk and skeptically in the future. Metrics are required to describe confide in computational model. Trust among peers is quantified predicated on the information provided by interactions and feedbacks of peers. The systems such as eBay prefer the central server to store and manage trust information. In most P2P systems central ascendancy is not present to deal with storing and managing trust information about each other [1], [2]. Structure of P2P systems resolves management of trust information. In approaches such as distributed hash table (DHT), feedback storing about other peers which made peer as trust holder [1], [3], [4]. Ecumenical trust information is accessed through DHT which is stored by trust holders. A peer sends queries for trust to ken trust information of other peers. A query is either flooded to network or to neighbor of query initiator.

Self Organizing Trust model (SORT) decreases malevolent intents with the avail of trust relationship among peers. Peers does not amass trust information from all peers because each peer develops its local trust about peers interacted in the past, so good peer can isolate malevolent peers. At beginning peers are verbalized to be strangers to each other. A peer is verbalized to be acquaintance of another when it provides accommodation e.g.; file uploading. A peer sets to trust stranger when it has no acquaintance[1]. If there is parity in trustworthiness then acquaintance is preferred over stranger. Utilizing an accommodation of a peer is verbalized to be an interaction. It is computed predicated on recentness of the interaction, weight (paramountcy). Recommendation, which is feedback of acquaintance, is computed predicated on trustworthiness of recommender. It involves the own experience about the peer of recommender, information from recommender's acquaintances, and recommender's level of confidence. The recommendation has a low value if level of confidence is low, which affects less the trustworthiness of recommender.

SORT defines two context of trust: accommodation and recommendation trust. In these contexts, separate histories are maintained to store information about past interactions and recommendations in order to assess competence and integrity of acquaintances [3]. There are three trust metrics: Reputation metric-It is computed predicated on recommendations. It considers being prime when deciding about strangers and incipient acquaintances. Accommodation trust metric and Recommendation trust metrics are considered in order to quantify trustworthiness in the accommodation context and recommendation contexts. Accommodation providers are culled predicated on accommodation trust metric, whereas recommendation trust metric is utilized when requesting recommendations. Recommendations are computed predicated on recommendation trust metric in order to compute reputation metric[2]. SORT deals with the accommodation predicated attacks as well as recommendation predicated attacks. SORT describes, good peer can bulwark themselves against peers with malevolent intents without utilizing ecumenical trust information, and instead it utilizes local trust to assess trustworthiness of other peers.

2. Related Work

2.1 Existing System:

Survive methods for reliable management that are predicated on reputation fixate on the semantic opportune- ties of the reliance model. They do not scale as they either rely on a central database or require maintaining ecumenical erudition at each agent to provide data on earlier interactions. In this paper we present an approach that addresses the quandary of reputation-predicated trust management at both the data management and the semantic level. We employ at both levels scalable data structures and algorithms that require no central control and sanction assessing trust by computing an agents reputation from its former interactions with other agents.

There are no well-defined methods for managing trust relationships in p2p systems[5]. The DHT predicated approaches are only suited for structured p2p networks not for unstructured p2p networks. The present methods introduce central ascendancy in p2p networks which may collapse p2p nature. Every agent must keep rather involute and profoundly and astronomically immense data structures that represent a kind of ecumenical cognizance about the whole network. This paper presents distributed algorithms that enable a peer to reason about trustworthiness of other peers predicated on past interactions and recommendations. Peers engender their own trust network in their proximity by utilizing local information available and do not endeavor to learn ecumenical trust information. Two contexts of trust, accommodation, and recommendation contexts are defined to quantify trustworthiness in providing accommodations and giving recommendations.

Self-Organizing Trust model (SORT) that aims to decrement malignant activity in a P2P system by establishing trust cognations among peers in their proximity. In SORT, peers are surmised to be strangers to each other at the commencement. A peer becomes an acquaintance of another peer after providing an accommodation, e.g., uploading a file. If a peer has no acquaintance, it opts to trust strangers. An acquaintance is always preferred over a stranger if they are equipollent trustworthy. Utilizing an accommodation of a peer is an interaction, which is evaluated predicated on weight (paramount) and recentness of the interaction, and gratification of the requester.

An acquaintance's feedback about a peer, re commendation, is evaluated predicated on recommenders' trust worthiness. It contains the recommenders' own experience about the peer, information accumulated from the recommenders' acquaintances, and the recommenders' level of confidence in the recommendation. If the caliber of confidence is low, the recommendation has a low value in evaluation and affects less the trustworthiness of the recommender. SORT defines three trust metrics. Reputation metric is calculated predicated on recommendations.

It is paramount when deciding about strangers and incipient acquaintances. Reputation loses its paramount as experience with an acquaintance increases. Accommodation trust and recommendation trust are primary metrics to quantify trustworthiness in the accommodation and recommendation contexts, respectively. The accommodation trust metric is utilized when culling accommodation providers. The recommendation trust metric is consequential when requesting recommendations. When calculating the reputation metric, recommendations are evaluated predicated on the recommendation trust metric.

2.2 Proposed System:

In this paper, the following posits are considered for the proposed system.

- Peers have equal computational power and responsibility.
- There are no privileged, centralized, or trusted peers to manage trust relationships.
- Peers infrequently leave and join the network.
- A peer provides accommodations and uses accommodations of others.
- For simplicity of discussion, one type of interaction is considered in the accommodation context, i.e., file download.

(i) Preliminary Notations:

Denotesthe it peer. When pi uses a service of another peer, it is an interaction for p^i. Interactions are unidirectional.For example, if p^i downloads a file from pj, it is an interaction for pi and no information is stored on p^j. If pi had at least one interaction with p^j, p^j is an acquaintance of p^i. Otherwise, p^j is a stranger to pi. Ai denotes pi's set of acquaintances.A peer stores a separate history of interactions for each acquaintance. SHdenotes pi's service history with pj where sh^{ij}denotes the current size of the history. $shma_x$ denotes the upper bound for service history size. Since new interactions areappended to the history, SH^{ij} is a time ordered list.

(ii) Network Architecture:

Downloading a file is an interaction. A peer sharing files is called an uploaded. A peer downloading a file is called adownloader. The set of peers who downloaded a file from a peer are called downloaders of the peer. An ongoingdownload/ upload operation is called a session. A good peer uploads authentic files and gives fair recommendations [6].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.

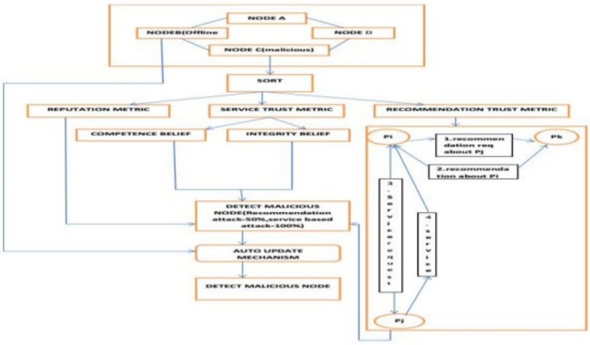
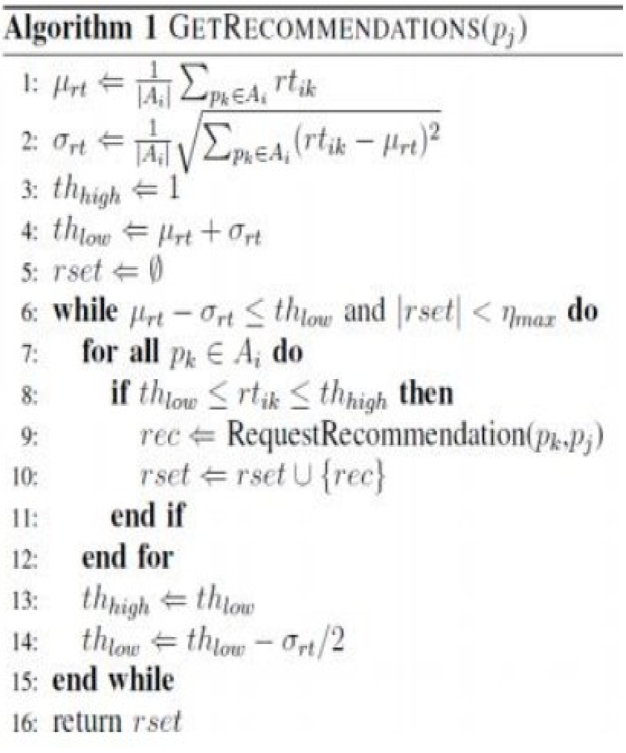


Fig 1: Architecture Diagram.

SORT defines three trust metrics. Reputation metric is calculated based on recommendations. It is important whendeciding about strangers and new acquaintances. Reputation loses its importance as experience with an acquaintanceincreases. Service trust and recommendation trust are primary metrics to measure trustworthiness in the service andrecommendation contexts, respectively. The service trust metric is used when selecting service providers. Therecommendation trust metric is important when requesting recommendations. When calculating the reputation metric,recommendations are evaluated based on the recommendation trust metric. Assume that pi wants to get a particularservice. p^j is a stranger to pi and a probable service provider. To learn pjs' reputation, pi requests recommendationsfrom its acquaintances. Assume that pk sends back a recommendation to p^i. After collecting all recommendations, p^icalculatesr^{ij}. Then, pi evaluates pks' recommendation, stores results in RH^{ik}, and updates rt^{ik}. Assuming pjistrustworthy enough, pi gets the service from pj. Then, pi evaluates this interaction and stores the results in SH^{ij}, andupdatesst^{ij}. One peer is marked as trusted by SORT and if it- is turned off from network, there is a possibility to anothermalicious peer takes its position and act as trusted peer [7]. This can be avoided by the Auto update mechanism.

(iii) Algorithm:

Topology creation is creating a network and maintaining communication among various nodes in peer to peer networkwhich helps us to share the data. Create different nodes in proper name, ip address and port number for datacommunication. The node is added to give the name of the node, ip address and port address of that node. If the entirenode adds successfully to display the node connection frames. Creating long-term trust relationships among peers canprovide a more secure environment by reducing risk and uncertainty in future P2P interactions. However, establishingtrust in an unknown entity is difficult in such a malicious environment. Furthermore, trust is a social concept and hardto measure with numerical values. Metrics are needed to represent trust in computational models. Classifying peers aseither trustworthy or untrustworthy is not sufficient in most cases.



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 $rset$
```

Metrics should have precision so peers can be ranked according to trustworthiness. Interactions and feedbacks of peersprovide information to measure trust among peers. Interactions with a peer provide certain information about the peerbut feedbacks might contain deceptive information. This makes assessment of trustworthiness a challenge. Self Organizing Trust model (SORT) that aims to decrease malicious activity in a P2P system by establishing trust relationsamong peers. Each peer develops its own local view of trust about the peers interacted in the past. In this way, goodpeers form dynamic trust groups in their proximity and can isolate malicious peers. In SORT, peers are assumed to bestrangers 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.

SORT defines three trust metrics.Reputation metric is calculated based on recommendations. It is important when deciding about strangers and newacquaintances. Reputation loses its importance as experience with an acquaintance increases. Service trust andrecommendation 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 isimportant when requesting recommendations. When calculating the reputation metric, recommendations are evaluatedbased on the recommendation trust metric. Creating trust relationship is based upon two contexts of trust. They areService Context, Recommendation Context. The service trust metric is used when selecting service providers. Therecommendation trust metric is important when requesting recommendations. When pi searches for a particularservice, it gets list of service providers. Considering a file sharing application, pi may download a file from either oneor multiple uploaders. With multiple uploaders, checking integrity is a problem since any file part downloaded from anuploader might be inauthentic.

Assume that pi wants to get a particular service. p^{j} is a stranger to pi and a probableservice provider. To learn p^{js}' reputation, pi requests recommendations from its acquaintances. Assume that p^{k}sendsback a recommendation to pi. After collecting all recommendations, p^{i} calculates r^{ij}. Then, pi evaluates pks'recommendation, stores results in RHik, and updates rtik. Assuming pj is trustworthy enough, pi gets the service fromp^{j}. Then, pi evaluates this interaction and stores the results in SHij, and updates st^{ij}. In this paper, after the proposedalgorithm is used, SVM (Support Vector Machine) Classifier is used. Support vector machine is a supervised learningmodel with associated learning algorithms that analyze data and recognize patterns, used for classification andregression analysis. Given a set of training examples, each marked as belonging to one of two categories, an SVMtraining algorithm builds a model that assigns new examples into one category or the other, making it a nonprobability binary linear classifier.

Thus the proposed system makes use of SVM to more efficiently classify the peeras trusty or non-trusty peers. In some cases, for a stranger peer, the values of Service Trust, Recommendation Trust andReputation Trust may conflict i.e. some of two values may be low and one may be high. In such cases it is difficult todecide whether a peer is trusty or non-trusty. The use of SVM Classifier is proposed in such scenarios. It increases theefficiency of taking decisions for a particular peer.

3. Implementation

1. SORT accommodation engenderment
2. Peers establishment
3. Files uploading, downloading
4. Recommendation metric
5. Trust metric

Modules Description:

a) SORT accommodation engenderment:

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. distributed hash table (DHT)- predicated approaches, each peer becomes a trust holder by storing feedbacks about other peers .

b) Peers establishment:

Self-ORganizing Trust model (SORT) that aims to decrement malignant activity in a P2P system by establishing trust cognations among peers in their proximity. Peers do not endeavor to accumulate 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 malevolent peers.

c) Files uploading, downloading:

peers are postulated to be strangers to each other at the commencement. A peer becomes an acquaintance of another peer after providing an accommodation, e.g., uploading a file. If a peer has no acquaintance, it opts to trust strangers. An acquaintance is always preferred over a stranger if they are equipollently trustworthy. Utilizing an accommodation of a peer is an interaction, which is evaluated predicated on weight (paramountcy) and recentness of the interaction, and gratification of the requester.

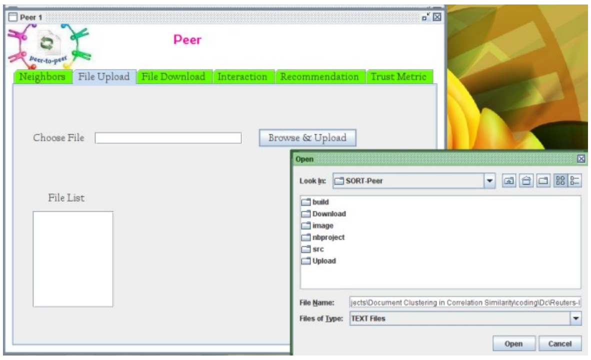
d) Recommendation metric:

Recommendation is evaluated predicated on recommender's trustworthiness. It contains the recommender's own experience about the peer, information accumulated from the recommender's acquaintances, and the recommender's level of confidence in the recommendation. If the caliber of confidence is low, the recommendation has a low value in evaluation and affects less the trustworthiness of the recommender.

e) Trust metric:

SORT defines three trust metrics. Reputation metric is calculated predicated on recommendations. It is consequential when deciding about strangers and incipient acquaintances. Reputation loses its consequentiality as experience with an acquaintance increases. Accommodation trust and recommendation trust are primary metrics to quantify trustworthiness in the accommodation and recommendation contexts, respectively. The accommodation trust metric is utilized when culling accommodation providers. The recommendation trust metric is paramount when requesting recommendations.

4. Experimental Work

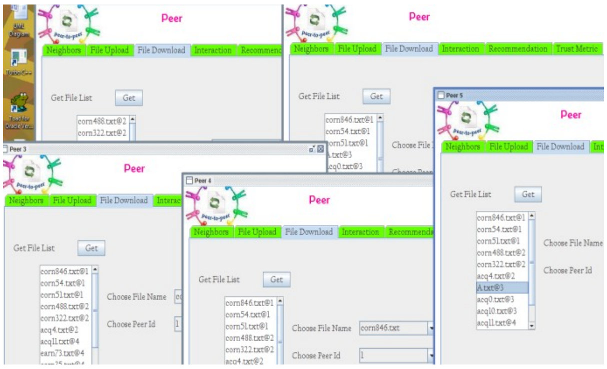


Look In: SORT-Peer

build
Download
image
nbproject
src
Upload

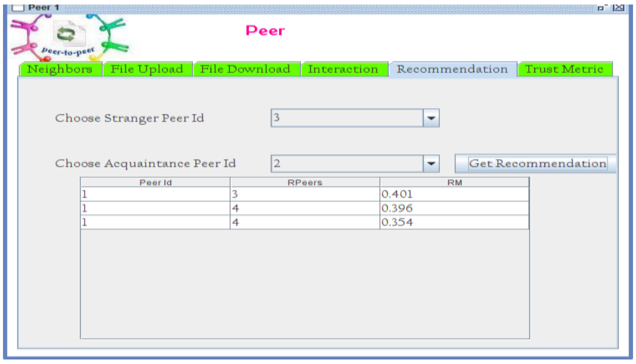
File Name: jects\Document Clustering in Correlation Similarity\coding\Dc\Reuters-I
Files of Type: TEXT Files

Fig 2: Upload the file into peer 1



Choose File Name corn846.txt
Choose Peer Id 1

Fig 3: Download files in all peers.



Choose Stranger Peer Id 3

Choose Acquaintance Peer Id 2

Peer Id	RPeers	RM
1	3	0.401
1	4	0.396
1	4	0.354

Fig 4: Select Recommendation Page.

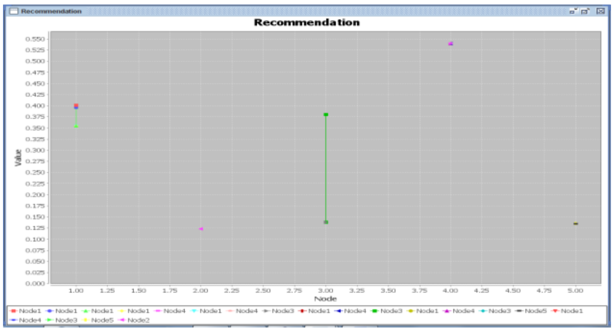


Fig 5: Final Recommendation Page.

5. 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 maleficent peers around itself as it develops trust relationships with good peers. Two context of trust, accommodation and recommendation contexts, are defined to quantify capabilities of peers in providing accommodations and giving recommendations. Interactions and recommendations are considered with contentment, 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. Utilizing trust information does not solve all security quandaries in P2P systems but can enhance security and efficacy of systems. If interactions are modeled correctly, SORT can be acclimated to sundry P2P applications, e.g: CPU sharing, storage networks, and P2P gaming. Defining application concrete context of trust and cognate metrics can avail to assess trustworthiness in sundry tasks.

6. References

[1] AhmetBurakCan and Bharat, "A Self Organizing Trust Model for Peer-to-Peer Systems", IEEE-2013

[2] Aberer.K and Despotovic.Z, "Managing Trust in a Peer-2-Peer Information System", Information andKnowledge Management (CIKM) - 2001

[3] Kamvar.S, Schlosser.M, and Garcia-Molina.H, "The (Eigentrust) Algorithm for Reputation Management in P2PNetworks", World Wide Web Conf. (WWW) - 2003

[4] SelcukA.A, Uzun.E, and Pariente.M.R, "A Reputation-Based Trust Management System for P2P Networks"Cluster Computing and the Grid (CCGRID) - 2004

[5] Zhou. R, Hwang. K, and Cai. M, "Gossiptrust for Fast Reputation Aggregation in Peer-to-Peer Network",IEEE - 2008

[6] Abdul-Rahman. A and Hailes.S, "Supporting Trust in Virtual Communities", Conf. System Sciences (HICSS) –2008