Enhanced Key Aggregation Technique For Climbable Knowledge Sharing In Cloud

[1] Priyanka Kale,[2] Mrunali Vaidya

[1] M.Tech Student ,Computer Science And Engineering, BIT, Ballarpur, Maharashtra, India
[2] Asst.professor, Computer Science And Engineering, BIT, Ballarpur, Maharashtra, India

***Abstract* - Cloud Emerging new computing technology cloud offers storage services. Data sharing with others in secure and efficient manner is important function in cloud storage. For this the new expandable public-key cryptosystems which derives fixed-size cipher texts. Using expandable public-key cryptosystems transform the multiple keys of the classes as single secret key. The generated single secret key sent to others or be stored in a smart card with very limited secure storage. If a user needs to classify his cipher texts into more than n classes, user can register for additional key pairs. Each class is indexed by a two-level index and the number of classes is increased by n for each added key. For the approach, at most two aggregate keys are needed. This key extension approach can also be seen as a key update process. In case a secret value is compromised, user can replace compromised public key with new public key. Eventually the approach is flexible and effective.**

***Index Terms* - cloud domain, key-aggregate cryptosystem, Identity key, secrete key, Attribute encrypt, TVES.**

I. INTRODUCTION

Nowadays, many large scale and small scale organizations outsource their large-scale data storage to the cloud for saving the cost in maintaining their storage. With cloud storage service, the members of an organization can share data with other members easily by uploading their data to the cloud. Examples of organizations which may benefit from this cloud storage and sharing service are numerous, such as international enterprises with many employees around the world, collaborative web application providers with a large user base, or institutions dealing with big data, healthcare researchers, patients, etc. While the economic benefits brought by outsourcing data can be attractive, security is one of the most significant factors that hinder its wide development. Since data operations in the cloud are not transparent to users, and security breaches or improper practices are common and inevitable, users still have a huge concern about the security of their data on the cloud, especially on data integrity. Cryptography is the method of storing and transmitting data in a form that only those intended for it can read and process the required data. It is technique of protecting information by encrypting the data it into an unreadable format using some encryption algorithm. Cryptography is an effective way of protecting sensitive information that is to be stored on media or transmitted through network communication paths. The main goal of cryptography is that to hide information from unauthorized individuals like intruders or hackers. Hackers now a day can hack most of the cryptography algorithms and the information can be revealed if the attacker has enough time and resources to hack the data. So a more realistic goal of cryptography is to decrypting the data to be difficult.
Considering data privacy, rely on the server to enforce the access control after authentication, if there is any unexpected privilege escalation will expose all data which is sensitive. In a shared- cloud computing environment, things become even worse because Data from different clients can be hosted on separate virtual machines (VMs) but reside on a single physical machine. Regarding availability of files, there is lot of cryptographic schemes which allows 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 owner's anonymity. Likewise, cloud users probably will not hold the strong belief that the cloud server is doing a good job in terms of confidentiality.

II. AIM AND OBJECTIVE

The main objective of the any citation analysis is to provide useful information to the scholars in searching of literature and to help the librarian in selecting relevant sources. The specific objectives of the present study are to find out the information sources cited by researchers in the field of enhanced key aggregation technique for climbable knowledge sharing in cloud storage,
(i) To find-out the authorship pattern and degree of collaboration in aggregate key for flexible choices,
(ii) To identify the core cryptosystem,
(iii) To examine the subject wise break up of citations,
(iv) To know the country wise distribution of CLOUD storage
(v) To identify the language wise distribution of journal citations.

III. LITERATURE SURVEY

Symmetric-Key Encryption With Compact Key:
Benaloh et al. [2] presented an encryption scheme which is originally proposed for concisely transmitting large number of keys in broadcast scenario [3]. The construction is simple and we briefly review its key derivation process here for a concrete description

of what are the desirable properties we want to achieve. The derivation of the key for a set of classes (which is a subset of all possible cipher text classes) is as follows. A composite modulus is chosen where p and q are two large random primes. A master secret key is chosen at random. Each class is associated with a distinct prime. All these prime numbers can be put in the public system parameter .A constant-size key for set can be generated. For those who have been delegated the access rights for S' can be generated. However, it is designed for the symmetric-key setting instead. The content provider needs to get the corresponding secret keys to encrypt data which is not suitable for many applications. Because method is used to generate a secret value rather than a pair of public/secret keys, it is unclear how to apply this idea for public-key encryption scheme. Finally, we note that there are schemes which try to reduce the key size for achieving authentication in symmetric-key encryption, e.g., [4]. However, sharing of decryption power is not a concern in these schemes.

IBE With Compact Key :

Identity-based encryption (IBE) (e.g., [5], [6], [7]) is a public-key encryption in which the public-key of a user can be set as an identity-string of the user (e.g., an email address, mobile number). There is a private key generator (PKG) in IBE which holds a master-secret key and issues a secret key to each user with respect to the user identity. The content provider can take the public parameter and a user identity to encrypt a message. The recipient can decrypt this cipher text by his secret key. Guo et al. [8], [9] tried to build IBE with key aggregation. In their schemes, key aggregation is constrained in the sense that all keys to be aggregated must come from different ―identity divisions‖. While there are an exponential number of identities and thus secret keys, only a polynomial number of them can be aggregated.[1] This significantly increases the costs of storing and transmitting cipher texts, which is impractical in many situations such as shared cloud storage. As Another way to do this is to apply hash function to the string denoting the class, and keep hashing repeatedly until a prime is obtained as the output of the hash function.[1] we mentioned, our schemes feature constant cipher text size, and their security holds in the standard model. In fuzzy IBE [10], one single compact secret key can decrypt cipher texts encrypted under many identities which are close in a certain metric space, but not for an arbitrary set of identities and therefore it does not match with our idea of key aggregation.

Scalable and Efficient Data Possession:

Storage outsourcing is a rising trend which prompts a number of interesting security issues, many of which have been extensively investigated in the past. However, Provable Data Possession (PDP) is a topic that has only recently appeared in the research literature. The main issue is how to frequently, efficiently and securely verify that a storage server is faithfully storing its client's (potentially very large) outsourced data. The storage server is assumed to be untrusted in terms of both security and reliability. (In other words, it might maliciously or accidentally erase hosted data; it might also relegate it to slow or off-line storage.) The problem is exacerbated by the client being a small computing device with limited resources. Prior work has addressed this problem using either public key cryptography or requiring the client to outsource its data in encrypted form. In this paper, we construct a highly efficient and provably secure PDP technique based entirely on symmetric key cryptography, while not requiring any bulk encryption. Also, in contrast with its predecessors, our PDP technique allows outsourcing of dynamic data, i.e, it efficiently supports operations, such as block modification, deletion and append. Two recent results PDP and POR have highlighted the importance of the problem and suggested two very different approaches. The first is a public key- based technique allowing any verifier (not just the client) to query the server and obtain an interactive proof of data possession. This property is called public verifiability. The interaction can be repeated any number of times, each time resulting in a fresh proof. The POR scheme uses special blocks (called sentinels) hidden among other blocks in the data. During the verification phase, the client asks for randomly picked sentinels and checks whether they are intact. If the server modifies or deletes parts of the data, then sentinels would also be affected with a certain probability.

IV. MODULES

- Cloud Admin Panel
- User Management
- File Sharing and Storage Management
- Encryption
- Identity key/Aggrigate Key Generation
- Decryption
- File Transfer Log

Cloud Admin Panel

In cloud admin panel firstly the user register for the purpose of files storage, usage tracking, payment tracking, account activation/deactivation and rent calculation.

User Management

In user management, the user firstly Edit Profile, Group Formation ,Communication between existing users, Upload Important Documents in encrypted format, File Storage Log, Download Files with identity key and secrete key, Remove Files with identity key and secrete key ,Share files to other recipients, Rent Monitoring as per the usage , Payment Summary that all thing are included it.

File Sharing and Storage Management

Encrypted File Upload

- While uploading users file will be encrypted using AES Algorithm and get stored on server

Secrete key Generation

- At the time of user registration, every user have to decide his/her unique secrete key.
- This key will be considered as secrete key of his all documents
- At the time of file upload actual secrete key for every document will be automatically generated and stored in data base in encrypted format.

Secure File Download

- Each user can download/decrypt the encrypted file if and only if he/she knows secrete key for the particular file
- Each user have to prove his identity before doing any file related operation
- Identity key generation

At the time of user registration, system will automatically choose one of the available algorithm definitions and generate unique identity key for every user and send it on users email

Access control automation

- It Ensure and automate the process by which only authorized, authenticated senders and recipients can access files.

File Transfer Log

- File storage size log
- Recipient tracking
- Existing Files log

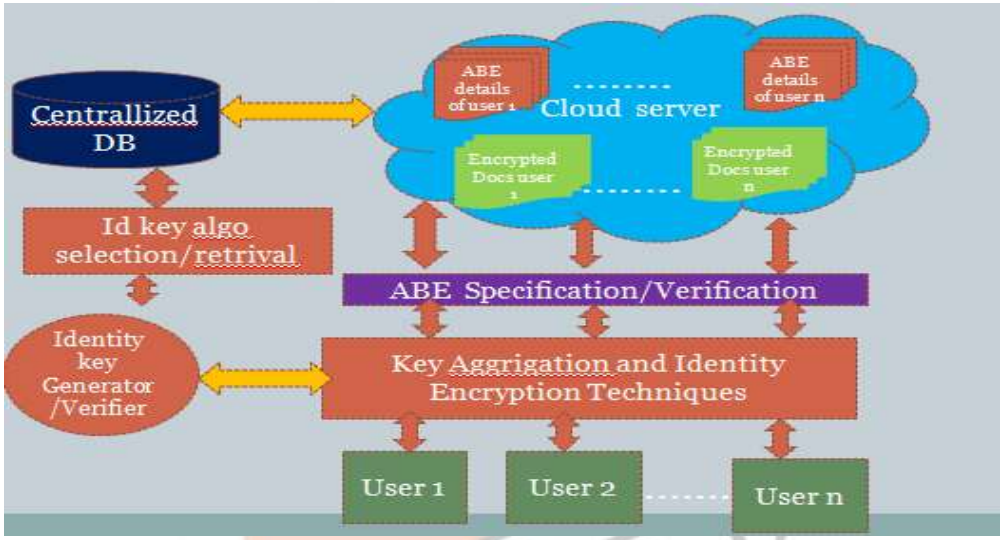


Fig -1: System Architecture

V. ALGORITHM USED

Id key Algorithm definations

Id	Algodef
1001	prof|123+S|$+mobile|124+S|#+id
1002	id+S|%+mobile|3254+S|*+dob|124
1003	mobile|5671+prof|345+S|@+id
1004	S|%+mobile|1849+S|*+id+dob|478
1005	dob|8742+id+S|%$+mobile|97341
1006	gender|12+S|$+mobile|1234+S|#
1007	id+S|%+mobile|9156+S|*+name|1
1008	mobile|5371+gender|341+S|@+id
1009	S|%+mobile|19+S|*+id+dob|4
1010	dob|874+S|%$+id+name|1+city|1

Table -1: Database table idkeyalgodef

Steps:

1. Fetched one algo id from 1 to 10 Randomly eg id=1005
2. fetch id key algorithm def for particular id say 1005
3. generate id key using algo

split algodef by + store separate def in string array

0	dob|8742
1	Id
2	S|%$
3	mobile|97341

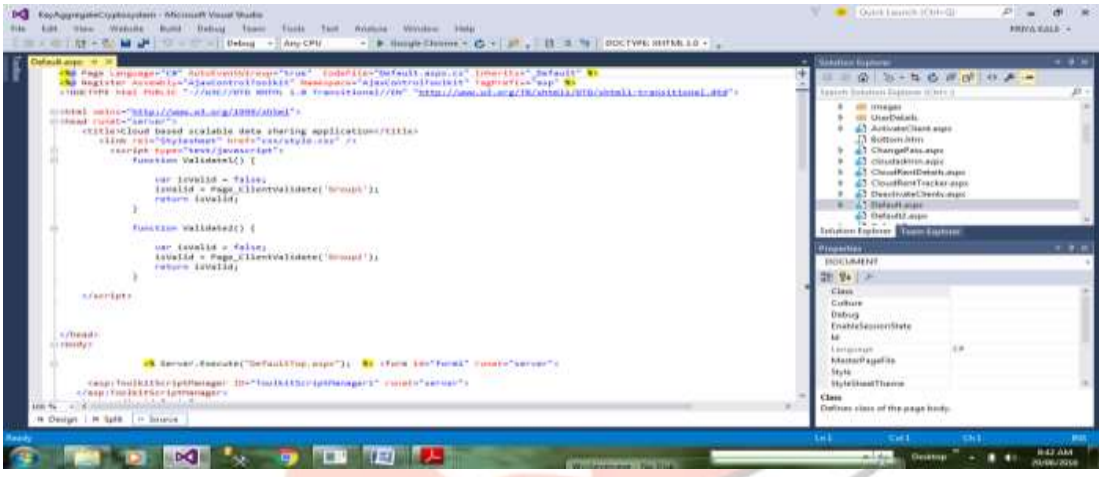
Table -2: split algodef by

0	Dob
1	8742

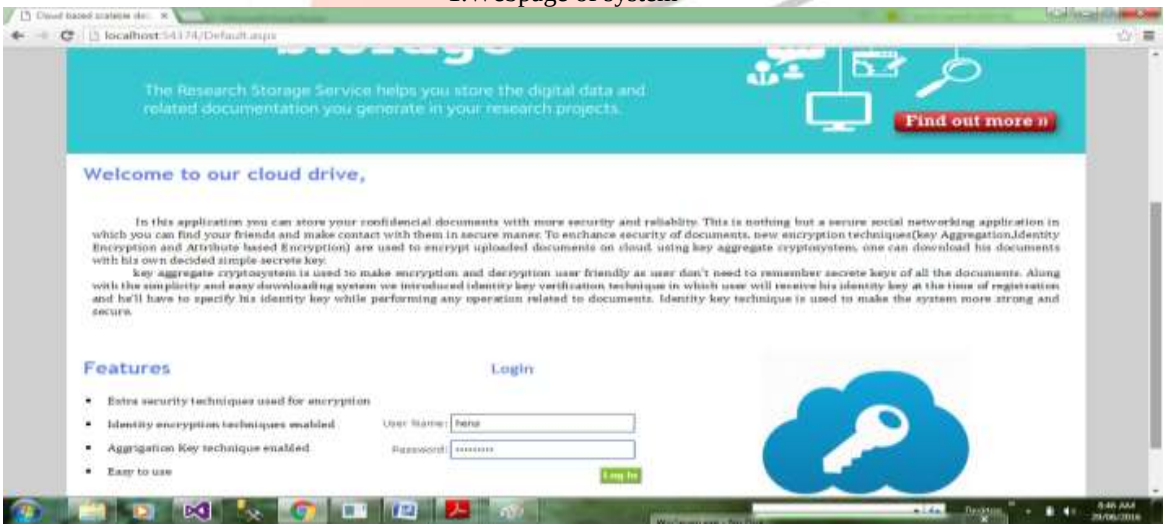
Table -3: Store separate def in string array

id=1002
dob=13/12/1987
mobile=9878765654
idkey=892/1002%$55789
In idkey algorithm fetched feched one algo id from 1 to 10 randomly id then id key algorithm def for particular id peek the one id and generate id key using algorithm.

VI. SCREEN SHOT



1.Webpage of system



2 .User Login

3. User Identity Authentication

4. Upload Document

5. Cloud rent details report

VII. CONCLUSIONS

This application is useful in cloud computing to scalable system using different cryptographic algorithm. How to defend users‟ data privacy is a central problem of cloud storage. With more mathematical concept, cryptographic schemes are getting more multipurpose and often involve multiple keys for a single system. In this System we consider how to compactl secret keys in public-key cryptosystems which support delegation of secret keys for different cipher text classes in cloud storage. Our approach is more flexible than hierarchical key assignment which can only save storage spaces if all key-holders share a similar set of privileges. A limitation in our work is the predefined bound of the number of maximum cipher text classes. In cloud storage, the number of cipher texts usually grows rapidly. So we have to reserve enough cipher text classes for the future extension.

VIII. ACKNOWLEDGMENT

I express my gratitude to all anonymous researchers for providing us such helpful opinion, findings, conclusions and recommendations.

REFERENCES

[1] Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage Cheng-Kang Chu, Shennan S. M. Chow, Wen-Guey Tzeng, Jianying Zhou, and Robert H. Deng, Senior Member, IEEE
[2] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, vol. 62, no. 2, pp. 362-375, Feb. 2013.
[3] B. Wang, S.S.M. Chow, M. Li, and H. Li, "Storing Shared D. on The Cloud via Security-Mediator," P. IEEE 33rd Int'l Conf.Distributed Computing Systems (ICDCS), 2013.
[4] S.S.M. Chow, C.-K. Chu, X. Huang, J. Zhou, and R.H. Deng, "Dynamic Secure Cloud Storage with Provenance," Cryptography and Security, pp. 442-464, Springer, 2012.
[5] D. Boneh, C. Gently, B. Lynn, and H. Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps," Proc. 22. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '03), pp. 416-432, 2003.
[6] M.J. Atallah, M. Blanton, N. Fazio, and K.B. Frikken, "Dynamic And Efficient Key Management for Access Hierarchies," ACM Trans. Infonnarion and System Security, vol. 12, no. 3, pp. 18:1-18:43, 2009.
[7] 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.
[8] F. Guo, Y. Mu, Z. Chen, and L. Xu, "Multi-Identity Single-Key Decryption without Random Oracles," P. Information Security and Cryptology (Inscrypt '07), vol. 4990, pp. 384-398, 2007.
[9] V. Goyal, 0. Pandey, A. Salmi, and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. 13th ACM Conf. Computer and Comm. Security (CCS '06), pp. 89-98, 2006.
[10] S. Singh,”Different Cloud Computing Standards a Huge Challenge”, The Economic times, 4 June 2009.