© 2017 IJSRST | Volume 3 | Issue 6 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X
Themed Section: Science and Technology

A Secure Anti-Collusion File Sharing System for Untrusted Cloud Storage

Kavitha G[1], Latchoumy P[2], Afreen Banu H[3]

[1]Department of Information Technology, B.S.A Crescent University, Chennai, Tamilnadu, India
[2]Department of Information Technology, B.S.A Crescent University, Chennai, Tamilnadu, India
[3]UG Student, Department of Information Technology, B.S.A Crescent University, Chennai, Tamilnadu, India

ABSTRACT

In recent years, cloud has emerged as a collaborative computing platform and it is essential to share the data among multiple users of a group. As organizations outsource the storage of data files on the cloud, security guarantees becomes a main constraint. As the data is shared among collaborative members of the group, preserving the privacy of data is a challenging task due to frequent changes in group membership. Hence, in an untrusted cloud environment, it is necessary to protect the data against collusion attack and avoid the revoked user of the group to access original data. In practice, it is difficult for the group manager to update all the files with a new key whenever a user is added or removed from the group. As the updated files are stored in the cloud storage, it leads to duplication of files. In this paper, a secure anti-collusion file sharing system is developed to protect the data against collusion attack and a deduplication scheme is presented to avoid duplication of files and utilize the cloud storage efficiently. In this work, the key distribution among dynamic group members is done without a secure communication channel. Any authorized user can get the individual key from group manager and after user's private key gets activated, the user is permitted to access the files of the group. In the proposed scheme, the revoked users are not permitted to access the files present in the cloud.

Keywords: Dynamic Group, File Sharing, Cloud Storage, Data Access, Data Security, Duplication.

I. INTRODUCTION

Cloud computing provides on demand access to shared computing resources and data with minimal effort and low cost [8]. In the digital era, it tends to be less expensive to store data in the cloud rather to invest in the infrastructure and maintain it day by day. Various cloud storage providers store and process the data either in a privately owned or a third-party data center that may be located far from the user, ranging in distance from across a city to across the world. There are a number of security threats associated with cloud data services including traditional security threats like network eavesdropping, illegal invasion, denial of service attacks and specific threats associated with cloud like side channel attacks, virtualization vulnerabilities, and abuse of cloud services.

Many organizations have a considerable concern on migrating to cloud storage as there is a lack of physical control over stored data [9]. To preserve the privacy of data, the user's data can be encrypted and then stored in the cloud. But, Cloud provides a dynamic collaborative environment and has intrinsic security challenges like who can access the data and what data files can be accessed because new members may join the group and some members can leave the group. Cloud file sharing provides end users with the ability to access files with any Internet-capable device from any location. Usually, the user or the group manager has the ability to grant access privileges to other users if they belong to the same group [1]. In this paper, we have concentrated on the collision attack that is possible to occur in dynamic collaborative groups and proposed a method for secure file sharing across a group and avoid duplication of files in cloud storage. When the user leaves the group or joins another group, the private information issued to existing members remains unaffected and only the public information is updated to change the group key. Hence, this paper addresses the issue of data sharing without disclosing confidential information to unauthorized entities while retaining the ability to provide controlled access to shared data.

The rest of the paper is organized as follows. Section 2 describes the literature work related to the data sharing and access control in collaborative computing environments. Section 3 presents the design of anti-collusion file sharing framework. The implementation and results are discussed in section 4. Finally, section 5 concludes the paper and provides directions for future work.

II. LITERATURE REVIEW

An attribute based system for securing and managing group keys is presented in [2]. The user can derive the group keys if the attributes satisfy the access control policy of the file. The revoked member of the group could not derive the group key individually or by pooling the attributes of other users. But, for secure collaborative applications, it is difficult for the current group key management techniques to manage and distribute the group keys. This attribute based system is not flexible and cannot provide fine grain access control among a group of users.

X.Liu et. al have presented a secure multi owner data sharing scheme in cloud [3]. As there is a frequent change in the membership of group, preserving data and identity privacy from an untrusted cloud is a challenging task. A cloud user can anonymously share the data with the other members by implementing group signature and dynamic broadcast encryption techniques. In this scheme, the computation cost of clients for file access operation increases linearly with the number of revoked users. Also, there is an extra storage overhead in the cloud as the files are encrypted using symmetrical encryption.

C.Piechotta et.al discusses about ensuring detailed data access control in the cloud [4]. As the data is migrated to the cloud, there is a need to focus on securing data in an untrusted cloud environment. A hybrid encryption technique that combines both symmetric and asymmetric encryption technique is applied to encrypt the shared data. In this method, every cloud user can have multiple signing keys if they have been added, revoked or reinstated. These multiple signing keys are kept at the cloud provider side and the user may not be aware of this and believe that only a single key exists for a user. Further, additional re-encryptions are required to allow users in the lower tier hierarchy to decrypt the data in a collaborative environment. This introduces additional computational overhead and requires much effort to revoke users from collaborative environment.

A group key management scheme for secure group communication and differentiated access control is presented by X.Zou et.al [5]. With multiple participating entities, there is a problem of insider threats in which the malicious parties inside the organizations access the shared data and abuse, tamper or damage the data stored in the cloud. To provide flexible access control with fine tuned granularity, an access control polynomial is constructed whenever there is a need to distribute a secret key to the specific user group. The access control polynomial function is constructed every time a user joins or leaves a group. There is a change in the group key and encryption is done with the new key to prevent the revoked users from accessing shared data. But, computing the group key and communicating it to the group members for each join or leave operation of users is very difficult and leads to an increase in the storage cost.

A multi authority access control system for enforcing fine grained access control to data stored in the cloud is proposed by Qi Li et.al [6]. The multi authority cipher text policy ABE implements forward secrecy to the revoked users and backward secrecy to the new users. A unique global identifier is issued to each user to avoid collusion attacks from unauthorized users. Any revoked user cannot decrypt the cipher text until it possesses the required attributes specified by access structure. But, in this method multiple unauthorized users may cooperate with each other and decrypt the cipher text by combining their attributes which is not possible with one unauthorized user.

The current work proposes a secure anti-collusion file sharing framework to overcome the above stated problems that exist in data sharing among group members. In this method, the users obtain the private key from the group manager without any secure communication channel. It provides a fine grain access control as any authorized user can access the file from the cloud and the revoked users are denied file access. The group keys are not recomputed whenever there is a change in the members of the group. Instead, the private key for the new user is generated and sent only to the

new user. This saves considerable amount of time spent in the dissemination of group keys. The collusion attacks are avoided in this proposed scheme and the de duplication technique reduces the storage cost for data owners.

III. ANTI-COLLUSION FILE SHARING FRAMEWORK

The cloud provides storage space to users from multiple service providers. The data maintained on the cloud must be protected and safe. To ensure data privacy, the data is encrypted at the user premises and stored in the cloud. But, there is a possibility of insider attacks on the stored data.

This effect can be adverse when the data is shared among the group members of a dynamic group. Secure data sharing among dynamic group members have to be established by avoiding collusion attacks. In this work, an anti-collusion file sharing framework is presented to preserve the data privacy among the members of the group. The user can securely obtain the private key from the group manager when he joins the group and the revoked user cannot access the cloud when he leaves the group.

The anti-collusion file sharing framework is depicted in Fig.1. The group manager is responsible for generation of private keys for the data files, user registration and user revocation. Initially, the user sends a request to the group manager to join the group. This request is validated by the group manager and the group key is sent to the user. The user can access the files stored in the cloud by using the group key. The files are encrypted using Advanced Encryption Standard (AES) algorithm and uploaded into the cloud. Similarly, the user with a valid group key can download the files from cloud and decrypt it. The assigned group key will be valid for accessing the data files until the user leaves the group or being marked as revoked user.

The group membership is subject to dynamic change due to new user registration and user revocation. Hence, to preserve the privacy of data stored in untrusted cloud, the members of the group will request for private key for each file to be accessed in addition to group key. The user request is validated by the group manager and the private keys will be generated for each file for an individual group member. Hence, secure file sharing is implemented by using group key and private key for accessing files and collusion attacks are avoided.

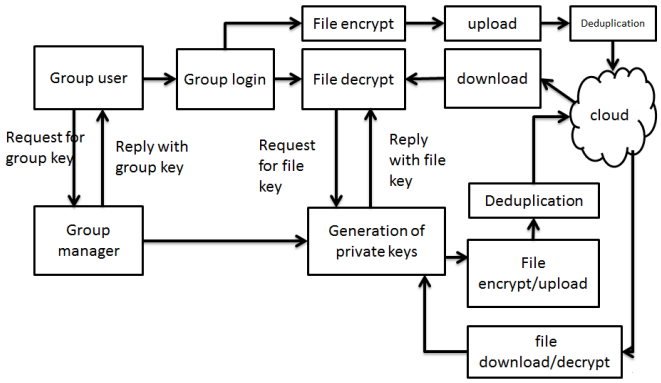

Figure 1. Anti-collusion File Sharing Framework

The cloud storage space is managed efficiently by implementing the deduplication technique. Deduplication is a technique that eliminates redundant data by keeping only one physical copy and referring the others as redundant data to that copy. Deduplication can take place at either the file level or the block level. For file level deduplication, it eliminates duplicate copies of the same file. If the file exists in the cloud storage, then it will not be uploaded. When a file is to be uploaded into cloud, it is checked for duplication by using file level deduplication technique. The File level deduplication method compares the name, size, type and date modified of every file stored in cloud and if a match is found, then they are treated as duplicate files. Hence, the deduplication technique avoids multiple users of the group to upload the same data file in multiple copies.

Deduplication Algorithm:

Step 1: Files are indexed by node and device, files with the same inode+device are considered equal. If the platform does not support inode ids, then this check is skipped.

Step 2: Files are then indexed by size, only file with same size are compared.

Step 3: During comparison, the files are read at block sizes increasing in powers of two, starting with 2k.The blocks are hashed and compared and if they do not match the comparison it is stopped early. If all hashes are equal then the files are considered to be equal.

Step 4: Hashes are only computed when needed and cached in memory. Since the hash block size increase in powers of two, only few dozen hashes are needed even for larger files.

Step 5: Non files always return false.

IV. RESULTS

The secure anti-collusion file sharing system is implemented using Net Beans as front end and SQL YOG as back end. The performance of the proposed key distribution algorithm and deduplication algorithm is analyzed in an untrusted cloud storage system. The computation cost of group member for file upload and file download for a file size of 10MB is calculated and compared with the existing Role Based Access Control Scheme (RBAC) [7]. The performance of the proposed Anti-collision File Sharing Scheme (AFS) for file upload is depicted in fig.2 and file download is shown in Figure 3.

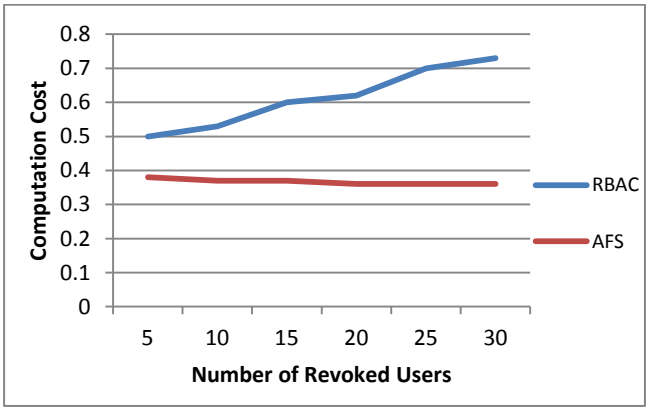

Figure 2. Computation cost of group member for file upload

In the existing RBAC scheme, when the number of revoked users is increasing, the computation cost for uploading a file for a group member has also increased. But, in the proposed AFS scheme, the computation cost is not dependent on the number of revoked users as the user revocation operation is done by the group manager and the group member will encrypt only their data files without considering the information of other users. Hence, the AFS scheme achieves a minimum computation cost compared to RBAC even when there is an increase in the number of revoked users.

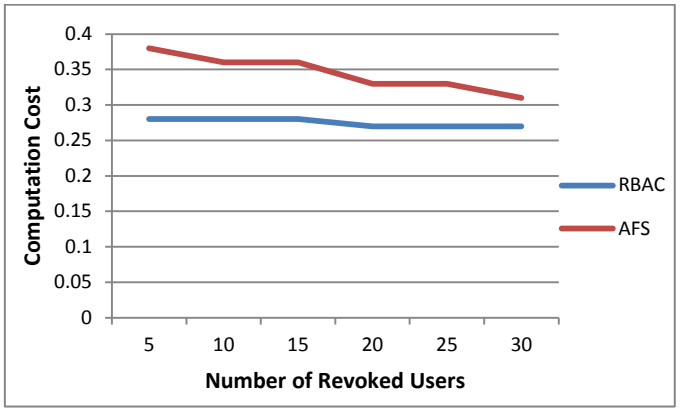

Figure 3. Computation cost of group member for file download

The computation cost for file download of a group member in the existing RBAC scheme is minimum and almost the same because the computation cost is irrelevant to the number of revoked users in the cloud. The decryption operation of the data files is the same for a group member even if there is an increase in the number of revoked users in the cloud. In the proposed AFS scheme, the computation cost decreases with the number of revoked users because the computation of the recovery of secret key decreases with the number of revoked users.

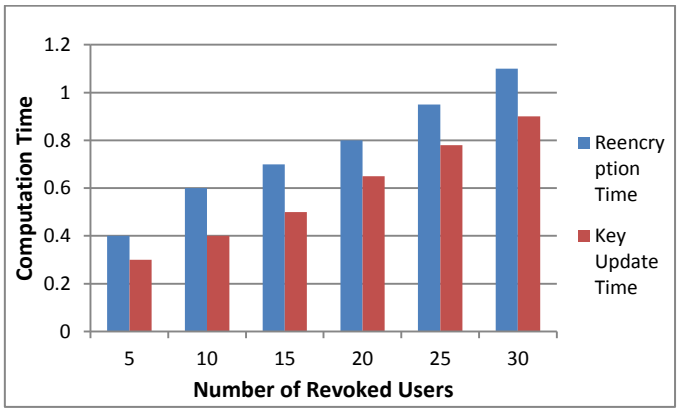

Figure 4. Re-encryption and Key Update Time

The computation cost of re encryption time and key update time for the data files is shown in Figure 4. In AFS, the collusion attack is avoided because the revoked users of the group cannot access the data files. Each user in the group is given a private key for a data file and if the user is revoked by the group manager, the private key is deactivated. The re-encryption time increases linearly with the number of revoked users because the new members of the group will encrypt the data files using the key given by the group manager. The

key update time by the group manager for the data files shared among dynamic group members in an untrusted cloud is depicted in figure 4. The keys need not be updated for each member of the group if one user is revoked from that group. The new key is to be generated only for the new members who join the group and the existing group member's key are not modified.

V. CONCLUSION

A secure anti-collusion file sharing framework is presented in this work. The proposed work enables secure file sharing among dynamic group members working in a collaborative project. The proposed AFS system protects the data files stored in an untrusted cloud platform both from insider, external and collusion attacks. As the user is revoked from the group, the key management procedure adopted by the group manager is simple and effective. The user who joins the group will securely obtain their private keys from the group manager. The computation cost of group members for file upload and file download are better compared to RBAC scheme. When some of the existing group members are revoked and new members join the group, the AFS scheme manages dynamic groups efficiently by computing the private keys only for the new users and the private keys for the other group members are not recomputed again. Hence, in an untrusted cloud platform, the revoked users are not allowed to access the data files even if they cooperate with other group members for accessing the original data files stored in the cloud. The storage space of the cloud is efficiently managed as the duplicate files are identified using the de-duplication technique and not uploaded in the cloud storage and thereby redundant copy of the files are avoided.

VI. REFERENCES

[1] Zhongma Zhu and Rui Jiang, "A Secure Anticollusion data sharing scheme for dynamic groups in cloud," IEEE Trans. Parallel Distribution System, vol. 27, no. 1, pp. 1182–1191, Jan. 2016.

[2] M. Nabeel, E. Bertino, "Attribute based group key management" Transactions on Data Privacy, vol.7, no.3,pp. 309-336, Dec.2014.

[3] X. Liu, Y. Zhang, B. Wang, and J. Yang, "Mona: Secure multiowner data sharing for dynamic groups in the cloud," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 6, pp. 1182–1191, Jun. 2013.

[4] C. Piechotta, M. G.Olsen, A.E.Jensen, J.W. Coleman and P.G.Larsen, "A secure dynamic collaboration environment in a cloud context" , Future Generation Computer Systems, Vol. 55, pp.165–175, 2016.

[5] X. Zou, Y.-S. Dai, and E. Bertino, "A practical and flexible key management mechanism for trusted collaborative computing," in Proc. IEEE Conf. Comput. Commun., 2008, pp. 1211–1219.

[6] Qi Li , Jianfeng Ma, Rui Li, Ximeng Liu, Jinbo Xiong and Danwei Chen, "Secure, efficient and revocable multiauthority access control system in cloud storage", Computers & Security, vol.59, pp.45-59, 2016.

[7] L. Zhou, V. Varadharajan, and M. Hitchens, "Achieving secure role-based access control on encrypted data in cloud storage," IEEE Trans. Inf. Forensics Security, vol. 8, no. 12, pp. 1947–1960, Dec. 2013.

[8] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proc. ACM Symp. Inf., Comput. Commun. Security, 2010, pp. 282–292.

[9] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Commun. ACM, vol. 53, no. 4, pp. 50–58, Apr. 2010.