Volume 3, Issue 3, March-2016, pp. 106-111 ISSN (O): 2349-7084

International Journal of Computer Engineering In Research Trends

Available online at: www.ijcert.org

Generating Aggregate Key for Group Data Sharing by Means of Cloud Data Storage

[1]Sandeep Srinivas Dwaram, [2] P.Venkateswara Rao.

[1]*M.Tech, CST Branch, Dept. of CSE, Adikavi Nannaya University, Rajahmundry, Andhra Pradesh, India.*

[2]*Associate Professor, Department of Computer Science and Engineering, Adikavi Nannaya University, Rajahmundry, Andhra Pradesh, India.*

Abstract: -Sharing of encrypted data with various kinds of users via public cloud storage always leads to greater security concerns over many types of data leaks in the cloud. To make it possible, efficient generation of keys and their management should take place in developing schemes. In the situation of sharing any class of documents to any class of users demands various numbers of encryption keys for each document used. Further, need a large number of keys for keyword query searching for each document used. This increases the complexity in generating various numbers of keys for decrypting the document and decreases the efficiency of keyword query searching. This also makes the user to store large number of keys for both encryption and search which makes the situation impractical. In this we approached, by suggesting concept of key aggregate searchable encryption (KASE) and initializing the idea through a real KASE scheme, in which the data owner will send a single aggregate key of group of documents to group of authorized users, and the user will send a single trapdoor to the cloud server for keyword query searching.

Keywords—data sharing, Searchable encryption, data privacy, cloud storage

1 INTRODUCTION

These days the capacity in the cloud has emerged as a proficient response for suitable and on-interest gets to immense measures of data shared over the Internet. Business clients are being focusing by cloud storage because of its few advantages, including lower cost, better dexterity, and enhanced asset usage. Ordinary clients are likewise sharing private information, for example, photographs and recordings, with their companions through interpersonal organization applications in view of cloud. Then again, while profiting from the convenience of sharing information through cloud storage, clients are additionally progressively stressed over inadvertent information uncover by the cloud.Such information uncovering, will be performed by malignant adversary or a devilish cloud administrator, can frequently direct to extreme infringement of private information or classified information in regards to business. To talk about clients tension over conceivable information uncover in cloud storage, a general methodology is for the information proprietor to encode all the information before transferring them into the Such that without further ado the scrambled information might be get back and decoded by people who contains the unscrambling keys. Such distributed storage is frequently called the cryptographic cloud data storage [6].Though; the encryption of information fabricates it requesting for clients to pursuit and after that ideal recover just the information including the given keywords. A typical arrangement is to utilize a searchable encryption (SE) plan in which the information owner is required to encode potential catchphrases and transfer them to the cloud together with scrambled information, such that, for recovering information coordinating a keyword, the client will send the coordinating keyword to the cloud to respond for the inquiry over the encoded information.

Even though mixing a searchable encryption Scheme with cryptographic cloud storage can

accomplish the essential security needs of a cloud storage, running such a system for large scale application relating huge number of users and large number of files may still be delayed by practical issues relating the well-organized management of encryption keys, which, to the Finest of our knowledge essentially, the need for specifically imparting encoded information to various clients for the most part requests distinctive encryption keys to be utilized for various documents. Then again, this includes the quantity of keys that should be spread to clients, both for them to seek over the encoded records and to unscramble the documents, will be in respect to the quantity of such documents. Such a substantial number of keys must not just be spread to clients by means of secure channels, additionally be safely put away and took care of by the clients in their gadgets. The understood prerequisite for secure correspondence, stockpiling, and computational trouble might bring about framework inadequacy.

In this paper, we propose the novel idea of key-aggregate searchable encryption (KASE), and instantiating the idea through a solid KASE technique. The proposed KASE plan identifies with any cloud storage those backings the searchable gathering information sharing component, which implies any client, might like to appropriate a gathering of documents which are specific with a gathering of selected clients, while allowing the last to do keyword look over the prior. To keep up searchable gathering information sharing the primary requirements for effective key administration is twofold. Essentially, an information proprietor needs to distribute a solitary aggregate key (rather than a gathering of keys) to a client for sharing any number of documents. Ensuing, the client needs to present a solitary aggregate trapdoor to the cloud for performing keyword seeks over any amount of shared records. KASE plan can guarantee both solicitations.

2. RELATED WORK:

1) Principally we depict a typical structure of key aggregate searchable encryption (KASE) gathered from a few polynomial calculations for security parameter setup, key generation, encryption, key extraction, trapdoor generation, trapdoor adjustment, and trapdoor testing. We then clarify both practical and security necessities for plotting a legitimate KASE scheme

2) We then instantiate the KASE skeleton by Scheming a solid KASE plan. In the wake of giving the full structure for the calculations, we investigate the viability of the plan, and set up its wellbeing through definite examination.

2.1 Searchable Encryption

Searchable encryption schemes divided into two types, i.e,Searchable Symmetric Encryption (SSE) and Public Key Encryption with Keyword Search (PEKS).Both can be described as the tuple SE= (Setup,Encrypt,Trapdoor(Trpdr),Test);

1. **Setup** (1^{λ}): Owner runs this algorithm toset up the plan. It takes as information a security Parameter (1^{λ}) and yields vital keys.
2. **Encrypt** (l;n): This algorithm is controlled by the ownerto encode the information and generate its keyword ciphertexts. It takes as input the info (n), owner's important keys including searchable encryption key (l) and data encryption key, outputs data ciphertext and keyword ciphertext (Cn).
3. **Trpdr**(l;x): This algorithm is controlled by a user toproduce a trapdoor (Trd) for a keyword (w) using key (l).
4. **Test**(Trd, Cn): This algorithm is controlled by the cloudserver to perform a keyword search over encode data. It takes as input trapdoor Trd and the keyword ciphertexts (Cn), yields whether (Cn) contains the specified keyword. For precision, it is required that, for a message (n) containing keyword

x and a searchable encryption key l, if (Cn $\leftarrow$ -Encrypt(l;n) and Tr $\leftarrow$ Trpdr(l;x)), then Test(Trd, Cn)=true.

3. THE KEY-AGGREGATE SEARCHABLE ENCRYPTION (KASE) SCHEMA

3.1 Problem Statement

In this paper, we propose the novel methodology of Key-aggregate searchable encryption (KASE) as an upgraded arrangement, as delineated in Fig.1 (b). , in KASE, seeta necessities to issue a single aggregate key, rather than {ki}mi=1 for imparting m reports to Ram, and Ram needs to issue a single aggregate trapdoor, rather than { Tri }mi=1, to the cloud server. The cloud server can use this

aggregate trapdoor and some open information to complete keyword seeks and return to the outcome to Ram. Therefore, in KASE, the assignment of keyword pursuit right can be accomplished by sharing the single aggregate key.

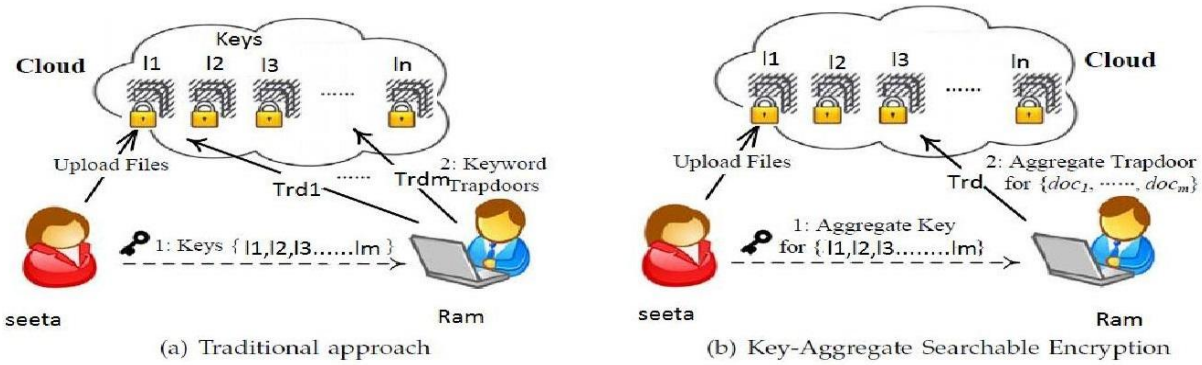


Fig. 1.keyword search in group data sharing system.

To outline a key-aggregate searchable encryption technique under which any subset of the keyword ciphertexts from any arrangement of reports is searchable with a consistent size trapdoor created by a steady size aggregate key

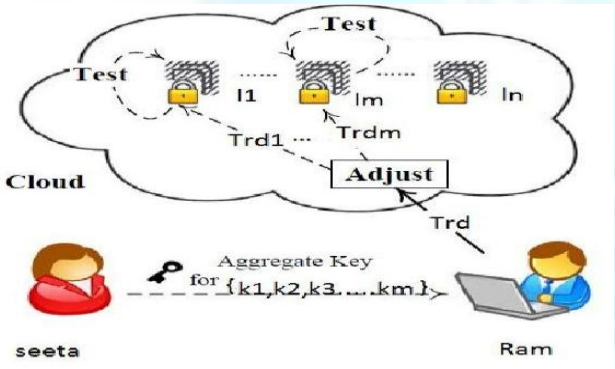


Fig.2 Framework of Key-aggregate searchable encryption.

3.2 The KASE Scheme Construction

The KASE development is made out of few algorithms. Exceptionally, to set up the strategy, the cloud server would create open parameters of the system during the **Setup** algorithm, and these open parameters can be reprocessing by divergent data owners to disperse their files. For each information owner, they should produce a public/master-secret key pair through the **Keygen** algorithm. Keywords of each documentcan be encoded through the **Encrypt** algorithm with the exclusive searchable encryption key. In that case, the information owner can apply the master-secret key to produce an aggregate searchable encryption key for a gathering of chose documents through the **Extract** algorithm.The aggregate key can be spread safely to affirm user

Who need to get those documents? After that, as appeared in Fig.2, an affirmed user can make a keyword trapdoor by means of the **Trapdoor** algorithm using this aggregate key, and present the trapdoor to the cloud. After getting the trapdoor, to carry out the keyword search over the specific arrangement of documents, the cloud server will run the **Adjust** algorithm to create the privilege trapdoor for each document, and after that run the **Test** algorithm to test whether the documentcontains the keyword.

This construction is summarized in the following.

1. Setup(1^{λ}, n): This algorithm is controlled by the cloudservice supplier to set up the scheme. On input of a security parameter 1^{λ} and the most extreme conceivable number n of documents which belongs to a data owner, it yields the public system parameter params.

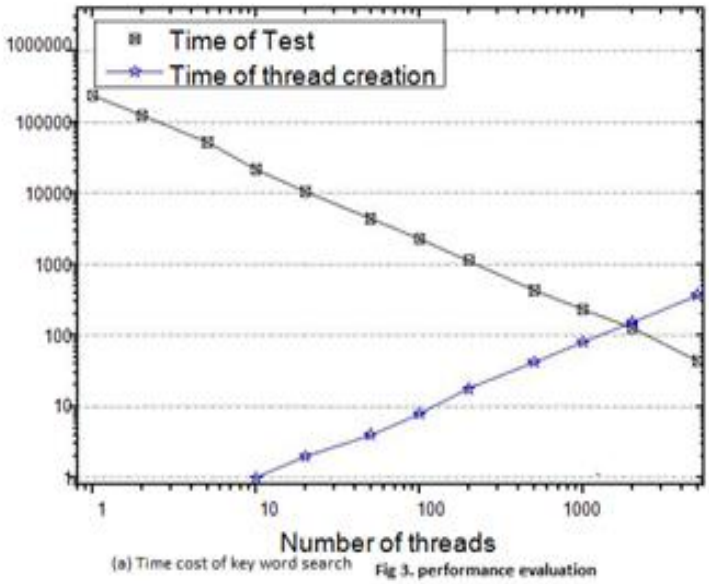
2. **Keygen**: This algorithm is controlled by the dataowner to produce a random key pair (pk,msk).

3. **Encrypt**(pk, i): This algorithm is controlled by the dataowner to encode the i-th document and produce its keywords ciphertexts. For each file, this algorithm will produce a delta Дi for its searchable encryption key ki. On input of the owner's public key (pk) and the file index i, this algorithm yields data ciphertext and keyword ciphertexts Ci.

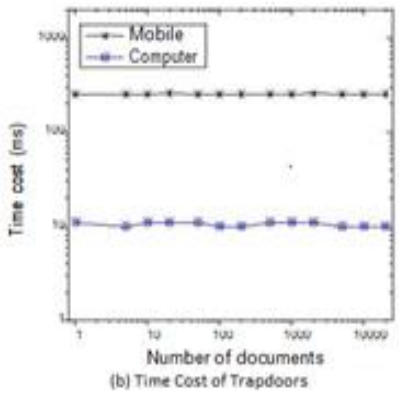
3. **Extract**(msk, S): This alogo controlled by the Data owner to produce an aggregate searchable encryption key for hand over the keyword search look a good fit for set of documents to other users. It takes as input the owner's master-secret key (msk) and a set (S) which enclose the directory of documents, and then outputs the aggregate key kagg.

3.3 Trapdoor Generation

1. **Trapdoor** (kagg, x): This algorithm is controlled By the user who got the aggregate key to perform a search. It uses as input the aggregate searchable encryption key (kagg) and a keyword (w), then yields only one trapdoor (Trd).
2. **Adjust** (params, i, S, Trd): this algorithm is controlled by cloud server to fit the aggregate trapdoor to produce the right trapdoor for each distinct document. It takes as input the system public parameters params, the set (S) of documents indices, the index (i) of target document and the aggregate trapdoor (Tr), then yields each trapdoor (Tri) for the i-th target document in(S).
3. **Test**(Tr_i, i): this algorithm is controlled by the cloud server to action keyword search over an encoded document. It takes as input the trapdoor (Tr_i) and the document index (i), then yields true or false to denote whether the document doc_i contains the keyword (w).



(a) Time cost of key word search



(b) Time Cost of Trapdoors

Fig 3. performance evaluation

Fig 3. performance evaluation

4. Performance Evaluation

Taking that: **1)** in a viable information sharing sys-tem taking into account cloud storage, the client can recover information by any conceivable gadget and the cell phones are broadly utilized now; **2)** the execution is profoundly dependent on the essential cryptographic operations particularly in the blending calculation, we ponder whether the cryptographic operations in light of matching computation can be productively executed utilizing both PCs and cell phones

1.The executable time of KASE.Trapdoor is a con-stant, i.e., 0.01 second in PC and 0.25 sec-ond in cell phones. Truth be told, the numerical operation in KASE.Trapdoor is the once multi-plication in G, so that the watch keyword pursuit can be performed proficiently in both cell phones and PC. Contrasted and different plans, there is a critical change in our plan.

2. The multi-thread procedure is accepted in our examination. To test the execution, we set the quantity of keyword ciphertexts as 10000. As appeared in Fig.3 (a), we can see that the execution time of KASE.Test will be decreased when we build the quantity of threads. At the point when the number grows up to 200, it just needs 1 second to complete the keyword look more than 10000 keyword ciphertexts. We additionally see that when the quantity of threads is vast, it would take more opportunity to make these threads. At the point when the number grows up to 1000, the season of thread creation will get to be 80 millisecond. Along these lines, the multi-thread strategy can give the assistance to enhancing execution, yet the quantity of threads ought to be chosen deliberately in the pragmatic applications.

5. CONCLUSION and FUTURE ENHANCEMENT

Thinking about of the practical issue of protection saving information sharing framework in view of public cloud storage which is need a data owner to allot a countless of keys to users to access them to permit the documents, In this proposed concept of key-aggregate searchable encryption (KASE) and build a solid KASE scheme. It can provide a productive solution to building practical data sharing system based on public cloud storage. In a KASE scheme, the data owner needs to share a single key to a user when contributing a lot of documents with the user, and the user needs to produce a single trapdoor when they queries over all files shared by the same owner. On the other hand, if a user wants to query over documents shared by multiple owners, that user must generate multiple trapdoors to the cloud. The future upgrade for this proposed work is to discover how to reduce the quantity of trapdoors under multi-owners attaining so as to set the security.

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. 6, No. 1, January 2014

[2] 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.

[3] X. Liu, Y. Zhang, B. Wang, and J. Yan. "Mona: secure multiowner data sharing for dynamic groups in the cloud", IEEE Transactions on Parallel and Distributed Systems, 2013, 24(6): 1182-1191.

[4] 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.

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

[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] P. Van,S. Sedghi, JM. Doumen. "Computationally efficient searchable symmetric encryption", Secure Data Management, pp.87-100, 2010.

[8] 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.

[9] D. Boneh, C. G, R. Ostrovsky, G. Persiano. "Public Key Encryption with Keyword Search", EUROCRYPT 2004, pp. 506C522,2004.

[10] 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.

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

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

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

[14] 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.

[15] 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.

[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.

[18] 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.

[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. 10thInt'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

