International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611

A Secured Dynamic Group Sharing with V^2 Signature and Encryption Techniques in Public Cloud

Rajani Sajjan[1], Vijay Ghorpade[2], Nayantara Yerate[3]

[1]Computer Science & Engineering, VVPIET, Solapur, India

[2]D.Y.Patil College of Engineering & Technology, Kolhapur, India

[3]Computer Science & Engineering, VVPIET, Solapur, India

Abstract: ***Cloud systems can be used to enable data sharing capabilities and this can provide an abundant of benefits to the user. As group data sharing in public cloud computing has gained lots of popularity, providing privacy and security to the data shared has become top most priority. The cloud has a semi-trust kind and so it needs a security model which has no confidential data being exposed to cloud providers and attackers. Another important factor in providing privacy and security is periodic removal of unwanted files which if not done regularly then, may become a part of interest for attackers and can be misused. For this purpose, a secure group sharing framework for public cloud is proposed which combines V^2Signature Admin and Broadcast Encryption Techniques. By applying the proxy signature technique, the group leader can effectively grant the privilege of group management to one or more chosen group members. The shared data will have a predefined life span which will be specified by data owner. When the life span expires, the group admin is responsible to ask the members whether they need that file anymore and take action accordingly. A very high authentication is required to achieve identity privacy.***

Keywords: Identity privacy; V^2 Signature; life span.

1. Introduction

Cloud Computing has become more popular these days and is adopted widely and demand of outsourcing data has greatly increased. Cloud servers generally store data at very low cost and make it available 24 hours over internet. Cloud servers satisfy the need for data storage and high performance computation. As cloud servers are maintained and managed by a semi-trusted third party, it needs more secure methodologies for security. Whenever a user wants to share data, that data should be received only by the intended recipients and no one other than them. [1] The Cryptographic mechanisms are used to secure the data by encrypting them. This encrypted data is stored in the cloud. Authorized users can download the encrypted files and decrypt them with the given keys. Storing and sharing in dynamic environment dumps huge amount of data files in the cloud, which remains in the cloud server for indefinite time period. The confidential data stored may be misused by service providers. So security should be provided for these data by using some encryption techniques. To provide more security, the data owner must provide a time stamp with the file which specifies the time of sharing that particular file in the cloud. Once the life span is over, the cloud server should notify about the file to the group admin, who will negotiate with the group members and ask them whether they need that file anymore. If yes, the group admin then again set the time stamp factor of that file. If no, the group admin should remove the file from the cloud server. The major problem of adopting cloud servers is Identity Privacy. Many people may be do not adopt the cloud because, if user privacy is not maintained properly, then the actual identities of the group member can be disclosed easily to various kinds of intruders and cloud service providers. Therefore a high-level user authentication is needed for such systems.

2. Literature Review

2.1 Cryptographic Cloud Storage

A proposed [1] a security for customers to store and share their sensitive data in the cryptographic cloud storage. It provides a basic encryption and decryption for providing the security. The revocation operation is a sure performance killer in the cryptographic access control system. To optimize the revocation procedure, they present a new efficient revocation scheme which is efficient, secure, and unassisted. In this scheme, the original data are first divided into a number of slices, and then published to the cloud storage. When a revocation occurs, the data owner needs only to retrieve one slice, and re-encrypt and re-publish it. They have applied the efficient revocation scheme to the cipher text-policy attribute-based encryption based cryptographic cloud storage. The security analysis shows that the scheme is computationally secure.

2.2 Proxy re-encryption

In Yu et al.'s [1] scheme, an encrypted file can be decrypted by a user only if he/she has all of the file's attributes. By using proxy re-encryption, the computing complexity of digital envelope generation for a session key of a sharing file decreases to only at the data owner's side. For each one-time session key, the data owner needs to compute only one digital envelope by using his/her own public key. Based on the proxy re-encryption algorithm, cloud servers can compute

digital envelopes for all intended recipient. The efficiency of Yu et al.'s scheme relies on that there is high attribute variability between different files and high attribute variability between different users. But in group applications, different group members usually have same or similar interests, and they usually have attributes in common between them. In the scenario of interest based group sharing, if using Yu et al.'s scheme, the communication and computing overhead of user revocation will be dependent on the size of the group. The efficiency of the schemes depends on the assumption that cloud servers must be absolutely trusted. Otherwise, cloud servers can launch the collusion attack with some curious leaving group members. So, in order to protecting files from the prying eyes of curious cloud servers and leaving group members, the data owner needs to re-generate his key pairs and re-generate N _ 1 proxy-re-encryption keys when revoking a group member. This computing overhead is very high for the data owner, especially in the scenario of user joining and leaving frequently in the group.

3. Proposed System

Identity Privacy is a must thing in dynamic group sharing for public cloud computing. Users of the group should be provided with highly secure authentication. For this a two-way security system has to be used. First User will access his account by simply text password which he has set during registration. Once, the user clears first level, he enters the second level where he will be mailed to his e-mail id a *randomly generated* ***One-Time password*** which will be valid for that particular session. He cannot use that password during next authentication process.

Cloud servers have powerful computing and storing capability with group key maintaining process but cannot leak private information of the group including data, group members security parameter information and so on. Each member in the group can leave or apply to join the group at his/her will. There are two kinds of users Group Admin, Group Member. Group Admin will create the group, add members. The group members can then share data by encrypting the data and the keys and upload the file, and then the users who wants to read it will decrypt the file using symmetric and asymmetric algorithm. For all the above some techniques will be used those are **V^2 Signature Admin** and **Broadcast Encryption** techniques.

In dynamic environment sharing data actually dumps huge amount of data files in the server which remains in the cloud for indefinite period of time. The confidential information stored in the cloud server can be misused by service providers or attackers. As the cloud computing is semi-trusted third party, where the cloud providers may follow all the rules but will always be curious to know the secret information about the group and the data shared. So, for this purpose, another important factor to be focused on is to remove unwanted shared files. The data owner while sharing a file will use a *life span*, i.e. set a particular time period for that particular file which denotes the until what duration the file should be stored onto the server. Once the life span is over, the group admin should communicate with the group members and ask them whether they need the file anymore. If yes, add some more time to the file, else, delete the file from the server.

V^2 Signature Admin is system which verifies (i.e checks whether the data owner belongs to the same group) and validates (to allow the data owner to continue sharing the file) the data owner before sharing it among the group so that no one other than the group members will be allowed for sharing the data into the group.

The **Dynamic Broadcast Encryption** techniques enabling the group manager to dynamically add new user and at the same time preserves the previously computed information. That is, newly joining users can directly decrypt data files without contacting with data owners. So that there is no need to update user decryption keys. The **identity-based broadcast encryption** scheme that has many desirable property. It is fully collusion resistant, its cipher texts excluding the information to specified receiver set is short constant length, each private keys is short constant length, and the length of its public key is proportional to only the maximum number of receiver sets. Although the required computational costs are proportional to the square of the number of receivers when the set of receivers is newly determined, the computational costs that is required for each sender and each receiver are small constants as long as messages are encrypted for the same set of receivers as previous one. When some receivers are added or removed from the set of receivers, the computational costs required for each sender and each receiver are proportional to the product of the number of receivers in this set and that of added and removed receivers.

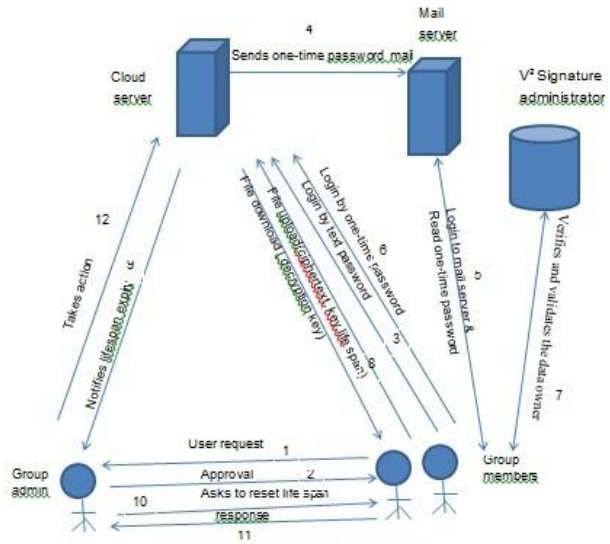

Figure 1: Scenario of Cloud Data Sharing Scheme

The different system model entities:

a) Cloud Sever

Cloud is the large repository of resources. Cloud is responsible for storing all user's data and granting access to the file within a group to other group members based on publically available revocation list which is maintained by Group Admin. We assume that the cloud server is honest but curious. That is, the cloud server will not maliciously delete

or modify user data, due to the protection of data auditing schemes

b) Group Manager
The Group Manager (admin) is acted by the administrator of the company. Therefore we assume that the Group Admin is fully trusted by the other parties Group Admin perform various operations such as system parameters generation, user registration, group creation, assign group signature, generation of private key and assign to the requested user, maintain revocation list and migrate this list into cloud for public use.

c) Group Member
Group members are a collection of registered users that will store their private data into the cloud server and share them with others in the group. Both Group Admin and group member can login using their login details. After successful login, Group Admin activates newly added members of the cloud by generating keys for each member and send it to the corresponding group members. He can also check the group details. After successful login, Group Members signature is verified. After successful verification, the member can upload, download and can modify the files. Group member must be encrypting data files before uploading to the cloud. The Group Members account can be revoked after he leaves the cloud by the Group Admin.

d) File Upload
File upload is the process of storing specified data files into the cloud for sharing in the group. Uploaded files remains in the cloud up to the time specified while uploading the file. Before uploading the file, file has to be encrypted to ensure security and privacy of the files. Then it is encapsulated with corresponding decryption key and *life span (ls)* value for the file and send it to cloud.

e) 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. User registered with their details such as identity (user name, mobile no and email-id). During registration process, user got unique identity and access structure. This generates secret key for the members. For registered users they will obtain private key, that private key is used for file encryption and decryption.

f) User Authentication
The user can login successfully only if user id and password are entered correctly. The login is a failure if the incorrect user id or wrong password is entered by the user. This helps in preventing unauthorized access.

g) Key Distribution
Means of distributing secret keys by the Group Admin that is valid only if the group members are not revoked from the group.

h) User Revocation
User revocation is the process of removal of user from system user list which is performed by group admin. Group admin can directly revoke multiple users through public revocation list at any time without affecting any non-revoked user. If the login credentials of the specified user matches with the details of revocation list then access is denied.

i) File Download
To access the data that are stored in the cloud, group member will give request as group id, data id. Cloud server will verify their signature, if the group member in the same group then allow to access file. Group member have rights to access data, but not having rights to delete or modify the data that are stored in the cloud.

j) File Deletion
Group Admin deletes the file when the time specified for that particular file is expired.

4. Equations

V^2 Signature Admin: Let A be an original signer (group manager) who has an authentic key pair (PrK_A and PuK_A), and B be a V^2 Signature Admin who has an authentic key pair (PrK_B and PuK_B). Let m_w be A's warrant information for the delegation, which has semantic means including the original signer's identity, some information about the V^2 signature admin (for example the identity), period of delegation validity, the qualification of messages on which the signature admin can sign, etc. Let $\delta_A = Sign(PrK_A, m_w)$ be A's signature on the warrant m_w using his/her private key PrK_A. A transmits δ_A to B. This scheme is described as follows:

- (Signature key generation) SKG is a signature key generating algorithm that takes original signer's signature δ_A and signature admin's private key PrK_B as inputs, and outputs a signature admin's key pair ($PPrK_B$, $PPuK_B$). It is executed by the signature admin:

$$(PPrK_B, PPuK_B) \leftarrow SKG(\delta_A, PrK_B). \quad (1)$$

- (Sign) Sign is a signing algorithm that takes signature admin's private key $PPrK_B$ and message m as inputs, and outputs proxy signature δ_P. It is executed by the signature admin :

$$\delta_P \leftarrow Sjgn(m\ PPrK_B). \quad (2)$$

- (Signature verifying) SV is a signature verifying algorithm that takes (δ_P , m, m_w, PuK_A, PuK_B) as inputs, and outputs either accept or reject.

It is executed by any signature admin:

$$PSV(\delta_P, m, m_w, PuK_A, PuK_B) = \text{accept or reject.} \quad (3)$$

DynamicBroadcastEncryption:
DBE=(Setup,Join,Encrypt,Decrypt).
Setup(λ):Takes as input the security parameter λ and outputs a manager key mk and an initial group encryption key ek. The group manager is given **mk** and **ek** is made public.
Join(mk,i): Takes as input the manager key mk and a user counter i. Join generates a user label $\mathbf{lab_i}$ abd a user

decryption key $\mathbf{dk_i}$. The user label lab_i is added to the group encryption key ek:=ek∪{labi} and the user decryption key dki is sent to the i-th user securely. We denote by n the total number of users (evolving over time) and by U = {1, . . . , n} the set of all users.

Encrypt(ek,R,ls): Takes as input the group encryption key ek and a set of revoked users R ⊆ U and outputs a random pair (hdr,K). When a message M ∈ {0, 1}∗ is to be broadcast to users in U \ R, the broadcaster generates (hdr,K) ← Encrypt(ek,R,ls), computes the encryption CM of M under the symmetric key K and broadcasts (hdr,R,CM). We will refer to hdr as the header or broadcast ciphertext, (hdr,R,ls) as the full header, K as the message encryption key, CM as the broadcast body and ls is the lifespan of the file.

Decrypt(dki,R, hdr): Takes as input a header hdr, a subset R ⊆ U and a user de-cryption key dki. If i ∈ U \R, the algorithm outputs the message encryption key K which is then used to decrypt the broadcast body CM and recover M.

References

[1] Kaiping Xue, Peilin Hong, "A Dynamic Secure Group Sharing Framework in Public Cloud Computing," IEEE Transactions on Cloud Computing, vol2, No.4, Oct-Dec 2014
[2] Delerablee,Paillier,Pointcheval, "Fully Collusion Secure Dynamic Broadcast Encryption with Constant-Size Ciphertexts or Decryption Keys," Springer Proc. 1st Int. Conf. on Pairing-based Cryptography, 2007, pp. 39-59
[3] Dan Boneh, Craig Gentry, and Brent Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In Victor Shoup editor, CRYPTO 2005, volume 3621 of LNCS,pages 258–275, Santa Barbara, CA, USA, August 14–18, 2005. Springer-Verlag, Berlin, Germany.
[4] P. Tysowski and M. Hasan, "Hybrid attribute-and re-encryption based key management for secure and scalable mobile applications in clouds," IEEE Trans. Cloud Comput., vol. 1, no. 2, pp. 172–186, Jul.–Dec. 2013.
[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. 13th ACM Conf. Comput. Commun. Security, 2006, pp. 89–98.
[6] M. Blaze, G. Bleumer, and M. Strauss, "Divertible protocols and atomic proxy cryptography," in Proc. Int. Conf. Theory Appl. Cryptographic Techn., 1998, pp. 127–44.
[7] P. Lee, J. Lui, and D. Yau, "Distributed collaborative key agreement and authentication protocols for dynamic peer groups," IEEE/ACM Trans. Netw., vol. 14, no. 2, pp. 263–276, Apr. 2006.
[8] Y. Amir, Y. Kim, C. Nita-Rotaru, and G. Tsudik, "On the performance of group key agreement protocols," ACM Trans. Inf. Syst. Security, vol. 7, no. 3, pp. 457–488, 2004.
[9] W. Yu, Y. Sun, and K. Liu, "Optimizing the rekeying cost for contributory group key agreement schemes," IEEE Trans. Dependable Secure Comput., vol. 4, no. 3, pp. 228–242, Jul.–Sep. 2007.
[10] M. G. G. Ateniese, K. Fu, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," ACM Trans. Inf. Syst. Security, vol. 9, no. 1, pp. 1–30, 2006.

Author Profile

Ms. Rajani S. Sajjan completed B.E.(Computer Science & Engineering) from Walchand College of Engineering from Sangli in 1999. Completed M.Tech(Computer Science & Engineering) from PDA College of Engineering, Gulbarga. Pursuing Ph.D. in Computer Science & Engineering from Shivaji University, Kolhapur.

Dr.Vijay R. Ghorpade completed Ph.D. from STRM University, Nanded. Specialized in Mobile Ad-hoc networks, Data Mining & Cloud Computing.

Ms.Nayantara S. Yerate completed B.E. in Information Technology from Solapur University and pursuing M.E. in Computer Science & Engineering from Solapur University.