ABSTRACT:

The capability of selectively sharing encrypted data with different users via public cloud storage may greatly ease security concerns over inadvertent data leaks in the cloud. However, this also implies the necessity of securely distributing to users a large number of keys for both encryption and search, and those users will have to securely store the received keys, and submit an equally large number of keyword trapdoors to the cloud in order to perform search over the shared data. The implied need for secure communication, storage, and complexity clearly renders the approach impractical. In this paper, we address this practical problem, which is largely neglected in the literature, by proposing the novel concept of key aggregate searchable encryption (KASE) and instantiating the concept through a concrete KASE scheme, in which a data owner only needs to distribute a single key to a user for sharing a large number of documents, and the user only needs to submit a single trapdoor to the cloud for querying the shared documents. The security analysis and performance evaluation both confirm that our proposed schemes are provably secure and practically efficient.

Keywords: data sharing, cloud storage

[1] INTRODUCTION

Cloud storage has emerged as a promising solution for providing ubiquitous, convenient, and on demand accesses to large amounts of data shared over the Internet. Today, millions of users are sharing personal data, such as photos and videos, with their friends through social network applications based on cloud storage on a daily basis. Business users are also being attracted by cloud storage due to its numerous benefits, including lower cost, greater agility, and better resource utilization. However, while enjoying the convenience of sharing data via cloud storage, users are also increasingly concerned about inadvertent data leaks in the cloud. Such data leaks, caused by a malicious adversary or misbehaving cloud operator,
can usually lead to serious breaches of personal privacy or business secrets (e.g., the recent high profile incident of celebrity photos being leaked in iCloud).

To address users’ concerns over potential data leaks in cloud storage, a common approach is for the data owner to encrypt all the data before uploading them to the cloud, such that later the encrypted data may be retrieved and decrypted by those who have the decryption keys. Such cloud storage is often called the cryptographic cloud storage. However, 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, such that, for retrieving data matching a keyword, the user will send the corresponding keyword trapdoor to the cloud for performing search over the encrypted data. Although combining a searchable encryption scheme with cryptographic cloud storage can achieve the basic security requirements of a cloud storage, implementing such a system for large scale applications involving millions of users and billions of files may still be hindered by practical issues involving the efficient management of encryption keys, which, to the best of our knowledge, are largely ignored in the literature.

[2] RELATED WORK

Data sharing systems based on cloud storage has attracted much attention recently. In particular, Chu et al. consider how to reduce the number of distributed data encryption keys. 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 on a traditional approach which is usually impractical. Aiming at this challenge, a key-aggregate Encryption (KAE) 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.

There is a rich literature on searchable Encryption, including SSE schemes Andes schemes inside contrast to those Existing work, in the context of cloud storage, keyword search under the thematic tenancy setting is a more common scenario. In such a scenario, the data owner would like to share document with a group of authorized users, and each user who has the access right can provide trapdoor to perform the keyword search over the shared document, namely “multiuser searchable encryption” (MUSE) scenario.

[3] PROBLEM STATEMENT
Sharing data via cloud storage, users are also increasingly concerned about inadvertent data leaks and malicious attackers in the cloud. Such data leaks, caused by a malicious adversary or a misbehaving cloud operator, can usually lead to serious breaches of personal privacy or business secrets.

[4] SYSTEM ARCHITECTURE
The KASE framework is composed of seven algorithms. Specifically, to set up the scheme, the cloud server would generate public parameters of the system through the Setup algorithm, and these public parameters can be reused by different data owners to share their files. For each data owner, he/she should produce a public/master-secret key pair through the KeygenAlgorithm. Keywords of each document can be encrypted via the Encrypt algorithm with the unique searchable encryption key. Then, the data owner can use the master-secret key to generate an aggregate searchable encryption key for a group of selected documents via the Extract algorithm. The aggregate key can be distributed securely (e.g., via secure e-mails or secure devices) to authorized users who need to access those documents. After that, as shown in Fig.2, an authorized user can produce a keyword trapdoor via the Trapdoor algorithm using this aggregate key, and submit the trapdoor to the cloud.

After receiving the trapdoor, to perform the keyword search over the specified set of documents, the cloud server will run the Adjust algorithm to generate the right trapdoor for each document, and then run the Test algorithm to test whether the document contains the keyword.
[5] MODULE DESCRIPTION

1. KEY GENERATION

In this module admin going to generate two keys for encryption and decryption process. By using Asymmetric algorithm, admin going to generate master secret key and public key.

2. ACCESS CONTROL

In this module admin going to give access control for the files he will going to upload, while uploading admin going to encrypt the file with the help of master secret key for the security purpose to the cloud.

3. KEYWORD INDEXING

Remove un-necessary words from the file and Find the keywords. Calculate the Content Weight age of keywords Convert the Keywords into hash code by using MD5 algorithm; place the hash code in Index Array.

4. SEND AGGREGATE KEY

Based on the categories selected by admin, system has to fetch the corresponding hash keys + fetch the Public Key. Generate the User Aggregate Key and finally send it to users.

5. SEARCH WITH KEYWORD

User has to select the aggregate Key then after that Input the search keyword. Convert the keyword into hash code. Decrypt the aggregate Key, Separate and get hash keys and separate and get public Key. Using Hash Key and keyword generate hash codes (Trapdoor). Send the Hash codes to server, based on the Hash codes received server has to check the keyword index and if any matching files are available, list all the file names to the user.
(Adjust & Test) View the shortlisted files from server, download the files and finally decrypt the file with owner public key.

[6] 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 overall 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.

REFERENCES

Keyword Trapdoor Search Over Shared Data in Cloud Storage

Sahana M.P  
Received BE. In vtu, Belgaum In 2014.  
M.Tech vtu, Belgaum. City Engineering  
College Bangalore2014-2016.  

Nandish A.C  
Associate Professor in the city  
Engineering college Bangalore.  

Sahana M.P d/o  
Papanna c  
#108 mangala mandya t, d.  
Pin code 571403  
Ph  

8722014843
Sahana0393@gmail.com