Public Integrity Auditing for Shared Dynamic Cloud Data with Group User Revocation

The advent of the cloud computing makes storage outsourcing become a rising trend, which promotes the secure remote data auditing a hot topic that appeared in the research literature. Recently some research consider the problemof secure and efficient public data integrity auditing for shared dynamic data. However, these schemes are still not secure against the collusion of cloud storage server and revoked group users during user revocation in practical cloud storage system. In this paper, we figure out the collusion attack in the exiting scheme and provide an efficient public integrity auditing scheme with secure group user revocation based on vector commitment and verifier-local revocation group signature. We design a concrete scheme based on the our scheme definition. Our scheme supports the public checking and efficient user revocation and also some nice properties, such as confidently, efficiency, countability and traceability of secure group user revocation. Finally, the security and experimental analysis show that, compared with its relevant schemes our scheme is also secure and efficient.

EXISTING SYSTEM:

- For providing the integrity and availability of remote cloud store, some solutions and their variants have been proposed. In these solutions, when a scheme supports data modification, we call it *dynamic* scheme, otherwise *static* one (or limited dynamic scheme, if a scheme could only efficiently support some specified operation, such as append). A scheme is *publicly verifiable* means that the data integrity check can be performed not only by data owners, but also by any third-party auditor. However, the dynamic schemes above focus on the cases where there is a data owner and only the data owner could modify the data.
- To support multiple user data operation, Wang et al. proposed a data integrity based on ring signature.
- To further enhance the previous scheme and support group user revocation, Wang et al. designed a scheme based on proxy re-signatures.
- Another attempt to improve the previous scheme and make the scheme efficient, scalable and collusion resistant is Yuan and Yu, who designed a dynamic public integrity auditing scheme with group user revocation. The authors designed polynomial authentication tags and adopt proxy tag update techniques in their scheme, which make their scheme support public checking and efficient user revocation.

DISADVANTAGES OF EXISTING SYSTEM:

- In the Wang et al. scheme, the user revocation problem is not considered and the auditing cost is linear to the group size and data size.
- However, the scheme assumed that the private and authenticated channels exist between each pair of entities and there is no collusion among them. Also, the auditing cost of the scheme is linear to the group size.
- However, in Yuan and Yu scheme, the authors do not consider the data secrecy of group users. It means that, their scheme could efficiently support plaintext data update and integrity auditing, while not ciphertext data. In their 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. Also, the data owner does not take part in the user revocation phase, where the

cloud itself could conduct the user revocation phase. 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.

PROPOSED SYSTEM:

- 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 ciphertext data base update among group users and efficient group user revocation respectively.
- Specifically, the group user uses 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:

- We explore on the secure and efficient shared data integrate auditing for multi-user operation for ciphertext database.
- 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.
- We provide the security and efficiency analysis of our scheme, and the analysis results show that our scheme is secure and efficient.

MODULES:

] Cloud server

] Group of users

] Public verifier

] Auditing Module

MODULES DESCRIPTION:

Cloud server

1. In the first module, we design our system with Cloud Server, where the datas are stored globally. Our mechanism, Oruta, should be designed to achieve following properties:
2. Public Auditing: A public verifier is able to publicly verify the integrity of shared data without retrieving the entire data from the cloud.

3. Correctness: A public verifier is able to correctly verify shared data integrity.
4. Unforgeability: Only a user in the group can generate valid verification metadata (i.e., signatures) on shared data.
5. Identity Privacy: A public verifier cannot distinguish the identity of the signer on each block in shared data during the process of auditing

Group of users:

- There are two types of users in a group: the original user and a number of group users. The original user initially creates shared data in the cloud, and shares it with group users. Both the original user and group users are members of the group. Every member of the group is allowed to access and modify shared data. Shared data and its verification metadata (i.e., signatures) are both stored in the cloud server. A public verifier, such as a third party auditor providing expert data auditing services or a data user outside the group intending to utilize shared data, is able to publicly verify the integrity of shared data stored in the cloud server.
- Owner Registration: In this module an owner has to upload its files in a cloud server, he/she should register first. Then only he/she can be able to do it. For that he needs to fill the details in the registration form. These details are maintained in a database.
- Owner Login: In this module, owners have to login, they should login by giving their email id and password.
- User Registration: In this module if a user wants to access the data which is stored in a cloud, he/she should register their details first. These details are maintained in a Database.
- User Login: If the user is an authorized user, he/she can download the file by using file id which has been stored by data owner when it was uploading.

Public verifier

- When a public verifier wishes to check the integrity of shared data, it first sends an auditing challenge to the cloud server. After receiving the auditing challenge, the Cloud server responds to the public verifier with an auditing proof of the possession of shared data.
- Then, this public verifier checks the correctness of the entire data by verifying the correctness of the auditing proof. Essentially, the process of public auditing is a challenge and- response protocol between a public verifier and the cloud server.

Auditing Module

- In this module, if a third party auditor TPA (maintainer of clouds) should register first. This system allows only cloud service providers. After third party auditor gets logged in, He/ She can see how many data owners have uploaded their files into the cloud. Here we are providing TPA for maintaining clouds.
- We only consider how to audit the integrity of shared data in the cloud with *static groups*. It means the group is pre-defined before shared data is created in the cloud and the membership of users in the group is not changed during data sharing.
- The original user is responsible for deciding who is able to share her data before outsourcing data to the cloud. Another interesting problem is how to audit the integrity of shared data in the

cloud with *dynamic groups* — a new user can be added into the group and an existing group member can be revoked during data sharing — while still preserving identity privacy.

SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

System	:	Pentium IV 2.4 GHz.
Hard Disk	:	40 GB.
Floppy Drive	:	1.44 Mb.
Monitor	:	15 VGA Colour.
Mouse	:	Logitech.
Ram	:	512 Mb.

SOFTWARE REQUIREMENTS:

Operating system	:	Windows XP/7.
Coding Language	:	JAVA/J2EE
IDE	:	Netbeans 7.4
Database	:	MYSQL