International Journal of Engineering Inventions
e-ISSN: 2278-7461, p-ISSN: 2319-6491
Volume 5, Issue 2 [Dec 2015] PP: 27-31

Advanced Sharing Reliable Data in the Cloud for the Multi User Group

*Prof.GVNKV Subbarao,** Prof.MD Sameeruddin Khan,*** Muneer Ahmed
,,*** Computer Science Engineering Department,Sree Dattha Institute of Engineering & Science*

Abstract: Although the cloud computing model is considered to be a very promising internet-based computing platform, it results in a loss of security control over the cloud-hosted assets. This is due to the outsourcing of enterprise IT assets hosted on third-party cloud computing platforms. Moreover, the lack of security constraints in the Service Level Agreements between the cloud providers and consumers results in a loss of trust as well. n this project, we propose a secure multi owner data sharing scheme, for dynamic groups in the cloud. By leveraging group signature and dynamic broadcast encryption techniques, any cloud user can anonymously share data with others.Proposing a new model for Sharing Secure Data in the Cloud for the Multiuser Group

Index Terms: *cloud computing, dynamic groups, broadcast, multiuser group*

I. INTRODUCTION

CLOUD computing is recognized as an alternative to service provider information technology due to its intrinsic resources. cloud computing are able to providing the various data sharing between multi to cloud users. cloud servers, users can enjoy quality services and save consequential investments on their local infrastructures sharing and low -maintenance characteristics.The major services offered by cloud providers is data storage and data applications. A organization allows its members in the same department to store sharing information among them in the cloud. released from the troublesome local data storage and maintenance. However, it also poses a significant risk to the confidentiality of those stored files. Cloud offers enormous opportunity for new innovation, and even disruption of entire industries. Cloud computing is the long dreamed vision of computing as a utility, where data owners can remotely store their data in the cloud to enjoy on demand high-quality applications and services from a shared pool of configurable computing resources. Identity privacy is one of the most significant obstacles for the wide deployment of cloud computing. Without the guarantee of identity privacy, users may be unwilling to join in cloud computing systems because their real identities could be easily disclosed to cloud providers and attackers. Although envisioned as a promising service platform for the Internet, the new data storage paradigm in “Cloud” brings about many challenging design issues which have profound influence on the security and performance of the overall system. One of the biggest concerns with cloud data storage is that of data integrity verification at untrusted servers. a basic solution is to encrypt data files, and then upload the encrypted data into the cloud . Unfortunately, designing an efficient and secure data sharing scheme for groups in the cloud is not an easy task due to the following challenging issues.

II. EXISTING SYSTEM

In existing methods for secure data sharing in cloud computing, however most methods failed to *achieved* 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 called MONA [1]. This approach presents the design of secure data sharing scheme, Mona, for dynamic groups in an untrusted cloud. In Mona, a user is able to share data with others in the group without revealing identity privacy to the cloud. Additionally, Mona 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 MONA outperforms the existing methods.

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 MONA 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.

III. PROPOSED SYSTEM

To achieve the reliable and scalable in MONA, in this paper we are presenting the new framework for MONA. 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.

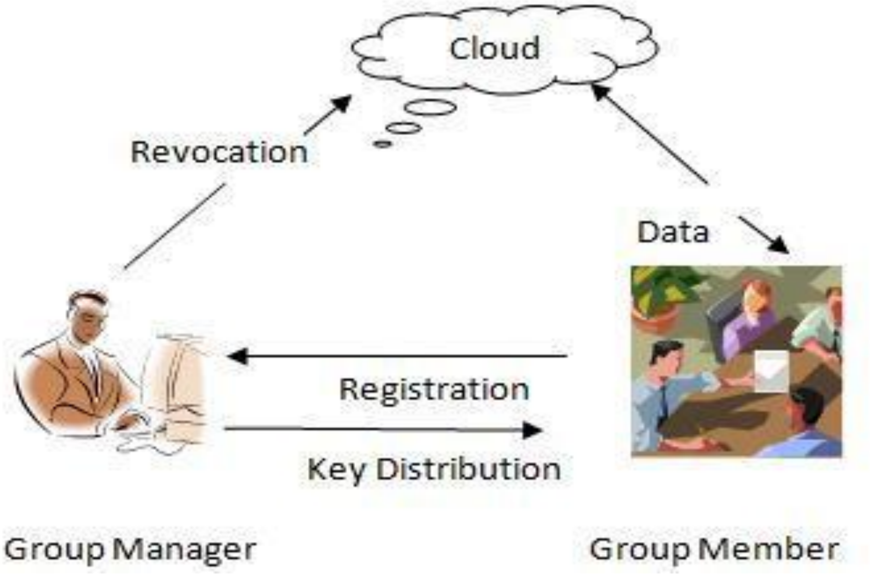


				Table 1
			Revocation List	
IDgroup	D1	y1	t1	P1
	D2	y2	t2	P2
	.	.	.	.
	Dr	yr	tr	PrWr tRL sig(RL)

Fig 4.1 Proposed System Model

Advantage

To overcome the disadvantage of existing system MONA, in the proposed MONA is 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 get extra time for accessing data after the time out by sending request to the cloud.

Scheme Description

This module describe the initialization, new multi user registration, user revocation, file uploadling , file deletion and file access.

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

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.

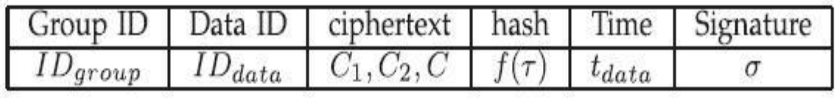
Revocation List

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.

File Generation

To store and share a data file in the cloud, a group member performs the following operations:

Getting the revocation list from the cloud. In this step, the member sends the group identity IDgroup 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.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).



Group ID	Data ID	ciphertext	hash	Time	Signature
ID_{group}	ID_{data}	C_1, C_2, C	$f(\tau)$	t_{data}	σ

Table 2: Message Format

$$\begin{cases} A_i = \dfrac{1}{\gamma + x_i} \cdot P \in G_1 \\ B_i = \dfrac{x_i}{\gamma + x_i} \cdot G \in G_1. \end{cases}$$

Uploading the data shown in Table 2 into the cloud server and adding the IDdata into the local shared data list maintained by the manager. On receiving the data, the cloud first check 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 Deletion

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 IDdata, the group manager computes a signature and sends the signature along with IDdata to the cloud.

IV. PERFORMANCE EVALUATION

In this section, we first analyze the storage cost of Mona, and then perform experiments to test its computation cost.

Storage

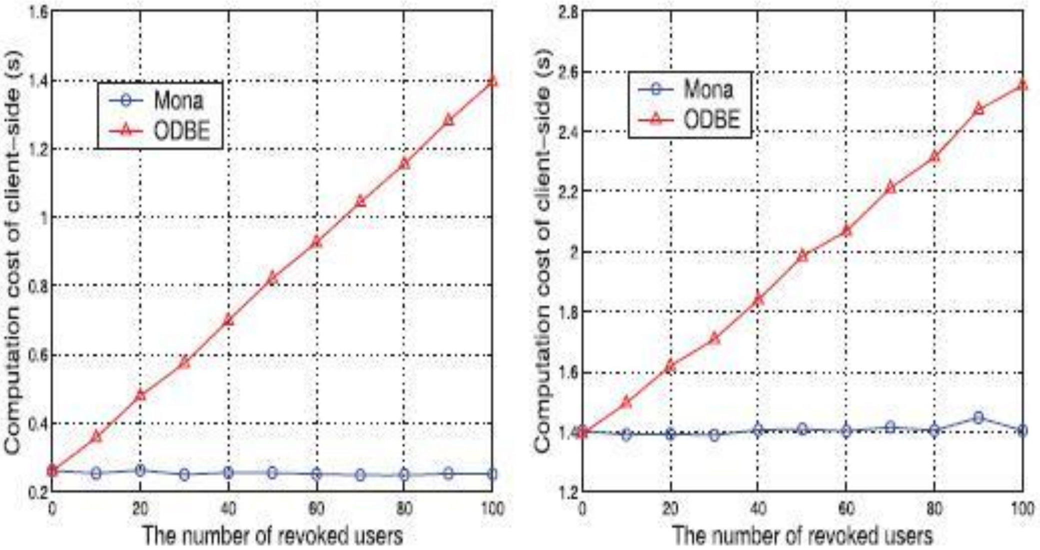
Without loss of generality, we set q=160 and the elements in G1 and G2 to be 161 and 1,024 bit, respectively. In addition, we assume the size of the data identity is 16 bits, which yield a group capacity of 2^{16} data files. Similarly, the size of user and group identity are also set as 16 bits.

Group manager. In Mona, the master private key of the group manager $(G, \gamma, \xi_1, \xi_2) \in G_1 \times \dot{Z}_q{}^3$ 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)* $10^{-3} \approx 28.5$ 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 precomputed 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. In Mona, the format of files stored in the cloud is shown in Table 2. Since C3 is the ciphertext of the file under the symmetrical encryption, the extra storage overhead to store the file is about 248 bytes,

which includes $(ID_{group}, ID_{data}, C_1, C_2, C_3, f(\tau), t_{data}, \sigma)$.



(a) Generating a 10 MB file (b) Generating a 100 MB file

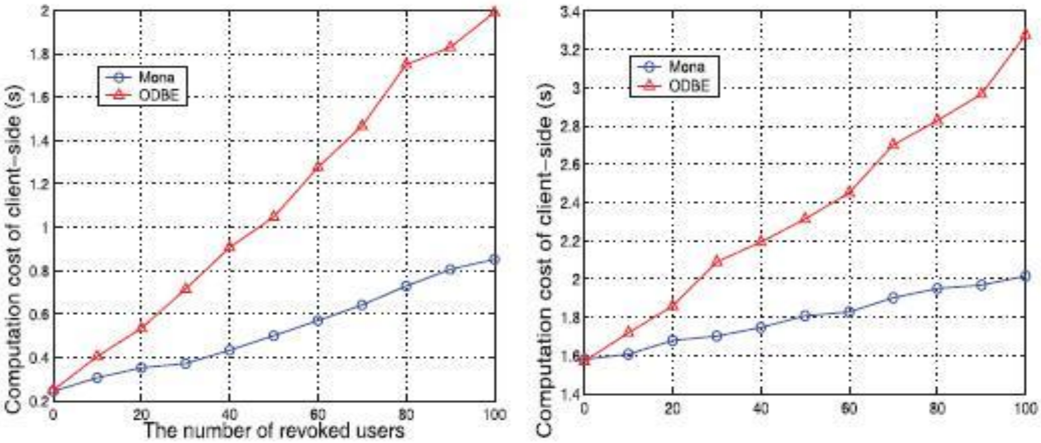
Fig. 5.1. Comparison on computation cost for file generation between Mona and ODBE.

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. 5.1, we list the comparison on computation cost of clients for data generation operations between Mona and the way that directly using the original dynamic broadcast encryption. It is easily observed that the computation cost in Mona 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 in Mona, 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 Mbytes are illustrated in Fig. 5.2. The computation cost in Mona increases with the number of revoked users, Besides the above operations, P1, P2, …, Pr need to be computed by clients in ODBE.



(a)Accessing a 10 MB file (b) Accessing a 100 MB file

Fig. 5.2. Comparison on computation cost for file access between Mona and ODBE.

Therefore, Mona is still superior than 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 Figs. 5.2a and 5.2b. In addition, the file deletion for clients is about 0.075 seconds, because it only costs a group signature on a message (IDdata, T) where T is a 160-bit number in Z*q.

V. CONCLUSION

In 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 reliable and scalable in MONA, in this paper we are presenting the new framework for MONA. 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.

[3]. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010. 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, 2

[13]. B. Sheng and Q. Li, "Verifiable Privacy-Preserving Range Query in Two-Tiered Sensor Networks," Proc. IEEE INFOCOM, pp. 46-50, 2008.