ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

Scalable Data Sharing with Outsourced Revocation in Cloud Computing

Rachana Gangwani, Prof. H. A. Hingoliwala

M.E Student, Department of Computer Engineering, JSCOE, Handewadi Road, Hadapsar, Pune, India

H.O.D., Department of Computer Engineering, JSCOE. Handewadi Road, Hadapsar, Pune, India

ABSTRACT: Sharing of data is major functionality in cloud storage. The approach implies a special type of public key encryption in which user can encrypt their data files with the public key but also under the identifier of the group. Any group member in the group can securely share data with other group members in the un trusted cloud. The group manager holds a master-secret called secret key, which can be used to extract secret keys for different classes. The extracted key can be an aggregate key which is as compact as a secret key for a single group, but aggregates the power of many such keys, that is the decryption power for any set of classes. Efficiently, the new group members can directly decrypt data files uploaded before their participation without contacting with group manager. User revocation facility is achieved through a novel revocation list without any updation of the secret keys for the remaining group members. The size and computation overhead of encryption are constant and independent with the number of revoked users. The other data files outside the particular group remain confidential. Unfortunately, sharing of the data in a multi-owner manner while preserving identity and data privacy from an untrusted cloud is still a challenging.

KEYWORDS: Cloud computing, Revocation, Outsourcing

I. INTRODUCTION

Cloud computing has become a widely accepted paradigm for providing services over the internet. With the increasing popularity of cloud storage, the risks for security, data integration, and confidentiality of data are implicitly increasing. Therefore, the cloud provider must consider the security and confidentiality as the challenging factors for data sharing functionality. The care has to be taken to protect data, as cloud storage is storing of the data remotely which is regulated by third party. The third party takes the responsibility for keeping data accessible and available to user's all the time. In today's world, it has become easy to go for free accounts to upload or store the data, photos, files or folders with storage capacity more than 25GB.

Along with the fast growing internet, user's can access and utilize all their files and mails from any place in the world. Instead of storing the data into the hard drive, user can save the data on the cloud which makes him avail all the data accessible for him from any corner of the world using internet. But considering the privacy of data, the traditional techniques for authentication are not reliable, because the unavoidable privilege escalation will disclose the confidentiality of data. For protecting the confidentiality of the data stored in cloud storage, the care has to be taken for encrypting those data before uploading them on the cloud by using some or the other cryptographic algorithms.

Cloud storage is a cloud computing model in which data is stored on remote servers accessed from the internet, or cloud. It is maintained, operated and managed by a cloud storage service provider on a storage server that is built on the virtualization techniques. Cloud storage is also known as utility storage. Cloud storage works through a data centre virtualization, providing data owner and applications with a virtual storage architecture that is scalable according to the application requirements. Clients will hesitate to store data in cloud if it is a matter of their data security and integrity. For this reason, the Third Party Auditor (TPA) was introduced which is nothing but a software which plays an important role in auditing the integrity and privacy of the data. Many methods were proposed for data sharing with regard to security issues.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

The basic idea of this project comes from the fact that cloud is a big platform to store and to retrieve the data in huge capacity and to provide privacy to the user's data. The challenging problem is how to securely share encrypted data without increasing the cost and complexity of sharing of the secret keys. Data user's can download the encrypted data from the storage, decrypt them, then send them to others for sharing, but it degrades the value of cloud storage. Data user's should be able to delegate the access rights of the sharing data to others so that they can access these data from the server.

II. LITERATURE SURVEY

As the previous section reveals various methodologies for enabling cloud storage auditing, but still there is a huge gap to meet the perfection. So, as a step towards this, this paper tried to grab many concepts so that a new and efficient system can be proposed. The detailed studies are as follows. In [1], the author describes new public-key cryptosystems that produce constant-size cipher texts such that efficient delegation of decryption rights for any set of cipher texts is possible. The novelty is that one can aggregate any set of secret keys and make them as compact as a single key, but encompassing the power of all the keys being aggregated. In other words, the secret key holder can release a constant-size aggregate key for flexible choices of cipher text set in cloud storage, but the other encrypted files outside the set remain confidential.

In [2], Author explains the public auditability for cloud storage is of critical importance so that user's can resort to a third-party auditor (TPA) to check the integrity of outsourced data. User's should be able to just use the cloud storage as if it is local, without worrying about data integrity. Such TPA with secure cloud storage system supporting privacy preserving public auditing. To enable the TPA to perform audits for multiple user's concurrently and efficiently. The technique of public key- based homomorphic linear authenticator (HLA) which enables TPA to perform the auditing without demanding the local copy of data and thus reduces the communication and computation overhead as compared to the straightforward data auditing approaches.

In [3], the main goal is to provide secure patient-centric PHR access and efficient key management. The key idea is to divide the system into multiple security domains (namely, public domains and personal domains) according to the different user's data access requirements. The PUDs consist of user's who make access based on their professional roles, such as doctors and medical researchers. For each PSD, its user's are personally associated with a data owner (such as family members or close friends), and they make accesses to PHRs based on access rights assigned by the owner.

In [4], Identity based encryption is a scheme of shared key encryption. The shared key of a user will be set as an identity sting of the user. Secret key generator is a trusted third party who holds master secret key. In this approach a secret key is issued to each user by the trusted party. Encryption of original data takes place by considering the identity of the user and public parameter. The receiver will decrypt the cipher text by his own secret key.

In [5], Attribute Based Encryption (ABE) is a scheme developed for encrypted data sharing using cryptosystem. In this scheme an attribute will be allocated to each cipher text, and the cipher text will be decrypted by master secret key holder by extracting secret key for policy of these attributes. It also deals with collusion resistance but size of the key increases with respect to the number of attributes in the sense size of the cipher text is not constant

In [7], Proxy re-encryption allows a proxy to transform a cipher text computed under Cams public key into one that can be opened by Appys secret key. Cam might wish to temporarily forward encrypted email to her colleague Bob, without giving him her secret key. In this case, the delegator could designate a proxy to re-encrypt her incoming mail into a format that the delegate can decrypt using his own secret key. Cam could simply provide her secret key to the proxy, but this requires an unrealistic level of trust in the proxy. It presents several efficient proxy re-encryption schemes that offer security improvements over earlier approaches. The primary advantage of our schemes is that they are unidirectional and do not require delegators to reveal all of their secret key to anyone or even interact with the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

delegatee in order to allow a proxy to re-encrypt their cipher texts. In our schemes, only a limited amount of trust is placed in the proxy.

III. PROBLEM STATEMENT

To develop an efficient scheme that supports flexible delegation that decrypts any subset of cipher texts produced by the encryption scheme by a single fixed size decryption key generated by the owner of the master-secret key for the particular set of groups.

IV. PROPOSED METHODOLOGY

A. Overview

Cloud is a big platform to store and to retrieve the data in huge capacity. There is a greater possibility of leakage of the user's data. Also, due to the availability of many data access entities in cloud, there has always been a threat of data theft which happens by the delegation of the key of the file or data.

So, in order to overcome these issues, our proposed system put forwards an idea for making the system more and more secure. The system provides a facility to achieve a security on a group of user's instead of a single user. In our system, the groups are been defined that is group of user's uploading and downloading the different documents stored over the cloud. Whenever the user is uploading the documents, these files are encrypting using the 16 bit secret keys. The key is generated by the group manager which is shared over the group members. If any user wants to share their data on the group. Then the user can share the data on group and system sends that key to the all member's of group. At the receiving end all members of a group should have that key. If any of the member's from the group is missing with the key then no member from the group will have access to that file. The access policies are being established for the access over the files for each member within the group this provides privacy to the data of the user.

B. System Architecture

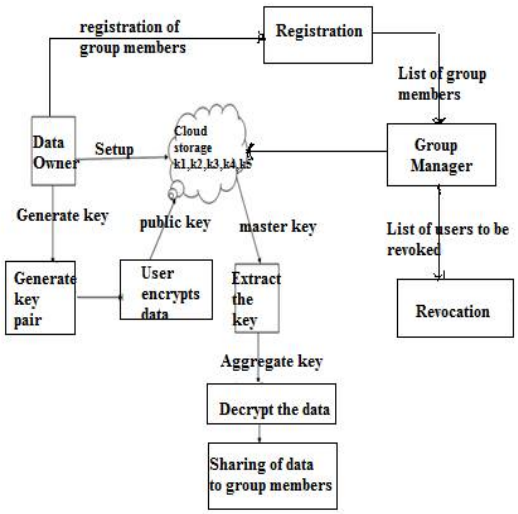

Fig 1: System architecture

Here we describe our framework for data sharing in cloud using aggregate key method in the following steps:

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

The data owner establishes the public system parameter via Setup phase and generates a key via KeyGen phase. Files can be encrypted via Encrypt phase by anyone who also decides the file needs to be encrypted. The data owner can use the master-secret key to generate an aggregate decryption key for a set of files via Extract phase. The generated keys can be passed to delegatees within the group securely via secure e-mails. Finally, any user with an aggregate key

within the group can decrypt any documents provided that the document is contained in the aggregate key via Decrypt phase. Sharing of data is done according to the access policies assigned for a particular group over the group of user's. Revocation phase includes access policies for the particular user whether the user wants to exit from the group then access policies with that user will be removed.

Step 1: Setup (1^, n): In this step, executed by the data owner to setup an account on an untrusted server. On input a security level parameter 1^ and the number of cipher text classes n, that is the class index should be an integer bounded by 1 and n, it outputs the public system parameter param, which is omitted from the input of the other algorithms for brevity.

Step 2: KeyGen (pk, msk): In this step, executed by the data owner to randomly generate a key pair. The key pair that is generated is the public key and master-secret key pair. The public key is used by the data owner for encrypting the message in the encrypt phase. The master-secret key is being used to extract the aggregate key for a corresponding cipher text classes according to the assigned access policies.

Step 3: Encrypt (pk, i, m): In this step, encryption of data is done using AES algorithm. This step is executed by anyone who wants to encrypt data i.e. message m. On input a public-key pk, an index i denoting the cipher text class, and a message m, it outputs a cipher text C. The encryption is done using Advanced Encryption Standard algorithm.

Step 4: Extract (msk, S): In this step, the aggregate key is extracted using the master secret key with the class index, access policies need to be assigned properly. This is executed by the data owner for delegating the decrypting power for a certain set of cipher text classes to a delegatee. On input the master-secret key msk and a set S of indices corresponding to different classes, it outputs the aggregate key for set S denoted by K_S.

Step 5: Decrypt (K_S, S, i, C): In this step, decryption of data is done using the AES algorithm. This step is executed by the delegate within the group, who received an aggregate key K_S.
The aggregate key is used which is generated in extract phase. On input as the aggregate key K_S, the set of cipher text classes S, an index i denoting the cipher text class the cipher text C belongs to, and C, it outputs the decrypted data i.e. the message m if i belongs to S.

Step 6: Revoke: In this step, the group manager handles this phase as in such it includes revocation of the users. If any member in the group wants to exit from the group then this is done only by the group manager as the access policies are defined.

C.Mathematical Model

Let S be the system to perform Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage

[1]Graph Representation and its Functions:

Let G be a closed graph that represents our system;
Such that G = {E, V}

Where,

E represents the set of edges; E = {e1, e2, e3 ....., e10} and

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

V is a set of vertices; V = {v1, v2, v3...., v5}.

In the graphical representation of the system, vertices in the set V represent the modules which are connected through directed edges in the set E representing the input/output of modules.

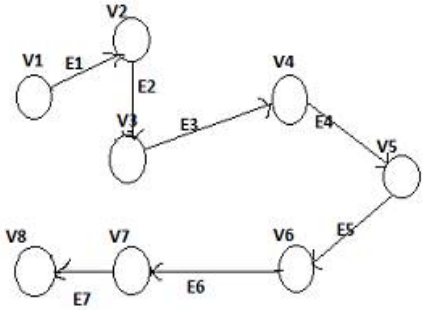

Fig 2: Graph Representation

Let fe be a rule of E into V such that for given edge; it returns vertices. fe (E) — V.
Thus, for our system,
fe (e1) = v2.....v2 is called using e1 for encryption of the data

fe (e2) = v3..... data is passed to v3 using e2 to generate an key.

fe (e3) = v4..... data is passed to v4 using e3 for encryption using the key that is being generated.

fe (e4) = v5..... v5 is called using e4 for uploading the data over the cloud.

fe (e5) = v6.....the key for decryption that is the aggregate key e5 is passed to v6.

fe (e6) = v7.....the key is used for decryption for downloading the data from the cloud.

fe (e7) = v8.....original message is retrieved from v8 is called by e7 when the key is entered.

V. RESULTS AND DISCUSSIONS

To show the effectiveness of the proposed system, some experiments are conducted on Java based Windows machine using Apache tomcat as the server. And a developed system is put under hammer in many scenarios to prove its authenticity as mentioned in below tests.

A. Key Complexity

To measure the performance of the system we set the bench mark by considering the system with more number of operating nodes (i.e. users) uploading files over the cloud. To determine the performance of the system, we examined how many relevant keys are been generated on the rise of the number of users in the scenario. So the available result is shown in figure 3.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

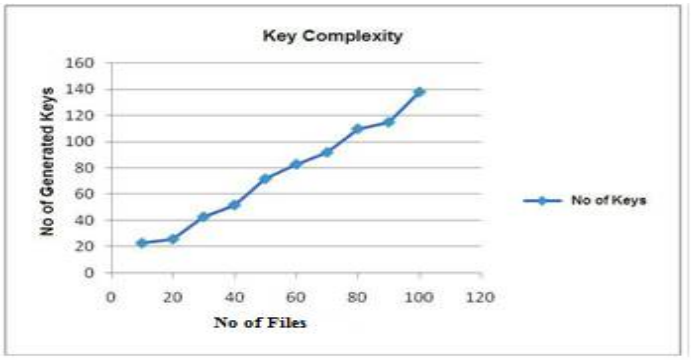

Fig 3: Key Complexity

The plot in figure 3 clearly indicates that the number of keys generated are always directly proportional to the number of the active users in the web system i.e. directly proportional to the number of files uploaded by the users in a particular group. This actually shows a good behavior of our model in Cloud system.

B. Delegation Ratio

We assume that there are exactly 2^h cipher text classes, and the delegatee of concern is entitled to a portion r of them. That is, r is the delegation ratio, the ratio of the delegated cipher text classes to the total classes. Obviously, if r ¼ 0, n_a should also be 0, which means no access to any of the classes; if r ¼ 100%, n_a should be as low as 1, which means that the possession of only the root key in the hierarchy can grant the access to all the 2^h classes.

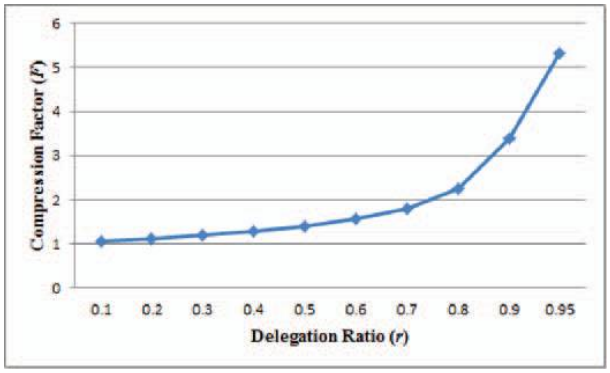

Fig 4: Compression achieved for tree based approach for delegating ratio of the classes

The compression factor F for a certain h, i.e., the average number of delegated classes that each granted key can decrypt. Specifically, it is the ratio of the total number of delegated classes (r2h) to the number of granted keys required (na). Certainly, higher compression factor is referable because it means each granted key can decrypt more cipher texts.

VI. CONCLUSION

In this paper, we consider how to generate the dynamic groups in the cloud computing. Also we have consider how to compact the secret keys of the data user's in public key crypto systems which supports delegation of secret keys with respect to the access policies in the cloud storage. The central focus is on the user revocation which should solve the problems of efficiency and storage. In addition, we analyze the security of our scheme with rigorous proofs, and

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 6, June 2016

demonstrate the efficiency of our scheme in experiments. Meanwhile, the storage overhead and encryption computation cost of our scheme are independent with the number of revoked users.

ACKNOWLEDGEMENT

It is a pleasure for me to thank many people who have supported me in completion of this paper work. Firstly, I would like to thank my Guide, Prof. H. A. Hingoliwala for his support during the entire work. He highlighted the key areas in this topic that helped get the right content in the topic. Also I extend my thanks to our HOD, principal, teachers and college to provide us the facilities in the department and their guidance.

REFRENCES

[1] Cheng-Kang Chu, Sherman S.M. Chow, Wen-Guey Tzeng, Jianying Zhou, and Robert H. Deng, "Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage," IEEE Trans. Parallel and distributed systems, Vol. 25, no. 2, Feb 2014.
[2] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, vol. 62, no. 2, pp. 362- 375, Feb. 2013.
[3] J. Benaloh, M. Chase, E. Horvitz, and K. Lauter,"Patient Controlled Encryption: Ensuring Privacy of Electronic Medical Records," in Proceedings of ACM Workshop on Cloud Computing Security (CCSW 09). ACM, 2009, pp. 103 114.
[4] F. Guo, Y. Mu, Z. Chen, and L. Xu,"Multi-Identity Single-Key Decryption without Random Oracles," Proc. Information Security and Cryptology (Inscrypt 07), vol. 4990, pp. 384-398, 2007.
[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-Based Encryption for Fine- Grained Access Control of Encrypted Data," Proc. 13th ACM Conf. Computer and Comm. Security (CCS 06), pp. 89-98, 2006.
[6] A. Sahai and B. Waters,"Fuzzy Identity-Based Encryption," Proc. 22nd Intl Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT 05), vol. 3494, pp. 457-473, 2005.
[7] G. Ateniese, K. Fu, M. Green, and S. Hohenberger,"Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage," ACM Trans. Information and System Security, vol. 9, no. 1, pp. 1-30, 2006.
[8] A. B. Lewko and B. Waters (2011)"Decentralizing attribute-based encryption," in Proc. EUROCRYPT, pp. 568588.
[9] R.S. Sandhu, "Cryptographic Implementation of a Tree Hierarchy for Access Control," Information Processing Letters, vol. 27, no. 2, pp. 95-98, 1988.
[10] Q. Zhang and Y. Wang, "A Centralized Key Management Scheme for Hierarchical Access Control," Proc. IEEE Global Telecomm. Conf. (GLOBECOM 04), pp. 2067-2071, 2004.

BIOGRAPHY

Rachana Gangwani is currently pursing M.E. (Computer) from Department of computer Engineering, Jayawantrao Sawant College of Engineering, Pune, India. Savitribai Phule Pune University, Pune, Maharashtra, Pune-411007.She received her B.E. (Computer) Degree from Jayawantrao Sawant College of Engineering, Pune, India. Savitribai Phule Pune University, Pune, Maharashtra, Pune- 411007.

Prof. H.A Hingoliwala, M.E (Computer) Head of Department and Asst Prof (Computer) Department of Computer Engineering, Jayawantrao Sawant College of Engineering, Pune, India. Savitribai Phule Pune University, Pune, Maharashtra, India-411007. He is awarded with the degree of B.E. (Computer) and M.E. (Computer). He has 17 years of teaching experience. His area of interest is image processing.