IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

EFFICIENT MULTI RESOURCE SHARING MECHANISM IN CLOUD BASED APPLICATIONS

[1]Srinivasa Rao Chevala, [2]Dr K Raja Kumar

[1]P.G. Student, [2]Assistant Professor,
[1]Department of Computer Science and System Engineering,
[1]Andhra University College of Engineering(A), Visakhapatnam, Andhra Pradesh, India

Abstract: Cloud computing impacts on different sectors, like Software Industry, E-learning, E-commerce, and health care. To purchase and maintain own resources for small industries is an economical burden to them. Cloud is providing the more services with minimal charges. By using cloud user can share a file to the other cloud user. Sigle ownership is not good for cloud-based applications. In this project, we are going to introduce the new mechanism Shared Ownership Reducing Space Complexity in The Cloud. In our Mechanism, the Owner will get the space by using access key control mechanism, we extended the existing model in SOM Authenticate protocol mechanism to share the files and, in this mechanism, the shared owner wants to delete the file he needs to get the permission from cloud service. By using SLA (Service Level Agreement). We will implement the Space sharing mechanism. In addition, we will reduce the burden on to the CSP (Cloud Service Provider), by using a cloud mechanism; we are going to increase the scalability of the CSP.

Index Terms **- SOM, SLA, CSP, Cloud Security, Shared Ownership, AOM, and SFD.**

I.INTRODUCTION

By using cloud user can share a file efficiently to other user in the cloud without any extra efforts. Due to this we are holding the individual file ownership. In the individual file ownership, file owner can grant the access permissions to requested user unilaterally. However, individual ownership is not suitable for varied cloud-based applications and collaborations. Let's take an example, in the research project, which is collaborate working by Educational institutions and Other Organizations. Both are working on cloud repository. But there are two problems here. First, a single proprietor can mislead the other by taking wrong access control decisions. Second, Community can elect one person as leader in among them. But newly an elected owner is not accountable for the access policies of other users in the community.

We address our research as follows: We define the shared ownership with Object Access Model (AOM) and by using this we are solving the access control problem in the existing cloud. We propose a primary solution, called shared community which distributive enforces AOM. Commune ensures that (1). a user can read a file from a shared repository if he has read access given by the owners, and (2). An Authorized user can edit a file if he has written permission granted by the owners of the cloud resources. We propose a second solution, dubbed Comrade, can get the access control decisions by succeed in consensus. If Cloud has the ability to translate access control decisions, then we can improve the performance of AOM.

II. RELATED WORK

In the single ownership system, all the access control is with single user only. If he wants to give permission then he can. So, it leads to less accessibility to the resources. Suppose the single owner will misuse his rights, then the entire system will become insecure. Above mention points were drawbacks in single ownership system.
By using Multi-Authority Attribute Based Encryption (MA-ABE) tool we are allowing the multiple users has access control on shared resource. But MA-ABE requires bilinear map, it is expensive and rely on novel cryptographic assumptions.
In this paper we are using Collision Resistant Secret Sharing (CRSS) instead of MA-ABE. CRSS allows only threshold policies. Control access to a large file need to different way such as combine the CRSS and Secure File Dispersal (SFD).
Advantages of Shared Mechanism: -

1. Space sharing mechanism is developed.
2. Increase the security and privacy in proposed mechanism.
3. Design Fractional shared ownership resources

III. METHODOLOGY

There are three modules in my project implementation. They are Data Owner, Data User and Admin. Data Owner can grant access to the Data User, who in collaboration. Here accesses are permission to the Data User to read or write or delete the file. At the time of deletion of file by the Data User, he will get an OTP to his mail for the confirmation deletion of the file. Data User can also request space from Data Owner by using Space Share Mechanism. Admin can see the who are the Data Owners, Data Users and Cloud files. Below We mentioned the protocols which are using to implement above functionalities.

Shared Ownership File Access Control Model (SOM)Authentication Protocol: -

In this module, we define the file sharing mechanism in cloud service. We implemented SOM (Shared Object Model) new way to overcome the existing issue. In this SOM model, we include the authentication protocol with the help of protocol the delete operation was redefining. The authentication protocol will give the information to cloud service whenever the file was share file at the time the cloud will generate a key for deletion purpose, and these should be based on grant permission among shared owners.

Collusion Resistant Secret Sharing (CRSS) Protocol: -
CRSS allows one user can send secret to a designated shareholder, in order that any subset of shareholders can reconstruct the key. Other users can get the permissions from the shareholders to reconstruct the key. If a user collects sufficient delegations, he will reconstruct the key.

Space Share Mechanism: -
In this section, we implemented space-sharing mechanism in cloud computing by using SLA (Service Level Agreement) the cloud tenant can share the space. By using the SLA, CSP will allow the tenant can share the space the authorized tenant based upon the request the space allocate and we implemented Load Balancing algorithm to decrease the burden on the CSP. In the load, balancing mechanism will check the access key, which is allocated by CRSS, and then it will balance the space.

IV. RESULTS AND DISCUSSION

Fig:1 Main Page

Fig 1 is the main page of our project. It has three components. They are Data Owner, Data User and Admin.

Fig:2 Data Owner

Fig:2 is the Data Owner Component. Inside the component there are owner request, user file request, file upload and download subcomponents present.

Below mentioned fig:3 is the admin module. Admin has all the rights to do.

Fig:3 Admin Module

V. CONCLUSION

In this project, we concentrate on the shared repositories in cloud services. In the existing mechanism also using shared repositories, but in the existing mechanism we find some of drawbacks in existing mechanism they concentrated single notation shared mechanism in SOM model. These mechanisms will drawback to the shared owner, because the actual owner can delete the file without notice of shared owner. Also, in the existing share cloud mechanism using, block chain technology, it will become burden to the cloud owners. Majorly the existing schemas are not concentrating on the shared space mechanism.

To overcome these problems, we proposed Shared Ownership Reducing Space Complexity in The Cloud. We implemented new way of SOM Authentication protocol mechanism we can grant access to the shared owner and we overcome the Delete operation issue and we increase the security level. In the existing mechanism, they do not concentrate on space to sharing mechanism by using SLA mechanism we implemented Load Balancing and Space Sharing mechanism to reduce burden on CSP and it will profitable to cloud owners.

REFERENCES

[1] M. Y. Becker, C. Fournet, and A. D. Gordon, "SecPAL: Design and Semantics of a Decentralized Authorization Language," in Journal of Computer Security (JCS), 2010, pp. 597–643.

[2] M. Blaze, J. Ioannidis, and A. D. Keromytis, "TrustManagement for IPsec," in ACM Transactions on Information and System Security (TISSEC), 2002.

[3] N. Li, B. N. Grosof, and J. Feigenbaum, "Delegation logic: A Logic-based Approach to Distributed Authorization," in TISSEC, 2003.

[4] C. Soriente, G. O. Karame, H. Ritzdorf, S. Marinovic, and S. Capkun, "Commune: Shared ownership in an agnostic cloud," ser. SACMAT '15, 2015.

[5] "Amazon Simple Storage Service(S3)," http://aws.amazon.com/s3/.

[6] S. Ceri, G. Gottlob, and L. Tanca, "What you always wanted to know about Datalog (and never dared to ask)," in Knowledge and Data Engineering, IEEE Transactions on, 1989.

[7] Y. Gurevich and I. Neeman, "DKAL: Distributed-Knowledge Authorization Language ," in CSF '08.

[8] J. DeTreville, "Binder, a Logic-based Security Language," in Proceedings of IEEE Symposium on Security and Privacy, 2002, pp. 105 – 113.

[9] "The Respect Network," https://www.respectnetwork.com/.

[10] "WDMy Cloud," http://www.wdc.com/en/products/products.aspx?id=1140.

[11] M. O. Rabin, "Efficient Dispersal of Information for Security, Load Balancing, and Fault Tolerance," in Journal of the Association for Computing Machinery.

[12] J. K. Resch and J. S. Plank, "AONT-RS: Blending Security and Performance in Dispersed Storage Systems," in FAST, 2011.

[13] R. L. Rivest, "All-or-Nothing Encryption and the Package Transform," in International Workshop on Fast Software Encryption (FSE), 1997.

[14] V. Boyko, "On the Security Properties of OAEP as an All-or-nothing Transform," in Procedings of CRYPTO, 1999, pp. 503–518.

[15] J. Daemen, and V. Rijmen, "AES Proposal: Rijndael," http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf.

[16] J. H. van Lint, Introduction to Coding Theory. Secaucus, NJ, USA:Springer-Verlag New York, Inc., 1982.

[17] M. vanDijk,A. Juels,A.Oprea,R. L.Rivest, E. Stefanov, andN. Triandopoulos, "Hourglass Schemes: how to prove that cloud files are encrypted," in CCS, 2012.

[18] A. B. Lewko and B.Waters, "Decentralizing Attribute-Based Encryption," in EUROCRYPT, 2011.

[19] P. Rogaway and M. Bellare, "Robust computational secret sharing and a unified account of classical secret-sharing goals," in CCS, 2007.

[20] [20] C. Charnes, J. Pieprzyk, and R. Safavi-Naini, "Conditionally secure secret sharing schemes with disenrollment capability," in CCS, 1994.

[21] Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System."

[22] F. Armknecht, J.-M. Bohli, G. O. Karame, Z. Liu, and C. A. Reuter,"Outsourced proofs of retrievability," ser. CCS '14.

[23] F. Armknecht, J.-M. Bohli, G. O. Karame, and F. Youssef, "Transparent data deduplication in the cloud," ser. CCS '15, 2015.

[24] Ethereum, "A Next-Generation Smart Contract and Decentralized Application Platform," White Paper, Tech. Rep., 2016. [Online]. Available: https://github.com/ethereum/wiki/wiki/White-Paper

[25] F. Zhang, E. Cecchetti, K. Croman, A. Juels, and E. Shi, "Town Crier: An Authenticated Data Feed for Smart Contracts," ser. CCS '16, 2016.

[26] D. F. Ferraiolo and D. R. Kuhn, "Role-based access controls," in Proc. of 15th NIST-NSA National Computer Security Conference, 1992.

[27] E. Heilman, A. Kendler, A. Zohar, and S. Goldberg, "Eclipse attacks on bitcoin's peer-to-peer network," ser. SEC'15. USENIX Association, 2015.