An Aggregate Key Sharing Mechanism For Sharing Data Between Different Groups Via Cloud

Author[1]

Miss.Sharayu Lande ,Computer Engineering ,pravara rural engineering college Loni ,India
sharayulande07@gmail.com

Author[2]

Prof N.B.Kadu ,Computer Engineering ,pravara rural engineering college Loni ,India

kamleshkadu@gmail.com

ABSTRACT

Today Cloud Computing has became common and most confidential data is changing into centralized into clouds hence for maintaining privacy and security of shared data primarily searchable cryptography is wide used. Sharing of knowledge is a crucial property of cloud computing. we are able to store information on varied information storage servers like e- mail servers and files servers in encrypted form to minimize security and privacy threats however this implies that if we want to get increased amount of security we have to limit the functionality. For instance, if a client desires to retrieve solely documents containing few words, it absolutely was not noted however information storage server perform the search and answer the question while not loss of knowledge and data. As per the property of low maintenance cost, cloud computing delivers an economically suitable and efficient way for sharing various information among cloud users. In this paper, we show how to securely, efficiently and flexibly we share data with others in cloud storage. Unfortunately, sharing data in a multi-user manner while preserving data and data privacy from an untrusted cloud is still a large limitation, due to the change of the membership. In this paper we are introducing an aggregate key for communication in different groups of cloud.

Keywords: *cloud computing, Searchable encryption, Aggregate key, Trapdoor.*

I.INTRODUCTION

Cloud storage is turning into additional widespread these days. In enterprise settings, we have a tendency to see the increase in demand for knowledge outsourcing, that advantages within the field of company knowledge and its management. It is additionally helpful as a core technology for different online technologies for individual applications[2].Cloud computing is thought as an alternate to ancient technology owing to its higher resource-sharing and low-maintenance capabilities. The main aim of cloud computing is to provide high performance energy of computing for various field like military and research organization for performing billions of computations at each second. It is also used in customer oriented areas like portfolios to transfer confidential information. In cloud computing the cloud service providers(CSP) like Amazon are able to give varied services to users with the assistance of powerful information servers. Moving the native information management systems into cloud servers, users will profit of high quality services and store vital investments on their local infrastructure. However sharing information through cloud storage , users area unit at the same time aware of the information leakages in cloud storage[5]. One of the basic services delivered by cloud service suppliers is information storage. consider a knowledge application. There is a corporation which allows its staffs within the same cluster or department to store and share documents or files within the cloud. Identification of privacy is most vital drawback for wide development of cloud computing. Without the proof of identity privacy users are not able to utilize the cloud service as a result of they don't want to show their real identity. To maintain information privacy, a basic plan is to encode files so transfer the encrypted data into the cloud. In this paper, we demonstrate cryptographic scenarios for the problem of searching on encrypted data and provide result of security for the resulting crypto systems[4].

2.RELATED WORK

Searchable encryption is widely used method for maintaining security over the shared data. There is a large amount of literature on searchable encryption, including SSE and PEKS 's schemes . In contrast to those existing schemes ,in the cloud storage, keyword search under the multi-tenancy is a more used scenario. In such a scenario, the data owner will to share a document with a group of authorized users ,and each user who has the access authority can provide a trapdoor to perform the process of keyword search over the shared document, namely, the multiple-users searchable encryption (MUSE) scenario[1]. schemes are

created by sharing the documents searchable encryption key with all users who have access on it, and broadcast encryptions used to reach coarse-grained access control. As a result, in MUSE, the big problem is how to manage which users can access which documents, whereas how to decrease the number of shared keys and trapdoors is not taken in account. Key aggregate searchable encryption can provide efficient solution and it can make MUSE more efficient and practical. Second approach is Multi-Key Searchable Encryption in which the number of trapdoors is equivalent to the number of documents to search over the documents (if user provides to the server a keyword trapdoor under every key along which a matched document can be encrypted). The objective of MKSE is to assure the cloud service provider can perform keyword search by using only one trapdoor over different documents. Next approach is Attribute based encryption which contains every cipher text to be associated with an attribute, and the master-secret key holder can be extract a secret key for a policy of these attributes so that the cipher text can be decrypted by this key if its associated attribute confirms to the policy. In this technique the user's secret key and cipher text is dependent on attributes. Another is searchable symmetric encryption that allows a client to encrypt its data in such a way that this data can get searched still. The most significant application of SSE into the cloud storage is where it enables a client to securely transfer its data to an untrusted cloud provider without losing the ability to search over it[1].SSE is active research and various functionalities of schemes can achieve various levels of security and efficiency. Any practical SSE scheme, however, should satisfy the following properties: sub linear searching time, security, indexes and the ability to modify files efficiently[7]. Previous existing-known SSE schemes cannot achieve all these properties at the simultaneously. This limits the practical value of SSE and reduces its chance of deployment in real-world cloud storage system.

3.PROPOSED SCHEME

In proposed system first of all ,data owner uploads document to the cloud and then it is get encrypted by data owner and keywords are generated. After that these keywords are encrypted and aggregate key is generated .Then data owner will select t he user with which he wants to share the aggregate key and after selection aggregate key will be sent to the user's email id. Then user will search for respected keywords and trapdoor will get generated and it will get send to cloud. After receiving trapdoor, Cloud server will adjust trapdoor for each document. And finally using test algorithm user will download respective document. Nowadays Cloud storage is known as a promising solution for providing universal , convenient, and on-demand access to greater amounts of information shared on the Internet. Today, billions of users are sharing pers onal data such as photos, videos, confidential documents with their friends via social networking applications based on the cloud storage on a daily basis. Business users are also getting attracted by cloud storage due to its numerous advantages, including lower price, greater agility, and better resource utilization capabilities[1].

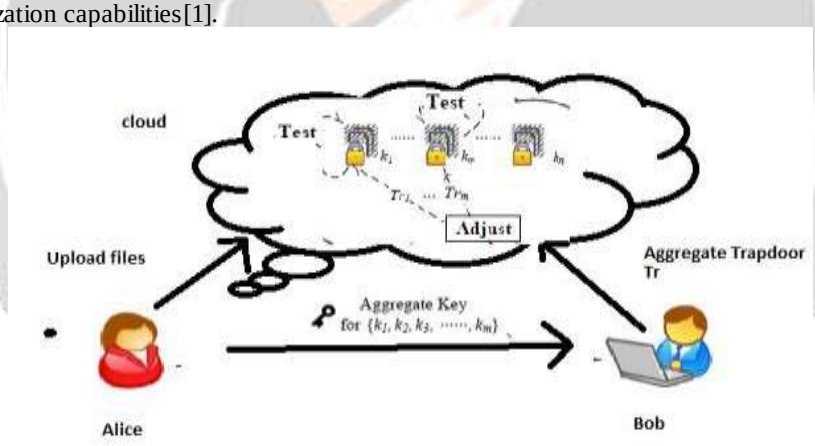

Fig 1. Architecture for KASE

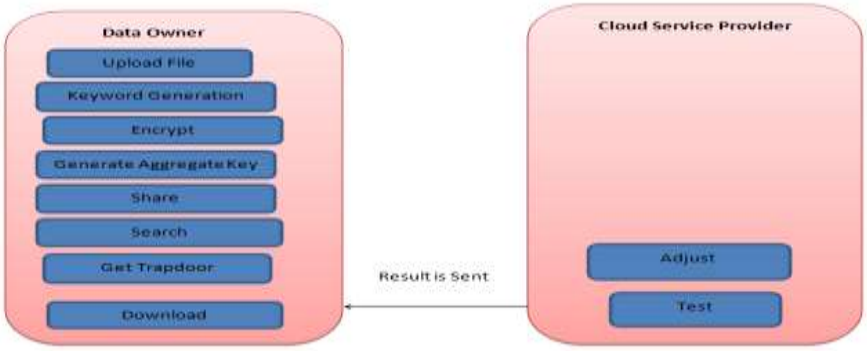

Fig 2.Working of system

4.IMPLEMENTATION DETAILS

4.1 ALGORITHM STRATEGY:

1]Setup(Generation of system parameter): It is the first step for setup the scheme. It accepts security parameter and n number of documents as i input and it gives public system parameter. This is the first algorithm in aggregate key mechanism and usually it is run by cloud service provider.

2]Key Generation{pk,msk}(public and private keys are generated):In this algorithm a random pair of keys can be produced. When a data owner wants to upload a file on cloud then he should generate his public key or secret key. For this purpose key generation algorithm is used which is mostly run by the data owner.

3]Encryption{pk,i}(Documents are encrypted and cipher text is generated):As we know that encryption process changes the documents plain data into the cipher, here also data owner uses encryption to encrypt i -th document and generate its keywords ciphertext. It produces searchable encryption key for each document.

4]Extract{msk,s}(Aggregate key is generated):In key generation algorithm a secret key is generated from which an aggregate searchable encryption key can be created by using extract algorithm. It provides the right of searching a specific keyword for particular set of documents to the other users. It takes the secret key and group of documents and outputs an aggregate key and it is run by the data owner.

5]Trapdoor{k_{agg},w}(Aggregate trapdoor is generated): This algorithm is used for producing an aggregate trapdoor and it is run by user who want to make a search over the n number of documents. It accepts aggregate key generated in above algorithm and set of keyword.

6]Adjust{params,i,S,T_r}(Right trapdoor for each document is generated): In this algorithm the aggregate trapdoor is adjusted to produce a right trapdoor to each document in document set and managed by the cloud server. It accepts various parameters like params,indices set,target document index and aggregate trapdoor.

7]Test{T_{ri} ,i}(Document is retrieved):Test algorithm is used to perform search of keyword over the encrypted document. By taking the trapdoor and indexof document in account it returns the answer by indicating true or false for whether the document contains given keyword or not respectively.

4.2 WORKFLOW OF SYSTEM:

As we know that cloud contains lot of information hence for managing the required information it uses databases and database tables. Suppose our system has four database tables e.Group(grpID,grpName,Param for storing the parameters), Member(memName,memId,Password,Public key for storing information of group member),Document(docID, docName,docPath,enckey,ownerId for storing all the document uploaded through data owner) and shared documents etc.

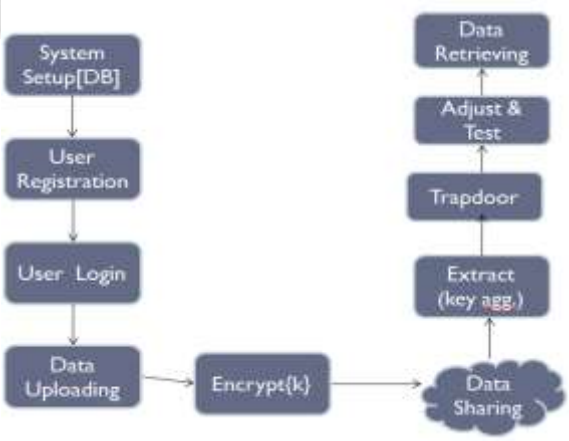

Fig 3. System Workflow

1]Setting up of system: Whenever an business or any other organization sends a request to cloud the cloud will produce above database tables for maintaining a record. Cloud will also assign a unique group id for this organization and it will select manager to control all system work. By using very first setup algorithm it will produce parameters params and these params will get inserted in database table group and update all its fields.

2] Sign up /Registration of user: If a new member want to get add in system then member will go through the registration activity and then manager will assign memberID,memberName,password,and a secret key for new member and manager will store all the record in table member.

3]Login User: After ensuring registration process now user can login in system. Login process is mostly used for authentication of user and verification of password.

4]Uploading data to cloud: When a data owner wants to upload any data or document to cloud then he have to encrypt all the data before uploading and hence it will run encryption algorithm and will produce cipher text. Whenever this new document is get uploaded then cloud server will assign a unique id for document as docId and store it in file path and will update a record in table documents.

5]Data Sharing: For sharing documents with the target member data owner will run extract algorithm for creating aggregate key and to send it to the member and finally maintain the record in shared docs table.

6]Performing Keyword Search: For retrieving the target document which contains the expected keyword a user will run trapdoor to produce keyword trapdoor and submits the trapdoor and ownerId to cloud. When cloud receives that request cloud will run adjust algorithm for producing right trapdoor for each document. And finally it will run test algorithm for searching related keyword. Lastly cloud will return the document in encrypted form.

7]Retrieving data: When user will receive the encrypted document he/she will run decrypt algorithm using aggregate key.

5.RESULT AND DISCUSSION

The performance is highly dependent on the basic cryptographic operations especially in the pairing computation, we study Whether the cryptographic operations based on pairing computation can be efficiently executed using computers.

1]The execution time of KASE Setup is 0.004 seconds and when the maximum number grows up to 10000, it is reasonable that KASE.Setup algorithm only needs 40seconds.

2] The execution time of KASE Encryption is 0.003seconds
and when the maximum number grows up to 10000, it is reasonable that KASE Encryption algorithm only needs 30seconds.

3] The execution time of KASE Extract 0.002 seconds and when the maximum number grows up to 10000, it is reasonable that KASE Encryption algorithm only needs 20seconds.

4] The execution time of KASE Trapdoor 0.005 seconds and when the maximum number grows up to 10000, it is reasonable that KASE trapdoor algorithm only needs 0.005 seconds because it is constant.

5]The execution time of KASE Adjust 0.008 seconds and when the maximum number grows up to 10000, it is reasonable that KASE adjust algorithm only needs 80 seconds.

6]The execution time of KASE Test is 0.003 seconds and when the maximum number grows up to 10000, it is reasonable that KASE Encryption algorithm only needs 30seconds.

Algorithm	Setup	Encryption	Extract	Trapdoor	Adjust	Test
No of documents	10000	10000	10000	10000	10000	10000
Time in Seconds	40	30	20	0.005	80	30

Table 1.Execution time

6.CONCLUSION

Due to the characteristic of low maintenance, cloud computing provides financially suitable and efficient solution for sharin g group resource among cloud users. Cloud storage is known as a promising solution for providing universal , convenient, and on-demand access to greater amounts of information shared on the Internet Our scheme is also very flexible, and it can be simply extended to support more advanced searching query . Here we conclude that KASE system provides a tremendous building block for the construction of secure services in the cloud storage which are not trusted by user . As we will share only sing le key the storage space required will become less and more efficient.

REFERENCES

[1] Baojiang Cui, Zheli Liu_ and Lingyu Wang "Key-Aggregate Searchable Encryption (KASE)for Group Data Sharing via Cloud Storage" PP : 99 , 2015.

[2] C. Chu, S. Chow,W. Tzeng, et al. “Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage”,IEEE Transactions on Parallel and Distributed Systems, 2014, 25(2): 468-477.

[3] Y. Hwang, P. Lee. “Public Key Encryption with Conjunctive Keyword Search and Its Extension to a Multi-user System”, In: Pairing-Based Cryptography C Pairing 2007, LNCS, pp. 2-22, 2007.

[4] X. Song, D.Wagner, A. Perrig. “Practical techniques for searches on encrypted data”, IEEE Symposium on Security and Privacy, IEEE Press, pp. 44C55, 2000.

[5] X. Liu, Y. Zhang, B. Wang, and J. Yan. “Mona: secure multiowner data sharing for dynamic groups in the cloud”, 1191. IEEE Transactions on Parallel and Distributed Systems, 2014, 25(2): 468-477.

[6] 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.

[7] 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.

[8]J. Li, Q. Wang, C. Wang. “Fuzzy keyword search over encrypted data in cloud computing”, Proc. IEEE INFOCOM, pp. 1-5, 2010.

[9] C. Bosch, R. Brinkma, P. Hartel. “Conjunctive wildcard search over encrypted data”, Secure Data Management. LNCS, pp. 114-127, 2011.

[10] C. Dong, G. Russello, N. Dulay. “Shared and searchable encrypted data for untrusted servers”, Journal of Computer Security, pp. 367-397, 2011.

[11] F. Zhao, T. Nishide, K. Sakurai. Multi-User Keyword Search Scheme for Secure Data Sharing with Fine-Grained Access Control. Information Security and Cryptology, LNCS, pp. 406-418, 2012.

[12] J. W. Li, J. Li, X. F. Chen, et al. “Efficient Keyword Search over Encrypted Data with Fine-Grained Access Control in Hybrid Cloud”, In: Network and System Security 2012, LNCS, pp. 490- 502, 2012.

[13] J. Li, K. Kim. “Hidden attribute-based signatures without anonymity revocation”, Information Sciences, 180(9): 1681-1689, Elsevier, 2010.

[14] X.F. Chen, J. Li, X.Y. Huang, J.W. Li, Y. Xiang. "Secure Outsourced Attribute-based Signatures”, IEEE Trans. on Parallel and Distributed Systems, DOI.ieeecomputersociety.org/10.1109/TPDS.2013.180, 2013.

[15] J.Li, X.F. Chen, M.Q. Li, J.W. Li, P. Lee, Wenjing Lou. “Secure Deduplication with Efficient and Reliable Convergent Key Management”, IEEE Transactions on Parallel and Distributed Systems, 25(6): 1615-1625, 2014.

[16] J. Li, K. Kim. “Hidden attribute-based signatures without anonymity revocation”, Information Sciences, 180(9): 1681-1689, Elsevier, 2010.

[17] X.F. Chen, J. Li, X.Y. Huang, J.W. Li, Y. Xiang.
"Secure Outsourced Attribute-based Signatures", IEEE Trans. on Parallel and Distributed Systems, DOI.ieeecomputersociety.org/10.1109/TPDS.2013.180, 2013.

[19] 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.

[20] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing", Proc. IEEE INFOCOM, pp. 525-533, 2010.

[21] B. Wang, B. Li, and H. Li, "Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud", Proc. 10th Int'l Conf. Applied Cryptography and Network Security, pp. 507-
525, 2012.

[22] D. Boneh, C. Gentry, B. Waters." Collusion resistant broadcast encryption with short ciphertexts and private keys", Advances in CryptologyCCRYPTO 2005, pp. 258-275, 2005.

[23] D. H. Phan, D. Pointcheval, S. F. Shahandashti, et al. "Adaptive CCA broadcast encryption with constant-size secret keys and ciphertexts", International journal of information security, 12(4):
251-265, 2013.