RESEARCH ARTICLE OPEN ACCESS

Self-Organizing Trust Model

Suyog Ashokrao Nagare [1], Prof. D.B. Kshirsagar [2]
Department of Computer Science and Engineering
SRES, College of Engineering, Kopargaon
India

ABSTRACT
As compared to other system Peer to peer system has open access, in networking domain. Each peer is capable of sharing information to other peer in peer to peer system. So there are chances of malicious activities increases. Along with recommendations one peer must send trust parameters to another peer for better security. In this system, recommendations are derived based on priority, trustworthiness, history, and peer satisfaction. The peer will communicate with that peer only who is having more recommendations and trustworthiness values.
Keywords :- Distributed systems, trust management, reputation, and security.

I. INTRODUCTION

Peer to peer network is the collection of independent peers. These peers share data among them without using any centralize system. So security is the main issue. On this system any malicious user can attack easily. To avoid the malicious attack in peers we are maintaining trustworthiness. But the challenging task is to keeping trust on another peer. Because the opponent peer can be malicious. To state into the numerical format, the peer is very complex as the trust is logical and social phenomenon. For the file sharing between peers, classification of peer as trustworthiness or non-trustworthiness is not always efficient. So we are maintaining the matrices here for peer trust calculation.

The trustworthiness alone is not a sufficient approach to communicate with peer. That's why; along with trustworthiness of the peer we maintain the recommendation matrix and reputation. Here self-organizing trust model (sort) technique focuses by maintaining trust relations among peers in their surroundings to reduce malicious activity in a peer to peer distributed system. In this system from remaining all peers, it do not try to collect trust information. Here about the peers interacted in the past, every peer develops its own local computation of trust[9]. Like this, good peers form dynamic trust groups evaluated in their surroundings and from system it can remove malicious peers.

The three matrices are calculate here. The reputation metric is the first matric and based on peer's recommendations this matrix is calculated. Among all peers it is important while deciding strangers and new nodes. Second, the primary metrics to compute trust relation in the service and recommendation surroundings are service trust metrics and recommendation trust metrics. The service trust metric is used while deciding service providers. Recommendation trust metric is used while requesting recommendations. While we evaluate the reputation metric, trust metric recommendations are calculated on the basis of recommendation

II. EXISTING SYSTEM

To store and manage trust information, in the presence of an authority a central server is a preferred way. The trust information is securely stored by central server and defines trust metrics. According to the trust information the trust of peer is detected from the central server.

In peer to peer system there is central manager, so that the chances of malicious activities are increases. To improve efficiency and the accuracy in the distributed system there are many researches have been done already. Here by maintaining trust value and recommendation, we have propose new system which mitigates risk management.

By malicious behaviours of peer entities the possible utility loss is known as risk are caused by the potential security policy violations. We generate trust relationships between peers to guide security parameters with risk management[5]. It helps to keep security at fixed level to the organization..

A. Algorithm for Reputation Management

Here some background protocols are explained. We only explain the abstract model of some network level model protocol due to space limitations the detailed overview of each protocol.

In this paper, all techniques are represented in terms of modelled structure. On the evolutional parameters used in the system our system performance and integrated security is depend.

B. *Overlay routing model*

To describe the overlays such as CAN, Chord, Tapestry and Pastry the structured Distributed routing overlay have used here. Here, from a large id space of integer data-type, uniform node IDs are assigned by participating nodes. With unique keys some objects are assigned.

Every key is assigned by the overlay to a unique live node, which is called the key's root. Protocol routes packet is in between this root and other peer. The routing table is maintained the unique id of every protocol along with port number. For sending packet in between nodes effectively this routing table is very important. As well as a neighbor list is maintained by each node which consist of the unique id of each node and number of nodes present at that particular peer space.

C. *DMRep*

It is the system which is structural distributed system. Here all nodes maintain their trust values by sending trust value to the central authority. Two trust models which are defined aberer and despotovic's trust model, where by using p-grid, peers report their complaints. The reputation based problem is addressed here in data access layer. The peer is considered as the trustworthy peer, until there is any complaint rises about a peer. But this causes problem while adding new comer inside the existing network.

The main advantage of the peer to peer system is every peer is capable of storing and sharing data to other nodes. This lead uses of network bandwidth and reduction in storage costs. So as others it considered as scalable.

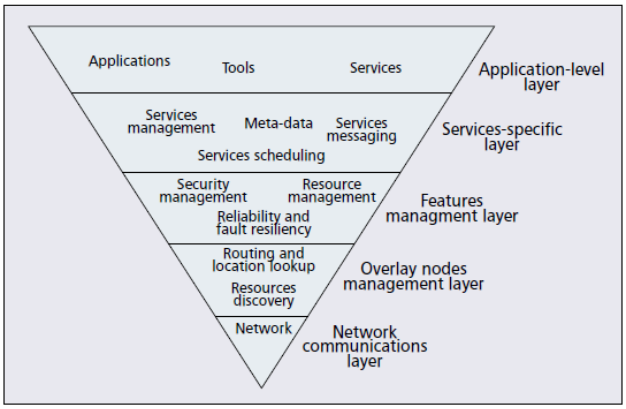


Fig 1. P2P overlay network Architecture.[6]

D. *Power Trust*

The distribution of peer feedbacks is Power Trust [5]. On the basis of their power nodes are ranked. Strongest node are low power node. From all adjacent peer, system calculates global feedback to which data sharing is dine. In this community context is improved using factors which are utilizing power nodes, feedback aggregation speed, and global reputation accuracy. It's having advantages like, power law distribution, fast reputation generation, ranking, system robustness and efficiency, disadvantages are Non deployment of power trust on unstructured nodes, fail to detect intrusions, collusions, and selfishness of peers and failed to calculate global trust value of each peer in a network.

To prevent malicious activity in the network, B. Bhargava A burak et al. [7] gives a self-organizing trust model (sort). No global information is used. Trust information of all peer adjacent in network does not collect peer. Two matrices of trust describes SORT, service and recommendation matrices are defined on the basis of services provided by peer and feedback received from peer. Feedback is considered as recommendation which consist of level of recommendation given by own peer. Local trust information is considered. In this paper. Reputation queries[1] send by peers only to peers interacted in the past. Disadvantages is that the system cannot detect its trust value if any peer is starts to become malicious after some time span.

III. PROPOSED SYSTEM

PEER to PEER algorithms enable a peer to reason about trustworthiness of other peers based on past interactions and recommendations. By using local information available, Peers create their own trust network in their proximity and do not try to learn global trust information. In proposed system peers do not collect information of all pairs in the network they only keep information of neighbours.

This systemhas following main roles:

A) Service trust matric
B) Reputation Trust Metric.
C) Recommendation Trust Metric.

A. *Service Trust Matric:*

Using the information in its service history a peer first calculates competence and integrity belief values when evaluating an acquaintance's trustworthiness in the service context. How well an acquaintance satisfied the needs of past interactions represented by Competence belief[10]? Let in the service context the friend request denote the competence

belief of Pi about Pj . Average behaviour in the past interactions is a measure of the competence belief. Consistency is as important as competence. Integrity belief is the level of confidence in predictability of future interactions.

Let in the service context, I bij denote the integrity belief of Pi aboutPj. A measure of the integrity belief is Deviation from average behaviour ($cbij$).

B. Reputation Trust Matric:

The reputation metric measures a stranger's trust worthiness Based on recommendations. We assume that Pj is a stranger to Pi and Pk is an acquaintance of Pi In the following two sections[4]. If Pi starts a reputation query to collect recommendations from its acquaintances, if it wants to calculate rij value[2]. Trustworthy acquaintances and requests their recommendations. Let the maximum number of recommendations denoted by max that can be collected in a reputation query and the size of a set S denoted by jSj. Pi sets a high threshold for recommendation trust values and requests recommendations from highly trusted acquaintances first, in the getting recommendation algorithm.

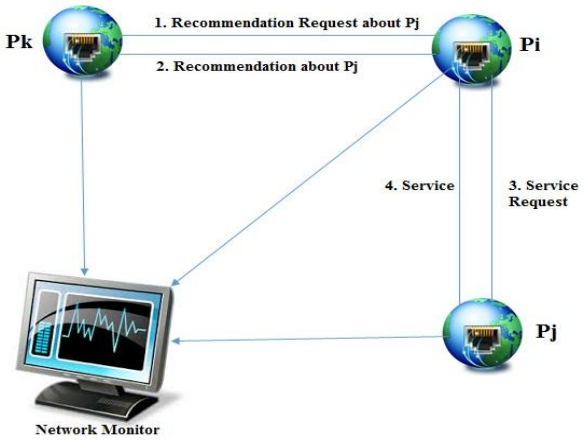


Fig 2. System Architecture.

C. Recommendation Trust Metric:

Assume that a particular service want to get to Pi. Pj a probable service provider and is a stranger to pi. Pi requests recommendations to learn Pj's reputation, from its acquaintances. Assume that recommendation send back to Pi from Pk[8]. after collecting all recommendations Pi calculates rij. Then, pk's recommendation evaluates Pi, and stores results in RH ik, and also updates rtik. Assuming Pj is trustworthy enough, Pi gets the service from Pj. Then, pi and stores the results in SH ij, and updates stij by evaluating this interaction.

IV. ALGORITHM

The Recommendation algorithm is used here which is having following steps.

1. First initialize the peers in the network.

2. Then initialize threshold value

3. Trust values Calculate threshold for recommendation.

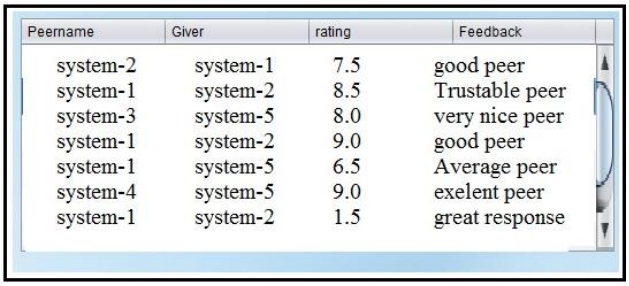
4. Calculate the threshold from highly trusted acquaintances for requests recommendations.

5. Evaluate Recommendation according to trust value of the recommender.

6. Decreases the threshold and repeats the same process.

7. When maximum recommendations are collected, if excessive network traffic then the algorithm stops.

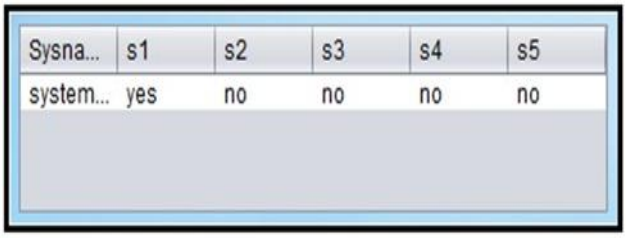
V. RESULTS



Peername	Giver	rating	Feedback
system-2	system-1	7.5	good peer
system-1	system-2	8.5	Trustable peer
system-3	system-5	8.0	very nice peer
system-1	system-2	9.0	good peer
system-1	system-5	6.5	Average peer
system-4	system-5	9.0	exelent peer
system-1	system-2	1.5	great response

Fig 3. All peer feedback

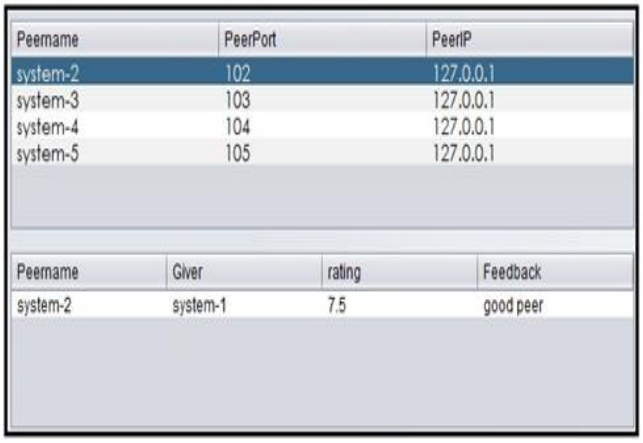
Above figure shows the module describing the feedbacks of all the peers.



Sysna...	s1	s2	s3	s4	s5
system...	yes	no	no	no	no

Fig 4. Neighborhood peer status

Above figure shows that, the one system is connected to how many nodes. Here, the system s1 connected with only itself.



Peername	PeerPort	PeerIP
system-2	102	127.0.0.1
system-3	103	127.0.0.1
system-4	104	127.0.0.1
system-5	105	127.0.0.1

Peername	Giver	rating	Feedback
system-2	system-1	7.5	good peer

Fig 5. Peer Review

The above figure shows that the list of peers along with peer port and peer IP.

VI. CONCLUSION

For security decisions a new trust which integrates risk management. Based security enforcement the unique feature of utility maximization through risk management such a new model offers. This is achieved by trust enhanced security making process using both the current state of knowledge on the trustworthiness of the entities and the risk allocation for the given interaction. In doing so to guide the security decisions, we enable the leveraging of the knowledge on trust relationships such that while keeping the security risk at a defined level, the underlying application gains maximum utility. To the best of our knowledge, this paper is the first which maximize the utility to integrate risk management for trust based security decisions.

ACKNOWLEDGMENT

I would like to take this opportunity to express my profound gratitude and deep regard to my Project Guide,Prof. D. B. Kshirsagar (Head of Computer Engineering Department), for his exemplary guidance, valuable feedback and constant encouragement throughout the duration of the project. I also want to thanks Prof. P.N. Kalwadekar(PG Co-ordinator) for his valuable suggestions were of immense help throughout my project work. His perceptive criticism kept me working to make this project in a much better way. Working under both of them was an extremely knowledgeable experience for me.

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.R. Caves, Multinational Enterprise and Economic Analysis, Cambridge University Press, Cambridge, 1982. (book style)

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

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

[4]] Reference 4 [4] L. Xiong and L. Liu, “Peertrust: Supporting Reputation-Based Trust for Distributed Ecommerce Communities,” IEEE Trans. Knowledge and Data Eng., vol. 16, no. 7, pp. 843-857, July 004

[5] B. Yu and M. Singh, “A Social Mechanism of Reputation management in Electronic Communities,” Proc. Cooperative Information Agents (CIA), 2000

[6] Z. Despotovic and K. Aberer, “Trust-Aware Delivery of mposite Goods,” Proc. First Int’l Conf. Agents and Distributed Computing, 2002

[7] F. Cornelli, E. Damiani, S.C. Vimercati, S. Paraboschi, and P. Samarati, A reputation-based approach for choosing reliable resources in Distributed networks ,In CCS02, Washington DC, USA 2002.

[8] K. Aberer, A. Datta, and M. Hauswirth, “P-Grid: Dynamics of Self- Organization Processes in Structured DISTRIBUTED Systems,” Distributed Systems and Applications, vol. 3845, 2005.

[9] R. Zhou and K. Hwang, “Powertrust: A Robust and Scalable Reputation System for Trusted Distributed Computing,” IEEE Trans. Parallel and distributed Systems, vol. 18, no. 4, pp. 460-473, Apr. 2007.

[10] M. Gupta, P. Judge, and M. Ammar, “A Reputation System for Distributed Networks,” Proc. 13th Int’l Workshop Network and Operating Systems Support for Digital Audio and Video (NOSSDAV), 2003.