ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

Achieving Secure, Scalable, Data Group Sharing Using Key Aggregate Searchable Encryption in Cloud

Rajeshree K. Duratkar, Prof. Sonali A. Patil

M.E Student, Department of Computer Engineering, JSPM's B.S.I.O.T.R, Wagholi, Pune University, India

Asst. Professor, Department of Computer Engineering, JSPM's B.S.I.O.T.R, Wagholi, Pune University, India

ABSTRACT: In Cloud Storage there is an important functionality called Data Sharing. Data sharing is a very essential in cloud storage. This article usually has a tendency to reveal the way too firmly and flexibly understanding with others in cloud garage. The energy of preferentially sharing encrypted data with no longer similar users by public cloud garage may extremely ease safety misery, with the aid of threat understanding disclose in the cloud. The popular flexibility of allocating any numbers of files with any number of users by accomplishing weight age totally exclusive coding keys to be used for diverse files. On the alternative hand, this includes the requirement of firmly distributing to users by way of an outsized style of keys for every coding and search, and data users were given to get to save the number of received keys. In this paper, we have a propensity to target, by means of suggesting the unconventional construct of key combination searchable coding (KASE) and instantiating the idea through KASE, throughout which an facts proprietor wishes to proportion out one Aggregate key to a user for downloading number of documents, and therefore the user has to present one trapdoor to the cloud for wondering the shared files. Data deduplication is also proposed to check the duplicate file, if uploaded further and remove the duplicate file.

KEYWORDS: Public key encryption, Aggregate key, Cloud computing, Group data sharing, Deduplication.

I.INTRODUCTION

Nowadays the storage in the cloud has materialized as a capable answer for suitable and on-demand accesses to huge amounts of information shared over the Internet. Business users are being paying attention by cloud storage due to its several benefits, including lower cost, better agility, and improved resource utilization. Everyday users are also sharing private data, such as photos and videos, with their friends through social network applications based on cloud. On the other hand, while benefiting from the expediency of sharing data through cloud storage, users are also gradually worried about accidental data reveal by the cloud. Such data revealing, will be performed by malicious opponent or a mischievous cloud operator, can habitually direct to severe violation of private data or confidential data regarding bussiness. To speak about users anxiety over possible data reveal in cloud storage, a general approach is for the data owner to encrypt all the data before uploading them in to the cloud, such that presently the encrypted data may be get back and decrypted by individuals who contains the decryption keys. Such cloud storage is often called the cryptographic cloud storage [6].Though; the encryption of data builds it demanding for users to search and then preferable retrieve only the data including the given keywords. A common solution is to employ a searchable encryption (SE) scheme in which the data owner is required to encrypt potential keywords and upload them to the cloud together with encrypted data, such that, for retrieving data matching a keyword, the user will send the matching keyword to the cloud to react for the search over the encrypted data. Even though merging a searchable encryption Scheme with cryptographic cloud storage can accomplish the essential security needs of a cloud storage, executing such a system for large scale application relating huge number of users and large number of files may still be delayed by realistic issues relating the well-organized management of encryption keys, which, to the finest of our knowledge. Primarily, the want for selectively sharing encrypted data with different users usually demands different encryption keys to be used for different files. On the other hand, this involves the number of keys that need to be spread to users, both for them to search over the encrypted files and to decrypt the files, will be relative to the number of such files.

DOI: 10.15680/IJIRCCE.2017. 0505319

Such a large number of keys must not only be spread to users via secure channels, but also be securely stored and handled by the users in their devices. The implicit requirement for secure communication, storage, and computational difficulty may cause system ineffectiveness.

In this paper, we propose the novel concept of key aggregate searchable encryption (KASE), and instantiating the concept through a concrete KASE method. The proposed KASE scheme relates to any cloud storage that supports the searchable group data sharing feature, which means any user may prefer to distribute a group of files which are selective with a group of selected users, while permitting the final to carry out keyword search above the earlier. To maintain searchable group data sharing the main needs for efficient key management are double. Primarily, a data owner wants to allocate a single aggregate key (instead of a group of keys) to a user for sharing any number of files. Subsequent, the user needs to submit a single aggregate trapdoor to the cloud for performing keyword search over any quantity of shared files. KASE scheme can assure both requests. Data owner if uploaded same file then duplicate copy will not be encrypted by our proposed work deduplication. Deduplication concept is used to save the space in cloud.

II.LITERATURE REVIEW

S. Yu, C. Wang, K. Ren, and W. Lou[2]proposed Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing. In which cloud computing is developed computing paradigm in which resources of the computing infrastructure are provided as services over the Internet. As to assure as it is, this paradigm also brings forth many new challenges for data security and access control when users outsource annoyed data for sharing on cloud servers, which are not within the same trusted influence, as data owners. To keep sensitive user data confidential against untrusted servers, existing solutions usually apply cryptographic methods by to cause to appear data decryption keys only to authorized users. The problem of simultaneously accomplish fine grained access, scalability, and data confidentiality of access control actually still remains not resolved.

R. Lu, X. Lin, X. Liang, and X. Shen[3] proposed Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing. In which Success of data forensics in cloud computing is based on secure place that records ownership and process history of data objects. But it is the still challenging issue in this paper. In this paper, they proposed a new secure provenance scheme based on the bilinear pairing techniques .As the essential bread and butter of data forensics and post investigation in cloud computing, the proposed scheme is characterized by providing the information confidentiality on sensitive documents stored in cloud. Secure authentication on user access, and place tracking on disputed documents is provided in this paper. With the provable security techniques, this paper formally demonstrates the proposed scheme is secure in the standard model.

X. Liu, Y. Zhang, B. Wang, and J. Yan[4] in this paper proposed Mona: Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud. Where character of low maintenance, cloud computing provides an economical and efficient solution for sharing group resource among cloud users. Due to the frequent change of membership sharing data in multi-owner manner while preserving data and identify privacy from untrusted cloud is still a challenging issue.

C.Chu, S.Chow, W.Tzeng,etal[5] proposed Key-Aggregate Crypto system for Scalable Data Sharing in Cloud Storage. In which data sharing is large functionality in cloud storage In this article, we show how to securely, efficiently, and adaptable share data with others in cloud storage. The novelty is that one can aggregate any set of secret keys and make them as compact as a single key, but to enclose the power of all the keys being aggregated. In other words, the secret key something that holds or secures can release a constant-size aggregate key for flexible choices of cipher text set in cloud storage, but the other encrypted files not inside the set unchanged confidential. This compact aggregate key can be suitable sent to others or be stored in a smart card with very limited secure storage.

III.ARCHITECTURAL DESIGN

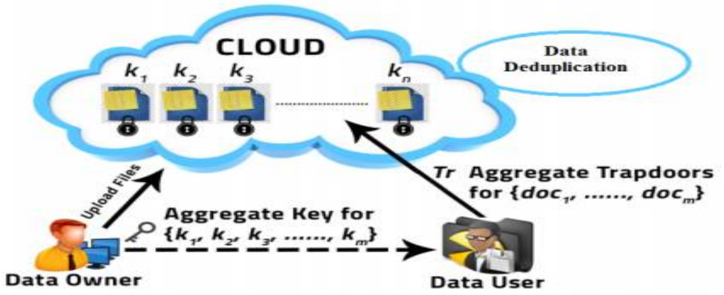

CLOUD
k_1
k_2
k_3
k_n
Data Deduplication
Upload Files
Aggregate Key for {k_1, k_2, k_3, ..., k_m}
Tr Aggregate Trapdoors for {doc_1, ..., doc_m}
Data Owner
Data User

Fig: KASE Architectural diagram

IV.PROPOSED SYSTEM

Through the concrete KASE scheme, the proposed work is to support searchable cluster knowledge sharing the most needs for economical key management a twofold. First, an information owner solely has to distribute one combination key to a user for sharing any number of files within a group. Second, the user solely has to submit one combination trapdoor to the cloud for keyword search over any range of shared files. Thirdly, to ignore the duplicate file to be uploaded on cloud. The KASE architecture Fig.1 it consists of a data owner, data user and cloud. Data owner first upload files to cloud. Same file will not be upload because of deduplication. After uploading file on cloud it get encrypted using aggregate key i.e combination of owner public key and randomly generated master secret key for sharing number of documents of that owner. This single aggregate key created will then sent to the data user through a secure communication channel. Then, the data user can perform searching for file through keyword. Then download the file by providing the aggregate key. User also perform group data sharing in which user has to select a group then only the aggregate key is shared with all the group of users or selected group user via secure channel. Hence the group user by only entering the aggregate key can access the documents of that owner.

V.ALGORITHMS

A. Proposed Algorithm

The KASE construction is composed of several algorithms.
1. Setup(1, n): This algorithm is run by the cloud service provider.
2. Keygen(pk,msk): This algorithm is run by the data owner to generate a key and cloud generate a random master secret key pair.
3. Encrypt(pk, i): This algorithm is run by the data owner to encrypt the i-th document and generate its keyword ciphertexts. Encryption is done by AES algorithms.

i. AES (Advance Encryption Standard):

The following AES steps of encryption for a 128-bit block:

Step 1. Derive the set of round keys from the cipher key.

Step 2. Initialize the state array with the block data (plaintext).

Step 3. Add the initial round key to the starting state array.

Step 4. Perform nine rounds of state manipulation.

Step 5. Perform the tenth and final round of state manipulation.

Step 6. Copy the final state array out as the encrypted data (ciphertext).

Step 7. Each round of the encryption process requires a series of steps to alter the state array. These steps involve four types of operations called:

- SubBytes
- ShiftRows
- MixColumns
- XorRoundKey

The following AES steps for decryption of a 128-bit block:

Step 1. Perform initial decryption round:

- XorRoundKey
- InvShiftRows
- InvSubBytes

Step 2. Perform nine full decryption rounds:

- XorRoundKey
- InvMixColumns
- InvShiftRows
- InvSubBytes

Step 3. Perform final Xor Round Key and The same round keys are used in the same order.

4. Extract(msk,S): This algorithm is run by the data owner to generate an aggregate searchable encryption key for keyword search for a certain set of documents to other users.

i. Algorithm for Aggregate Key Generation Process.

```
Input: key1: Master-Key (msk),
       key2: Public Key(pk),
       key3: Data.
Output: Aggregate key (kagg).
Process:
Step 1. First Setup Data.
Step 2. All the keys like k1, k2, k3 are in string format then it will converted into bytes using
        Byte  Encoder.
Step 3. Then every string converted in string to number like, k1=123, k2=564, k3=356
Step 4. All set key combine then it can give separator for that different key like 12 05640.356
          here  no value consider as separator.
Step 5. Master Secrete key i.e,msk.
Step 6. Key convolution: we are use,
        F(x) = (k1x + k2 x2 + k3) Here the x is considered as any number.
Step 7. Display Aggregate key.
```

5. Trapdoor(kagg, x): This algorithm is run by the user who has the aggregate key to perform a search.

6. Adjust(params, i, S, Trd): This algorithm is run by cloud server to adjust the aggregate trapdoor to generate the right trapdoor for each different document
7. Test(Tri, i):This algorithm is run by the cloud server to perform keyword search over an encrypted document.

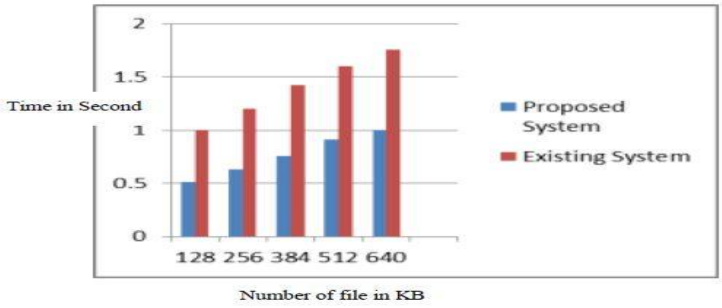
VI. PSEUDO CODE

Mathematical Module

Let =S‘ be the final setup of KASE Framework.
This will include user, resources, system.
S = {. . . . . }
Identify the inputs as I
I = {F}
F = {F1, F2, F3, F4,F5, F6, F7 =I‘. . . | files to be uploaded}
Identify the outputs as O
O = {T}
K= {Aggregate key . . . | K‘ given aggregate key for files}
Identify the functions as =F‘
S = {. . .
F = {F1 (), F2(), F3(), F4(), F5(), F6(), F7()}
F1 (I) = Upload file
F2 (I) = Request file to data owner
F3 (O) = Encryption
F4 (O) = key generation
F5 (O) = Trap door
F6 (O) = Matched keyword with targeted document
F7 (O)= Get authenticated document

VII.SIMULATION RESULTS

The proposed KASE scheme is implemented with Java. Performance of system show the comparison between KASE and existing system. We get more secured system than existing one. And user get aggregate key which will avoid confusion. Also proposed system is scalable than previous technique. For decrypting documents in KASE we required nearly half of the existing system time. As in the result, we observe that our proposed system time efficiency calculated are greater than the existing system. The efficiency should also be increased as instead of single owner, the scheme is applied for multiple owners together. In existing system owner has less flexibility in its functionality but in proposed one they have given more scope for their operations and so improved efficiency.



Graph.1 Performance Comparison graph of data decryption time required between KASE(proposed system) and existing system.

Experimental results show the table of comparison between KASE(proposed system) and existing system, the time complexity for decrypting the file in seconds. The decryption times required of KASE is nearly half of the existing system. When KASE is used in the cloud system, the decryption time for 128 KB file is 0.51second whereas existing system used in the cloud system, the decryption time for 128 KB file is 1second and so on . Outcome success work in KASE gives secured file. These file are secured using encryption. In this technique we are perform combination of keys which will avoid confusion for user.

VI.CONCLUSION

In this paper we have tried to resolve the problem of privacy of key for data sharing in cloud storage. By achieving secure single aggregate key for downloading number of files and also reduce storage complexity and improve efficiency of search over shared data. Secondly, we studied model for Key Aggregate Searchable Encryption scheme, a practical system for efficient management of encryption keys for sharing large number of documents with users any group of users In KASE scheme this goal is been achieved by different cryptographic techniques and searchable encryption method, which enables the data owner for the distribution of single aggregate key instead of group of keys to a user for sharing any number of files and at the user end submission of single aggregate trapdoor to the cloud for performing keyword search over any number of shared files. Thus reducing the large number of keys to a single aggregate key enhances the key management requirements. Thirdly reduce the burden of duplicate file uploading through deduplication to save the memory space.

REFERENCES

[1] Baojiang Cui, Zheli Liu and Lingyu Wang : Key-Aggregate Searchable Encryption for Group Data Sharing via Cloud Storage IEEE Transactions on Computers VOL: PP NO: 99, 2015.
[2] S. Yu, C. Wang, K. Ren, and W. Lou, Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing, Proc. IEEE INFOCOM, pp. 534-542, 2010.
[3] R. Lu, X. Lin, X. Liang, and X. Shen, Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing, Proc. ACM Symp. Information, Computer and Comm. Security, pp. 282-292, 2010..
[4] X. Liu, Y. Zhang, B. Wang, and J. Yan. Mona:secure multi- owner data sharing for dynamic groups in the cloud, IEEE Transactions on Parallel and Distributed Systems, 2013, 24(6): 1182- 1191..
[5] C.Chu,S.Chow,W.Tzeng,etal.Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage, IEEE Transactions on Parallel and Distributed Systems, 2014, 25(2): 468-477..
[6] X.Song,D.Wagner,A.Perrig.Practical techniques for searches on encrypted data, IEEE Symposium on Security and Privacy, IEEE Press, pp. 44C55, 2000.
[7] R. Curtmola, J. Garay, S. Kamara, R. Ostrovsky. Searchable symmetric encryption: improved definitions and efficient constructions, In: Proceedings of the 13th ACM conference on Computer and Communications Security, ACM Press, pp. 79- 88, 2006.
[8] P. Van,S. Sedghi, JM. Doumen. Computationally efficient searchable symmetric encryption, Secure Data Management, pp. 87-100, 2010.
[9] S. Kamara, C. Papamanthou, T. Roeder. Dynamic searchable symmetric encryption, Proceedings of the 2012 ACM conference on Computer and communications security (CCS), ACM, pp. 965- 976, 2012.
[10] Y. Hwang, P. Lee. Public Key Encryption with Conjunctive Keyword
[11] J. Li, Q. Wang, C. Wang. Fuzzy keyword search over encrypted data in cloud computing", Proc. IEEE INFOCOM, pp. 1-5, 2014.
[12] C. Bosch, R. Brinkma, P. Hartel. Conjunctive wildcard search over encrypted data", Secure Data Management. LNCS, pp. 114- 127, 2011.
[13] Z. Liu, Z. Wang, X. Cheng, et al. "Multi-user Searchable Encryption with Coarser-Grained Access Control in Hybrid Cloud, Fourth International Conference on Emerging Intelligent Data and Web Technologies (EIDWT), IEEE, pp. 249-255, 2013.

BIOGRAPHY

Rajeshree Kawdu Duratkar is a M.E student in the computer engineering Department, College of JSPM"s B.S.I.O.T.R, Wagholi, Pune University, Maharashtra, India. She received bachelor of Computer science and engineering (B.E) degree in 2010, from Sipna C.O.E.T, Amravati, MS, India. Her research interests in cloud computing.

DOI: 10.15680/IJIRCCE.2017. 0505319