Periodic Updation in TPA for Secure Dynamic Audit Services of Outsourced Storages in Clouds

N. Siva Priya,
Dept. of Computer Science and Engg, University College of Engineering, Nagercoil, India,

Abstract – The main objective of this project is secure dynamic auditing of shared data in the cloud. In existing we use Oruta, a privacy-preserving public auditing mechanism for shared data in the cloud. In this, we utilize ring signatures to construct holomorphic authenticators, so that a public verifier is able to audit shared data integrity without retrieving the entire data, yet it cannot distinguish who is the signer on each block. Our mechanism, Oruta, should be designed to achieve following properties by Public Auditing, Correctness, Unforgeability and Identity Privacy. In our proposed system, an audit service is constructed based on the technique called fragment structure and supporting provable updates to outsourced data and timely anomaly detection. In addition, we propose a method based on periodic verification for improving the performance of audit services.

Index Terms—Public auditing mechanism, Correctness, Unforgeability, Identity Privacy, Ring signatures, Data Fragmentation.

1. INTRODUCTION

Cloud service providers offer users efficient and scalable data storage services with a much lower marginal cost than traditional approaches. It is routine for users to leverage cloud storage services to share data with others in a group, as data sharing becomes a standard feature in most cloud storage offerings, including Dropbox, iCloud and Google Drive.

The data integrity in cloud storage, however, is subject to skepticism and scrutiny, as data stored in the cloud can easily be corrupted due to the inevitable hardware/ software failures and human errors. To make this matter even worse, cloud service providers may be reluctant to inform users about these data errors in order to maintain the reputation of their services and avoid losing profits. Therefore, the integrity of cloud data should be verified before any data utilization, such as search or computation over cloud data the traditional approach for checking data correctness is to retrieve the entire data from the cloud, and then verify data integrity by checking the correctness of signatures or hash values of the entire data. Certainly, this conventional approach is able to successfully check the correctness of cloud data.

However, the efficiency of using this traditional approach on cloud data is in doubt. The main reason is that the size of cloud data is large in general. Downloading the entire cloud data to verify data integrity will cost or even waste user amounts of computation and communication resources, especially when data have been corrupted in the cloud. Besides, many uses of cloud data (e.g., data mining and machine learning) do not necessarily need users to download the entire cloud data to local devices. It is because cloud providers, such as Amazon, can offer users computation services directly on large-scale data that already existed in the cloud.

Recently, many mechanisms have been proposed to allow not only a data owner itself but also a public verifier to efficiently perform integrity checking without downloading the entire data from the cloud, which is referred to as public auditing.

2. RELATED WORK

- System Overview
- Ring Signatures
- Fragment structure

2.1 System Overview

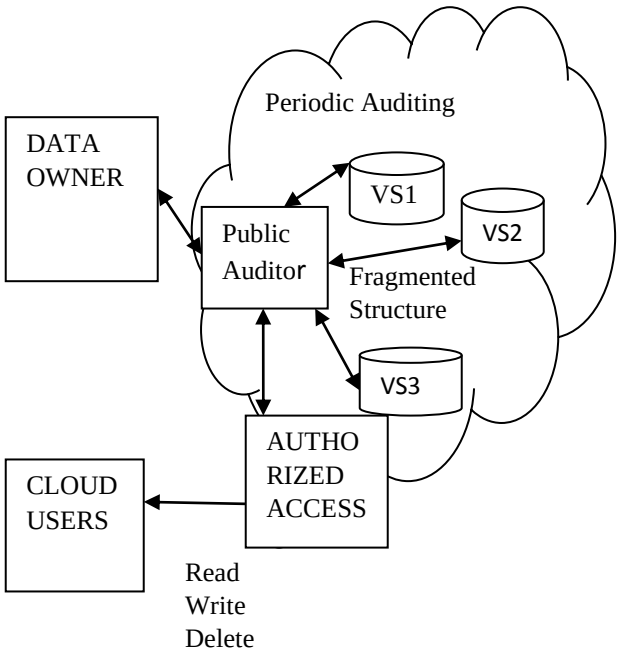

Figure 1 System Overview

The system model involves the cloud server, a group of users and a public verifier. There are original user and a number of group users. The original user initially creates shared data in the cloud, and shares it with group users. We design the volunteer storage to store the data by splitting it into fragments and store it in the cloud storage. Each and every fragment that is stored in the cloud will be encrypted individually.

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 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.

The Public Auditor dynamically audits the data in this module. If the data is modified by user in cloud storage, the TPA updates periodically. It shows the information about the user who modified the data in the cloud. This will make the system more secured and safe from unauthorized persons.

2.2 Ring Signatures

We intend to utilize ring signatures to hide the identity of the signer on each block, so that private and sensitive information of the group is not disclosed to public verifiers. However, traditional ring signatures cannot be directly used into public auditing mechanisms, because these ring signature schemes do not support block less verifiability.

Without block less verifiability, a public verifier has to download the whole data file to verify the correctness of shared data, which consumes excessive bandwidth and takes very long verification times. Therefore, we design a new holomorphic authenticable ring signature (HARS) scheme, which is extended from a classic ring signature scheme.

The ring signatures generated by HARS are not only able to preserve identity privacy but also able to support block less verifiability. We will show how to build the privacy-preserving public auditing mechanism for shared data in the cloud based on this new ring signature scheme.

HARS contains three algorithms:

KeyGen

RingSign

RingVerify

In KeyGen, each user in the group generates his/her public key and private key.

In RingSign, a user in the group is able to generate a signature on a block and its block identifier with his/her private key and all the group members' public keys.

In RingVerify, a block identifier is a string that can distinguish the corresponding block from others. A verifier is able to check whether a given block is signed by a group member**.**

2.3 Fragment structure

We design the volunteer storage to store the data by splitting it into fragments and store it in the cloud storage. Each and every fragment that is stored in the cloud will be encrypted individually. We consider that a data storage service involves: Data Owner, who has a large amount of data to be stored in the cloud; who provides data storage service and has enough storage space and computation resources; Public Auditor, who has capabilities to manage or monitor the outsourced data under the delegation of Data Owner; and authorized applications, who have the right to access and manipulate the stored data. To maximize the storage efficiency and audit performance, our audit system introduces a general fragment structure for outsourced storages. A probabilistic auditing is preferable to realize the anomaly detection in a timely manner. To support dynamic data operations, it is used to record the changes of file blocks. It shows which user modifies the data at which time it has been modified will be known. Hence this data or information will be more secure.

3. PROPOSED MODELLING

- User Authentication
- Update the Cloud Storage
- Public Auditing Mechanism
- Periodic Updation in TPA
- Response from Cloud Storage

3.1 User Authentication

In this module the user has to register in the cloud service for authentication purpose which is to improve more security in the cloud storage environment.

The Authorized applications should be cloud application services inside clouds for various application purposes, but they must be specifically authorized by Data owners for manipulating outsourced data.

Any unauthorized modifications for data will be detected in audit processes or verification processes. This kind of strong authorization-verification mechanism, we assume neither Cloud Service Providers is trusted to guarantee the security of stored data.

3.2 Update the Cloud Storage

In this module the data owners update the data in the cloud storage. Data owner, who has a large amount of data to be stored in the cloud.

Cloud Service Providers, who provide data storage service and has enough storage space and computation resources.

The Public Auditor, who has capabilities to manage or monitor the outsourced data under the delegation of data owner and authorized applications, who have the right to access and manipulate the stored data.

Finally, application users can enjoy various cloud application services via these Authorized Applications.

The volunteer storage is designed to store the data by splitting it into fragments and store it in the cloud storage. Each and every fragment that is stored in the cloud will be encrypted individually.

The Public Auditor dynamically audits the data in this module. If the data is modified by user in cloud storage, the TPA updates periodically. It shows the information about the user who modified the data in the cloud. This will make the system more secured and safe from unauthorized persons.

3.3 Public Auditing Mechanism

A public auditing mechanism is used for integrity verification of untrusted and outsourced storages.

They also suffer from the lack of trust on Cloud Service Providers because the data change may not be timely known by the cloud users, even if these disputes may result from the users own improper operations.

Therefore, it is necessary for Cloud Service Providers to offer an efficient audit service to check the integrity and availability of stored data.

An audit service can provide public auditability without downloading raw data and protect privacy of the data. Hence we are using a Public Auditor (PA) that checks the data dynamically in a ring structure.

Ring signatures are utilized to construct holomorphic authenticators, so that a public verifier is able to audit shared data integrity without retrieving the entire data.

3.4 Periodic Updation in TPA

In this module we design the volunteer storage to store the data by splitting it into fragments and store it in the cloud storage. The Third parity Auditor audits dynamically the data in this module. If the data are modified in cloud storage by user, the TPA updates periodically. It shows which user modifies the data. Hence this system will be more secure.

Our audit service can provide public auditability without downloading raw data and protect privacy of the data.

Hence we are using a Third Party Auditor (TPA) that checks the data dynamically. To detect data errors or losses in outsourced storage, as well as anomalous behaviors of data operations in a timely manner.

3.5 Response from Cloud Storage

The response from cloud storage is more efficient because the data owners are entitled to utilize the audit service without additional costs.

The public auditor is used to verify the correctness of cloud data on demand without retrieving a copy of the whole data or introducing additional online burden to cloud services.

Hence it also minimizes the computation and communication costs and also used to download the data more efficiently.

Efficiently verify the shared data integrity in a ring structure without retrieving the entire file.

4. RESULTS AND DISCUSSIONS

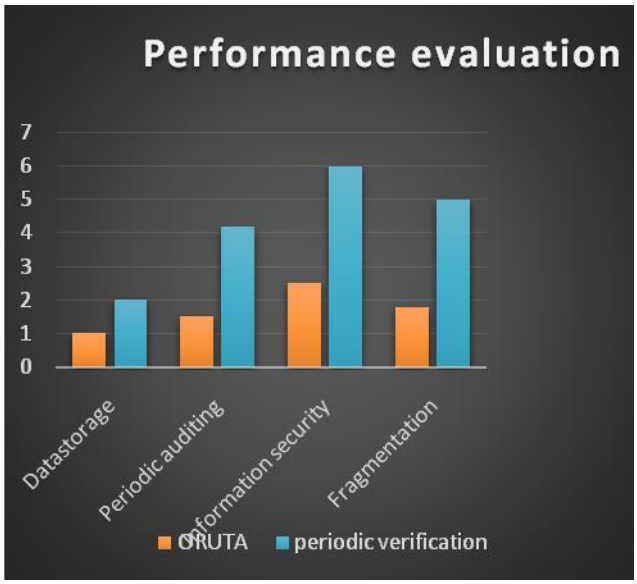

Figure 1 Resultant Graph of the Proposed System

Factors	ORUTA	Periodic verification using TA
Data storage	1	2
Periodic auditing	1.5	4.2
Information security	2.5	6
Fragmentation	1.8	5

5. CONCLUSION

The privacy-preserving public auditing mechanism that supports public auditing on shared data stored in the cloud. We utilize ring signatures to construct holomorphic authenticators, so that a public verifier is able to audit shared data integrity without retrieving the entire data, yet it cannot distinguish who is the signer on each block. Oruta, should be designed to achieve following properties by Public Auditing, Correctness, Unforgeability and Identity Privacy. An audit service is constructed based on the technique called fragment structure and supporting provable updates to outsourced data and timely anomaly detection, In addition a method is proposed based on periodic verification for improving the performance of audit services. A probabilistic auditing is preferable to realize the anomaly detection in a timely manner. To support dynamic data operations, it is used to record the changes of file blocks. It shows which user modifies the data at which time it has been modified will be known. Hence this data or information will be more secure.

REFERENCES

[1] Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security inCloud Computing," in Proc. IEEE International Conference on Computer Communications (INFOCOM), 2010, pp. 525–533.

[2] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing," in Proc. IEEE International Conference on Computer Communications (INFOCOM), 2010, pp. 534–542.

[3] N. Cao, S. Yu, Z. Yang, W. Lou, and Y. T. Hou, "LT Codesbased Secure and Reliable Cloud Storage Service," in Proc. IEEE International Conference on Computer Communications (INFOCOM), 2012.

[4] S. D. C. di Vimercati, S. Foresti, S. Paraboschi, G. Pelosi, and P. Samarati, "Efficient and Private Access to Outsourced Data," in Proc. IEEE International Conference on Distributed Computing Systems (ICDCS), 2011, pp. 710–719.

[5] B. Chen, R. Curtmola, G. Ateniese, and R. Burns, "Remote Data Checking for Network Coding-based Distributed Stroage Systems," in Proc. ACM Cloud Computing Security Workshop (CCSW), 2010, pp. 31–42.

[6] M. Franz, P. Williams, B. Carbunar, S. Katzenbeisser, and R. Sion, "Oblivious Outsourced Storage with Delegation," in Proc. Financial Cryptography and Data Security Conference (FC), 2011, pp. 127–140.

[7] Juels and B. S. Kaliski, "PORs: Proofs of Retrievability for Large Files," in Proc. ACM Conference on Computer and Communications Security (CCS), 2007, pp. 584–597.

[8] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," in Proc. International Conference on Security and Privacy in Communication Networks (SecureComm), 2008.

[9] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic Provable Data Possession," in Proc. ACM Conference on Computer and Communications Security (CCS), 2009, pp. 213–222.

[10] H. Shacham and B. Waters, "Compact Proofs of Retrievability," in Proc. International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT). SpringerVerlag, 2008, pp. 90–107.