ISSN 23499842(Online), Volume 1, Special Issue 2(ICITET 15), March 2015
International Journal of Innovative Trends and Emerging Technologies

ENSURE DATA INTEGRITY FOR SHARED CLOUD DATA USING TRUST BASED ACCESS CONTROL

[1] ENIYA S, [2]JEYARANI R ,[3] PANDITHURAI O,[4] VELUMADHAVARAO R
[12]UG Department of computer science,Rajalakshmi Institute of Technology Chennai,Tamil Nadu,India
[34]Assistant professor Department of computer science,Rajalakshmi Institute of Technology Chennai,Tamil Nadu,India
[1]iniyasubramani@gmail.com, [2]jeyarani36@gmail.com, [3]pandics@ritchennai.edu.in,[4]velumadhavarao.r@ritchennai.edu.in

Abstract-

Cloud computing is a fast mounting technology owing to its resource sharing characteristics and proficient storage services. The flip side to this technology is security issues of confidentiality as data stored in cloud environment are outsourced. Hence, the need to uphold integrity resulted in the advancement of various privacy preserving mechanisms. To check the level of integrity being provided by these mechanisms tracking of shared cloud data are done continuously by the data owner. Intruders are identified by the modification of data that is been shared. Trust based access control is provided for the intruders through ant colony optimization and the intruders will be blocked from accessing the shared cloud data. To verify the level of integrity achieved owner endlessly monitors the shared cloud data.

KEYWORDS: Cloud computing, Integrity, Trust based access, Ant colony optimization.

I. INTRODUCTION

A Network is a group of two or more devices linked together to share information. This information is restricted to computers within the connected network. The information shared in the network cannot be accessed by any device outside the network. In such cases, each system outside network must have a local copy of the data which will affect efficient sharing of

data. Overcoming the drawback of network systems, cloud computing provides its users with on-demand network access to a shared pool of computing resources, thus supplying users with various services in the cloud with accessibility from anywhere around the world using any device.

The cloud providers such as Microsoft, Google, and Amazon store the user's applications in a database at a remote server. Cloud computing provides an effective way of sharing resources and data to multiple users. Open cloud computing environment does not support identity based security Cloud data leakage is a top issue in cloud security and trust. To address this problem, all stakeholders need to have the ability to track their data and information in the cloud. Tracking requires strong transparency management and accountability Via event management.

Access control is one of the important measures to ensure the security of cloud computing. Earlier access control technology can not only ensure valid users but also prevents unauthorized users and also solve security problems. Trust mechanism will be introduced into access control model in this paper. Mutual trust between users and cloud service nodes are ensured through trust mechanism. Only trusted users have access to the Cloud, and simultaneously users can select the most credible cloud service nodes. Trust based access control not only considers

user's behavior trust and ensure that user's access request poses no malicious threat to cloud server, and also takes cloud service node's credibility into account.

II. RELAED WORK

Boyang Wang,[20] deals with, data integrity can be audited publicly, users need to signatures on all the blocks. Different blocks are signed by different users because of data modifications done by different users**.** This paper proposes Public Auditing for Shared Data with Efficient User Revocation in the Cloud to allow the cloud to re-sign blocks on existing users during user revocation, ensure that existing

users do not need to download and re-sign blocks by themselves. And public verifier is able to audit the integrity of shared data without retrieving the entire data from the cloud, even if some part of shared data has been re-signed by the cloud. Results show that our mechanism can significantly improve the efficiency of user revocation.

Cong Wang,[18] proposes that the public audit ability for cloud storage is important so that users can resort to a third party auditor to check the integrity of outsourced data. To introduce an effective TPA, the auditing process should bring no new vulnerabilities towards user data privacy, and no additional on-line burden to user. In this paper, propose a secure cloud storage supporting privacy-preserving public auditing.

Ryan K L Ko,[3] deals with ,ability to track data from its creation to its current state or its end state will enable the full transparency and accountability in cloud . By analyzing and utilizing provenance, it is possible to detect data leakage threats and alert data administrators and owners; and increasing needs of trust and security for customers' data.

Wang Danrul,[4] proposes that trust relationships between users and cloud service nodes are established by mutual trust based mechanism. Security problems of access control are solved by implementing MTBAC model into cloud computing environment.

.

III. EXISTING SYSTEM

In existing system, Data and files are shared by data owner in an open cloud environment. Data are divided into many small blocks, where each block is independently signed by the owner .Identity of data owner is preserved by various privacy preserving mechanisms. Data user is able to publicly audit the shared data integrity. The identity of the signer on each block is kept private from public verifiers, who are able to verify shared data integrity without retrieving the entire file. Many mechanisms allow not only a data owner itself but also a public verifier to efficiently perform integrity checking without downloading the entire data from the cloud.

LIMITATIONS:

Privacy preserving mechanism unconditionally protects the identity and also restricts the ability of the data owner to reveal the identity of the intruder. We do not address the traceability concept.

We do not address the question of how to prove data freshness although still preserving identity privacy.

IV. PROPOSED SYSTEM:

This paper proposes a novel system for checking the level of integrity provided by various privacy preserving mechanisms for shared cloud data. The data owner keeps on monitoring the particular file that is been shared by the owner. The continuous monitoring is performed using file system watcher. This will generate an event log report. Data owner will analyze the report to identify the intruders who will modify the original data and reduce its integrity. To overcome this Users are monitored using traceability mechanism to determine whether the shared data is modified by intruders. To restrict the entry of intruders, trust based access control model is proposed based on ant colony optimizing algorithm. It is a computational method that is inspired from the way of ant colony seeking the shortest path from the food resource to the nest without visual aid. It is feasible to apply ACO to the field of trust management in cloud computing environment. Fig1.1 explains clearly about overall architecture of the paper. Data owner initially shares the data. After proper authentication into the system the data owner acquires the privilege to track the particular file being shared. Based on the analysis of report that is been generated during tracking process trust based access is provided to the shared cloud data. After providing trust access monitoring will be continuously to ensure data integrity.

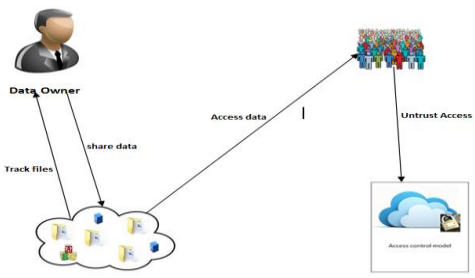


Fig 1.1: Architecture

A. Tracking

Data owner after successful authentication uploads the file into the cloud sharing environment.

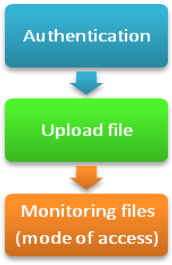


Fig 1.2: Process involved in tracking

Fig 1.2 explains about the process involved in tracking process. New users register to the system by filling the registration form. Registered users enter into the system using individual username and password On successful login, users obtain privilege to access the shared files. Admin enter into the system and choose the file to be monitored. Access mode of the file chosen is continuously monitored by the owner. Monitoring is done by the function File System Watcher. Log report is generated at the end of this monitoring process.

B. Trust based access

Trust based access to the intruders is provided using ant colony optimization (ACO).

PSEUDO CODE

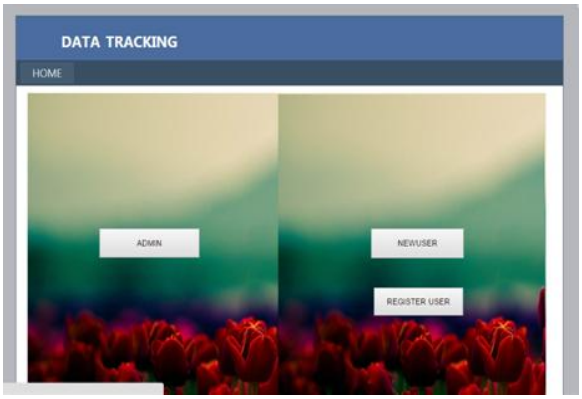
```
Input: Problem Size n
Output:
  Create Heuristic Solution(n)
  Cost()

Pheromone  Initialize Pheromone()
While (Stop Condition())
   For ( To )
        Construct Solution(Pheromone, n)
        Cost()
      If ( )

  End
      Local Update And Decay  Pheromone(Pheromone,,,)
   End
   Global Update And Decay Pheromone(Pheromone,,, )
End
Return ()
```

between ants regarding a good path between the colony and a food source in an environment. This mechanism is called stigmergy. The probabilistic step-wise construction of solution makes use of both history (pheromone) and problem-specific heuristic information to incrementally construct a solution piece-by-piece. Each component can be selected if it has not already been chosen (for most combinatorial problems), and for those components that can be selected from given current component, their probability for selection is defined as the maximizing contribution to the overall score of selecting the component, is the heuristic coefficient (commonly fixed at 1.0). The pheromone value for the component, is the history coefficient, and is the set of usable components. A greediness factor () is used to influence when to use the above probabilistic component selection and when to greedily select the best possible component. A local pheromone update is performed for each solution that is constructed to dissuade following solutions to use the same components in the same order. It represents the pheromone for the component (graph edge) (), is the local pheromone factor, and is the initial pheromone value. At the end of each iteration, the pheromone is updated and decayed using the best candidate solution found thus far (or the best candidate solution found for the iteration), as follows: where represents the pheromone for the component (graph edge) (), is the decay factor, and is the maximizing solution cost for the best solution found so far if the component is used in the globally best known solution, otherwise it is 0.

V. EXPERIMENTAL WORK

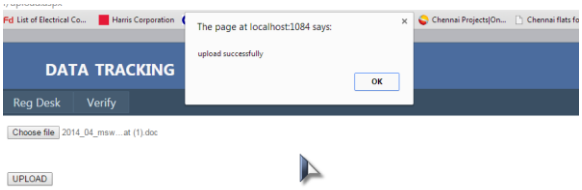
AUTHENTICATION:

Data owner and user authentication is done.

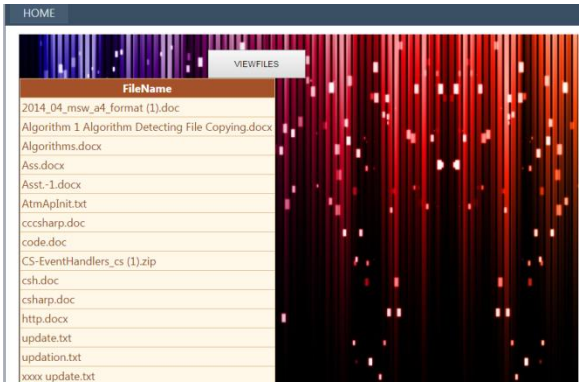


UPLOAD AND SHARE FILES:

On successful authentication data owner upload the file that is to be shared.

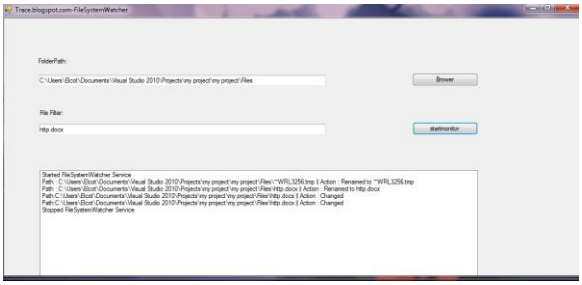


The uploaded files shared by the owner are listed to the users for accessing, but the user do not have the privilege to modify it.



TRACKING:

Data owner continuously monitors the shared file using File System Watcher and generates an event log report. Based on this intruders will be identified and trust based access will be provided.



VI. CONCLUSION AND FUTURE ENHANCEMENT

In this paper, the level of integrity of shared cloud data is checked by tracking using File System Watcher. Trust based access protect shared data from intruders and also will improve data integrity to greater extent. Trust model of the cloud service node is based on the ant colony optimization. Access control in cloud computing environment on the basis of the trust between users and the cloud service nodes, and protect users and cloud servers security efficiently. Problem for our future work is how to prove data freshness (prove the cloud possesses the latest version of shared data) while still preserving high level of shared data integrity.

VII. REFERENCES

[1]"Panda: Public Auditing for Shared Data with Efficient User Revocation in the Cloud," *IEEE Transactions on Services Computing*, 2014, accepted.

[2] C. Wang, S. S. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," *IEEE Transactions on Computers*, vol. 62, no. 2, pp. 362–375, 2013.

[3] B. Wang, B. Li, and H. Li, "Public Auditing for Shared Data with Efficient User Revoation in the

Cloud," in *the Proceedings of IEEEINFOCOM 2013*, 2013, pp. 2904–2912.

[4]B.Wang, S. S. Chow, M. Li, and H. Li, "Storing Shared Data on the Cloud via Security-Mediator," in *the Proceedings of ICDCS 2013*,2013, pp. 124–133.

[5]Guoyuan Lin, Yuyu Bie, Min Lei. ACO-BT M: A Behavior Trust Model in Cloud Computing Environment [J].International Journal of Computational Intelligence Systems, 2013 (ahead-ofprint):1-11.

[6]Guoyuan Lin, Yuyu Bie, Min Lei. Trust Based Access Control Policy in Multi-Domain of Cloud Computing. Journal of Computers. 2013, 8(5): 1357-1365.

[7]B. Wang, B. Li, and H. Li, "Oruta: Privacy-Preserving PublicAuditing for Shared Data in the Cloud," in *Proceedings of IEEE Cloud 2012*, 2012, pp. 295–302.

[8]K. Ren, C. Wang, and Q. Wang, "Security Challenges for the Public Cloud," *IEEE Internet Computing*, vol. 16, no. 1, pp. 69–73 2012.

[9]D. Song, E. Shi, I. Fischer, and U. Shankar, "Cloud Data Protection for the Masses," *IEEE Computer*, vol. 45, no. 1, pp. 39–45, 2012.

[10]Guoyuan Lin, Shan He, Hao Huang. Access Control Security Model Based on Behavior in Cloud Computing Environment [J]. Journal of China Institute of Communications, 2012, 33(3): 59-66.

[11]Zhanjiang Tan, Zhuo Tang, Renfa Li, Ahmed Sallam, Liu Yang. Research on Trust-Based Access Control Model in Cloud Computing [C]//Information Technology and Artificial Intelligence China Communications· April 2014Conference (ITAIC), 2011 6th IEEE Joint International.
IEEE, 2011, 2: 339-344.

[12]R. K. L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, and B. S. Lee, "Trust Cloud: A framework for accountability and trust in cloud computing," in IEEE ICFP 2011: the 2nd IEEE Cloud Forum for Practitioners,2011.

[13]Jong P. Yoon, Z. Chen. Using Privilege Chain for Access Control and Trustiness of Resources in Cloud Computing [M]//Networked Digital Technologies. Springer Berlin Heidelberg, 2010: 358-
368.

[14]K.-K. Muniswamy-Reddy and M. Seltzer, "Provenance as first class cloud data," SIGOPS Oper. Syst. Rev., vol. 43, pp. 11–16, January 2010.