Imperial Journal of Interdisciplinary Research (IJIR)
Vol-2, Issue-7, 2016
ISSN: 2454-1362, http://www.onlinejournal.in

File Sharing in Public Cloud Using Aggregate Key Encryption

Syed Ahmed Mohiuddin Peerzade[1] & Aparna. R[2]
[1]Student, CSE department, MSRIT
[2]Assistant Professor, Department of CSE, MSRIT.

Abstract*: As Cloud computing gets to be predominant, more sensitive information is brought in the public cloud like users personal records, individual wellbeing information, government archives, so on. By putting away their information in the public cloud, Information proprietors are alleviated of having the pain of storing information and to maintain such that the users can be happy with storage of their personal data in a very good quality and they in turn can access whenever they need it.*

In any case, if the cloud users and the servers present in cloud are not present in the same local domain and if they keep the information which will become risky, which means that the servers present in cloud could never be trusted fully. Afterwards it takes delicate information for the most part ought to be encrypted preceding outsourcing for information protection and privacy risks in the projects. Today's mail servers, for example, IMAP servers, document servers and other information storage servers normally should be completely trusted they have admittance to the information, and consequently should be trusted not to uncover it without approval which presents undesirable security and protection dangers in applications. Past work demonstrates to fabricate encoded record frameworks and secure mail servers, yet commonly one must yield usefulness to guarantee security. The crucial issue is that moving the computation to the information storage appears to be extremely troublesome when the information is distributed, and numerous calculation issues over encoded information beforehand had no pragmatic arrangements.

1. Introduction

Information sharing is a critical usefulness in distributed storage. For instance, bloggers can allow their fellow mates to have access to their personal information not completely but some subset of their personal information, an undertaking allows representatives who can access some part of the sensitive information[5]. It is obvious that the cloud users can download the information which is encrypted with some limit, decrypt it and afterwards they can send it to others which in turn other people can share with others, but chances are there that it might lose some estimation of storage of cloud in a distributed format. The Clients who gets the information which is sensitive from others can in turn has the access privileges to share with others by keeping in mind that they can later access the data from the cloud server. So it's clear that sharing the information with others which is in fact partial in the cloud servers is not so important.

Storage of data in cloud is becoming popular day by day. In major business which are getting outsourced it's been seen the popularity of the storage of the information, which will be helpful in making decisions for the business information. And we can for sure say that this field has become innovation for major of the online small applications which will be helped for individuals. These days for everything but difficult in applying with the expectation of complimentary records for email, images records, having the estimation of capacity of something around 30 GB (or a couple of dollars for more than 1 TB)[1,2].

Computation in cloud is a promising processing worldview which as of late has drawn broad consideration from both the educated community and industry. By joining an arrangement of existing and new methods from examination zones, for example, Service-Oriented Architectures (SOA) furthermore, virtualization, conveyed processing is seen all things considered a registering worldview in which assets in the processing base are given as administrations over the Internet. A standout amongst the most basic administrations offered by cloud suppliers is information stockpiling[3].

Notwithstanding, encryption of data or information makes viable data to use exceptionally troublesome undertaking, having said that there can be a considerable measure of making the documents to be outsourced. Furthermore, figuring in Cloud, data proprietors can grant the information with many of the other users[4]. The individual cloud users can in turn access the some of the records i.e. it can be recoverable. A standout amongst the most prominent courses is to specifically recover documents through

search phrase based pursuit as opposed to recovering all the encoded records back which is totally unrealistic in distributed computing situations. Such type of key phrase based pursuit allows the cloud users to recover their individual data of their interest and it has been generally that is the data can be seek which is present in the plain text format, for example, search performed in Google. Lamentably, information encryption limits client's capacity to perform keyword inquiry and in this manner makes the customary plaintext scan strategies inadmissible for Cloud Computing.

2. Aggregate Key Encryption Framework

We are describing the problems with other framework and then we will define our approach of Aggregate key.

2.1. Problem Statement

A canonical application of KAC is sharing of the information. The concept of aggregation of key property will be specifically useful when it's expected that the delegation to be efficient. This enables a content provider for sharing the confidential data in a selective way, having a fixed cipher text expansion, by giving to each authorized user a single and small aggregate key[7]. The working of this algorithm is illustrated in the figure below.

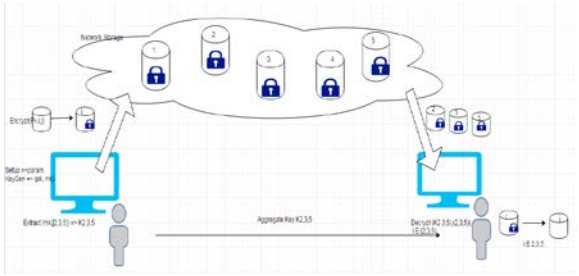

Fig. 1. Sharing encrypted data using single key encryption

The disadvantage of this type are, although the parameter can be downloaded having encrypted data, its better if the size of it is independent of all encrypted classes of text. Later, when the user has the appropriate keys in any of the personal device not utilizing any of the trusted H/W, they aggregate key can be leaked which is against the cryptosystem.

2.2. Motivation

To start with, identity protection is a standout amongst the most noteworthy deterrents sending the computation in distributed format. Without having the assurance of getting the individual security, clients might not want to be participating in public cloud in light of the fact that their genuine personalities could be effortlessly uncovered to cloud suppliers and aggressors. On the other hand, individuals unusual privacy might occur which denies the privacy. For instance, few of the cloud users can trouble other cloud users of the same organization by sharing the dummy or unrecognized records and to whom we cannot trace. Consequently, tracing these type of individuals, which empowers gathering administrator getting uncover genuine personality of the cloud user or a client, it is likewise profoundly alluring.

Second, it is very suggested that any person in a group should have the ability to totally appreciate the information securing and sharing organizations gave by the cloud, which is characterized as the numerous proprietor way safely look over distributed information, searchable encryption strategies have been produced as of late. Searchable encryption plots as a rule to develop a record for each of the key word which we are interested in and associating with the list along with the information which has the key phrase associated with the document[6]. Having incorporated the key phrases and trapdoors within a file, compelling catchphrase hunt can be acknowledged while both record content and key phrase protection are very much saved. Despite the fact that taking into consideration performing search is safely and viably, the current searchable encryption systems sometimes fall short for distributed processing circumstance since they reinforce simply exact keyword look.

2.3. Objectives

The objective of this project is to design a system which helps users to share their documents in a most secure way in public cloud having the below aspects

To introduce the concept of sharing of aggregate key among users to access the documents in public cloud.

To reduce the number of keys that gets shared between owner and other users.

To keep the contents of the document in a secure most way in public cloud.

To give the users better operational tool to effectively solve the problem.

To build all the capabilities in an economical way using open source tools.

To make the user easier to share any document from anywhere having connectivity to internet.

2.4. Description of Framework

The following seven algorithms will be implemented [1,5].

Setup(): This setup step gets run by the administrator of the cloud which sets up the basic parameters. The parameter of the security m and the number of documents to be as of which some place has with an information proprietor, it yields public param also.

Key_generation(): The owner of the data will run this method which generates a pair of primary key and master secret key.

Encryption (primary_key, j): This calculation is run by the cloud user who shares the data to encrypt jth record and also it creates the key word cipher texts. For every report, this calculation will make a small variable j for making the file searchable encryption key. With contribution owner's primary key along with the file j, this calculation yields information encrypted text and keyword encrypted text, Dj.

Decrypt (msk, S): Calculation of this is controlled by the user who shares the data to create aggregate key of encryption appointing key phrase searching on some specific number of files for different clients. The input to this algorithm is owner's private key and a list T which has all the indexes of the files and it gives back the aggregate key kagg.

Trapdoor generation (kagg, w): Calculation of this is done by the user of the public cloud, the user who has the permission to download the file and he does the keyword searching over the cloud. The input to this is the aggregate key and the key phrase w which yields trapdoor Qr.

Adjust (attr, j, T, TD): Calculation of this is done by the cloud which confirm the aggregate trapdoor produces unique trapdoor for all the documents shared by the user in the public cloud. The input to this algorithm is attributes generated by the setup algorithm and also the archives records, which is the list of j also the trapdoor, which in turn gives back the trapdoor for each index j in the list.

Testing (TD, j): Calculation of this is done by the cloud server which performs key phrase match on the encrypted file. The input to this is the trapdoor TD and the index j from the list T, on successful match it gives Boolean True or False.

Below framework should also be implemented

System setup: whenever the company requests to setup their personal public cloud in which users can share the data, the setup is done to install database having all the required tables, and we allocate a UUID for that particular company. And even an administrator has to be set to grant permissions for genuine users. Administrator will be the main controller [6]. The framework parameters attr, admin runs the setting up algorithm and the tables gets generated.

Registration of user. Whenever a new user wants to join the circle, it creates an account and the administrator should grant this new user to share and get the data from the public cloud. The new user has to give the email id, password and an email is sent to the registered email id for any updates.

Login of users in Public cloud. User has to login to use the service, the user has to give the appropriate username password to log into his account[8].

Sharing Data. Whenever a user wants to share the data in the public cloud then he encrypt the data or file and the encrypted file and the given keyword cipher text, gets transferred to the cloud database. The cloud administrator has the document id of the file and the cloud stores the file or data in the encrypted format.

Sharing of Data. Whenever the user wants to share the data with a group of other cloud users then it runs the Extract method which in turn produces aggregate key, and this key is stored in the database of the owner [4]. And the single aggregate which gets generated using both the encrypted text and the key phrase is sent to the user. An automated mail is sent to the user with whom the data gets shared.

Search based on Keyword. To download the data the user has the aggregate key and the searchable keyword which will be used to generate the trapdoor and a single trapdoor is sent to the public cloud and this trapdoor gets adjusted by the cloud admin and the appropriate file is matched based on the index based on the keyword provided by the user. After matching the appropriate file the public cloud server gives back the file in the encrypted format.

Retrieval of Data. The user gets the file in the encrypted format and now he runs the Decrypt algorithm to decrypt all the file contents utilizing the aggregate key sent by the user who had shared the file.

2.5. Implementation

Below is the description how it will be implemented.

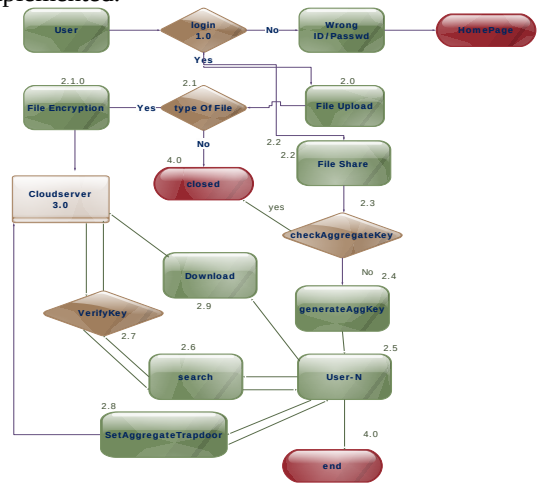

Fig. 2 Flow chart

Interaction between user and cloud server, user is requesting to cloud server, and Response to user via cloud server[10].

User should be create user account with the help of username, email-id and mobile number before login, once user account has created. , cloud server will generate UID, password for every public cloud user, after that cloud user can login with user ID and password.

User has created the user account, initially user account will be deactivated we should make active using ADMIN user.

User is selecting the file to upload in the cloud server. During this stage first user has to entered keyword which is used as a trapdoor. It will create public key, private key automatically then finally going for the encryption of the file.

Once the file is encrypted then user need to generate Cipher text of the file contents. Once Aggregate key is generated then file will be shared to the other user. User will receive the file information along with aggregate key along via mail[4,5]. User need to check the mail for the further information.

Below is the Key Generation Algorithm

1. Whenever this algorithm is get called, in turn it calles the method generateKeys() with the parameter file content in bytes.

String keys = generateKeys(fileContents);

2. In step 2, keyGen method is called and the same file content values are passed
3. Calculate the length of the array

Count = (int)fileContent.length

4. Calculating initial index position of the array

int initialIndex = new Random().nextInt(count-10)

5. Calculating last index position of the array

Int lastIndex = initialIndex +3

6. Copying the array with initialIndex and lastIndex range

Byte[] documentsKey = Arrays.copy(content,(initialIndex),(lastIndex))

7. Transforming array to string and return it.

generatedKey = new String (KeyValueFromDocument)

Return generatedKey

3. CONCLUSION

By looking at the problem of maintaining the information as private when it comes sharing in distributed or public cloud environment which requires that the public cloud user or owner has to share the multiple keys to the group of people with whom he'll be sharing the document has to share the multiple keys, each one for a single file. So we'll be the pioneer in maintaining only one aggregate if the owner of the file wants to share numerous files in the public cloud.

Performance of the public cloud server depends on the application developed, we are utilizing caching which will increase the performance and efficiency for keyword searching technique.

From the results discussed in previous section, it is clear that all the functional and nonfunctional requirements are met. Thus the System is tested against all the Requirements. The process of implementation started from identifying the scope, defining problem statement, defining requirements, followed by design, implementation, testing and results.

4. Future Enhancement

In any case, if a cloud user needs the query over the reports which gives the data of the files which is shared by different owners, the cloud owner should generate numerous reports. The trapdoors generation should be reduced for multi owners and it comes under the future work. Support for files having images will also be added in the future. Also, federated clouds has been attracted nowadays, however our Aggregate key encryption will not be applicable for such type of situation straight forwardly.

The proposed system is only for the sharing of documents. As a whole the system can be enhanced to bring various other type of documents like images in different formats and other type of documents like pdf's and zipped files.

5. References

[1] 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.*

[2] Curtmola, Reza, et al. "Searchable symmetric encryption: improved definitions and efficient constructions." *Journal of Computer Security* 19.5 (2011): 895-934.

[3] Kamara, Seny, Charalampos Papamanthou, and Tom Roeder. "Dynamic searchable symmetric encryption." *Proceedings of the 2012 ACM conference on Computer and communications security.* ACM, 2012.

[4] Bösch, Christoph, et al. "Conjunctive wildcard search over encrypted data."*Workshop on Secure Data Management.* Springer Berlin Heidelberg, 2011.

[5] Zhao, Fangming, Takashi Nishide, and Kouichi Sakurai. "Multi-user keyword search scheme for secure

data sharing with fine-grained access control."*International Conference on Information Security and Cryptology*. Springer Berlin Heidelberg, 2011.

[6] Yu, Shucheng, et al. "Achieving secure, scalable, and fine-grained data access control in cloud computing." *Infocom, 2010 proceedings IEEE*. Ieee, 2010.

[7] Lu, Rongxing, et al. "Secure provenance: the essential of bread and butter of data forensics in cloud computing." *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*. ACM, 2010.

[8] Boneh, Dan, Ben Lynn, and Hovav Shacham. "Short signatures from the Weil pairing." *International Conference on the Theory and Application of Cryptology and Information Security*. Springer Berlin Heidelberg, 2001.

[9] Chen, Xiaofeng, et al. "Secure outsourced attribute-based signatures." *IEEE Transactions on Parallel and Distributed Systems* 25.12 (2014): 3285-3294.

[10] Li, Jingwei, et al. "Efficient keyword search over encrypted data with fine-grained access control in hybrid cloud." *International Conference on Network and System Security*. Springer Berlin Heidelberg, 2012.