International Journal of Advanced Technology and Innovative Research

www.ijatir.org

ISSN 2348–2370
Vol.08,Issue.14,
October-2016,
Pages:2741-2745

Scalable Data Sharing in Cloud Storage with Key Aggregate Cryptosystem

NABEELA SUMEEN[1], S. SHALINI[2], M. SWAPNA[3]

[1]PG Scholar, Dept of CSE, Aurora's Scientific Technological & Research Academy, Hyderabad, TS, India.
[2]Assistant Professor, Dept of CSE, Aurora's Scientific Technological & Research Academy, Hyderabad, TS, India.
[3]Associate Professor, Dept of CSE, Aurora's Scientific Technological & Research Academy, Hyderabad, TS, India.

Abstract: An efficient cryptographic approach for data sharing where data is shared among a group of users as Data sharing is an important functionality in cloud storage. How to securely and efficiently share a collection of data related to any subject areas with others in cloud storage. Development of new novel concept of Key- Aggregate Searchable Encryption (KASE). This concept is implemented through development of concrete key-aggregate searchable encryption framework scheme. This scheme is described as where a data owner only needs to generate and distribute a single aggregate key to a data user for sharing a large number of documents and on the other side user only needs to submit a single aggregate trapdoor to the cloud server, so that he/she can query over the shared documents by the help of generated single aggregate trapdoor. This proposed scheme is perfectly more secure and practically efficient. It is an effective method which is considered as best solution to build a practical data sharing system based on public cloud storage. A detailed review of various methods used for data access controls and encryption is presented and a brief comparison among the discussed methods is given.

Keywords: Cloud Storage Provider, Outsourcing, attribute based Encryption, Key-Aggregate Cryptosystem.

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 business. 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. 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.

II. EXISTING SYSTEM

There is a rich literature on searchable encryption, including SSE schemes and PEKS schemes. In contrast to those existing work, in the context of cloud storage, keyword search under the multi-tenancy setting is a more common scenario. In such a scenario, the data owner would like to share a document with a group of authorized users, and each user who has the access right can provide a trapdoor to perform the keyword search over the shared document, namely, the "multi-user searchable encryption" (MUSE) scenario. Some recent work focus to such a muse scenario, although they all adopt single-key combined with access control to achieve the goal. In, muse schemes are constructed by sharing the document's searchable encryption key with all users who can access it, and broadcast encryption is used to achieve coarse-grained access control. Attribute based encryption (ABE) is applied to achieve fine-grained access control aware keyword search. As a result, in muse, the main problem is how to control which users can access which documents, whereas how to reduce the number of shared keys and trapdoors is not considered. Key aggregate searchable encryption can provide the solution for the latter, and it can make muse more efficient and practical.

III. PROPOSED SYSTEM

In this paper, we address this challenge by proposing the novel concept of key-aggregate searchable encryption (KASE), and instantiating the concept through a concrete kase scheme. The proposed KASE scheme applies to any cloud storage that supports the searchable group data sharing functionality, which means any user may selectively share a group of selected files with a group of selected users, while allowing the latter to perform keyword search over the former. To support searchable group data sharing the main requirements for efficient key management are twofold. First, a data owner only needs to distribute a single aggregate key (instead of a group of keys) to a user for sharing any number of files. Second, the user only needs to submit a single aggregate trapdoor (instead of a group of trapdoors) to the cloud for performing keyword search over any number of shared files. To the best of our knowledge, the kase scheme proposed in this paper is the first known scheme that can satisfy both requirements (the key-aggregate cryptosystem [4], which has inspired our work, can satisfy the first requirement but not the second). Contributions. More specifically, our main contributions are as follows.

- we first define a general framework of key aggregate searchable encryption (KASE) composed of seven polynomial algorithms for security parameter setup, key generation, encryption, key extraction, trapdoor generation, trapdoor adjustment, and trapdoor testing. We then describe both functional and security requirements for designing a valid KASE scheme.
- we then instantiate the KASE framework by designing a concrete KASE scheme. After providing detailed constructions for the seven algorithms, we analyze the efficiency of the scheme, and establish its security through detailed analysis.
- we discuss various practical issues in building an actual group data sharing system based on the proposed kase scheme, and evaluate its performance. The evaluation confirms our system can meet the performance requirements of practical applications

IV. LITERATURE SURVEY

Description: In this paper, it describes the cryptographic schemes for the problem of searching on encrypted data and provides proofs of security for the resulting crypto systems. These techniques have a number of crucial advantages. They are provably secure, they provide provable secrecy for encryption, in the sense that the un trusted server cannot learn anything about the plaintext when only given the ciphertext; they provide query isolation for searches, meaning that the un trusted server cannot learn anything more about the plaintext than the search result; they provide controlled searching, so that the un rusted server cannot search for an arbitrary word without the user's authorization; they also support hidden queries, so that the user may ask the un trusted server to search for a secret word without revealing the word to the server. The algorithms we present are simple, fast (for a document of length n, the encryption and search algorithms only need O(n)stream cipher and block cipher operations), and introduce almost no space and communication overhead, and hence are practical to use today.

Title: Searchable Symmetric Encryption: Improved

Description: Searchable symmetric encryption (SSE) allows a party to outsource the storage of his data to another party in a private manner, while maintaining the ability to selectively search over it. This problem has been the focus of active research and several security definitions and constructions have been proposed. In this paper, it begins by reviewing existing notions of security and propose new and stronger security definitions. It then present two constructions that we show secure under our new definitions. Interestingly, in addition to satisfying stronger security guarantees, our constructions are more efficient than all previous constructions. Further, prior work on SSE only considered the setting where only the owner of the data is capable of submitting search queries. It consider the natural extension where an arbitrary group of parties other than the owner can submit search queries. It formally define SSE in this multi-user setting, and present an efficient construction

V. COMPARISON OF VARIOUS METHODS

Considering the case of data stored under the cloud storage, the serious issues like confidentiality, integrity and access control should be checked, whether they are meet or not. There are plenty of access control schemes like Attribute Based Encryption (ABE), Key Policy-Attribute Based Encryption (KP-ABE), Ciphertext PolicyAttribute Based Encryption(CP-ABE) and Key-Aggregate Searchable Encryption (KASE). Let us compare and analyze these access control schemes in detail. Comparison of all access

control schemes is as follows in Table 1. Sahai and Waters who introduced the Attribute Based Encryption (ABE) scheme, it is a public key based encryption which is giving more security and better access control. The main specialty of this scheme, it provides the encryption and decryption by means of their user attributes. Generation of ciphertext and secret keys depends on their attributes. If the attribute of secret key is different from the attribute of ciphertext, than decryption process is not possible. Considering the value of threshold as t. If atleast't' numbers are matching, then performing decryption. Advantage of ABE is it has very complex access control and no need of list of users in this process, only required is the access policy. Disadvantage of ABE is that the data owner needs to use all the available set of user's public keys, so as to perform the data encryption.

TABLE I: Comparison of ABE, KP-ABE, CP-ABE and KASE

Parameters	ABE	KP-ABE	CP-ABE	KASE
Efficiency	Average	Low	Average	Average
Data confidentiality	Present	Present	Present	Present
User accountability	Absent	Absent	Present	Present
Fine grained access control	Low	Low	Average	Average
Computational overhead	High	Low	Average	Average
Collusion resistant	Average	Good	Good	Excellent
Revoke users	Absent	Present	Present	Present

Speaking about the Key Policy-Attribute Based Encryption (KPABE), it is another type of ABE. It can perform one to many communications. In this KP-ABE scheme, each private key will be linked with an access tree structure. This type of access tree structure will explain the type of ciphertext which can be decrypted by using the key. Here, the ciphertext is represented with the set of attributes and the key is represented with the access structure, this scheme is called as KP-ABE. This scheme gives a fine grained access control and it can also provide better flexibility than ABE. Problem with KP-ABE is that who can decrypt the encrypted data decision can't be taken by the data owner. The Ciphertext Policy Attribute Based Encryption (CP-ABE) runs in reverse order of KP-ABE. This will eliminate the main disadvantage of KP-ABE. In CP-ABE, the data owner will decide the policy about who can perform decryption on the encrypted data. Disadvantage of CP-ABE is how to manage the attributes of data users and their respective access policy. The Key-Aggregate Searchable Encryption (KASE), it is a public key encryption scheme which is adapted from key-aggregate cryptosystem scheme and Multi-key searchable encryption scheme. Advantage is that in place of sharing the documents, data owner send the single aggregate key. By using this key, he/she can access all the documents will is meant for him/her. It eliminates the main disadvantage of KP-ABE, CP-ABE. In KASE, the data owner will generate a single aggregate key and transmit it to the user. Data user can submit generated single aggregate trapdoors to the cloud server. Cloud server than perform adjust and test algorithms to retrieve the relevant documents shared with him/her.

A. Performance Evaluation

Considering the studies of various cryptographic operations based on pairing computation. Which can be efficiently executed and be tested on both computers (Intel(R) Core(TM)i5-3337U CPU @ 1.80GHZ with OS as Windows7) and mobile devices(Samsung G3502U phone) is shown as under in Table-2.

TABLE II: Pairing Based Computation Execution Times

Tested on	Pairing	pow(in $\mathcal{G}$)	pow(in $\mathcal{G}_1$)	pow(in Z_p)
Samsung G3502U	485	243	74	0.8
Computer	10.2	13.3	1.7	0.05

Implementation of this system is done, by means of two libraries: jpbc (for mobile phones) and pbc library (for computer). In case of mobile devices, it takes about 5 seconds for pairing computations. But the sensor nodes and Personal Digital Assistant (PDA) requires only 1.5 and 0.5 seconds respectively. The above depicts the average time required by mobile device and computer for performing pairing based computations. Computers have faster average time for pairing as compared to mobile devices.

B. Kase Algorithm Evaluation

Considering all the steps (Setup, Keygen, Encrypt, Extract, Trapdoor, Adjust, and Test) which were present in KASE scheme and this scheme is evaluated on both mobile devices and computers.

KASE Setup: Generally setup algorithm requires a linear execution time against the maximum number of documents which were belonging to a particular data owner. When the maximum number of documents reaches a value of 20000, the KASE Setup algorithms requires 259 seconds (computers).

KASE Encrypt: Execution time of this is also linear against the number of keywords generated. Considering the case when the number of keywords reaches a value of 10000, the KASE Encrypt algorithms require 206 seconds in computers, whereas in mobile devices it takes 10018 seconds. By above values, two conclusions can be made: not to use mobile devices for uploading the documents associated with large number of keywords, keyword based searching can be executed more quickly in computers with the help of pairing based computation.

KASE Extract: Execution time against the number of shared documents is also linear. When the number of keywords reaches a value of 10000, the KASE Extract algorithms require 132 seconds in computers, whereas in mobile devices it takes 2430 seconds. Considering the above values, it is not suggested to use mobile devices for this stage. Since, the KASE Extract runs along with the KASE Encrypt algorithm.

KASE Trapdoor: Execution time is a constant value for both the mobile devices and computers. Considering the values such as 0.01 seconds in computers, whereas in mobile devices it takes 0.25 seconds. Considering the above

values, keyword searching can be done more efficiently in both mobile devices and computers. Also comparing with other available schemes, 5 International Journal of Computer Applications (0975 - 8887) Volume 139 - No.2, April 2016 KASE scheme is having substantial improvements in trapdoor generation.

KASE Adjust: It also provides a linear relation, when plotted execution time against the number of documents available to perform adjusting operation. It can be improved in practical applications more efficiently.

KASE Test: Execution time cost against the number of keyword ciphertexts is also linear. Considering the execution of KASE Test algorithm is twice the execution of pairing based computations. When the number of keyword ciphertexts grows to a value of 20000, computers takes 467 seconds for execution.

V. RESULTS

Results of this paper is as shown in bellow Figs.1 to 5.

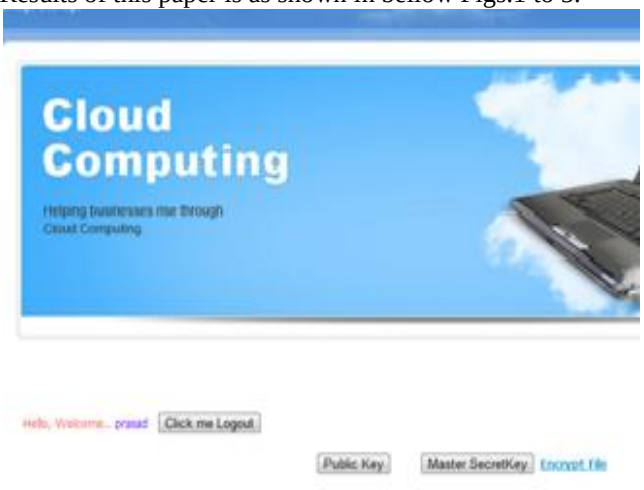

Fig.1.Key-Generation.

Fig.2. Data Encryption.

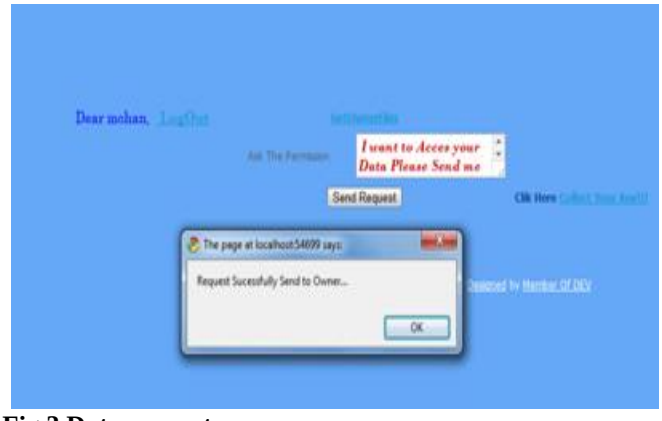

Fig.3.Data request.

Fig.4.Collecting Aggregate Key.

Fig.5.Decrypting the Data.

VI. CONCLUSION

Considering the practical problem of privacy preserving data sharing system based on public cloud storage which requires a data owner to distribute a large number of keys to users to enable them to access his/her documents, we for the first time propose the concept of key-aggregate searchable encryption (KASE) and construct a concrete KASE scheme. Both analysis and evaluation results confirm that our work can provide an effective solution to building practical data sharing system based on public cloud storage. In a KASE scheme, the owner only needs to distribute a single key to a user when sharing lots of documents with the user, and the user only needs to submit a single trapdoor when he queries over all documents shared by the same owner. However, if a user wants to query over documents shared by multiple owners, he must generate multiple trapdoors to the cloud. How to reduce the number of trapdoors under multi-owners setting is a future work. Moreover, federated clouds have attracted a lot of attention nowadays, but our KASE cannot be applied in this case directly. It is also a future work to provide the solution for KASE in the case of federated clouds.

VII. REFERENCES

[1] 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.

[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.