International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

ANALYSIS OF SECURED GROUP DATA USING CLOUD COMPUTING TECHNIQUES

G. MAHINDAR
Assistant Professor, Department of Computer Science and Engineering, Siddhartha Institute of Technology and Sciences, Narapally, Hyderabad, Telangana, India

Abstract :
Cloud computing is giving the accessibility for the users to deploy many number of files to the cloud and can also share with many users. It always with more security and very flexible to the users. The owner have to encrypt the selected files before uploading the data and decrypt it before end users. Every system wants secure keys for storing but the number of files get increased then key management becomes more complex. Now we proposed the new system called as(KASE). In this we proposed a key aggregate for sharing the files in groups and searchable encryption. Then we create the trapdoors manually for specific files is becomes more difficult so we applied the TF-IDF technique to avoid it manually.

Keywords: searchable encryption, aggregate key , cloud computing , data sharing for group

1.Introduction:

Cloud computing systems are used to enable the data sharing capacity and provides a more benefits to the user. currently there is a push for many IT organizations to increase their data sharing efforts. in enterprise edition demand for the data outsourcing is increasing every day. It should be assist in strategic managing of the corporate data. Many online services is used this scheme as a core technology. These services are used for online applications. This scheme was very easy to apply for free accounts like mail, album for photos , sharing the file with the storage size more than 25GB of data. Currently using the wireless technology ,cloud users can access almost all of the files ,directories , folders and emails by a mobile phone from anywhere in the world.Major requirements for secure data sharing in cloud are as follows. Mainly the owner of the data can be able to identify a group of users that are to be allowed to view the particular data. Within the group any member can able to access the data at anytime, anywhere without the data owner interference. no one other than the owner of the data and group members can access the data ,including the cloud service provider(CSP). the owner of the data can be able to add new users to the group. they can also be able to revoke the access rights against any member of the group over his shared data. any other member of the group can be allowed to join new users to the group or revoke(change) the permissions. to achieve the secure data sharing in cloud we use trivial solution, the data owner is to encrypt his or her data before storing it into the cloud, and the data remain secure against the cloud provider as well as other unknown users. whenever the data owner wants to share his data to group ,he sends the key used for data encryption to each member present in the group. the members in the group can get the encrypted data from cloud and decrypt the data using the key and does not require any involvement of data owner. It is computationally inefficient and makes too much burden on the data owner by considering the user revocation problem . whenever the data owner revokes the access rights to a member of the group, that particular case the data owner has to re encrypt the data with a new key, reading the revoked members key becomes useless. whenever the data is re encrypted he should distribute the new generated key to the remaining members in the group and it seems to be inefficient computationally and creates more burden on the data owner . In case the group size is large the millions of users are present. In real world the above solution is impractical and cannot be deployed in any business, government and medical related data. the traditional technology of authentication is not enough for data privacy because any unexpected privilege will expose all the data . the best way is to encrypt all the data before uploading it to the server with users own key. the important functionality of cloud storage is data sharing because the user can share their data from anywhere and anytime to anyone. Consider a example in an organization we may grant permission to access the sensitive data to their employee. the challenging task is that how the encrypted data is to be shared. We use traditional way is the user can download the encrypted data from storage .decrypt that Particular data and send it to share with other members ,then it may lose the importance of cloud storage.

2.Body Text:

1. Information :

It provides the purpose of the feasibility study, the history of the proposed project and the methodology used for doing this process and reference material used for conducting the feasibility study in the project to check the system feasibility we need two methods ,they are brain storming and surveying. Literature survey means studying various ice papers and other related references. it has conducted to determine projects visibility. the researches of study will be used to take decision .

2.Existing Methodologies:

In upcoming years, a growing number of researchers has been taken in studying the searchable encryption and various schemes which are efficient over encrypted cloud data has been proposed. there are many technical schemes related to cloud computing are proposed by researchers. the regained sharing cryptosystem of encrypted data was introduced in [2] . this scheme is called key policy attribute based encryption [KPABE]. In this cryptosystem cipher tasks are taken by set of attributes and private keys.These are associated with access structures that control that which cipher texts are able to decrypt [2].multi- identity single key decryption without random oracles, can produce multi single key decryption [MISKD].it is purely an identity based encryption[IBE] system in this a private decryption key can map multiple public keys (identities).exactly in MISKD, a single private key can be used to decrypt multiple cipher texts and encrypted with different public keys associated to private key[3].efficient and dynamic key management for access hierarchies. it has the following properties.

1.For a node we need only has functions to derive a descendants key from its own key.

2.The space complexity is same for both public information as well as for storing the hierarchy.

3.The private information of a class can consist of a single key associated with that class. Updates can be handled locally in the hierarchy , this scheme is proved as secure against illusion and key derivation by a node of its descendant key is bounded by the number of bit operations line as to take the length of the path between nodes .

4.This dynamic scheme can achieve a worst average case number of bit operation for key derivation that is exponentially better than the depth of balanced hierarchy.

3.problem definition:

The cloud computing has given the users to accessibility and deploy many number of files to the centralized cloud and share those particular files with number of users. the cloud computing is very flexible for the security concerns. the data owner need to encrypt the files before uploading it and must decrypt before end user. this cloud system needs secure storage of keys but as number of files increased then key management becomes very complex. then we proposed the system called as [KASE]. then it proposed a new key aggregation for file sharing in groups and searchable encryption . Then we observed to create trapdoors manually for some files it becomes very difficult and then we applied the TF-IDF or Cosine similarity technique to avoid the manual process.

4.Methodology/ Approach :

In this we describe the general problem and then define a generic frame work for key aggregate searchable encryption (KASE) and to provide specific requirements for designing a valid KASE scheme. Its aim is to propose a novel approach of key aggregate searchable encryption that satisfies several security and functional requirements. we build a key aggregate system can be securely share with the groups of users .After that we apply attribute based broadcast encryption system for file to be encrypted before uploading to the cloud .we should also apply auto keyword extraction technique mainly TF-IDF to create a trapdoor for searching the file .trapdoors and encrypted cipher texts will be uploaded to the cloud access control technique will be applied to give access only for the authorized user .we can perform the data sharing with clouds using advanced frame work of KASE algorithm. It is composed with seven steps of algorithm for security purpose setup parameters ,key generation ,encryption, key extraction, trapdoor generation , adjustment for trapdoor and trapdoor testing. For further we describe the system in detail and describe its main work flows.

System setup:

If an organization submits a request to the cloud then it will create a data base containing the above four tables and assigns a groupID for the organization and to insert a record into the table company .It also assigns and administrator account for the manager then group data sharing system will work under the manager control .Generating the system parameters param , manager runs the algorithm KASE. It updates and setup the old parameter in table company.

User registration: When we add a new member then the manager has to assign memberID ,membername ,password and a key pair generated by the public key encryption (PKE) scheme for him , and then store the wanted information into the table member .A users private key should be distributed through a secure channel.

User login : The most popular data sharing products for e.g.: citric and drop box are the free clouds . our system is based on password verification for authenticating users to improve further the security ,multi factor authentication or digital signature may be used when available.

Data uploading:

To upload any document or data the owner runs a (KAE) key aggregate encryption to encrypt the data , kase and also keyword cipher texts and then upload them to the cloud. at that time cloud assigns a docID for the document and store the encrypted data in path file then insert a record into the table documents. the owner can encrypt the keys using his private key and store them into the table documents.

Data sharing:

To share a group of documents with the particular target members .the owner has to run key aggregate encryption(KAE) extract and to generate the key aggregate keys and distribute them to the target members , and then insert or updates a record into the table shared documents. the shared documents for this member are changed at that time the owner must re-extract the keys and update the old documentID set in table shared documents.

keyword search: It means we can search the data through the related keywords . to retrieve any documents with an expected keyword , and then runs KASE. We use trapdoor to generate the keyword trapdoor for documents shared by the each owner, and then submit each trapdoor and related owners identity ownerID to the cloud. after receiving the request from each trapdoor the cloud will run KASE and adjust that trapdoor for each and every document in the docID set and run KASE. Test to perform keyword search then cloud will return the documents which are encrypted and contains a expected keyword to the member.

Data retrieving :

To retrieve any data or document from the cloud ,After receiving the encrypted documents the member will run KAE. decrypt the document using the aggregate key given by the document owner.

4.1 Major constraint: A key aggregate searchable encryption (KASE) scheme should satisfy three main functional requirements as compactness, search ability and delegation .Now a day's federated cloud is being a problem that can be attracted a lot of attention.

4.2 Approach for solving the problem and efficiency issues:

Our cloud system contains four different phases. Mainly to define a general framework of key aggregate searchable encryption (KASE) can be composed of 7 polynomials algorithm i.e security parameter setup , key generation, encryption , key extraction , trapdoor generation ,adjustment for trapdoor and trapdoor testing . then we describe both functional and security requirements for design a valid KASE scheme. After that detailed constructions are to be provided for seven algorithm and efficiency analyzing of scheme and establish its security through detailed analysis .various pratical issues is to be discussed in building an actual group data sharing system based on proposed KASE scheme and performance is to be evaluated .this evaluation says that our system can meet the performance requirement of pratical application. both evaluation and analysis result conforms that our work can give an effective solution to build pratical data sharing system based on public cloud storage.

Broadcast Encryption : In the broadcast encryption (BE) scheme ,a broadcaster encrypts a message for some subset s of user those who are listening on a broadcast channel [1]. any user in subset s can use his private key to decrypt the broadcast. A broadcast encryption (BE) scheme can be described as a tuple of three polynomial time algorithm BE =(setup, encrypt, decrypt) as follows:

setup(1 ; n) :

It is to be run by the system to set up the scheme . It takes the input as a security parameter 1 and the number of receivers n , outputs n private keys [d1;....;dn] and taken as a public key pk.

Encrypt (pk; S):

This algorithm is run by the broadcaster to encrypt the particular message for a subset of users .It takes a input as public key (pk) and a subset of users S1;......;n and outputs a pair of (Hdr,k), where Hdr is called the header and K is a message encryption key and which is encapsulated in Hdr .It is often referred to Hdr as the broadcast cipher text .The concrete message will be encrypted by K and broadcasted to the users in S.

Decrypt (pk, S ,I ,di , Hdr):

This algorithm is to be run by the user to decrypt the received messages. It takes as input a public key (pk), a subset of users S1; ..; n, a user id i21;; n, the private key di for user i and a header Hdr, and outputs the message encryption key K . The key K will be used to decrypt the received messages.To ensure the system to be correct, it is required that, for all S1; ..; n and all i2S, if (pk; (d1; ..; dn) R Setup(1; n)and(Hdr;K)REncrypt(pk; S)), then Decrypt(pk; S; i; di;Hdr) = K.

Searchable Encryption:

Searchable encryption schemes is divided into two categories, i.e., and public key encryption with keyword search(PEKS) searchable symmetric encryption (SSE). Both PEKS and SSE can be taken as the tuple SE = (Setup, Encrypt, Trapdoor, and Test):

Setup (1): This algorithm is to be run by the owner to set up the scheme. It takes input as a security parameter 1, and outputs the necessary keys.

Encrypt (k;m): This algorithm is to be run by the owner to encrypt the data and generate its keyword cipher texts. It takes the input as data m, owners necessary keys are included searchable encryption key k and data encryption key, outputs keyword cipher texts (Cm) and data cipher text.

Trapdoor(k;w): This algorithm is to be run by a user to generate a trapdoor Tr for a key-word w using key k.

Test (Tr; Cm): This algorithm is run by the cloud server (cs) to perform a search for keyboard over encrypted data. It takes a input trapdoor Tr and the keyword cipher texts (Cm), outputs whether cipher text(Cm) contains the specified keyword.

TF-IDF: The Auto keywords extraction technique is to create the trapdoors for searching the file . correctly , it is required that for a message (m) containing keyword w and a searchable encryption key k, if (Tr_Trpdr(k;w)) and (Cm_Encrypt(k;m)) then Test(Tr;Cm) = true.

5. Result and Discussion:

The main purpose is to provide document for the design and implement the New Algorithm for sharing the data over cloud using key aggregate . Both high-level requirements and details of the implementation are gathered here to ensure successful completion of the project and continuity for future project development. This document is to be a detailed design supplement to the Terms of Reference for the development of applications. This document will provide detailed information for designing, implementing, and configuring application for KASE search results using TF-IDF for fitting trapdoor algorithm, but it will not include end-user documentation. The intended audience of this specification includes project managers or developers or research oriented may use or extend this project in the future. Details in the document may be helpful for technically-oriented end-users of the cloud application.

6. Conclusions :

By Considering the real problem of privacy preserving data sharing system is purely based on public cloud storage and requires a data owner is to distribute a large number of keys to users to enable them to access their documents, we firstly propose the concept of key-aggregate searchable encryption (KASE) and construct a concrete KASE scheme. Both evaluation and analysis results confirm that our work can provide an better solution to build the practical data sharing system based on the public cloud storage.

7. Future scope :

In future the extending approach is to decrease the number of trapdoors under multi owner phenomena and to provide the better solution for KASE in the case of federated clouds.

References :

- Baojiang Cui, Zheli Liu and Lingyu Wan "Key-Aggregate Searchable Encryption (KASE) for Group Data Sharing via Cloud Storage", IEEE TRANSACTIONS ON COMPUTERS VOL: PP NO: 99 YEAR 2015 .
- Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine Grained Access Control of Encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS 06) ACM, 2006, pp. 89-98 .
- F. Guo, Y. Mu, Z. Chen, and L. Xu, "Multi Identity Single-Key Decryption without Random Oracles", in Proceedings of Information Security and Cryptology (Inscrypt 07), ser. LNCS, vol. 4990. Springer, 2007, pp. 384-398 .
- M. J. Atallah, M. Blanton, N. Fazio, and K. B. Frikken, "Dynamic and Efficient Key Management for Access Hierarchies", ACM Transactions on Information and System Security (TISSEC), vol. 12, no. 3, 2009 .
- 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 .