Key-Aggregate Searchble Encryption(KASE) for Group Data Sharing

[1]Salman Mujawar, [2]Prof. Dr. P. K. Deshmukh
[1]Student at RSCOE Pune,[2]Professor of Computer Engg. at RSCOE Pune,
[1]Computer Science Department,
[1]Rajarshi Shahu College of Engineering Pune, Pune, India

***Abstract*— In distributing computing the ability of specifically imparting scrambled information to distinctive clients through open distributed storage might significantly ease security worries over incidental information spills. The productive administration of encryption keys contains key test to outlining such encryption plans. The looked for flexibility of offering any social affair of picked reports to any get-together of customers solicitations unmistakable encryption keys to be used for various documents. Now a days, the need of safely dispersing to clients a substantial number of keys for both encryption and look becomes an important, and those clients will need to safely store the acquired keys. What's more, present a pretty much as broad number of watchword ,trapdoors to the cloud with a particular deciding objective to perform look for over the shared data. The suggested requirement for secure correspondence, stockpiling, and many-sided quality obviously renders the methodology unreasonable. In this paper, we address this viable issue, which is to a great extent dismissed in the proposing so as to write, the novel idea of key aggregate searchable encryption (KASE). The instantiating the idea through a solid KASE plan, in which an information proprietor just needs to convey a aggregate key to a client for sharing documents, and the client just needs to present a solitary trapdoor to the cloud for questioning the mutual reports. The security examination and execution appraisal both assert that our proposed arrangements are provably secure and in every way that really matters successful**.

***Index Terms*— cloud storage, data privacy , Key aggregation, Searchable encryption.**

I. INTRODUCTION

Considering the issue of security shielding data sharing system in light of open cloud stockpiling which obliges a data proprietor to pass on a generous number of keys to customers to engage them to get to his/her report. We interestingly propose the thought of key-aggregate searchable encryption (KASE) and develop a solid KASE arrangement. Both examination and evaluation results certify that our work can give a successful response for building down to earth information sharing framework in light of open cloud stockpiling. In a KASE arrangement, the data owner just needs to a proper a aggregate key for all his/her document to a customer when offering heaps of archives to the client, and the client just needs to present solitary trapdoor when he inquiries over all records shared by the same data owner Regardless, if a customer needs to request over reports shared by various information proprietor, he should make diverse trapdoors to the cloud. The most effective method to lessen the quantity of trapdoors under multi-data owner setting is a future work What's more, united mists have pulled in an impressive measure of thought Now a days, yet our KASE can't be joined for this circumstance direct It is furthermore a future work to give the response for KASE by virtue of joined mists. There is a rich writing on searchable encryption, counting SSE plans and PEKS plans. Instead of those ebb and flow work, in the setting of conveyed stockpiling, watchword seek under the multi-occupancy setting is a more typical situation.

A. Motivation

Presently number of clients store number of records on distributed storage. We are going to produce single aggregate key for all reports for each client. This motivate us to produce a single aggregate key for all documents of every user and gives security.

II. RELATED WORK

In [1] the creator only needs to distributed a single to user when sharing lots of documents with the user, and the user need only to submit a single trapdoor(keyword) when the user required all documents shared by the same user. This paper issues like for multiple owners generates multiple trapdoors to the cloud.

In [2] the creator addresses the issues like fine-graininess, versatility, and data security of access control challenging open issue by, on one hand, characterizing and enforcing access policies based on information attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in fine grained information access control to untrusted cloud servers without disclosing the fundamental information substance.

In [3] creator proposed a another secure provenance plan light of the bilinear pairing techniques. As the fundamental bread and butter of information forensics and post investigation in distributed computing, the proposed plan is characterized by giving the

information confidentiality on sensitive documents stored in cloud, anonymous authentication on user access, and provenance tracking on disputed documents. With the provable security techniques, we formally show the proposed plan is secure in the standard model.

In [4] the creator proposed a secure multi owner information sharing plan, named Mona. By utilizing bunch signature and dynamic shows encryption techniques, any cloud client can secretly impart information to others. Then, the capacity overhead and encryption calculation expense of this plan are autonomous with the quantity of denied users.

In [5] the creator demonstrate to safely, productivity, and adaptability share information with others in distributed storage. They depict new public-key crypto framework which create consistent-size cipher texts such that productive appointment of decryption rights for any set of cipher texts are possible. The novelty is that one can aggregate any set of secret keys and make them as compact as a single key, but encompassing the power of all the keys being aggregated.

In [6] the author described cryptographic schemes for the problem of searching on encrypted data and provides proofs of security for the resulting crypto systems. This techniques have a number of crucial advantages. The author conclude that they provides a powerful new building block for the construction of secure services in the untrusted infrastructure.

III. ISSUES

Implementation of all these aforementioned algorithms in section II provides the need for selectively sharing encrypted data with different users (eg., sharing of files with a certain friends in social network, or sharing of business documents with colleagues on a cloud) usually demands different encryption keys to be used for different files. A large number of keys is not distributed to users via secure channel, as well as is not securely stored and managed by the users in their personal device. In this paper we address this challenge by proposing concept of key –aggregate searchable encryption (KASE).

IV. PROBLEM STATEMENT

The individuals who have key can decipher the information before putting away information to the cloud. Such dispersed cache is occasionally called the cryptographic passed on cache. Regardless, the encryption of information makes it making progress toward clients to ask for and after that especially recovers just the information containing given fundamental words. An ordinary strategy is to utilize a searchable encryption (SE) plan in which the information proprietor is obliged to scramble potential indisputable words and trade them to the cloud together with encoded information, such that, for recovering information arranging a watchword, the client will send the narration out charm word trapdoor from the cloud for performing pursue over the blended information.

V. IMPLEMENTATIONS DETAILS

This framework contains following phases Setup, Keygen, Encrypt, Extract, Trapdoor, Adjust and Test.

A. System model

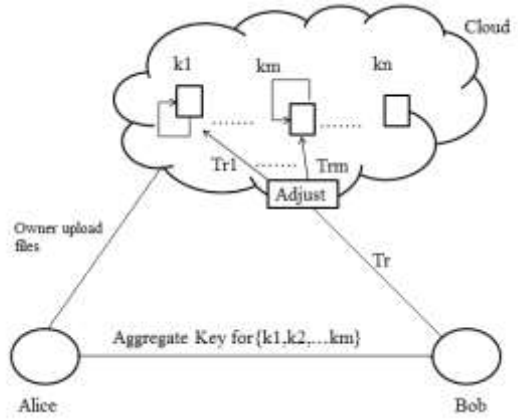


Fig.1. System Architecture

After careful analysis the system has been known to possess the subsequent modules:

• **Group generation**: Initially data owner generates particular group according to files going to upload by him. Data owner gives individual value for group and key for grop.

• **Registration**: User have to register his details and select group in which he is interested.

• **Acceptance:** Data owner accepts request send by the user. At this stage data owner get details of user who are going to download his files.

• **File upload**: Data owner upload files to particular group.

•Encrypt: this phase is bound by the data owner to encrypt the i^{th} paper and undergo its keywords' cipher texts. For every paper, this algorithm will build a delta Δi for its exploration- able encryption underlying. On input of the owner's give birth to fundamental and the file possession, this algorithm outputs statistics cipher text and keyword cipher texts.

• **Keygen**: this phase is deliver by the materials organization to carry a random essential pair.

• **Decrypt**: this phase is bound by the data owner to originate an aggregate searchable encryption key for delegating the keyword search right for a certain set of documents to other users. It takes as input the owner's master-secret key and a set which contains the indices of paper, then outputs the aggregate key.

• **Trapdoor:** this phase is bound by the user who has the put key to finish a search. It takes as input the lay away searchable encryption key and a keyword , outputs only one trapdoor.

• **Adjust:** this phase is bound by cloud server to adjust the aggregate trapdoor to originate the right trapdoor for each different paper. It takes as input the system public parameters, the set of documents' indices, the index i of target paper and the aggregate trapdoor, then outputs each trapdoor for the i^{th} target paper in paper indices.

• **Test**: this phase is run by the allay salver to perform keyword search over an encrypted document. It takes as input the trapdoor and the document like mad easily, adapted outputs genuine to denote whether the document contains the keyword.

• **File download**: in this phase user have to enter individual key, aggregate key, speke prime and speke value. After successfully entering these value file will be downloaded.

B. Mathematical Model

Let S be whole System,
S={I, P, O}
I-input,
P-procedure,
O- Output.
I={S, K, E, D, Tr, Ak, T}
D={d1, d2, dn}
U={u1, u2, un}
K={k1, k2, kn}
DO -upload {d1, d2, dn}
DO -generate {k1, k2, kn}
DO -generate {AK}
AK –E{AK}
DO -send E{AK} to {U}
User send Tr {EAK, w} to CS
CS –Test {Tr, Di}
O = {U - Download (D) if (Tr, Di) matched at CS}

C. Memorization Parameters

I: Memorization Parameters

Symbol	Meaning
S	Setup.
KG	KeyGen
E	Encrypt
Tr	Trapdoor
T	Test
U	Number of users
K	Number of keys
DO	Data owner
AK	Aggregate key
EAK	Encrypted aggregate key

CS	Cloud storage
Di	Document index

D. Algorithm

Algorithm 1 RSA.

Choose two very large random prime integers:

1. p and q
2. 2. Compute n and f(n):
3. n = pq and f(n) = (p-1)(q-1)
4. 3. Choose an integer e, 1 < e < f(n) such that:
5. gcd(e, f(n)) = 1(where gcd means greatest common denominator)
6. 4. Compute d, 1 < d < f(n) such that:
7. ed = 1 (mod f(n))
8. the public key is (n, e) and the private key is (n, d)
9. the values of p, q and f(n) are private
10. e is the public or encryption exponent
11. d is the private or decryption exponent

Algorithm 2: Speke

1. A and B agree to use an appropriately large and randomly selected safe prime p, as well as a hash function H().
2. A and B agree on a shared password Ïє.
3. A and B both construct g = H(Ïє)2 mod p.
4. A chooses a secret random integer a, then sends B
ga mod p.
5. B chooses a secret random integer b, then sends A gb mod p.
6. A and B each abort if their received values are not in the range [2,p-2], to prevent small subgroup confinement attack.
7. A computes K = (gb mod p)a mod p.
8. B computes K = (ga mod p)b mod p.

VI . EXPERIMENTAL SETUP

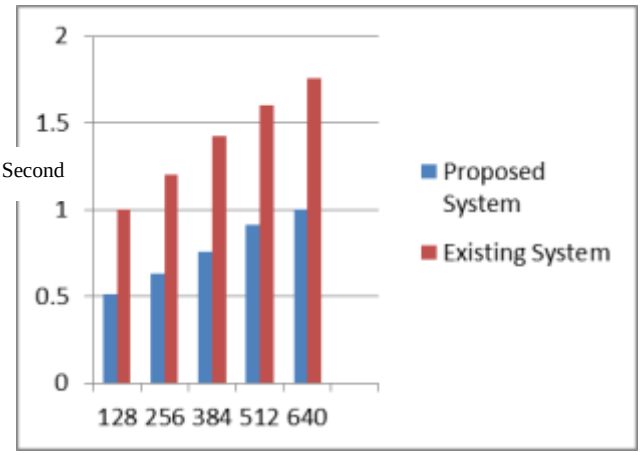
The system is built using Java and J2EE framework (version J2SDK 1.5), web technologies (JSP, HTML, CSS), Apache Tomcat, MySQL database on Microsoft Windows platform. The experiments have been performed on the machine with the following specifications: Intel Core 2 dual processor with 2.5 GHz CPU, 2 GB RAM, 180 GB Hard disk and running Microsoft Windows 7 Home Basic x64 OS.

VII. RESULT AND DISCUSSION

In this section we provide a concept of key aggregate searchable encryption. The data creator needs to distributed only a single aggregate key of his all document, the user needs to submit a single trapdoor when he needs to download a documents shared by the same creator. Our proposed system provides greater time efficiency than the existing system.

Data in KB	Time complexity for decrypt [Proposed System] in second	Time complexity for decrypt [Existing System] in second
128	0.51	1
256	0.63	1.2
384	0.76	1.42
512	0.91	1.6

640	1	1.76



VIII. CONCLUSION

Considering the even minded issue of security sparing data sharing structure considering open cloud cache which obliges a data owner to proper a generous number of keys to customers to enable them to get to his/her records, we propose the thought of key-total searchable encryption (KASE) and add to a strong KASE arrangement. Both examination and evaluation results assert that our work can give an effective response for building practical data sharing system considering open cloud cache. In a KASE arrangement, the owner simply needs to scatter a lone key to a customer when granting heaps of reports to the customer, and the customer simply needs to display a single trapdoor when he request over all records shared by the same owner. Then again, if a customer needs to address over documents shared by various proprietors, he should produce different trapdoors to the cloud. The best system to diminish the amount of trapdoors under multi-owner setting is a future work. Furthermore, brought together mists have pulled in a lot of thought nowadays, yet our KASE can't be associated for this circumstance particularly. It is also a future work to give the response for KASE on description of united cloud.

IX. ACKNOWLEDGMENT

We would like to thanks JSPMs Rajarshi Shahu College of Engineering, Computer Engineering Department as well as publishers for making their resources available and teachers for their guidance. We are thankful to the authorities of Board Of Studies, Computer Engineering in Savitribai Phule Pune University for their constant guidelines and support. We also thank the college authorities for providing the required infrastructure and support. Finally, we would like to extend a heartfelt gratitude to friends and family members.

REFERENCES

[1] Baojiang Cui, Zheli Liu and Lingyu Wang, "Key-aggregate searchable encryption(KASE) for group data sharing via cloud storage," IEEE transactions on computer 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 multiowner 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, " 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 - 27 Key -Aggregate Searchable Encryption(kASE) For Group Data Sharing, " Proceedings of the 2012 ACM conference on Computer and communications security (CCS), ACM, pp. 965-976, 2012.
[10] D. Boneh, C. G, R. Ostrovsky, G. Persiano, " Public Key Encryption with Keyword Search," EUROCRYPT 2004, pp. 506C522,2004.

[11] 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.
[12] J. Li, Q. Wang, C. Wang, "Fuzzy keyword search over encrypted data in cloud computing," Proc. IEEE INFOCOM, pp. 1-5,2010.
[13] C. Bosch, R. Brinkma, P. Hartel, "Conjunctive wildcard search over encrypted data, Secure Data Management", LNCS, pp. 114-127, 2011.
[14] C. Dong, G. Russello, N. Dulay, "Shared and searchable encrypted data for untrusted servers," Journal of Computer Security,pp. 367-397, 2011.
[15] 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.
[16] J. W. Li, J. Li, X. F. Chen, " 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.