ISSN : 2231-0711

P.Sailaja et al | International Journal of Computer Science Engineering and Technology( IJCSET) | April 2015 | Vol 5, Issue 4,73-75

# Key Based Secured Data and Data Storage in Cloud

P.Sailaja M.E, S.Sarath kumar B.Tech,K.Jagadeesh B.Tech,A.T.Vignesh B.Tech

*Department of Information Technology*
*Velammal Institute of Technology,Chennai*

***Abstract-*** **Cloud computing is used to store, manage, and process data using a network which is hosted rather than a local server or a personal computer.We proposed how data are stored and shared in the cloud ,we use a key aggregate cryptosystem which uses both public and private key to encrypt and decrypt the data we are uploading in the cloud. Then using cpabe algorithm the attribute policy is created so that unauthorized person cannot modify the data and share data to the others in offline.The user who is downloading the uploaded data has to send the aggregate key request to the user who has uploaded the data file,then the user has to send the aggregate key to decrypt the downloaded file,by checking the authenticated user.**

***Key words*— Data sharing, key-aggregate encryption, patient-controlled encryption , cpabe.**

## I. INTRODUCTION

CLOUD storage is gaining popularity recently. In enterprise settings, we see the rise in demand for data outsourcing, which assists in the strategic management of corporate data. It is also used as a core technology behind many online services for personal applications. Nowadays, it is easy to apply for free accounts for email, photo album, file sharing and/or remote access, with storage size more than 25 GB (or a few dollars for more than 1 TB). Together with the current wireless technology, users can access almost all of their files and emails by a mobile phone in any corner of the world. Considering data privacy, a traditional way to ensure it is to rely on the server to enforce the access control after authentication, which means any unexpected privilege escalation will expose all data. In a shared-tenancy cloud computing environment, things become even worse. Data from different clients can be hosted on separate virtual machines (VMs) but reside on a single physical machine. Data in a target VM could be stolen by instantiating another VM co resident with the target one. Regarding availability of files, there are a series of cryptographic schemes which go as far as allowing a third-party auditor to check the availability of files on behalf of the data owner without leaking anything about the data, or without compromising the data owners anonymity. Likewise, cloud users probably will not hold the strong belief that the cloud server is doing a good job in terms of confidentiality. A cryptographic solution, for example, with proven security relied on number-theoretic assumptions is more desirable, whenever the user is not perfectly happy with trusting the security of the VM or the honesty of the technical staff. These users are motivated to encrypt their data with their own keys before uploading them to the servers. Data sharing is an important functionality in cloud storage. The challenging problem is how to effectively share encrypted data. Of course users can download the encrypted data from the storage, decrypt them, then send them to others for sharing, but it loses the value of cloud storage. Users should be able to delegate the access rights of the sharing data to others so that they can access these data from the server directly. However, finding an efficient and secure way to share partial data in cloud storage is not trivial.

## II. BACKGROUND AND RELATED WORK

In this section, we describe and discuss notable work related to our study. This section describes the different ideas regarding the processes involved in the project by various authors It also includes the various implementation methods to resolve the issues involved in the exisisting system

Kaitai Liang1, Qiong Huang2?, Roman Schlegel1, Duncan S. Wong1[1]. To allow a delegator not only to delegate the keyword-controlled decryption rights of a broadcast encryption to a set of speci_ed recipi-ents, but also to control when the decryption rights will be delegated, in this paper, for the _rst time, we introduce a new notion called Timed-Release Conditional Proxy Broadcast Re-Encryption (TR-CPBRE).

Dan Boneh,Craig Gentry,Ben Lynn and Hovav Shacham[2].The concept of an aggregate signature, present security models for such signatures, and give several applications for aggregate signatures. We construct an e_cient aggregate signature from a recent short signature scheme based on bilinear maps due to Boneh, Lynn, and Shacham. Aggregate signatures are useful for reducing the size of certi_cate chains (by aggregating all signatures in the chain) and for reducing message size in secure routing protocols such as SBGP.

Cong Wang, Sherman S.M. Chow, Qian Wang, Kui Ren and Wenjing Lou[3]. The fact that users no longer have physical possession of the outsourced data makes the data integrity protection in Cloud Computing a formidable task, especially for users with constrained computing resources. Moreover, users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity. Thus, enabling public auditability for cloud storage is of critical importance so that users can resort to a

third party auditor (TPA) to check the integrity of outsourced data and be worry-free.

Boyang Wang, Sherman S. M. Chow, Ming Li, and Hui Li [4]. In this paper, we propose a simple, efficient, and publiclyverifiable approach to ensure cloud data integrity without sacrificing the anonymity of data owners nor requiring significant overhead. Specifically, we introduce a security-mediator (SEM),which is able to generate verification metadata (i.e., signatures) on outsourced data for data owners. Our approach decouples the anonymity protection mechanism from the PDP.

## III. METHODOLOGY

The methodology describes the various methods involved in the project. The methodology includes five methods **(i)**Authentication And Authorization,(ii) File Encryption by KAC,(iii)Cloud data sharing and (iv)File Decryption by KAC

*1) Authentication and Authorization:*

Authentication and Authorization process are the required of the Verifying the user Originality and appropriate Session Activities of the Registered User.

*2) File Encryption by KAC:*

After user login, the user can able to store the files into the cloud in an encryption manner. So that the encrypted files which are stored in the cloud cannot be decrypted normally by other users or hacker's.

*3) Cloud data sharing:*

The user initially upload's files data to the cloud, and shares it with other users. The shared files are encrypted by the owner. So whenever the other user's want to access or decrypt the file required keys permission to be accessed.

*4) File Decryption by KAC:*

The aggregate keys are sent in mail to other user by the original user. The file will be decrypted by the aggregate key's which was generated by the KAC. Finally, any user with an aggregate key can decrypt any cipher text provided that the cipher text's class is contained in the aggregate key via Decrypt.

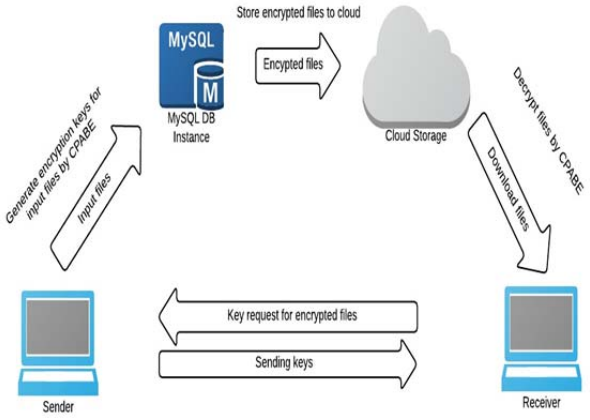



Fig) **Architecture diagram**

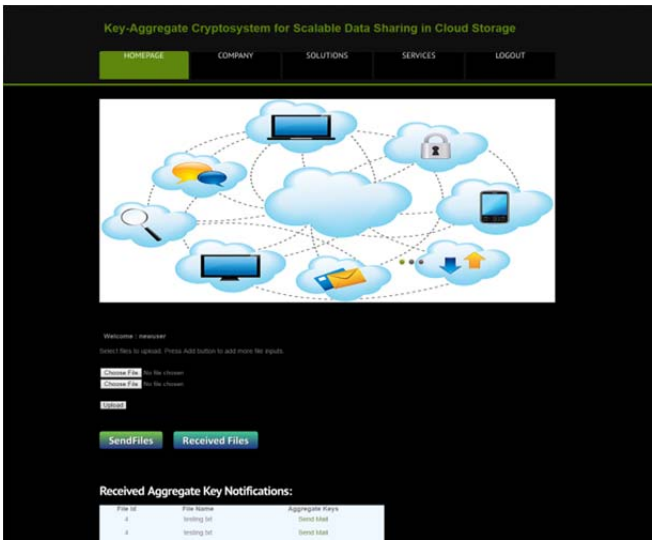
## IV. EXPERIMENTAL RESULTS

We propose key-aggregate cryptosystem (KAC), a special type of public-key encryption. In KAC, users encrypt files not only under a public-key and also with private key. We proposed how to "compress" secret keys in public-key cryptosystems which support delegation of secret keys for different cipher text classes in cloud storage. No matter which one among the power set of classes, the delegatee can always get an aggregate key of constant size. Our approach is more flexible than hierarchical key assignment which can only save spaces if all key-holders share a similar set of privileges.


a)home page


b)registration page


c)login page




d)file uploading page


e)sending aggregate request

## V. CONCLUSION

With more mathematical tools, cryptographic schemes are getting more versatile and often involve multiple keys for a single application. In this project, we propose key aggregate cryptosystem which uses both public and private key to encrypt and decrypt the data we are uploading in the cloud. Then using cpabe algorithm the attribute policy is created so that unauthorized person cannot modify the data and share data to the others in offline.

## REFERENCES

[1] S.S.M. Chow, Y.J. He, L.C.K. Hui, and S.-M. Yiu, "SPICE – Simple Privacy-Preserving Identity-Management for Cloud Environment," Proc. 10th Int'l Conf. Applied Cryptography and Network Security (ACNS), 2012.

[2] L. Hardesty, Secure Computers Aren't so Secure. MIT press,http://www.physorg.com/news176107396.html, 2009.

[3] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, Feb. 2013.

[4] B. Wang, S.S.M. Chow, M. Li, and H. Li, "Storing Shared Data on the Cloud via Security-Mediator," Proc. IEEE 33rd Int'l Conf. Distributed Computing Systems (ICDCS), 2013.

[5] S.S.M. Chow, C.-K. Chu, X. Huang, J. Zhou, and R.H. Deng, "Dynamic Secure Cloud Storage with Provenance," Cryptography and Security, 2012.

[6] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps," Proc. 22nd Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '03), 2003.

[7] M.J. Atallah, M. Blanton, N. Fazio, and K.B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies," ACM Trans. Information and System Security,2009.

[8] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter, "Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records," Proc. ACM Workshop Cloud Computing Security (CCSW '09), pp. 103-114, 2009.

[9] F. Guo, Y. Mu, Z. Chen, and L. Xu, "Multi-Identity Single-Key Decryption without Random Oracles," Proc. Information Security and Cryptology (Inscrypt '07), vol. 4990, pp. 384-398, 2007.

[10] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. 13th ACM Conf. Computer and Comm. Security (CCS '06),2006.

[11] S.G. Akl and P.D. Taylor, "Cryptographic Solution to a Problem of Access Control in a Hierarchy," ACM Trans. Computer Systems, 1983.

[12] G.C. Chick and S.E. Tavares, "Flexible Access Control with Master Keys," Proc. Advances in Cryptology (CRYPTO '89), vol. 435, pp. 316-322, 1989.

[13] W.-G. Tzeng, "A Time-Bound Cryptographic Key Assignment Scheme for Access Control in a Hierarchy," IEEE Trans. Knowledge and Data Eng. 2002.

[14] G. Ateniese, A.D. Santis, A.L. Ferrara, and B. Masucci, "Provably-Secure Time-Bound Hierarchical Key Assignment Schemes," J. Cryptology, 2012.

[15] R.S. Sandhu, "Cryptographic Implementation of a Tree Hierarchy for Access Control," Information Processing Letters, 1988.

[16] Y. Sun and K.J.R. Liu, "Scalable Hierarchical Access Control in Secure Group Communications," Proc. IEEE INFOCOM '04, 2004.

[17] Q. Zhang and Y. Wang, "A Centralized Key Management Scheme for Hierarchical Access Control," Proc. IEEE Global Telecomm. Conf. 2004.

[18] J. Benaloh, "Key Compression and Its Application to Digital Fingerprinting," technical report, Microsoft Research, 2009.

[19] B. Alomair and R. Poovendran, "Information Theoretically Secure Encryption with Almost Free Authentication," J. Universal Computer Science, 2009.