Volume 2, Issue 9, September 2012 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Usage of distributed identities and Decentralized Recommendation chains in Peer to Peer Reputation Mgt.

Dr. R. V. Krishnaiah
Principal,
DRK Institute of science& Tech.
Hyderabad, India

J. Yeshwanth
M.Tech Pursuing
DRK Institute of science& Tech.
Hyderabad, India

ABSTRACT-- One of the world habituated system is p2p in this Files can be shared directly between systems on the network without the need of a central server. In other words, each computer on a P2P network becomes a file server as well as a client. We describes the Reputation Systems for P2P networks more ambitious approach to protect the P2P network without using any central component, and thereby harnessing the full benefits of the P2P network. The reputations of the peers are used to determine whether a peer is a malicious peer or a good peer. Once detected, the malicious peers are ostracized from the network as the good peers do not perform any transactions with the malicious peers. Expulsion of malicious peers from the network significantly reduces the volume of malicious activities [1]. All peers in the P2P network are identified by identity certificates. The reputation of a given peer is attached to its identity. The identity certificates are generated using self-certification, and all peers maintain their own certificate authority which issues the identity certificates to the peer. Each peer owns the reputation information pertaining to all its past transactions with other peers in the network, and stores it locally. A two-party cryptographic protocol not only protects the reputation information from its owner, but also facilitates secure exchange of reputation information between the two peers participating in a transaction.

Index Terms-- peer, digital certificate, Encryption, Cryptography.

I. INTRODUCTION

"Peer to Peer." In a P2P network, the "peers" are computer systems which are connected to each other via the Internet. Files can be shared directly between systems on the network without the need of a central server. In other words, each computer on a P2P network [2] becomes a file server as well as a client.

The only requirements for a computer to join a peer-to-peer network are an Internet connection and P2P software. Common P2P software programs include Kazaa, Limewire, BearShare, Morpheus, and Acquisition. These programs connect to a P2P network, such as "Gnutella," which allows the computer to access thousands of other systems on the network [3] [11].

Once connected to the network, P2P software allows you to search for files on other people's computers. Meanwhile, other users on the network can search for files on your computer, but typically only within a single folder that you have designated to share. While P2P networking makes file sharing easy and convenient, is also has led to a lot of software piracy and illegal music downloads. Therefore, it is best to be on the safe side and only download software and music from legitimate websites[12].

Peer-to-peer (P2P) computing or networking is a distributed application architecture that partitions tasks or workloads among peers. Peers are equally privileged, equipotent participants in the application. They are said to form a peer-to-peer network of nodes. Peers make a portion of their resources, such as processing power, disk storage [1]

Reliability: The malfunction on any given node will not affect the whole system Peer-to-peer network are classified into two types

a. Structured Systems
b. Unstructured Systems

Structured P2P networks employ a globally consistent protocol to ensure that any node can efficiently route a search to some peer that has the desired file, even if the file is extremely rare. Such a guarantee necessitates a more structured pattern of overlay links. [10] By far the most common type of structured P2P network is the distributed hash table, in which a variant of consistent hashing is used to assign ownership of each file to a particular peer, in a way analogous to a traditional hash table's assignment of each key to a particular array slot. An unstructured P2P network is formed when the overlay links are established arbitrarily. Such networks can be easily constructed as a new peer that wants to join the network can copy existing links of another node and then form its own links over time. In an unstructured P2P network **[3]**, if a peer wants to find a desired piece of data in the network, the query has to be flooded through the network to find

as many peers as possible that share the data.

Abdul-Rahman and Hailes **[4]** have proposed another trust model and the corresponding metrics. They argue that Bayesian probability may not be the best metric for representing degree of trust, because probability is inher-ently transitive while trust is not. In addition, the authors provide methods for combiningrecommendations and use the context of recommendations and recommender weights to evaluate the reputations from recommendations. Aberer and Despotovic have proposed completely distributed solution for trust management over the P-Grid peer-to-peer network. They store reputation data in the form of a binary search tree, over the network. Any agent looking for the recommendation data of another agent searches the P2P network and computes the reputation from the recommendations received.

Chen and Singh Scheinetal. also provide trust models, similar to those mentioned above. Dellarocas has enumerated the design challenges in the online reporting systems. Dellarocas surveys online reputation, reporting mechanisms, and the corresponding issues. In addition, the author provides a good overview of recommendation repositories, professional rating sites, collaborative filtering systems , and regression approaches. Dellarocas also enumerates the attacks on reputation systems and techniques for foiling those attacks.

II. APPROACHES

In our proposed system we have seven modules they are

- Login Module
- Active Node in Dynamic root
- Group Controller
- Trusted Group Members
- Data Transfer
- Find Group Key
- Block Untrusted User

A. Login Module

In this module it is responsible for user authentication and registration of new users.

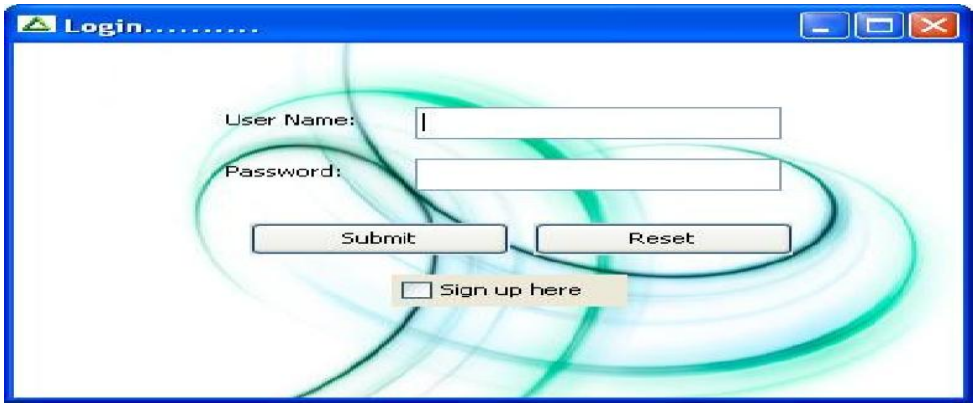


Fig 1. Login

B. Active Node in Dynamic root

In our communication group have number of client[9] nodes are interconnected in the server. when a new number joins or leaves the communication group only it's reflecting for local group each group have separate group key for communication in between who are in the communication group in that time.

C. Group Controller

Two types of secrecy are provided by this module they are forward and backward secrecy[9] .the forward secrecy is used to prevent a leaving user or expelled group member to continue accessing the group communication .the backward secrecy is used to prevent a new member from decoding the messages exchanged before it joined the group.

D. Trusted Group Members

We divide the multicast group into regional subgroups .each subgroup is independently managed by a subgroup controller like a separate multicast group with its own sub group key[10] .our protocol is responsible for reducing the overload of the group controller.

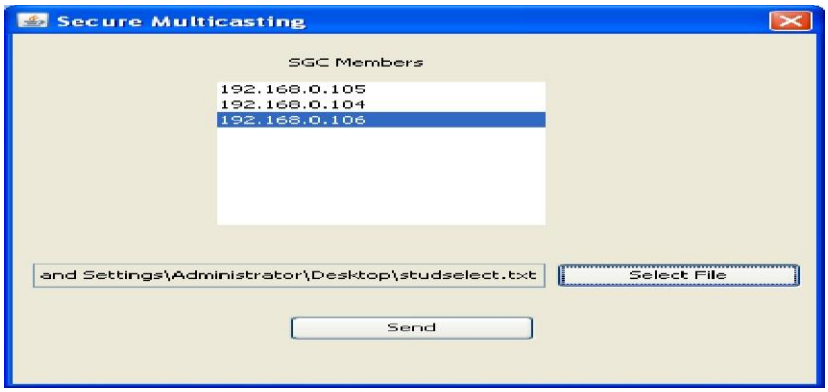


Fig 2: Trusted Group Members

Data Transfer In our multicast communication group mainly concentrate[6] on enabling the data transfer among the server and multiple clients in the network communication.

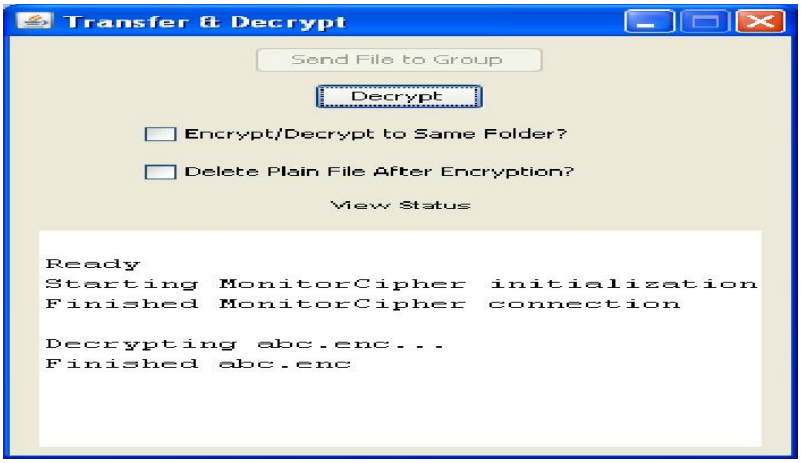
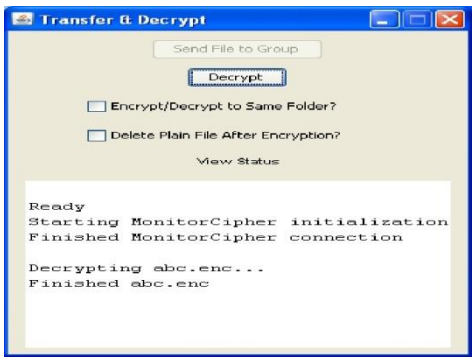


Fig 3: Data Transfer Find Group Key

When a new member joins in the communication group then we create a new sub group key for only for its local group as well as existing member leaves from the communication group after that they don't want to access the local sub group so it need to be refreshed. Block Untrusted User.



Trusted users are formed a group. If a new member request to join in group, the IP Address will be validated[8]. IP address will be validate with the help of subnet masking, such as the Class A, Class B, Class C Part of the IP address. If the IP is not matched with the trusted group then it will not be allowed to enter into the trusted group.

III. CONCLUSION

This paper presents self-certification, an identity management mechanism, reputation model, and a cryptographic protocol that facilitates generation of global reputation data in a P2P network, in order to expedite detection of rogues. A reputation system for peer-to-peer networks can be thwarted by a consortium of malicious nodes. Such a group can maliciously raise the reputation of one or more members of the group. There is no known method to

protect a reputation system against liar farms and the absence of a third trusted party makes the problem of liar farms even more difficult.

The self-certification-based identity generation mechanism reduces the threat of liar farms by binding the network identity of a peer to his real-life identity while still providing him anonymity. The Identity mechanism is based on the fundamental that the ranks of the peers are more relevant than the absolute value of their reputation. The cost of this security is the difference in the ranks of the providers because of the use of the proposed mechanism. The global reputation data are protected against any malicious modification by the third party peer and are immune to any malicious modifications by their owner. The proposed protocol reduces the number of malicious transactions and consumes less bandwidth per transaction than the other reputation systems proposed in its category. It also handles the problem of highly erratic availability pattern of the peers in P2P networks. Currently, the reputation of the provider is considered and the reputation of the requester is ignored.

IV. FUTURE ENHANCEMENT

This system can be extended to encapsulate the reputations of both the provider and the requester. In addition, instead of generic number values, the reputation values can be modified in accordance with the context of the reputation.

REFERENCES

[1] Continuous Neighbor Discovery in AsynchronousSensor NetworksReuven Cohen and Boris Kapchits, IEEE/ACM TRANSACTIONS ON NETWORKING, VOL. 19, NO. 1, FEB 2011.

[2] L. Gu, et al., "Lightweight detection and classification for wireless sensor networks in realistic environments," ACM Sensys, 2005.

[3] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," IEEE J. Sel. Areas Commun., vol. 18, no. 3, pp. 535-547, 2000.

[4] T. A. Henzinger, "The theory of hybrid automata," Verification Digital Hybrid Syst., vol. 170, pp. 265-292, 2000.

[5] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. B. Srivastava, "Coverage problems in wireless ad-hoc sensor networks," IEEE Infocom, Anchorage, AK, 2001.

[6] M. Hefeeda and M. Bagheri, "Randomized k-coverage algorithms for dense sensor networks," IEEE Infocom, Anchorage, AK, 2007.

[7] O. Dousse, C. Tavoularis, and P. Thiran, "Delay of intrusion detection in wireless sensor networks," ACM MobiHoc, Florence, Italy, 2006.

[8] L. Stabellini and A. Proutiere, "Evaluating delay and energy in senetworks with sporadic and correlated traffic," 7th Scandinavian Workshop on Wireless Ad-Hoc Networks, Johannesbergs Slott, Sweden, 2007.

[9] K. Shuaib, M. Alnuaimi, M. Boulmalf, I. Jawhar, F. Sallabi, and A.Lakas, "Performance evaluation of IEEE 802.15.4: experimental and simulation results," J. Commun., vol. 2, no. 4, pp. 29-37, 2007.

[10] F. Cuomo, S. Della Luna, P. Todorova, and T. Suihko, "Topology formation in IEEE 802.15.4: cluster-tree characterization," IEEE PerCom, Hong Kong, 2008.

[11] F. Shu, M. Zukerman, T. Sakurai, and H. L. Vu, "Packet loss analysis of the IEEE 802.15.4 MAC without acknowledgments," IEEE Commun. Lett., vol. 11, no. 1, pp. 79-81, 2007.

[12] X. Ling, Y. Cheng, J. W. Mark, and X. (Sherman) Shen, "A general analytical model for the IEEE 802.15.4 contention access period," IEEE WCNC, Hong Kong, 2007.

