NOVATEUR PUBLICATIONS
INTERNATIONAL JOURNAL OF INNOVATIONS IN ENGINEERING RESEARCH AND TECHNOLOGY [IJIERT]
ISSN: 2394-3696
VOLUME 2, ISSUE 5, MAY-2015

SECURE MULTI-OWNER DATA SHARING FOR DYNAMIC GROUPS IN THE CLOUD

R.Rathidevi
Research student
Computer Science and Engineering Department,
National Engineering College, India.

N.Nachiyar
Research student
Computer Science and Engineering Department,
National Engineering College, India.

J.Naskath
Professor
Computer Science and Engineering Department,
National Engineering College, India.

ABSTRACT

Cloud computing is associate degree raising computing paradigm within which resources of the computing infrastructure area unit provided as services over the net. As promising because it is, this paradigm conjointly brings forth several new challenges for knowledge security and access management once users confidential against un trusted servers, existing solutions sometimes apply crypto logical ways by revealing knowledge cryptography keys solely to approved users. One among the largest considerations with cloud knowledge storage is that of knowledge integrity verification at un trusted servers. To preserve knowledge privacy, a basic resolution is to encode knowledge files, so transfer the encrypted knowledge into the cloud. To resolve this downside recently the simplest economical technique Anglesey given for secured multi owner knowledge sharing in but we have a tendency to known some limitations within the same approach in terms of responsibleness and quantifiability. Therefore during this paper we have a tendency to area unit more extending the fundamental Anglesey by adding the responsibleness still as raising the quantifiability by increasing the quantity of cluster managers dynamically.

KEYWORDS: Dynamic groups, Multi owner, Data Sharing, Cloud Computing, integrity, scalability.

INTRODUCTION

Maintaining the integrity information plays an important role within the institution of trust between data subject and repair supplier. Though unreal as a promising service platform for the web, the new information storage paradigm in "Cloud" brings concerning several difficult style problems that have profound influence on the safety and performance of the system. One in all the most important issues with cloud information storage is that of information integrity verification at entrusted servers. What's a lot of serious is that for saving cash and space for storing the service supplier would possibly neglect to stay or deliberately delete seldom accessed information files that belong to a normal shopper. take into account the massive size of the outsourced electronic information and also the client's unnatural resource capability, the core of the matter may be generalized as however will the shopper notice associate economical thanks to perform periodical integrity verifications while not the native copy of information files. To preserve information privacy, a basic resolution is to encode information files, and so transfer the encrypted information into the cloud [2]. CS2 provides security against the cloud supplier, shoppers square measure still in a position not solely to with efficiency access their information through a research interface however additionally to feature and delete files firmly. many security schemes for information sharing on entrusted servers are planned secure filing system designed to be bedded over insecure network and P2P file systems like NFS, CIFS, Ocean Store, and Yahoo! case. The remaining paper is organized as follows. The related works are discussed in section II. The existing system is presented in section III. Proposed system is presented in section IV. Performance evaluation presented in section V and Section VI concludes the paper.

RELATED WORK

E. Goh [4] the use of SiRiUS is compelling in situations where users have no control over the file server (such as Yahoo! Briefcase or the P2P file storage provided by Farsite). They believe that SiRiUS is the most that can be done to secure an existing network file system without changing the file server or file system protocol. Key management and revocation is simple with minimal out-of-band communication. File system freshness guarantees are supported by SiRiUS using hash tree constructions. SiRiUS contains a novel method of performing file random access in a cryptographic file system without the use of a block server. Extensions to SiRiUS include large scale group sharing using the NNL key revocation construction B. Wang[5] in this paper, we propose Knox, a privacy-preserving auditing scheme for shared data with large groups in the cloud. They utilize group signatures to compute verification information on shared data, so that the TPA is able to audit the correctness of shared data, but cannot reveal the identity o f the signer on each block. With the group manager's private key, the original user can efficiently add new users to the group and disclose the identities of signers on all blocks. The efficiency of Knox is not affected by the number of users in the group. M. Armbrust [2] the data centers hardware and software is what we will call a cloud. When a cloud is made

available in a pay-as-you-go manner to the general public, they call it a public cloud; the service being sold is utility computing. They use the term private cloud to refer to internal data centers of a business or other organization, not made available to the general public, when they are large enough to benefit from the advantages of cloud computing that we discuss here. Thus, cloud computing is the sum of SaaS and utility computing, but does not include small or medium-sized data centers, even if these rely on virtualization for management. People can be users or providers of SaaS, or users or providers of utility computing. They focus on SaaS providers (cloud users) cloud providers, which have received less attention than SaaS users. S. Kamara [3] in this paper consider the problem of building a secure cloud storage service on top of a public cloud infrastructure where the service provider is not completely trusted by the customer. They describe, at a high level, several architectures that combine recent and non-standard cryptographic primitives in order to achieve our goal. Survey the benefits such architecture would provide to both customers and service providers and give an overview of recent advances in cryptography motivated specifically by cloud storage. A. Fiat [6] they introduce new theoretical measures for the qualitative and quantitative assessment of encryption schemes designed for broadcast transmissions. The goal is to allow a central broadcast site to broadcast secure transmissions to an arbitrary set of recipients while minimizing key management related transmissions. They present several schemes that allow centers to broadcast a secret to any subset of privileged users out of a universe of size so that coalitions of users not in the privileged set cannot learn the secret. V. Goyal, O. Pandey, A. Sahai, and B. Waters [7] they develop a new cryptosystem for One-grained sharing of encrypted data that call Key-Policy Attribute-Based Encryption (KP-ABE). In cryptosystem, cipher texts are labeled with sets of attributes and private keys are associated with access structures that control which cipher texts a user is able to decrypt. They demonstrate the applicability of our construction to sharing of audit-log information and broadcast encryption. Our construction supports delegation of private keys which subsumes Hierarchical Identity-Based Encryption (HIBE). The data owner uses a random key to encrypt a file, where the random key is further encrypted with a set of attributes using KP-ABE. Then, the group manager assigns an access structure and the corresponding secret key to authorized users, such that a user can only decrypt a cipher text if and only if the data file attributes satisfy the access structure. To achieve user revocation, the manager delegate's tasks of data file reencryption and user secret key update to cloud servers. However, the single owner manner may hinder the implementation of applications with the scenario, where any member in a group should be allowed to store and share data files with others.

EXISTING SYSTEM

In the literature study we have seen many methods for secure data sharing in cloud computing, however most methods failed to achieve the efficient as well as secure method for data sharing for groups. To provide the best solutions for the problems imposed by existing methods, recently the new method was presented through this paper

[1]. This approach presents the design of secure data sharing scheme, for dynamic groups in an un trusted cloud. Here the user is able to share data with others in the group without revealing identity privacy to the cloud. Additionally, this supports efficient user revocation and new user joining. More specially, efficient user revocation can be achieved through a public revocation list without updating the private keys of the remaining users, and new users can directly decrypt files stored in the cloud before their participation. Moreover, the storage overhead and the encryption computation cost are constant. Therefore practically in all cases outperforms the existing methods.

REVOCATION LIST

User revocation is performed by the group manager via a public available revocation list (RL), based on which group members can encrypt their data files and ensure the confidentiality against the revoked users. The revocation list is characterized by a series of time stamps (t1 < t2 <… tr). Let ID group denote the group identity. The tuple (Ai, xi, ti) represents that user i with the partial private key (Ai, xi) is revoked at time ti. P1, P2, …, Pr and Zr are calculated by the group manager with the private secret as follows: here x1=y1, x2=y2 and xr=yr.

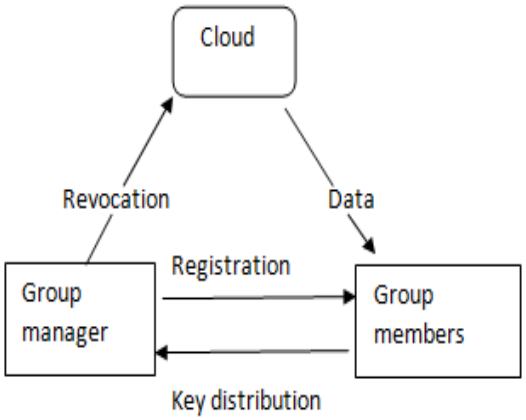

Figure 1: Existing system

(Ai, xi, ti) represents that user i with the partial private key (Ai, xi) is revoked at time ti. P1, P2, Pr and Zr are calculated by the group manager with the private secret as follows: here x1=y1, x2=y2 and xr=yr.

$$P_r = \frac{1.\,(P \times G_n)}{(\Upsilon + x_{(n+1)}) \ldots (\Upsilon + x_r)}$$

Motivated by the verifiable reply mechanism in [13], to guarantee that users obtain the latest version of the revocation list, we let the group manger update the revocation list each day even no user has being revoked in the day. In other words, the others can verify the freshness of the revocation list from the contained current date CRL. In addition, the

revocation list is bounded by a signature sig(RL) to declare its validity. The signature is generated by the group manager with the BLS signature algorithm [14]. Finally, the group manager migrates the revocation list into the cloud for public usage. Disadvantage However as per reliability and scalability concern this method needs to be workout further as if the group manager stop working due to large number of requests coming from different groups of owners, then entire security system of this failed down. In revocation list the time given for each user is fixed after time expire user cannot access the data until group manager update the revocation list and give it to the cloud.

PROPOSED SYSTEM

To achieve the reliability and scalability, in this paper we are presenting the new framework for data sharing. In this method we are further presenting how we are managing the risks like failure of group manager by increasing the number of backup group manager, hanging of group manager in case number of requests more by sharing the workload in multiple group managers. This method claims required efficiency, scalability and most importantly reliability. Advantage To overcome the disadvantage of existing system, in the proposed system if the group manager stop working due to large number of requests coming from different groups of owners, then backup group manager will remains available. Here user gets extra time for accessing data after the time out by sending request to the cloud.

Scheme Description: This section describes system, initialization, user registration, user revocation, file generation, file deletion and file access.

System Initialization: The group manager takes charge of system initialization as follows: Generating a bilinear map group system S=(q, G1, G2,e(.,.)).The system parameters including (S, P, H, H0 ,H1 ,H2, U, V , W , Y , Z, f, f1, Enc()), where f is a one-way hash function: {0,1}* —> Z*q ; f1 is hash function: {0,1}* —> G1; and Enck() is a secure symmetric encryption algorithm with secret key k.

User Registration

For the registration of user i with identity IDi, the group manager randomly selects a number xi belong to Z*q and computes Ai, Bi as the following equation:

Then, the group manager adds (Ai, xi, IDi) into the group user list, which will be used in the traceability phase. After the registration, user i obtains a private key (xi, Ai, Bi), which will be used for group signature generation and file decryption.

Table 1: Message format

Group ID	Data ID	Cipper text	Hash	Time	Signature
ID group	ID data	C1,C2, C	f(r)	T data	Key

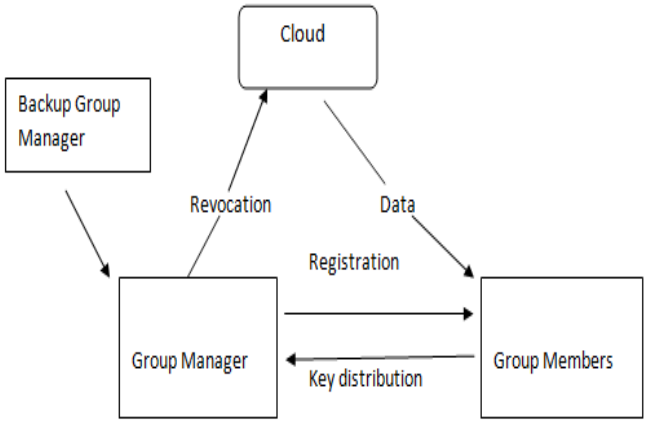

Figure 2: Proposed System Model

$$Y1 = \frac{1}{y + x1} . p \in G1.$$

$$Y2 = \frac{1}{y + x2} . q \in G2.$$

REVOCATION LIST

User revocation is performed by the group manager via a public available revocation list (RL), based on which group members can encrypt their data files and ensure the confidentiality against the revoked users. The list is characterized by time stamp t1,t2,…tr. In the proposed system once the user time stamp over does not wait for the group manager to update the time stamp or revocation list here once the time over the user immediately send request for extra time for access the data to the cloud. Then the cloud will send that request to the group manager once the see it and give permission then the cloud will time to access the data but if the group manager did not give permission then the cloud will not give permission for access of the data. Getting the revocation list from the cloud. In this step, the member sends the group identity ID group as a request to the cloud. Then, the cloud responds the revocation list RL to the member. Verifying the validity of the received revocation list.

Table 2: Revocation list

Groups	Data	Upload	Download from same group	Download from different group
	D1	U1	S1	T1
	D2	U2	S1	T1
	Dr	Ur	Sr	Tr

File Generation & Deletion:

To store and share a data file in the cloud, a group member performs the following operations: First, checking whether the marked date is fresh. Second, verifying the contained signature sig(RL) by the equation e(W, f1 (RL)) = e(P, sig(RL)). If the revocation list is invalid, the data owner stops this scheme. Encrypting the data file M. Selecting a random number T and computing fT. The hash value will be used for data file deletion operation. In addition, the data owner adds (IDdata, T) into his local storage. Constructing the uploaded data file as shown in Table 2, where tdata denotes the current time on the member, and a group signature on (IDdata, C1, C2, C, f(T); tdata) computed by the data owner through private key (A, x). Uploading the data shown in Table 2 into the cloud server and adding the ID data into the local shared data list maintained by the manager. On receiving the data, the cloud first checks its validity. If the algorithm returns true, the group signature is valid; otherwise, the cloud abandons the data. In addition, if several users have been revoked by the group manager, the cloud also performs revocation verification. Finally, the data file will be stored in the cloud after successful group signature and revocation verifications. File stored in the cloud can be deleted by either the group manager or the data owner (i.e., the member who uploaded the file into the server). To delete a file ID data, the group manager computes a signature and sends the signature along with ID data to the cloud.

PERFORMANCE EVALUATION

In this section, we first analyze the storage cost of this paper, and then perform experiments to test its computation cost. ***Storage:*** Without loss of generality, we set q=160 and the elements in G1 and G2 to be 161 and 1,024 bit, respectively. In addition, we assume the size of the data identity is 16 bits, which yield a group capacity of 2^{16} data files. Similarly, the size of user and group identity are also set as 16 bits. Group manager. In Mona, the master private key of the group manager

$$(G, E1, E2) \in G_1 \text{ X } z_q^3$$

Additionally, the user list and the shared data list should be stored at the group manager. Considering an actual system with 200 users and assuming that each user share 50 files in average, the total storage of the group manager is (80.125+42.125*200+2*10,000)* 10^{-3} 28.5 Kbytes, which is very acceptable. Group members. Essentially, each user in our scheme only needs to store its private key (Ai, Bi, xi) $\in G_1^2 \times Z_q$ which is about 60 bytes. It is worth noting that there is a tradeoff between the storage and the computation overhead. For example, the four pairing operations including (e(H, W), e(H, P), e(P, P), e(Ai, P)) $\in G_2^4$ can be precompiled once and stored for the group signature generation and verification. Therefore, the total storage of each users is about 572 bytes. The extra storage overhead in the cloud. Here , the format of files stored in the cloud is shown in Table 2. Since C3 is the cipher text of the file under the symmetrical encryption, the extra storage overhead to store the file is about 248 bytes, which includes.(ID (Group) ,ID (data),C1,C2,C3,f(r), t(data), Key)

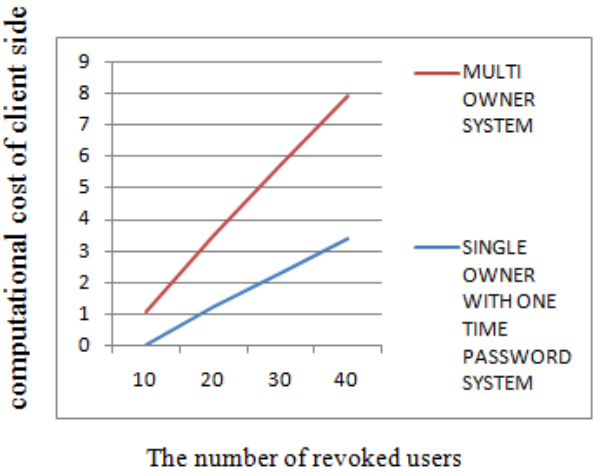

Figure 3: Comparison on computation cost for file generation between multi owner and ODBE.

SIMULATION

The simulation consists of three components: client side, manager side as well as cloud side. Both client-side and manager-side processes are conducted on a laptop with Core 2 T7250 2.0 GHz, DDR2 800 2G, Ubuntu 10.04 X86. The cloud-side process is implemented on a machine that equipped with Core 2 i3-2350 2.3 GHz, DDR3 1066 2G,Ubuntu 12.04 X64. In the simulation, we choose an elliptic curve with 160-bit group order, which provides a competitive security level with 1,024-bit RSA.

CLIENT COMPUTATION COST

In Fig.3, list the comparison on computation cost of clients for data generation operations between multi owner and the way that directly using the original dynamic broadcast encryption. It is easily observed that the computation cost here is irrelevant to the number of revoked users. On the contrary, the computation cost increases with the number of revoked users in ODBE. The reason is that the parameters (Pr, Zr) can be obtained from the revocation list without sacrificing the security , while several time-consuming operations including point multiplications in G1 and exponentiations in G2 have to be performed by clients to compute the parameters in ODBE. From Figs. 5.1a and 5.1b, we can find out that sharing a 10 Mbyte file and a 100-Mbyte one, cost a client about 0.2 and 1.4 seconds in our scheme, respectively, which implies that the symmetrical encryption operation domains the computation cost when the file is large. The computation cost of clients for file access operation with the size of 10 and 100

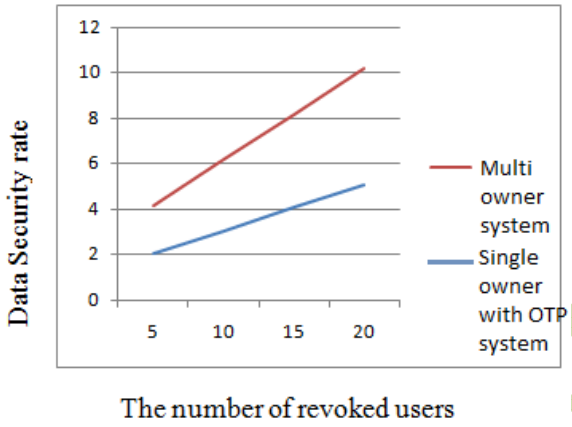

Figure 4: Comparison on computation cost for file access & Multi owner and ODBE.

Mbytes are illustrated in Fig. 3. The computation cost here increases with the number of revoked users, besides the above operations, P1, P2, Pr need to be computed by clients in ODBE. Therefore, this is still superior to ODBE in terms of computation cost. Similar to the data generation operation, the total computation cost is mainly determined by the symmetrical decryption operation if the accessed file is large, which can be verified from

CONCLUSION

Cloud computing is very attractive environment for business world in term of providing required services in a very cost effective way. However, assuring and enhancing security and privacy practices will attract more enterprises to world of the cloud computing In Thus to achieve the reliability and scalability here, in this paper we are presenting the new framework for multi owner. In this method we are further presenting how we are managing the risks like failure of group manager by increasing the number of backup group manager, hanging of group manager in case number of requests more by sharing the workload in multiple group managers. This method claims required efficiency, scalability and most importantly reliability. Extensive analyses show that our proposed scheme satisfies the desired security requirements and guarantees efficiency as well. Here we also show that how user gets extra time even after the time out this also one of the advantage of proposed schema.

REFERENCES

[1] Xuefeng Liu, Yuqing Zhang, Boyang Wang, and Jingbo Yan, "Mona: *Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 24, NO. 6, JUNE 2013.*

[2] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee,

D.A. Patterson, A. Rabkin, I.Stoica, and M. Zaharia, "*A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.*

[3] S. Kamara and K. Lauter, "*Cryptographic Cloud Storage," Proc. Int'l Conf. Financial Cryptography and Data Security (FC), pp. 136- 149, Jan. 2010.*

[4] E. Goh, H. Shacham, N. Modadugu, and D. Boneh, "*Sirius: Securing Remote Untrusted Storage," Proc. Network and Distributed Systems Security Symp. (NDSS), pp. 131- 145, 2003.*

[5] B. Wang, B. Li, and H. Li, "Knox: *Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud," Proc. 10th Int'l Conf. Applied Cryptography and Network Security, pp. 507-525, 2012.*

[6] A. Fiat and M. Naor, "*Broadcast Encryption," Proc. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 480-491, 1993.*

[7] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "*Attribute- Based Encryption for Fine-Grained Access Control of Encrypted Data," Proc. ACM Conf. Computer and Comm. Security (CCS), pp. 89-98, 2006.*

[8] D. Pointcheval and J. Stern, "*Security Arguments for Digital Signatures and Blind Signatures," J. Cryptology, vol. 13, no. 3, pp. 361-396, 2000.*

[9] R. Lu, X. Lin, X. Liang, and X. Shen, "*Secure Provenance: The Essential of Bread and Butter of Data Forensics in Cloud Computing," Proc. ACM Symp. Information, Computer and Comm. Security, pp. 282-292, 2010.*

[10] D. Naor, M. Naor, and J.B. Lotspiech, "*Revocation and Tracing Schemes for Stateless Receivers," Proc. Ann. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 41-62, 2001.*

[11] D. Boneh and M. Franklin, "*Identity-Based Encryption from the Weil Pairing," Proc. Int'l Cryptology Conf.Advances in Cryptology (CRYPTO), pp. 213-229, 2001.*

[12] D. Boneh, X. Boyen, and H. Shacham, *"Short Group Signature," Proc. Int'l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 41-55, 2004.*