A Reputation Based Resource Allocation And Resource Monitoring In Peer To Peer Networks

U.Sahana[1] and G. Selvavinayagam [2]
[1] Pg scholar & [2] Assistant Professor
Department of Information Technology, SNS College of Technology, Coimbatore, India

Abstract— **Distributed algorithms used by a peer to reason about trustworthiness of other peers based on the offered local information which includes past interactions and recommendations received from others. Peers work together to establish trust among each other without using a priori information or a trusted third party. A peer's trustworthiness in given that services, e.g., uploading files, and giving recommendations is evaluated in service and recommendation contexts. Three main trust metrics, reputation, service trust, and recommendation trust, are defined to accurately measure trustworthiness in these contexts. An interaction is evaluate based on three parameter: satisfaction, weight, and fading effect. When evaluating a recommendation, including to these parameters, recommender's trustworthiness and confidence about the information provided are considered. A file sharing application is pretend to understand capability of the proposed algorithms in mitigating attacks. For realism, peer and resource parameters are based on several empirical studies. Service and recommendation based attacks are simulated.Nine different behavior models representing individual, collaborative, and identity varying malicious peers are studied in the experiments. Observations reveal that malicious peers are identified by good peers. The attacks are mitigate even if they increase high reputation. Collaborative recommendation-based attacks might be victorious when malicious peers make unfairness among good peers. Identity changing is not a good attack approach.**

Keywords
Trust management, Reputation, peer to peer systems, security

I.INTRODUCTION

PEER-TO-PEER (P2P) systems rely on collaboration of peers to complete tasks. Ease of the stage malicious activity is a threat for security of P2P systems. create long-term trust relationships among peers can offer a more secure environment by dropping risk and ambiguity in future P2P interactions. However, establish trust in an strange entity is hard in such a malicious environment. Furthermore, trust is a social notion and rigid to calculate with numerical values. Metrics are needed to characterize trust in computational models.

classify peers as both trustworthy or untrustworthy is not adequate in most cases. Metrics should have exactitude so peers can be ranked according to trustworthiness. Interactions and feedbacks of peers afford information to measure trust among peers. Interactions with a each peer provide certain information about the particular peer but feedbacks might contain certain deceptive information. This makes appraisal of trustworthiness a challenge. In the presence of an authority, a central server is a prefer way to accumulate and manage trust information, e.g., eBay.

The central server steadily stores trust information and defines trust metrics. Since there is no central server in most P2P systems, peers arrange themselves to store and manage trust information about each other. Management of trust information is needy to the structure of P2P network. In distributed hash table (DHT) based approach, each peer becomes a trust container by storing feedbacks about other peers. Global trust information stored by trust holders can be accessed through DHT efficiently. In unstructured networks, each peer supplies trust information about peers in its neighborhood or peers interact 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.

This propose a Self-organizing Trust model (SORT) that aims to reduce malicious activity in a P2P system by establish trust relations among peers in their proximity. No a priori information or a trusted peer is used to influence trust establishment. Peers do not aim to collect trust information from all peers. Each peer develops its hold 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 separate 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 implicit to be strangers to each other at the beginning. A peer become an coworker of another peer after providing a service, e.g., uploading a file.

Peer-to-peer (P2P) network is a kind of decentralized and distributed network architecture in which individual nodes in the system (called *"peers"*) perform as both suppliers and consumers of resources, in contrast to the centralized client server model where client nodes demand access to resources provide by central servers.In a peer-to-peer network, tasks (such as searching for files or streaming audio/video) are mutual amongst multiple interconnected peers who each make a part of their resources (such as processing power, disk storage or network

ISSN: 2278 – 7798

bandwidth) directly presented to other network participants, without the need for centralized coordination through servers.

The peer-to-peer movement permitted millions of Internet users to connect "directly, form groups and collaborating to suit user-created search engines, virtual supercomputers, and file systems." The basic idea of peer-to-peer computing was envisioned in earlier software systems and networking discussions, reaching back to ethics stated in the first Request for Comments, RFC .

II. RELATED WORK

In this section, literatures of several highly related research areas to SORT, A secure and effective reputation based distributed P2P trust management model has been proposed which has advantages in struggle various malicious behaviors and uses Self Certified Cryptographic Exchanges between the peers. This can successfully track each peer's contribution in the system and allows peers to store their reputations locally and switch that information with other peers by a two-party cryptographic protocol[1]. this reputation system, called EigenTrust, has been shown to significantly reduce the number of inauthentic files on the network, even under a variety of conditions where malicious peers collaborate in an attempt to intentionally subvert the system[2].

The open and unknown nature of a P2P network makes it an ideal average for attackers to spread malicious content. In this paper, we describe a reputation-based trust management protocol for P2P networks where users rate the reliability of parties they deal with, and share this information with their peers. The protocol helps establishing trust among good peers as well as identifying the malicious ones. [3] The scheme has a considerably low gossiping message overhead, i.e. O(nlog2 n) messages for n nodes. Bloom filters reduce the memory overhead per node to 512 KB for a 10,000-node network. We evaluate the performance of GossipTrust with both P2P file-sharing and parameter-sweeping applications. The simulation results demonstrate that GossipTrust has small aggregation time, low memory demand, and high ranking accuracy. These results suggest promising advantages of using the GossipTrust system for trusted P2P computing[4]. Gnutella is not a pure power-law network, its current configuration has the benefits and drawbacks of a power-law structure, and the Gnutella virtual network topology does not match well the underlying Internet topology, hence leading to ineffective use of the physical networking infrastructure.[5] formalism for trust which provides us with a tool for precise discussion. The formalism is implementable: it can be embedded in an artificial agent, enabling the agent to make trust-based decisions. Its applicability in the domain of Distributed Artificial Intelligence (DAI) is raised.[7] PariSync is formed by two modules: a topology module, that chooses for each node a small subset of neighbors with which to exchange timing information and an estimation module, that assembles the information into an estimate of the node's offset and drift from a global virtual clock emerging from the consensus of all peers[9]. A hopping sequence is used to determine the spreading of data streams across the channels (i.e. the order and the amount of time each channel is occupied). This may be realised either probabilistically based on an attribute such as network distances or routing metrics, or it may be simply implemented on a per packet basis[10].

III.SYSTEM DESIGN

A file sharing simulation program is implemented in .NET to observe results of using SORT in a P2P environment. SORT handle attacks, how much attacks can be mitigate, how much recommendations are not accommodating in correctly identify malicious peers, and what kind of attackers are the most harmful. The simulation runs as each cycles. Each cycle represents a time of period. When a peer Downloading a file is know as an interaction. A peer distributing files is called an uploader. When each peer downloading a particular file is called a downloader. The set of peers who downloaded a file starting a peer are called downloaders of the peer. An current download/upload operation is called a session.

A file search demand reaches up to 40 percent of the network and income online uploaders only. A file is downloaded from one uploader to simplify integrity inspection. All peers are tacit to have antivirus software so they can sense infected files. Four special cases are studied to understand things of trust computation methods under attack situation

There No trust Trust information which is not used for uploader collection. An uploader is chosen according to its bandwidth. This method is the base case to recognize if trust is useful to mitigate attacks.In No reputation query An uploader is elected based on trust information but peers do not demand recommendations from further peers. Trust computation is through based on SORT equations but reputation (r) value is constantly zero for a peer. This method will help us to assess if recommendations are helpful.

The outcome of experiments of individual attackers. For each type of individual attacker, two part network topologies are formed: one with 10 percent malicious and one with 50 percent malicious. Each network topology is experienced with four trust computation methods. In the experiments, a hypocritical attacker behave malicious in 20 percent of each and every one interactions. Using trust information does not answer all security problems in P2P system but can improve security and efficiency of systems.

IV.SYSTEM COMPUTATIONAL MODEL

We make the next assumption. Peers are like in computational power and accountability. Here there are no privileged, centralized, or trusted peers to handle trust relationships. Peers irregularly leave and join the network. A peer provide services and uses services of others. For ease of conversation, one type of interaction is measured in the service framework, i.e., file download.

4.1 Network construction:

ISSN: 2278 – 7798

Client-server computing or networking is a spread application construction that partition tasks or workloads among service providers (servers) and service requesters, called clients. Often clients and servers function over a computer network on part hardware. A server machine is a high-performance host that is running successively one or more server programs which it can share its resources with clients. A client also can shares any of its resources; Clients therefore begin communication session by servers which await (listen to) incoming requests

The node name is nothing but the system name, which can be given by the user. The next value is host number which can be get from our network configuration details. The next one is the IP address of the system.These can be identified by a simple command on DOS environment. The command 'netstat' helps to get all details about the network configuration.

Topology Construction:

Here it we use mesh topology because of its unstructured nature. Topology is constructed by reaching the names of the nodes and the acquaintances between the nodes as input from the user. While realization each of the nodes, their associated port and ip address is also obtained. For successive nodes, the node to which it should be coupled is also received from the user. While adding nodes, contrast will be done so that there would be no node duplication. Then it identify the source and the destinations

.

4.2 Server Module

The module contains both the server authentications. The admin would have the privilege to view the whole process processed by the user. Once the user registers, user would be able to view only the authenticated page. The personal information and the data which are transferred by the user can be viewed by the user. The login in the secure module is non-dynamic and secure. Once logged in the server would be able to receive the data packets. The server has much number process and tasks such as listed below.

The network is classified by workgroups. The active and the connected systems over the network are obtained with the use of this module. Once logged in to the process, the module obtains the active systems and displays to the user. The user would be able to choose the system to which the data needs to be transmitted by file transfer.

Optimal Distribution

This module implements the optimized node selection among different topologies and nodes. This module helps to select optimal node by considering the resources. An algorithm has proposed for spreading the messages, so that all the available paths are effectively utilized. Recall that a row is a collection of nodes in the contour that are at the same distance from the source.

Transitory Counters and security analysis

Passing counters is counting message transmission between sources to destination. Every time checking how many packets handled in this link. The counters check whether the packets travelled in these links and path reached the destination successfully. The counter then calculates the maximum amount of packets transmitted in the particular path. When it founds the route which transmits the maximum amount of packets successfully, it stores the path details to select the path to transmit the packets again. Hence the packets can be transmitted through this predetermined path with the maximum transfer rate. The server module contains the security analysis over a link and topology.

4.3 Behavioural Modal and Reconfiguration scheme:

The behavioral model and link gradient module have introduced to reroute the dates with the history and effectiveness of the transferred data. It requires a list of the application's nodes (host, process) and the location of the end-to-end response time data as well as security guarantee.

The link incline mechanism quantifies how a change in the link latency, security for each link affects the response time and is defined as a vector. Intuitively, the link gradient of a link is a partial derivative that specifies the rate at which the system's response time changes per unit change in the link latency of communication link, assuming that the latencies of all other links remain constant. The link analysis can be used to approximate how the response time of the system would be affected by a change in link latencies.

4.4 Peer Clock Analysis

In this module we are going to implement the individual port based clock monitoring port, this port calculating the every transactions of the packet transmission and procedural cost of the node, at the same time calculating the neighbor node communication level, each and every clock based hopping process are updating in centralized unit, the port clock declarations are depending on the contact-initiation part, the data transmission part, and the resynchronization/adjustment part, these process are all controlled by CCPH algorithm.

4.5 Port Hopping Method

The extension to multiple clients per server is based on communication since each client considers the server's clock as the reference clock, it can interact with the server independently of the other clients. For scalability reasons it is desirable that the server has more than one worker ports open in each time period, so we are going to measure the sequences or even by the same sequence calculating through the CSS algorithm

4.6 Searching module:

The proposed project is designed for the purpose of searching for files and folders which are shared over a Local Area Network. It is particularly useful in organizations where a lot of folders are shared on individual systems. Searching files and downloading from a shared folder won't give any details about the file movement (i.e.) the user couldn't identify the users who are all downloaded the files for the shared folder. This project also concentrates on tracking those file movement in the LAN. It gives the number files download ,source machine IP address and Destination machine IP address , time and date etc., so the admin can view the list of IP's connected in the network and downloaded files. And the admin can upload files to the shared folder.

ISSN: 2278 – 7798

4.8 Object tracking:

This module provides the implementation previous module. That is analysis the moving files on two phase. In this module get the frequent sequential files and group of relationship in a distributed manner on the detection. The network partitions the trajectories of file moving is "identify the files" from our algorithm. And finally we discovered the track of moving files efficiently.

4.7 Attack Model:

The attacker module describes the creation and identification of global adversary over network. This helps to identify the adversary who is doing misbehaving activities and other attack in the network

4.8 Reports

All the data transactions and impostor information are familiar to the administrator. The administrator can view all the reports and observe the network paths. The whole histories of data are maintain by the administrator. So that, the administrator can able to construct the denial of service of the intruder since the reports module, one communication port might be kept open continuously by various hopping.

V. METHODOLOGY

Encryption using DES algorithm:

I have employed DES algorithm for encrypting the files.

- DES is a *block cipher*— it returns a ciphertext exactly of the same size as the specified plaintext (64 bits).
- DES uses 56-bits key amount and operate on 64-bit blocks.
- Get input files and spilt into blocks and Assign For Encryption
- That Must be 64 bits, 8 bytes. allocate this key to the user who will decrypt this file.
- Get the Key for the file to Encrypt.
- Encrypt the file using secret key. Starts to encrypt a file
- Remove the Key from memory.
- send the converted stream to Output File

File transfer:

To transfer the encrypted files, the following techniques are used.

- Socket
- TCP Listener
- TCP Client
- Network Stream

1. First specify the destination IP/ System Name.
2. The IP and System Name can be retrieved from DNS(which provides Domain nameresoultion functionality)
3. TCP Listener listens for connections from TCP network clients.
4. TCP client Proides client connection for TCp networks.
5. Socket creates endpoint to the communication.
6. In the proposed model berkeley socket interface has been implemented.
7. For socket implementation the following parameteres should be used.
    a. Address family
    b. Socket type
    c. Protocol Type

The encrypted files will be converted into bytes format, and it will be stored in an array. TCP client starts network services with the help of sockets.

Through the network stream option file receive stream may get. DNS gets the host name and receives the data. Network stream writes the received data into the input files and encodes the data.

No trust:

Trust information is not used for uploader selection. An uploader is preferred according to its bandwidth. This process is the base case to know if trust is helpful to mitigate attacks.

No reputation query:

An uploader is preferred based on trust information but peers do not request recommendations from other peers. Trust computation is done based on SORT equations but reputation(r) value is always zero for a peer. This technique will help us to assess if recommendations are cooperative.

Flood reputation query:

SORT equations are use but a reputation query is flooded to the whole network. This process will help us to understand if receiving more recommendations is helpful to mitigate attacks. A peer may request a recommendation from strangers

Attack Model:

Against the following attacks:

1.Naive:

The attacker constantly uploads infected/ inauthentic files and gives unfairly low recommendations about others [22].

2. Discriminatory:

The attacker selects a group of fatalities and always uploads infected/inauthentic files to them [22], [5]. It gives unfairly low recommendations about fatalities. For other peers, it behave as a good peer.

3. Hypocritical:

The attacker uploads infected/iauthentic files and gives unfairly low recommendations with x percent probability [3], [5]. In the other period, it behaves as a good peer.

4. Oscillatory:

The attacker builds a high reputation by being good for a long time period. Then, it behaves as a naive attacker for a short period of occasion. After the malicious period, it becomes a good peer yet again.

VI. CONCLUSION AND FUTURE WORK

A trust model for P2P networks is existing, in which a peer can build up a trust network in its closeness. A peer can separate malicious peers in the region of itself as it develop trust relationships with good peers. There is two

ISSN: 2278 – 7798

context of trust, one is service and other recommendation contexts, are defined to calculate capabilities of peers in given that services and giving recommendations. Interactions and recommendations are measured with satisfaction, weight, and fading effect parameters. A recommendation contain the recommender's own knowledge, information from its friends peer, and level of self-reliance in the recommendation. These parameters provided us a improved estimation of trustworthiness. Individual, collaborative, and pseudonym changing attackers are considered in the experiments. Damage of collaboration and pseudospoofing is reliant to attack actions. Although recommendations are vital in hypocritical and oscillatory attackers, pseudospoofers, and collaborators, they are less valuable in naive and discriminatory attackers. By extend the trust model by execution of clock synchronization concept to avoid and reduce spoofing attacks and topology reconfiguration problems. Tracking node actions and logs in order to find the malicious ratio by using Css algorithm (clock synchronization algorithm) Priority based reconfiguration(protocol).

VII. REFERENCES

[1] K. Aberer and Z. Despotovic, "Managing Trust in a Peer-2-Peer Information System," Proc. 10th Int'l Conf. Information and KnowledgeManagement (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] 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

[12] Manolis Sifalakis, Stefan Schmid and David Hutchison "Network Address Hopping" A Mechanism to Enhance Data Protection for Packet Communications.

[13] P. Bertasi, M. Bonazza, N. Moretti, E. Peserico "PariSync: Clock Synchronization in P2P Networks"conf. Department of Information Engineering University of Padova.

[14] R. Zhou and K. Hwang, "Powertrust: A Robust and Scalable Reputation System for Trusted Peer-to-PeerComputing,"IEEETrans.Parallel and Distributed Systems, vol. 18, no. 4, pp. 460-473, Apr. 2007.

[15] Y. Wang and J. Vassileva, "Bayesian Network Trust Model in Peer-to-Peer Networks," Proc. Second Workshop Agents and Peer-to-Peer Computing at the Autonomous Agents and Multi Agent Systems Conf. (AAMAS), 2003.

[16] P. Victor, C. Cornelis, M. De Cock, and P. Pinheiro da Silva, "Gradual Trust and Distrust in Recommender Systems," Fuzzy Sets Systems, vol. 160, no. 10, pp. 1367-1382, 2009.

[17] G. Swamynathan, B.Y. Zhao, and K.C. Almeroth, "Decoupling Service and Feedback Trust in a Peer-to-Peer Reputation System," Proc. Int'l Conf. Parallel and Distributed Processing and Applications (ISPA), 2005.

[18] M. Gupta, P. Judge, and M. Ammar, "A Reputation System for Peer-to-Peer Networks," Proc. 13th Int'l Workshop Network and Operating Systems Support for Digital Audio and Video (NOSSDAV),2003.

[19] S. Staab, B. Bhargava, L. Lilien, A. Rosenthal, M. Winslett, M.Sloman, T. Dillon, E. Chang, F.K. Hussain, W. Nejdl, D. Olmedilla,and V. Kashyap, "The Pudding of Trust," IEEE Intelligent Systems,vol. 19, no. 5, pp. 74-88, 2004.

[20] M. Virendra, M. Jadliwala, M. Chandrasekaran, and S. Upadhyaya,"Quantifying Trust in Mobile Ad-Hoc Networks," Proc.IEEE Int'l Conf. Integration of Knowledge Intensive Multi-Agent Systems (KIMAS), 2005.

[21] E.J. Friedman and P. Resnick, "The Social Cost of Cheap Pseudonyms," J. Economics and Management Strategy, vol. 10, no. 2, pp. 173-199, 2001.

ISSN: 2278 – 7798

[22] S. Xiao and I. Benbasat, "The Formation of Trust and Distrust in Recommendation Agents in Repeated Interactions: A Process-Tracing Analysis," Proc. Fifth ACM Conf. Electronic Commerce (EC),2003.

[23] A. Habib, D. Xu, M. Atallah, B. Bhargava, and J. Chuang, "A Tree-Based Forward Digest Protocol to Verify Data Integrity in Distributed Media Streaming," IEEE Trans. Knowledge and Data Eng., vol. 17, no. 7, pp. 1010-1014, July 2005.

Authors

U.sahana

She Completed Her UG At Kalasalingam University And Now She Is Currently Pursuing Post Graduation (M.Tech) In Department Of InformationTechnology, SNS College Of Technology, Coimbatore-35.

G.selvavinayagam

He Completed His UG At Bharathiar University And Done His PG At Anna University. At Present He Is Undertaking His Research Work At Anna University .Now He Currently Working As A Assistant Professor, Department Of Information Technology In SNS College Of Technology, Coimbatore. His Area Of Interest Are Automata Theory, Theoretical Computer Science And Network Security. He Attended Many Conference And Published 1 National And 12 International Research Paper In His Area Of Interest.

ISSN: 2278 – 7798