GROUP USER REVOCATION FOR SHARED DYNAMIC DATA ON AUDITING VIA CLOUD STORAGE

Roselinmary.A
Student,
Computer Science and Engineering,
Velammal Institute of Technology,
Chennai, India

Saranya.R
Student ,
Computer Science and Engineering,
Velammal Institute of Technology ,
Chennai,India

Jeevitha.B
Student,
Computer Science and Engineering,
Velammal Institute of Technology,
Chennai,India

Dr.Sugumar.R
Professer,
Computer Science and Engineering,
Velammal Institute of Technology,
Chennai,India

Abstract: Data auditing is the process of conducting a data review to measure how company's data is fit for agreed function. This engages profiling of data and assesses the collision of pitiable quality data on the organization's performance and profits. The systems are still not secure beside the collusion of cloud storage server as well as revoked group users during user revocation in practical cloud storage system. In this paper , we found out that the collusion attack in the exiting scheme .An efficient public integrity auditing scheme with secure group user revocation based on vector commitment plus verifier-local revocation group signature. We are providing each and every group user signature is unique and tracking user`s activities via session tracking methodology. If the users misuse the group signature mark as a hacker then intimate to cloud admin. Finally, the security and experimental analysis show that compared with its relevant schemes our scheme is also secure and efficient.

Keywords:*collusionattact,revocation,group,auditing,uniquesignature,tracking.*

I. INTRODUCTION

The cloud computing motivates enterprises and organizations to outsource their data to third-party cloud service providers (CSPs), which will improve the storage limitation of resource constrain local devices. Recently, some commercial cloud storage services, such as the simple storage servic on-line data backup services of Amazon and some practical cloud based software Google Drive, Dropbox, Mozy, Bitcasa , and Memopal, have been built for cloud application. Since the cloud servers may return an invalid result in some cases, such as server hardware/software failure, human maintenance and malicious attack, new forms of assurance of data integrity and accessibility are required to protect the security and privacy of cloud user's data. To overcome the above critical security challenge of today's cloud storage services, simple replication and protocols like Rabin's data dispersion scheme are far from practical application. The formers are not practical because a recent IDC report suggests that data-generation is outpacing storage availability. The later protocols ensure the availability of data when a quorum of repositories, such as k-out-of-n of shared data, is given. However, they do not provide assurances about the availability of each repositories, which will limit the assurance that the protocols can provide to relying parties.

Recently, the development of cloud computing boosted some applications where the cloud service is used as a collaboration platform. In these software development environments, multiple users in a group need to share the source code, and they need to access, modify, compile and run the shared source code at any time and place. The new cooperation network model in cloud makes the remote data auditing schemes become infeasible, where only the data owner can update its data. Designed a scheme based on proxy re-signatures. However, the scheme assumed that the private and authenticated channels exist between each pare of entities and there is no collusion among them.

The deficiency of above schemes motivates us to explore how to design an efficient and reliable scheme, while achieving secure group user revocation. To the end, we propose a construction which not only supports group data encryption and decryption during the data modification processing, but also realizes efficient and secure user revocation.

Our idea is to apply vector commitment scheme over the database. Then we leverage the Asymmetric Group Key Agreement (AGKA) and group signatures to support cipher text data base update among group users and efficient group user revocation respectively. Specifically, the group user use the AGKA protocol to encrypt/decrypt the share database, which will guarantee that a user in the group will be able to encrypt/decrypt a message from any other group users. The group signature will prevent the collusion of cloud and revoked group users, where the data owner will take part in the user revocation phase and the cloud could not revoke the data that last modified by the revoked user.

II.LITERATURE SURVEY

1.G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proc. of ACM CCS, Virginia, USA, Oct. 2007, pp. 598–609.

We introduce a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage systems. We present two provably-secure PDP schemes that are more efficient than previous solutions, even when compared with schemes that achieve weaker guarantees. In particular, the overhead at the server is low (or even constant), as opposed to linear in the size of the data. Experiments using our implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation.

2.K. D. Bowers, A. Juels, and A. Oprea, "Proofs of retrievability: theory and implementation," in Proc. of CCSW 2009, llinois, USA, Nov. 2009, pp. 43–54.

In a proof-of-retrievability system, a data storage center must prove to a verifier that he is actually storing all of a client's data. The central challenge is to build systems that are both efficient and*provably* secure—that is, it should be possible to extract the client's data from any prover that passes a verification check. In this paper, we give the first proof-of-retrievability schemes with full proofs of security against arbitrary adversaries in the strongest model, that of Juels and Kaliski.

Our first scheme, built from BLS signatures and secure in the random oracle model, features a proof-of-retrievability protocol in which the client's query and server's response are both extremely short. This scheme allows public verifiability: anyone can act as a verifier, not just the file owner. Our second scheme, which builds on pseudorandom functions (PRFs) and is secure in the standard model, allows only private verification. It features a proof-of-retrievability protocol with an even shorter server's response than our first scheme, but the client's query is long. Both schemes rely on homomorphic properties to aggregate a proof into one small authenticator value.

3.B. Libert, T. Peters, and M. Yung, "Scalable group signatures with revocation," in Proc. of EUROCRYPT 2012, CA, USA, Aug. 2002, pp. 61–76.

A group signature scheme allows any group member to sign on behalf of the group in an anonymous and unlinkable fashion. In the event of a dispute, a designated trusted entity can reveal the identity of the signer. Group signatures are claimed to have many useful applications such as voting and electronic cash. A number of group signature schemes have been proposed to-date. However, in order for the whole group signature concept to become practical and credible, the problem of secure and efficient group member revocation must be addressed. In this paper, we construct a new revocation method for group signatures based on the signature scheme by Ateniese et al. [ACJT]. This new method represents an advance in the state-of-the-art since the only revocation schemes proposed thus far are either: 1) based on implicit revocation and the use of fixed time periods, or 2) require the signature size to be linear in the number of revoked members. Our method, in contrast, does not rely on time periods, offers constant-length signatures and constant work for the signer.

4.W. Susilo "Asymmetric group key agreement," in *Proc. of EUROCRYPT Cologne, Germany, Apr. 2009*.

A group key agreement (GKA) protocol allows a set of users to establish a common secret via open networks. Observing that a major goal of GKAs for most applications is to establish a confidential channel among group members, we revisit the group key agreement definition and distinguish the conventional (*symmetric*) group key agreement from *asymmetric group key agreement* (ASGKA) protocols. Instead of a common secret key, only a shared encryption key is negotiated in an ASGKA protocol. This encryption key is accessible to attackers and corresponds to different decryption keys, each of which is only computable by one group member.

III. EXISITING SYSTEM

In these software development environments, multiple users in a group need to share the source code, and they need to access, modify, compile and run the shared source code at any time and place. In the scheme, the user revocation problem is not considered and the auditing cost is linear to the group size and data size. To further enhance the previous scheme and support group user revocation designed a scheme based on proxy re-signatures. However, the scheme assumed that the private and authenticated channels exist between each pair of entities and there is no collusion among them the auditing cost of the scheme is linear to the group size.

DISADVANTAGES OF EXISTING SYSTEM:

1.The user revocation problem is not considered and the auditing cost is linear to the group size and data size.
2.Their scheme could efficiently support plaintext data update and integrity auditing, while not cipher text data.
3.Intheir scheme, if the data owner trivially shares a group key among the group users, the defection or revocation any group user will force the group users to update their shared key.
4.Also, the data owner does not take part in the user revocation phase, where the cloud itself could conduct the user revocation phase.
5.In this case, the collusion of revoked user and the cloud server will give chance to malicious cloud server where the cloud server could update the data as many time as designed and provide a legal data finally.

IV.PROPOSED SYSTEM

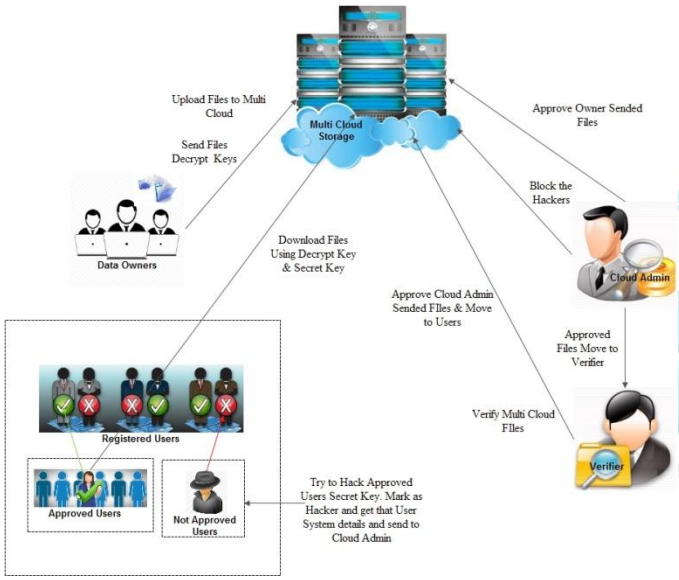
We propose a construction which not only supports group data encryption and decryption during the data modification processing, but also realizes efficient and secure user revocation. Our idea is to apply vector commitment scheme over the database. Then we leverage the Asymmetric GroupKey Agreement (AGKA) and group signatures to support ciphertext data base update among group users and efficient group user revocation respectively.Specifically, the group user use the AGKA protocol to encrypt/decrypt the

share database, which will guarantee that a user in the group will be able to encrypt/decrypt a message from any other group users. The group signature will prevent the collusion of cloud and revoked group users, where the data owner will take part in the user revocation phase and the cloud could not revoke the data that last modified by the revoked user.

ADVANTAGES OF PROPOSED SYSTEM:

1.We explore on the secure and efficient shared data integrate auditing for multi-user operation for ciphertext database.
2.By incorporating the primitives of victor commitment,asymmetric group key agreement and group signature, we propose an efficient data auditing scheme while at the same time providing some new features, such as traceability and countability.

V.ARCHITECTURE DIAGRAM



VI.MODULES DESCRIPTION

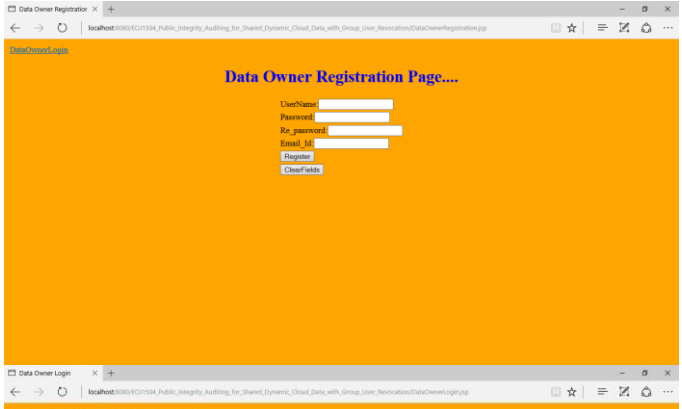
The project contains Four modules,

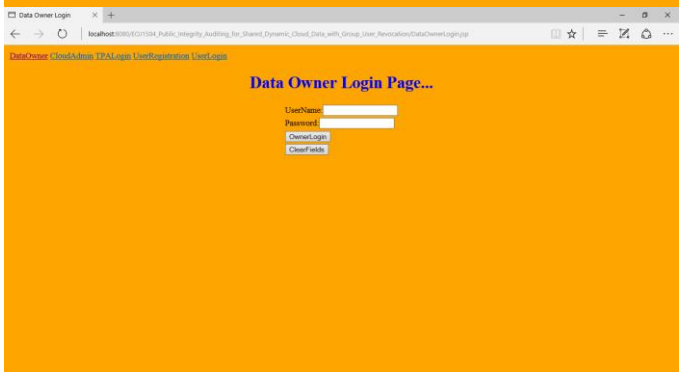
Data Owner
Third Party Auditor
Cloud Admin
Users

VII.IMPLEMENTATION:

Data Owner

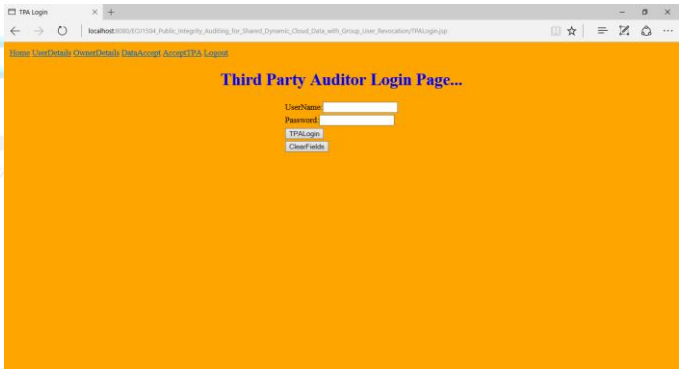
Data Owner uploads files to group users via Cloud Storage like Drop box. If Owner Upload a file in background that the file will be encrypted using Blowfish Cryptography and validation the Time based one time password and enter in to cloud storage.





Third Party Auditor:

Third Party Auditor verify/audits all the uploaded owner files in cloud storage and sends into the cloud admin approval. If admin approve the file after he will send the files to group users.

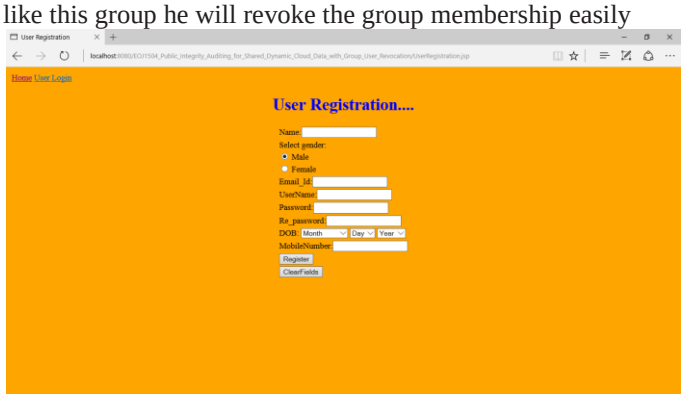


Cloud Admin:

Cloud Admin Verify the Auditor approve files after grant the final approval and sends that files to third party auditor and approve users and track hackers after block the hackers in our application.

Users:

Users register first, after login he must join any group. Each and Every Group Signature is Unique within the same group. User sign in the group login after download all files and not like this group he will revoke the group membership easily



VIII.CONCLUSION

The primitive of verifiable database with efficient updates is an important way to solve the problem of verifiable outsourcing of storage. We propose a scheme to realize efficient and secure data integrity auditing for share dynamic data with multi-user modification. The scheme vector commitment, Asymmetric Group Key Agreement (AGKA) and group signatures with user revocation are adopt to achieve the data integrity auditing of remote data. Beside the public data auditing, the combining of the three primitive enable our scheme to outsource ciphertext database to remote cloud and support secure group users revocation to shared dynamic data. We provide security analysis of our scheme, and it shows that our scheme provide data confidentiality for group users, and it is also secure against the collusion attack from the cloud storage server and revoked group users. Also, the performance analysis shows that, compared with its relevant schemes, our scheme is also efficient in different phases.

REFERENCE

[1] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in Proc. of CRYPTO 2004, CA, USA, Aug. 2004, pp. 41–55.

[2] U. M. Maurer and S. Wolf, "Diffie-hellman oracles," in Proc. of CRYPTO 1996, CA, USA, Aug. 1996, pp. 268–282.

[3] F. Bao, R. Deng, and H. Zhu, "Variations of diffie-hellman proble," in Information and Communications Security, Huhehaote, China, Oct. 2003, pp. 301–312.

[4] X. Chen, J. Li, J. Weng, J. Ma, and W. Lou, "Verifiable computation over large database with incremental updates," in Proc. of ESORICS 2014, Wroclaw, Poland, Sep. 2014, pp. 148–162.

[5] X. Chen, J. Li, X. Huang, J. Ma, and W. Lou, "New publicly verifiable databases with efficient updates," to appear in IEEE Transactions on Dependable and Secure Computing, Accepted.

[6] B. Lynn. (2006) The pairing-based cryptography library. [Online]. Available: http://crypto.stanford.edu/pbc/

[7] R. Gennaro, C. Gentry, and B. Parno, "Non-interactive verifiable computing: Outsourcing computation to untrusted workers," in Proc. of CRYPTO 2010, CA, USA, Sep. 2010, pp. 465– 482.

[8] C. Gentry, "Fully homomorphic encryption using ideal lattices," in Proc. of ACM STOC 2009, Washington DC, USA, May 2009, pp. 169–178.

[9] C. Gentry and S. Halevi, "Implementing gentrys fullyhomomorphic encryption scheme," in Proc. of EUROCRYPT 2011, Tallinn, Estonia, May 2011,pp. 129–148.

[10] S. Benabbas, R. Gennaro, and Y. Vahlis, "Verifiable delegation of computation over large datasets," in Proc. of CRYPTO 2011, CA, USA, Aug. 2011, pp. 111–131.