A Threshold Signature: Secure Data Sharing for Dynamic Groups in the Cloud

[1]B. Suganya
[1]PG Scholar,
CSE Department
P.S.R. Engineering College, Sivakasi
Affiliated to Anna University,
Chennai, India

[2]Dr. K. Vimala Devi
[2]Professor
CSE Department
P.S.R. Engineering College, Sivakasi
Affiliated to Anna University,
Chennai, India

***Abstract*:- Cloud computing is an emerging computing paradigm. It provides an economical and efficient solution for sharing group resource among cloud users. Due to frequent change of members in multi owner group, preserving user data and their identity privacy becomes a challenging issue in cloud. In this paper, we propose a secure multi-owner data sharing scheme, for dynamic groups in the cloud.By including a (t, n) Threshold Signature and stateless broadcast encryption techniques, any cloud user can anonymously share data with others. The security of the proposed threshold signature scheme is based on the difficulty of computing the discrete logarithm modulo for a composite number. The size of the group signature and the verification time of the group signature are equivalent to that of an individual signature.Meanwhile, the storage overhead and encryption computation cost of our scheme are independent with the number of revoked users.In addition, we analyze the security of our scheme with rigorous proofs, and demonstrate the efficiency of our scheme in experiments.**

Keywords – Cloud computing, Data Sharing, Identity Privacy, Privacy Preserving, Dynamic Groups, Threshold Signature.

I. INTRODUCTION

With cloud computing and storage, users are able to access and to share resources offered by cloud service providers at a lower marginal cost. Cloud computing is recognized as an alternative to traditional information technology due to its intrinsic resource sharing and low-maintenance characteristics. In cloud computing, the cloud service providers (CSPs), such as Amazon, are able to deliver various services to cloud users with the help of powerful datacenters. By migrating the local data management systems into cloud servers, users can enjoy high quality services and save significant investments on their local infrastructures. One of the most fundamental services offered by cloud providers is data storage. Let us consider a practical data application. A company allows its staffs in the same group or department to store and share files in the cloud. By utilizing the cloud, the staffs can be completely released from the troublesome local data storage and maintenance. However, it also poses a significant risk to the confidentiality of those stored files. Specifically, the cloud servers managed by cloud providers are not fully trusted by users while the data files stored in the cloud may be sensitive and confidential, such as business plans. To preserve data privacy, 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.

In the conventional digital signatures such as RSA andElGamal, a single signer is sufficient to sign a message and any verifier can verify the validity of the signature with the signer's public key. However, the responsibility of signing messages for many applications needs to be shared by a set of signers. It is a policy for some applications and occasions that at least t persons rather than one person generate cooperatively a signature. The multi-signature schemes [12], the (t, n) threshold signature schemes and the (t, n) threshold multi-signature schemes were proposed for slightly different concepts. In these schemes, it is required that several signers cooperate to generate a valid group signature for a message on behalf of the group.

In a multi-signature scheme, several signers can generate a signature for a message; any verifier may check the validity of the multi-signature with the public keys of the signers. In threshold multi-signature schemes, t or more members in the group can cooperate to generate a valid group signature on behalf of the group. One verifier is sufficient to verify a given a given signature and the verifier needs the public keys of the signers for verification. In the above schemes, the signers are not anonymous. Finally, the (t, n) threshold signature scheme has the feature that t or more members of the group can cooperate to generate a valid group signature on behalf of the group. And the verifier can check the validity of the group signature without identifying the identities of the signers. That is, the signers are anonymous.

A. *Challenging Issues*

First, 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. On the other hand, unconditional identity privacy may incur the abuse of privacy. For example, a misbehaved staff can deceive others in the company by sharing false files without being traceable. Therefore, traceability, which enables the group manager (e.g., a company manager) to reveal the real identity of a user, is also highly desirable.

Second, it is highly recommended that any member in a group should be able to fully enjoy the data storing and sharing services provided by the cloud, which is defined as the multiple-owner manner. Compared with the single-owner manner, where only the group manager can store and modify data in the cloud, the multiple-owner manner is more flexible in practical applications. More concretely, each user in the group is able to not only read data, but also modify his/ her part of data in the entire data file shared by the company.

Last but not least, groups are normally dynamic in practice, e.g., new staff participation and current employee revocation in a company. The changes of membership make secure data sharing extremely difficult. On one hand, the anonymous system challenges new granted users to learn the content of data files stored before their participation, because it is impossible for new granted users to contact with anonymous data owners, and obtain the corresponding decryption keys. On the other hand, an efficient membership revocation mechanism without updating the secret keys of the remaining users is also desired to minimize the complexity of key management.

II. EXISTING SYSTEM

Several security schemes for data sharing on untrusted servers have been proposed. In these approaches, data owners store the encrypted data files in untrusted storage and distribute the corresponding decryption keys only to authorized users. Thus, unauthorized users as well as storage servers cannot learn the content of the data files because they have no knowledge of the decryption keys. However, the complexities of user participation and revocation in these schemes are linearly increasing with the number of data owners and the number of revoked users, respectively. By setting a group with a single attribute, Lu et al proposed a secure provenance scheme based on the cipher text-policy attribute-based encryption technique, which allows any member in a group to share data with others. However, the issue of user revocation is not addressed in their scheme. Yu et al presented a scalable and fine-grained data access control scheme in cloud computing based on the key policy attribute based encryption (KP-ABE) technique.

Disadvantages of Existing System:

- Without Guarantee of Identity Privacy, users may be unwilling to join in cloud computing systems.
- The changes of membership make secure data sharing extremely difficult the issue of user revocation is not addressed.
- Only the group manager can store and modify data in the cloud.

III. RELATED WORK

In [3], Yu et al. presented a scalable and fine-grained data access control scheme in cloud computing based on the KPABE technique. 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 groupmanager assigns an access structure and the corresponding secret key to authorized users, such that a user can only decrypt a ciphertext if and only if the data file attributes satisfy the access structure. To achieve user revocation, the manager delegates 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.

In [4], Kallahalla et al. proposed a cryptographic storage system that enables secure file sharing on untrusted servers, named Plutus. By dividing files into file groups and encrypting each file groups with a unique file-block key, the data owner can share the file groups with others through delivering the corresponding lockbox key, where the lockbox key is used to encrypt the file-block keys. However, it brings about a heavy key distribution overhead for large-scale file sharing. Additionally, the file-block key needs to be updated and distributed again for a user revocation.

In [5], files stored on the untrusted server include two parts: file metadata and file data. The file metadata implies the access control information including a series of encrypted key blocks, each of which is encrypted under the public key of authorized users. Thus, the size of the file metadata is proportional to the number of authorized users. The user revocation in the scheme is an intractable issue especially for large-scale sharing, since the file metadata needs to be updated. In their extension version, the NNL construction [10] is used for efficient key revocation. However, when a new user joins the group, the private key of each user in an NNL system needs to be recomputed, which may limit the application for dynamic groups. Another concern is that the computation overhead of encryption linearly increases with the sharing scale.

In [6], Ateniese et al. proposed leveraged proxy reencryptions to secure distributed storage. Specifically, the data owner encrypts blocks of content with unique and symmetric content keys, which are further encrypted under a master public key. For access control, the server uses proxy cryptography to directly reencrypt the appropriate content key(s) from the master public key to a granted user’s public key. Unfortunately, a collusion attack between the untrusted server and any revoked malicious user can be launched, which enables them to learn the decryption keys of all the encrypted blocks.

In [7], Lu et al. proposed a secure provenance scheme, which is built upon group signatures and ciphertext-policy attribute-based encryption techniques. Particularly, the system in their scheme is set with a single attribute. Each user obtains two keys after the registration: a group signature key and an attribute key. Thus, any user is able to encrypt a data file using attribute-based encryption and others in the group can decrypt the encrypted data using their attribute keys. Meanwhile, the user signs encrypted data with her group signature key for privacy preserving and traceability. However, user revocation is not supported in their scheme.

IV. THE PROPOSED SCHEME

To solve the challenges presented above, we propose Mods, a secure multi-owner data sharing scheme for dynamic groups in the cloud. The main contributions of this paper include:

1. We propose a secure multi-owner data sharing scheme. It implies that any user in the group can securely share data with others by the untrusted cloud.
2. Our proposed scheme is able to support dynamic groups efficiently. Specifically, new granted users can directly decrypt data files uploaded before their participation without contacting with data owners.
3. User revocation can be easily achieved through a novel revocation list without updating the secret keys of the remaining users. The size and computation overhead of encryption are constant and independent with the number of revoked users.
4. We provide secure and privacy-preserving access control to users, which guarantees any member in a group to anonymously utilize the cloud resource. Moreover, the real identities of data owners can be revealed by the group manager when disputes occur.

Advantages of Proposed System:

- Any user in the group can store and share data files with others by the cloud.
- The encryption complexity and size of ciphertexts are independent with the number of revoked users in the system.
- User revocation can be achieved without updating the private keys of the remaining users.
- A new user can directly decrypt the files stored in the cloud before his participation.

A. A THRESHOLD SIGNATURE

In this section, a new (t,n) threshold signature scheme is presented with the assistance of a mutually trusted center. The scheme consists of three phases: the system initiation phase, the threshold signature generation phase, and the threshold signature verification phase. We describe the three phases in details as follows:

[System Initialization Phase] The system contains a mutually trusted center, who is responsible for selecting all parameters. Assume that there are n members in a group, let A be the set of all group members. Any t or more members in the group can sign a message on behalf of the group, let B be any subset in A of size t. The mutually trusted center selects the following parameters:

- A number $N = p \cdot q = (2p'+1)\cdot(2q'+1)$, where p, q, p' and q' are distinct large primes.
- A generator g with order $v = p' \cdot q'$ in $Z_N{}^*$.
- A system public value e such that $\gcd(e, v) = 1$, where $e \cdot d = 1 \bmod v$ and d is a system secret value.
- A one-way hash function $h(\ )$.
- A secret polynomial function $f(x) = c_{t-1}x^{t-1} + \cdots + c_1 x + c_0 \bmod v$ with degree $t-1$, where $c_{n-1}, \cdots, c_1, c_0 \in Z_v{}^*$.
- A secret key x and a public key y, where $x = f(0) \bmod v$ and $y = g^x \bmod N$.

Thus, the mutually trusted center publishes e, y, N, g and $h(\ )$, and keeps d, x, v, p, q, p' and q' in secret.

For each group member U_i with a public value ID_i, for $i \in A$, the mutually trusted center computes U_i's secret key $x_i = (g^{f(ID_i)\cdot \ell_i})^d \bmod N$ and publishes his public key $y_i = g^{f(ID_i)\cdot \ell_i} \bmod N$, where $\ell_i = \prod_{j\in A, j\neq i} (ID_i - ID_j)^{-1} \bmod v$.

[Threshold signature generation phase] Without loss of generality, assume that there are t group members want to sign a message m on behalf of the group. The t group members can be denoted as $U_1, U_2, \ldots U_t$. The set of t group members is denoted as B. Each member U_i chooses a random number k_i and computes r_i as

$$r_i = g^{k_i \cdot e} \bmod N.$$

Thus, U_i makes r_i publicly available through a broadcast channel. After all r_i are available, each group member computes the product R as

$$R = \prod_{i\in B} r_i \bmod N.$$

Then, U_i uses his secret key x_i and the random number k_i to compute

$$s_i = (x_i)^{h(m,R)\cdot\prod_{j\in A, j\notin B}(ID_i - ID_j)\cdot\prod_{j\in B, j\neq i}(0-ID_j)} \cdot g^{k_i} \bmod N .$$

The user U_i sends $\{r_i, s_i\}$ to a designated clerk, who takes the responsibility of collecting the partial signatures. Besides, the clerk may authenticate the partial signatures by verifying the following equation

$$s_i^{\,e} = (y_i)^{h(m,R)\cdot\prod_{j\in A, j\notin B}(ID_i - ID_j)\cdot\prod_{j\in B, j\neq i}(0-ID_j)} \cdot r_i \bmod N ,$$

$$\text{for } i = 1,2,\ldots,t .$$

If the equation holds, the partial signature $\{r_i, s_i\}$ is valid.

[Threshold signature verification phase] Any verifier can use the group public key y to authenticate the validity of the group signature $\{R, S\}$ for the message m by checking the following equation

$$S^e \equiv y^{h(m,R)} \cdot R \bmod N .$$

If the equation holds, the group signature $\{R, S\}$ is valid.

B. DYNAMIC BROADCAST ENCRYPTION

Broadcast encryption [11] enables a broadcaster to transmit encrypted data to a set of users so that only a privileged subset of users can decrypt the data. Besides the above characteristics, dynamic broadcast encryption also allows the group manager to dynamically include new members while preserving previously computed information, i.e., user decryption keys need not be recomputed, the morphology and size of ciphertexts are unchanged and the group encryption key requires no modification. The first formal definition and construction of dynamic broadcast encryption are introduced based on the bilinear pairing technique, which will be used as the basis for file sharing in dynamic groups.

V. SYSTEM MODEL AND DESIGN GOALS

A. SYSTEM MODEL

We consider a cloud computing architecture by combining with an example that a company uses a cloud to enable its staffs in the same group or department to share files. The system model consists of three different entities: the cloud, a group manager (i.e., the company manager), and a large number of group members (i.e., the staffs) as illustrated in Fig .1.

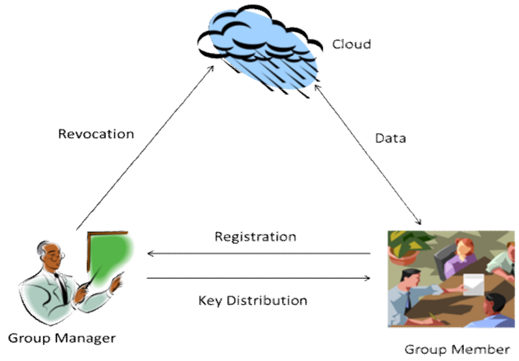


Fig .1 System Model

B. PROPOSED TECHNIQUES

My proposed system consists of following techniques:

Attribute Based Encryption: Proposed system use a Technique called Attribute Based Encryption (ABE) is a public key cryptography which allows Data Owner to encrypt and decrypt his files by using set of attributes along with private key. For each user a dedicated access tree structure will be defined by using data attributes. When relevant attributes provided by user that satisfies the access tree structure then, decrypted file can be downloaded.

System Setup: System initialization can be performed by forming a cloud architecture in which group manager creates an account with cloud server. Further, more users can join with to share files. This is possible through making a request to group manager. During registration process users need to fill their personal information which will be evaluated by group manager to provide an approval for data access in cloud. Once, user got registered with the cloud system, he is free to access any file until life time expiry or revocation on the basis of request. Initially, group manager collects attributes relevant to the data file units and are encrypted, then uploaded to cloud server. Policy engine used in the system automatically runs and generates access structure of the data file. Also, generates user's public key. Once the access structure satisfies the attributes given by the user the decrypted file can be downloaded by them.

User Registration: After successful creation of cloud setup, users need to get registered with the system through user registration process. While registering, users need to submit their personal details for completion of registration process. During registration process, user got unique identity and access structure. This generates secret key for the members. Data file can be encrypted by using Public Key to generate Cipher text.

User Revocation: User revocation is the process of removal of user from system user list which is performed by group manager. The system maintains Attribute History List (AHL) for each attributes. For the user to be revoked, his access structure is removed from AHL, so that they can't have more access to cloud.

File Upload: Before uploading files, group manager assign File identity to selected data files and then encrypts file using his public key. Along with encryption attributes for encryption is added.

File Access: Users can access data files if they have valid secret key. While accessing files, user's secret key is validated against access structure of the user. If it satisfies user's access structure, decrypted data file can be downloaded by Group Members.

File Deletion: This operation can be performed by Group Manager, if they no longer needed that files. For file deletion, Group Manager needs to provide File Identifier along with secret key. If owner's signature is verified successfully then cloud server successfully deletes the file with specified identity.

Dynamic Policy Updates*:* Group Manager can update their data attributes for a particular file whenever needed to achieve more security and integrity.

C. DESIGN GOALS

In this section, we describe the main design goals of the proposed scheme including access control, data confidentiality, anonymity and traceability, and efficiency as follows:

Access control: The requirement of access control is twofold. First, group members are able to use the cloud resource for data operations. Second, unauthorized users cannot access the cloud resource at any time, and revoked users will be incapable of using the cloud again once they are revoked.

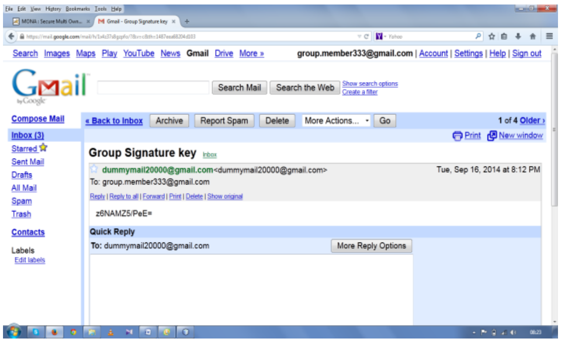
Data confidentiality: Data confidentiality requires that unauthorized users including the cloud are incapable of learning the content of the stored data. An important and challenging issue for data confidentiality is to maintain its availability for dynamic groups. Specifically, new users should decrypt the data stored in the cloud before their participation, and revoked users are unable to decrypt the data moved into the cloud after the revocation.

Anonymity and traceability: Anonymity guarantees that group members can access the cloud without revealing the real identity. Although anonymity represents an effective protection for user identity, it also poses a potential inside attack risk to the system. For example, an inside attacker may store and share a mendacious information to derive substantial benefit. Thus, to tackle the inside attack, the group manager should have the ability to reveal the real identities of data owners.

Efficiency: The efficiency is defined as follows: Any group member can store and share data files with others in the group by the cloud. User revocation can be achieved without involving the remaining users. That is, the remaining users do not need to update their private keys or reencryption operations. New granted users can learn all the content data files stored before his participation without contacting with the data owner.

VI. RESULTS

A Threshold Group Signature key will be sent to the corresponding User Email



File Edit View History Bookmarks Tools Help
MONA : Secure Multi Own...
Gmail - Group Signature key
Search Images Maps Play YouTube News Gmail Drive More »
group.member333@gmail.com | Account | Settings | Help | Sign out
Gmail by Google
Search Mail
Search the Web
Show search options
Create a filter
Compose Mail
Inbox (3)
Starred
Sent Mail
Drafts
All Mail
Spam
Trash
Contacts
Labels
Edit labels
« Back to Inbox
Archive
Report Spam
Delete
More Actions...
Go
1 of 4 Older ›
Print
New window
Reply | Reply to all | Forward | Print | Delete | Show original
More Reply Options

Group Signature key Inbox

dummymail20000@gmail.com<dummymail20000@gmail.com> Tue, Sep 16, 2014 at 8:12 PM

To: group.member333@gmail.com

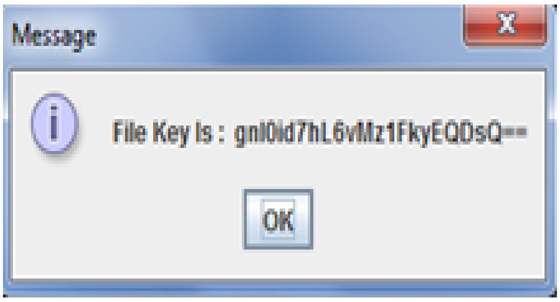
z6NAMZ5/PeE=

Quick Reply

To: dummymail20000@gmail.com

After Uploading File key will be generated:

Example:



Message

File Key Is : gnl0id7hL6vMz1FkyEQDsQ==

VI. CONCLUSION

In this paper, a secure data sharing scheme, Mods, for dynamic groups in an untrusted cloud is designed. In Mods, a user is able to share data with others in the group without revealing identity privacy to the cloud. Additionally, Modssupport efficient user revocation and new user joining. More specially, efficient user revocation can be achieved through a public revocation list without updating theprivate 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. Extensive analyses show that our proposed scheme satisfies the desired security requirements and guarantees efficiency as well.

REFERENCE

[1] 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.

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

[3] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-Grained Data Access Control in Cloud Computing," Proc. IEEE INFOCOM, pp. 534-542, 2010.

[4] M. Kallahalla, E. Riedel, R. Swaminathan, Q. Wang, and K. Fu, “Plutus: Scalable Secure File Sharing on Untrusted Storage,” Proc. USENIX Conf. File and Storage Technologies, pp. 29-42, 2003.

[5] 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.

[6] G. Ateniese, K. Fu, M. Green, and S.Hohenberger, “Improved Proxy Re-Encryption Schemes with Applications to Secure Distributed Storage,” Proc. Network and Distributed Systems SecuritySymp. (NDSS), pp. 29-43, 2005.

[7] R. Lu, X. Lin, X. Liang, and X. Shen, “Secure Provenance: The Essential of Bread and Butter of Data Forensics Cloud Computing,” pp. 282-292, 2010Proc. ACM Symp. Information, Computer and Comm. Security pp. 29-43, 2006

[8] 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.

[9] C. Wang, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,” Proc. IEEE INFOCOM, pp. 525-533, 2010.

[10] D. Chaum and E. van Heyst, “Group Signatures,” Proc. Int’l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), pp. 257-265, 1991.

[11] A. Fiat and M. Naor, “Broadcast Encryption,” Proc. Int’l Cryptology Conf. Advances in Cryptology (CRYPTO), pp. 480-491, 1993.

[12] L. Harn, "Group-oriented (t,n) threshold digital signature scheme and digital multisignature", IEE Proc. Computers and Digital Techniques, Vol. 141, No. 5, 1994, pp. 307-313.