Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-4, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

Survey on Key-Aggregate Searchable Encryption

Arathy S V
Department Computer Science, Rajiv Gandhi Institute of Technology, Kottayam

***Abstract**: In key-Aggregate Searchable Encryption data owner only needs to distribute a single aggregate key to a user for sharing any number of files and also user only needs to submit a single aggregate trapdoor to the cloud for performing keyword search over any number of shared files. In this survey describe different searchable encryption technique used for sharing group of documents to user.*

***Keyword** : KASE, Searchable Encryption, MUSE, MKSE.*

1. Introduction

Today, millions of users are sharing personal data with their friends through social network applications based on cloud storage . There is data leakage problem while using cloud storage . To overcome this problem the data owner encrypt the data before uploading to the cloud by using a key. Share this encryption key to the user through secure channel for decryption. Such a cloud storage is called the cryptographic cloud storage . The encryption of data makes it challenging for users to search and then selectively retrieve only the data containing 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. For retrieving data matching a keyword, the user will send the corresponding keyword trapdoor to the cloud for performing search over the encrypted data.

Searchable symmetric encryption (SSE) and public key encryption with keyword search (PEKS) used for the construction of searchable encryption (SE) for sharing data in single owner and performed secure search on remote server by submitting search query on keyword provided . There are different searchable Encryption technique are used.

Multi-user searchable Encryption(MUSE) scheme used single key with access control. The data owner shares documents with group of users and users who has access right can receive them by submitting trapdoor for keyword search on shared contents. It advances the single user SSC and PEKS schemes.

Another searchable encryption technique is Multi-key searchable encryption(MKSE) . In Multi-key searchable encryption a user provide a single search token to the server and allows the server to search for that token's word in documents encrypted with different keys. This reduces the number of trapdoors used for each document separately. MKSE provides a general approach to perform keyword search over a group of documents with only one trapdoor.

Another encryption technique is Key Aggregate Encryption for Data Sharing. It reduce the number of distributed data encryption scheme. Key-aggregate encryption scheme for data sharing generate an aggregate key for the user to decrypt all the documents. In this technique set of documents encrypted by different keys to be decrypted with a single aggregate key.

2. Searchable Encryption Technique

Here I discuss four methods used for Searchable Encryption. They are Multi-user Searchable Encryption(MUSE), Multi-key Searchable Encryption (MKSE), Key-Aggregate Encryption for Data Sharing and Key-Aggregate Searchable Encryption(KASE).

2.1. Multi-user Searchable Encryption (MUSE):

In Multi-User Searchable Encryption 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. Broadcast encryption is used to achieve coarse-grained access control. A attribute based encryption (ABE) is applied to achieve fine-grained access control aware keyword search. The main problem in MUSE is how to control which users can access which documents.

2.2. Multi-Key Searchable Encryption

Popa and Zeldovich [3] firstly introduces the concept of multi-key searchable encryption. In Multi-key searchable encryption a client to provide a single search token to the server and allows the server to search for that token's word in documents encrypted with different keys. Consider the key of user i with uk_i, and the key of document j with k_j . Consider that a user, say Alice, (with key uk_A) has n encrypted documents at the server, and each is encrypted under a key k_j for j = 1, . . . , n. Alice wants to search for a word w over all the documents

she has access to, so she uses uk_A to compute a token for a word w. In order to allow the server to match the token against words encrypted with $k_1, \ldots, k_n$, Alice gives the server some public information called delta. Alice provides one delta per key k_j , denoted $\Delta_{ukA,kj}$. The server can use $\Delta_{ukA,kj}$ to convert a search token under key uk $_A$ to a search token under k_j , a process called adjust. In this way, the server can obtain tokens for word w under $k_1, \ldots, k_n$ while only receiving one token from Alice, and then performing a traditional single-key search with the new tokens. Multi-key search provides efficiency guarantees over single-key search.

A multi-key search scheme MK is a tuple of algorithms *(MK.Setup, MK.KeyGen, MK.Delta, MK.Token, MK.Enc, MK.Adjust, MK.Match)* as follows:

- *params* ← *MK.Setup*(1^κ): Takes as input the security parameter and outputs system wide parameters.
- *K*←*MK.KeyGen(params)* : Takes as input the system parameters and outputs a secret key, which could be a key for a user or for a document.
- $\Delta \leftarrow MK.Delta(k_1, k_2)$: Takes as input two keys and outputs a delta.
- *tk* ← *MK.Token(k,w)* : Takes as input a key k and a word w and outputs a search token tk.
- *c* ← *MK.Enc(k,w)* : Takes as input a key *k* and a word w and outputs an encryption of the word *c*.
- *stk* ← *MK.Adjust(tk, Δ)* : Takes as input a token *tk* and a delta Δ and outputs a search token *tk*.
- *b* ← *MK.Match(stk, c)* : Takes as input a search token *stk* and a ciphertext *c* and outputs a bit *b*.

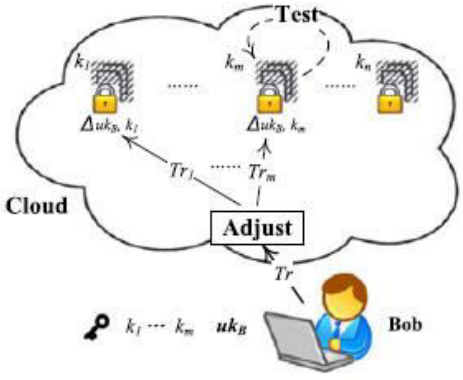

Figure 1 Multi-Key Searchable Encryption

2.3. Key-Aggregate Encryption for Data Sharing

To share several documents with different encryption keys with the same user, the data owner will need to distribute all such keys to him/her in a traditional approach which is usually impractical. Key-aggregate encryption scheme for data sharing is proposed to generate an aggregate key for the user to decrypt all the documents. To allow a set of documents encrypted by different keys to be decrypted with a single aggregate key, user could encrypt a message not only under a public-key, but also under the identifier of each document. The construction is inspired by the broadcast encryption scheme. Here the data owner can be regarded as the broadcaster, who has public key *pk* and master-secret key *msk*, each document with identifier *i* can be regarded as a receiver listening to the broadcast channel, and a public information used in decryption is designed to be relevant to both the owner's *msk* and the encryption key, the message encryption process is similar to data encryption using symmetric encryption in BE, but the key aggregation and data decryption can be simply regarded as the further mathematical transformation of *BE.Encrypt* algorithm and *BE.Decrypt* algorithm .The Key Aggregate encryption for data sharing allows efficiently delegating the decryption rights to other users. This scheme does not support any search over the encrypted data. For privacy-preserving data sharing, keyword search is a necessary. Key-Aggregate Searchable encryption support keyword ciphertext encryption, trapdoor generation and keyword matching.

2.4. Key-Aggregate Searchable Encryption (KASE)

In traditional approach for sharing encrypted data with different users different encryption key to be used for different files. Key to be distributed to the users for perform the search over the encrypted document and decryption proportional to the number of files. Also large number of trapdoor must be generated for perform keyword search over the document. In Key-Aggregate Searchable Encryption the data owner only need to share single aggregate key for share any number of files and the user only need to submit single aggregate trapdoor to perform key word search over the shared document. KASE Scheme composed of seven algorithms. First one is *setup,* this algorithm used for setup the system. This algorithm take security parameter as input and outputs public system parameter *param*. Second one is *Keygen* algorithm , it generate public/master-secret key pair for each data owner. Keywords of each document can be encrypted via the *Encrypt* algorithm with the unique searchable encryption key. Fourth algorithm is *Extract* . This algorithm is run by the data owner to generate 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 *msk* and a set *S* which contains the indices of documents, then outputs the aggregate key. Fifth algorithm is Tradoor . This algorithm takes input as aggregate searchable encryption key and keyword then outputs only one trapdoor. This algorithm is run by the user who has the aggregate key. Sixth algorithm is *Adjust*. This algorithm is run by cloud server to adjust the aggregate trapdoor to generate the right trapdoor for each different 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 outputs each trapdoor Tri for the i^{th} target document in *S*. Last and final algorithm is *Test*. This algorithm is run by the cloud server to perform keyword search over an encrypted document. It takes as input the trapdoor Tr_i and the document index *i*, then outputs *true* or *false* to denote whether the document doc_i contains the keyword *w*.

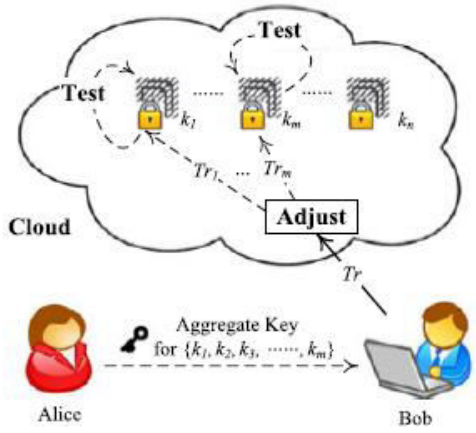

Figure 2 Key-Aggregate Searchable Encryption

3. Conclusion

In this paper discussed several methods for searchable encryption. Four methods are Multi-User Searchable Encryption (MUSE), Multi-Key Searchable Encryption (MKSE), Key-Aggregate Encryption for data sharing and Key-Aggregate Searchable Encryption (KASE). These each technique have advantage and disadvantage.

4. References

[1] Baojiang Cui, Zheli Liu, and LingyuWang, "Key-Aggregate Searchable Encryption (KASE) for Group Data Sharing via Cloud Storage ," ieee transactions on computers, vol. 65, no. 8, august 2016.

[2] X. Liu, Y. Zhang, B. Wang, and J. Yan, "Mona: Secure multi-owner data sharing for dynamic groups in the cloud," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 6, pp. 1182–1191, Jun. 2013.

[3] R. A. Popa and N. Zeldovich, "Multi-key searchable encryption," Cryptol. ePrint Archive, Rep. 2013/508, 2013.

[4] X. Liu, Y. Zhang, B. Wang, and J. Yan, "Mona: Secure multi-owner data sharing for dynamic groups in the cloud," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 6, pp. 1182–1191, Jun. 2013.

[5] C. K. Chu, S. Chow, W. G. Tzeng, J. Y. Zhou, and R. Deng, "Key-aggregate cryptosystem for scalable data sharing in cloud storage," IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 2, pp. 468–477, Feb. 2014.

[6] Fangming Zha, Takashi Nishide, Kouichi Sakurai, "Multi-User Keyword Search Scheme for Secure Data Sharing with Fine-Grained Access Control" Computer Security Symposium 2011 19-21 October 2011

[7] C. Dong, G. Russello, and N. Dulay, "Shared and searchable encrypted data for untrusted servers," J. Comput. Security, vol. 19, pp. 367–397, 2011

[8] F. Zhao, T. Nishide, and K. Sakurai, "Multi-user keyword search scheme for secure data sharing with fine-grained access control," in Proc. Int. Conf. Inf. Security Cryptol., 2012, pp. 406–418.

[9] J. W. Li, J. Li, X. F. Chen, C. F. Jia, and Z. L. Liu, "Efficient keyword search over encrypted data with fine-grained access control in hybrid cloud," in Proc. 6th Int. Conf. Netw. Syst. Security, 2012, pp. 490–502.