

# Self-Organizing Trust Model for Distributed System

Baradkar Shubhangi<sup>1</sup> Khamkar Dhanshri<sup>2</sup> Ithape Pranita<sup>3</sup>

<sup>1,2,3</sup>Student

<sup>1,2,3</sup>Bharati Vidyapeeths College Of Engineering For Women, Dhankawadi, Pune-043

**Abstract**— Distributed networking technologies have gained popularity as a mechanism for users to share files without the need for centralized servers. A Distributed network provides a scalable and fault-tolerant mechanism to locate nodes anywhere on a network without maintaining a large amount of routing state. This allows for a variety of applications beyond simple file sharing. Examples include multicast systems, anonymous communications systems, and web caches. We survey security issues that occur in the underlying Distributed routing protocols, as well as fairness and trust issues that occur in file sharing and other Distributed applications. We discuss how techniques, ranging from cryptography, to random network probing, to economic incentives, can be used to address these problems. Open nature of Distributed 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 a 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.

**Key words:** Distributed systems, reputation, security, trust management

## I. INTRODUCTION

Distributed systems rely on collaboration of peers to accomplish tasks. Ease of performing malicious activity is a threat for security of distributed systems. Creating long-term trust relationships among peers can provide a more secure environment by reducing risk and uncertainty in future distributed 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 distributed systems, peers organize themselves to store and manage trust information about each other [1], [2]. Management of trust information is dependent to the structure of distributed network. In distributed hash table (data) - based approaches, each peer becomes a trust holder by storing feedbacks about other peers [1], [3], [4]. Global trust

We propose a self-organizing trust model (sort) that aims to decrease malicious activity in a distributed 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 [7], forming trust relations in proximity of peers helps to mitigate attacks in a distributed system.

SORT defines three trust metrics. Reputation metric is calculated based on recommendations. It is important when deciding about 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.

## II. EXISTING SYSTEM

Abdul-rahman and Hailes evaluate trust in a discrete domain as an aggregation of direct experience and recommendations of other parties. They define a semantic distance measure to test accuracy of recommendations. Yu and Singh's model propagates trust information through referral chains. Referrals are primary method of developing trust in others. Mui et al. propose a statistical model based on trust, reputation, and reciprocity concepts. Reputation is propagated through multiple referral chains. Jøsang et al. discuss that referrals based on indirect trust relations may cause incorrect trust derivation. Thus, trust topologies should be carefully evaluated before propagating trust information. Terzi et al. introduce an algorithm to classify users and assign them roles based on trust relationships. Zhong proposes a dynamic trust concept based on McKnight's social trust model. When building trust relationships, uncertain evidences are evaluated using second-order probability and Dempster-Shaferian framework.

Self-Organizing Trust model (SORT) that was aims to decrease malicious activity in a P2P system by

establishing trust relations among peers in their proximity. 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.

SORT defines three trust metrics. Reputation metric was calculated based on recommendations. It was important when deciding about 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.

### III. PROPOSED SYSTEM

To propose a Self-Organizing Trust model (SORT) that aims to decrease malicious activity in a Distributed system by establishing trust relations among the nodes in their proximity.

We are going to establish trusted environment around every node in trusted network. Each node wishing to connect to network has to go through the process of trust value calculation based on service matrix, trust value, reputation matrix etc. Node having appropriate value of trust getting added to network and perform activities, but node performing less trust value cannot add to network.

A Distributed network provides a scalable and fault-tolerant mechanism to locate nodes anywhere on a network without maintaining a large amount of routing state. This allows for a variety of applications beyond simple file sharing. Examples include multicast systems, anonymous communications systems, and web caches. We survey security issues that occur in the underlying Distributed routing protocols, as well as fairness and trust issues that occur in file sharing and other Distributed applications.

Proposed system 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.

In proposed system peers do not collect information of all pairs in the network they only keep info of neighbours.

### IV. MODULES USED

- Service Trust metrics
- Reputation Metrics
- Recommendation Trust Metrics
- Selecting Service Provider
- Network Monitor

#### A. Service Trust Metrics:

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 interactions. Let friend request denote the competence belief of  $p_i$  about  $p_j$  in the service context. Average behavior in the past interactions is a measure of the competence belief. Consistency is as important as competence.

#### B. Reputation Metrics:

The reputation metric measures a stranger's trust worthiness based on recommendations. we assume that  $p_j$  is a stranger to  $p_i$  and  $p_k$  is an acquaintance of  $p_i$ . If  $p_i$  wants to calculate  $r_{ij}$  value, it starts a reputation query to collect recommendations from its acquaintances. Trustworthy acquaintances and requests their recommendations.

#### C. Recommendation Trust Metrics:

In a large distributed system, it is difficult to obtain knowledge about every entity in the network, let alone first hand knowledge and experience of them. Yet, any entity in the network is a potential target of communication. Human beings cope with unknown entities via first impressions, and word of mouth. Entities in the network represent human beings one way or another, and they exist in their own social system. Consequently, it makes sense for them to have, as far as possible, the capability to reason about trust in the same ways in which humans would.

Assume that  $p_i$  wants to get a particular service.  $p_j$  is a stranger to  $p_i$  and a probable service provider. To learn  $p_j$ 's reputation,  $p_i$  requests recommendations from its acquaintances.

Assume that  $p_k$  sends back a recommendation to  $p_i$ . After collecting all recommendations,  $p_i$  calculates  $r_{ij}$ . Then,  $p_i$  evaluates  $p_k$ 's recommendation, stores results in  $RH_{ik}$ , and updates  $rt_{ik}$ . Assuming  $p_j$  is trustworthy enough,  $p_i$  gets the service from  $p_j$ .

#### D. Selecting Service Provider:

When  $p_i$  searches for a particular service, it gets a list of service providers.

Service provider selection is done based on service trust metric, service history size, competence belief, and integrity belief values. When  $p_i$  wants to download a file, it selects an uploader with the highest service trust value.

#### E. Network Monitor:

The network monitor performs all the task related to the establishment of the network.

The network monitor keep watch on every activity happening in the network.

It is responsible for maintaining the sequential report of the functionality of other network elements.

#### F. Block Diagram:

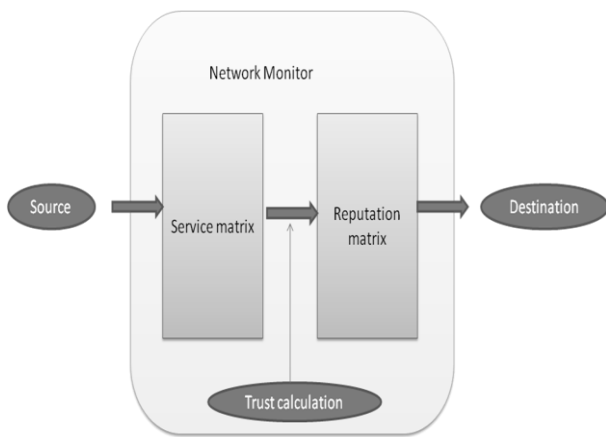


Fig.1: Block Diagram

## V. THE TRUST MODEL

The trust model manages the trust information in the system. For the purpose of this work, we consider that the trust information contain a set of pairs  $\{(a, t)\}$ , where  $a$  is an agent id and  $t$  ranging over  $[0, 1]$  is the trust value. While  $a \in A$  is a relatively static value,  $t \in T$  is a dynamic one which is updated according to past experiences  $e \in E$  by a trust update function:

$$\delta: T \times E \rightarrow T \quad (1)$$

Where  $T$  is a set of trust values,  $E$  is a set of experiences from previous interactions. The low level trust management design is out of scope of this paper. The interested readers can refer to work in [11] for general trust calculations and our work on trust management for mobile agent applications. When requested by the incoming agent for authorization.

$$\text{decision} : (t > t_{th}) \rightarrow \{0, 1\} \quad (2)$$

Where  $t_{th}$  is the trust threshold. When the above condition is tested true, then the agent is trusted and decision is set to 1; otherwise, the agent is regarded as malicious and decision is set to 0.

## VI. CONCLUSION

This project focus to develop a quantitative model that can provide exact calculations for precise risk allocations. That is to determine how much privileges/permissions should be granted for any given trust value of an entity. And it helps to decrease malicious activity in network based on trust value.

## REFERENCES

- [1] K. Aberer. P-grid: "A self-organizing access structure for p2p information systems". In Cooperative Information Systems, 9th International Conference, CoopIS-2001, 2001.
- [2] K. Aberer and Z. Despotovic. "Managing trust in a peer-to-peer information system". In 2001 ACM CIKM International Conference on Information and Knowledge Management, 2001
- [3] E. Adar and B. A. Huberman. Free riding on gnutella. First Monday, 5(10), 2000.
- [4] S. Ba and P. A. Pavlou. "Evidence of the effect of trust building technology "in electronic markets: Price premiums and buyer behavior. MIS Quarterly, 26(3), 2002.

- [5] M. Chen and J. P. Singh. "Computing and using reputations for internet ratings". In 3rd ACM Conference on Electronic Commerce, 2001.
- [6] F. Cornelli, E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati. "Choosing reputable servers in a P2P network". In Eleventh International World Wide Web Conference, 2002.
- [7] C. Dellarocas. "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior". In 2nd ACM Conference on Electronic Commerce, 2000