A Novel Privacy Preserving Public Appraising Mechanism for Shared Data Distribution in the Cloud

Kirti Panmand
PG Student
Dr. D.Y. Patil College of Engineering, Pune

Sandeep Kadam
HOD of Computer Department
Dr. D.Y. Patil College of Engineering, Pune

ABSTRACT

Cloud server is a model for enabling suitable, on-demand network access to a shared pool of resources that can be rapidly provisioned and released with cloud service provider communication. However, public appraising for such shared data while conserving uniqueness of system leftovers to be an open challenge. In this paper, the first privacy-conserving mechanism is proposed that allows public appraising on shared data stored in the cloud. In this, ring signature mechanism is used to calculate the authentication information needed to audit the integrity of shared data. With the help of this mechanism, the uniqueness of the signer on each block in shared data is kept secret from a third party auditor (TPA), who is still able to authenticate the integrity of shared data without accessing the complete file.

General Terms

Cloud storage, TPA, Privacy conserving

Keywords

Cloud server, public appraising, shared data, integrity

1. INTRODUCTION

Cloud offers massive opportunity for new revolution and disruption of industries. In this data owners can remotely store their data in the cloud to enjoy on demand high-quality applications and services from a shared pool of computing resources. It is habitual for users to use cloud storage services to share data with others in a team, as data sharing becomes a standard aspect in most cloud storage assistances like Dropbox and Force. The integrity of data in cloud storage is subject to uncertainty and scrutiny, as data stored in an untrusted cloud can easily be lost or corrupted, due to hardware failures and human errors [1]. To protect the integrity of cloud data, it is best to perform public auditing. Therefore, a third party auditor (TPA) is introduced to perform this task, who offers its auditing service with more powerful calculation and communication abilities than regular users. The first provable data possession (PDP) mechanism [2] to perform public appraising is designed to check the accuracy of data stored in an untrusted server, without retrieving the complete data. Wang *et al.* [3] (referred to as WWRL in this paper) is designed to construct a public appraising mechanism for cloud data, so that during public appraising, the content of private data belonging to a personal user is not disclosed to the third party auditor. Sharing data among multiple users is one of the most attractive features that motivate cloud storage. But the difficulty introduced during the process of public appraising for cloud data is how to preserve identity privacy from the TPA, because the identities of signers on shared data may designate that a particular user in the group or a special block in shared data is a higher valuable target than others. Such information is confidential to the group and should not be revealed to any third party. However, no existing mechanism in the literature is able to perform public auditing on shared data in the cloud while still preserving identity privacy. This drawback, if not properly addressed, could hamper the successful deployment of the cloud server's design. As users data on remote storage, traditional cryptographic primitives for the purpose of data security protection cannot be adopted directly.

In this paper, a new privacy conserving public appraising mechanism is proposed for shared data in an untrusted cloud. In this, ring signatures mechanism [4], [5] is utilized to construct homomorphic authenticators [2], [6], so that the third party auditor is able to verify the integrity of shared data for a group of users without accessing the entire data. The advantage of ring signature is, TPA won't be able to identify signer of each block i.e. privacy & integrity will be preserved. It also supports batch auditing; which can audit multiple shared data simultaneously in a single auditing task. Random masking [3] is used to support data privacy during public auditing, and sql is used [7] to support fully dynamic operations on shared data. A dynamic operation indicates an insert, delete or update operation on a single block in shared data.

1.1 Integrity in Cloud Storage

The integrity of data in cloud storage is subject to ambiguity. Data stored in the cloud can easily be lost or corrupted due to the unavoidable hardware/ software failures and human errors. To make this matter even bad, cloud service providers may be reluctant to inform users about these data faults in order to sustain the good status of their services and avoid profit loss. Therefore, the integrity of cloud data should be verified before any data operation.

1.2 Efficient Cloud Processing

The efficiency of processing the cloud was very big challenge. The core reason is that the size of cloud data is very massive in general. Downloading the complete cloud data to verify data integrity will increase cost also waste user's amounts of computation and communication resources, especially when data have been corrupted in the cloud. Besides, many scheme like data mining and machine learning does not essentially need cloud users to download the whole cloud data to local devices.

1.3 Correctness of Data

The traditional method for checking the correctness of data in cloud includes two steps. The first step is to retrieve the entire data from the cloud, and the second step is to authenticate data integrity by checking the accuracy of signatures by RSA or hash values by MD5 of the whole data. Advantage of this approach is able to effectively check the accuracy of cloud data.

2. RELATED WORK

Provable data possession (PDP), first proposed by Ateniese et al. [2], allows a verifier to ensure the accuracy of a client's data stored at an untrusted server. By utilizing RSA-based homomorphic authenticators and sampling strategies, the verifier is able to publicly audit the integrity of data without retrieving the entire data, which is referred to as public verifiability or public appraising. Unfortunately, their mechanism is only suitable for auditing the integrity of static data.

In 2010 ,B. Chen, R. Curtmola, G. Ateniese, and R. Burns, worked on **"**Remote Data Checking for Network Coding-Based Distributed Storage Systems**"** they have introduced a mechanism for checking the accuracy of data under the multi-server situation, where these data are encrypted by network coding instead of using destruction codes. This scheme reduces communication overhead in the phase of data repair.

Juels and Kaliski defined another similar model called proofs of retrievability (POR), which is also able to check the correctness of data on an untrusted server. The original file is added with a set of randomly-valued check blocks called sentinels. The verifier challenges the untrusted server by specifying the positions of a collection of sentinels and asking the untrusted server to return the associated sentinel values.

Shacham and Waters [6] designed two improved POR schemes. The first scheme is built from BLS signatures, and the second one is based on pseudorandom functions.

To support dynamic operations on data, Ateniese et al. [8] presented an efficient PDP mechanism based on symmetric keys. This mechanism can support update and delete operations on data; however, insert operations are not available in this mechanism. Because it exploits symmetric keys to verify the integrity of data, it is not public verifiable and only provides a user with a limited number of verification requests.

Zhu et al. exploited the fragment structure to reduce the storage of signatures in their public auditing mechanism. In addition, they also used index hash tables to provide dynamic operations for users. The public mechanism proposed by Wang et al. [3] is able to preserve users' confidential data from the TPA by using random masking. In addition, to operate multiple auditing tasks from different users efficiently, they extended their mechanism to enable batch auditing by leveraging aggregate signatures [5].

Wang et al. [9] leveraged homomorphic tokens to ensure the correctness of erasure codes-based data distributed on multiple servers. This mechanism is able not only to support dynamic operations on data, but also to identify misbehaved servers. To minimize communication overhead in the phase of data repair.

3. CONSTRUCTION MECHANISM

It includes five algorithms: KeyGen, SigGen, Modify, ProofGen and ProofVerify. In **Key-Gen**, users generate their own private key. In **SigGen**, a user (either the owner or a group user) is able to compute ring signatures on blocks in shared data. Each user in the group is able to perform an insert, delete or update operation on a block, and compute the new ring signature on this new block in Modify. **ProofGen** is operated by the TPA and the cloud server together to generate a proof of data possession. In **ProofVerify**, the TPA verifies the proof and sends an auditing report to the user. The group is pre-defined before shared data is created in the cloud. Before the original user outsources shared data to the cloud, she decides all the group members, and computes all the initial ring signatures of all the blocks in shared data with her private key and all the group members' public keys. After shared data is stored in the cloud, when a group member modifies a block in shared data, this group member also needs to compute a new ring signature on the modified block.

ProofGen: To audit the integrity of shared data, a user first sends an auditing request to the TPA. After receiving an auditing request, the TPA generates an auditing message as follows:

1) The TPA randomly picks an element to locate the selected blocks that will be checked in this auditing process.

2) TPA sends an auditing message to the cloud server. After receiving an auditing message, the cloud server generates a proof of possession of selected blocks with the public aggregate key. After the computation, the cloud server sends an auditing proof.

ProofVerif: With an auditing proof, an auditing message, public aggregate key and all the group members public keys the TPA verifies the correctness of this proof.

Modify: Performing one of the following three operations:

• Insert: This user inserts a new block into shared data. Then computes the new identifier of the inserted block for the rest of blocks; the identifiers of these blocks are not changed.

• Delete: This user deletes block, its identifier and ring signature from the cloud server. The identifiers of other blocks in shared data are remaining the same. The total number of blocks in shared data decreases to n − 1.

• Update. This user updates the block in shared data with a new block.

Table 1. Summary of Characteristics

S. NO.	Methods	Existing Scheme	Proposed Scheme
1	Technique	Provable data possession (PDP) Proofs of Retrievability (POR)	Privacy preserving and Third party auditing
2	Identity of signer	Kept public to public verifier	Kept private from public verifier
3	Auditing	Single auditing	Batch auditing

Table2. Result & Data Set

Format	Input	Output
Authentication	username/password dialog box	HTML Pages served after successful authentication
	Insert Command	Insertion of records

Query	Delete Command	One can also delete records inserted into the table
	Update Command	Modifying data already entered into the table

3.1 Batch Auditing

With the usage of public auditing in the cloud, the TPA may receive amount of auditing requests from different users in a very short time. Unfortunately, allowing the TPA to verify the integrity of shared data for these users in several separate auditing tasks would be very inefficient. Therefore, with the properties of bilinear maps, this mechanism is extended to support batch auditing, which can improve the efficiency of verification on multiple auditing tasks. More concretely, assume that there are B auditing tasks need to be operated, the shared data in all the B auditing tasks are denoted as M1, ...,MB and the number of users sharing data Mb is described as db, where 1 _ b _ B. To efficiently audit these shared data for different users in a single auditing task, the TPA sends an auditing message to the cloud server. After receiving the auditing message, the cloud server generates an auditing proof for each shared data Mb as presented in ProofGen, where 1 _ b _ B, 1 _ l _ k, if two blocks are in the same shared data, these two blocks have the same identifier of shared data. As before, when a user modifies a single block in shared data Mb, the identifiers of other blocks in shared data Mb are not changed. After the computation, the cloud server sends all the B auditing proofs together to the TPA. To allow most of auditing proofs to still pass the verification when there is only a small number of incorrect auditing proofs; we can utilize binary search [3] during batch auditing. More specifically, once the batch auditing of the B auditing proofs fails, the TPA divides the set of all the B auditing proofs into two subsets, which contains B/2 auditing proofs in each subset, and rechecks the correctness of auditing proofs in each subset using batch auditing. If the verification result of one subset is correct, then all the auditing proofs in this subset are all correct. Otherwise, this subset is further divided into two sub-subsets, and the TPA rechecks the correctness of auditing proofs in the each sub-subsets with batch auditing until all the incorrect auditing proofs are found.

3.2 Digital Signature

A digital signature is a mathematical scheme for representing the validity of a digital message or document. A valid digital signature gives a receiver reason to believe that the message was created by a known sender, such that the sender cannot deny having sent the message and that the message was not altered in transit. Digital signatures are commonly used for software distribution, financial transactions, and in other cases where it is important to detect forgery or tampering. Digital signatures can also provide non-repudiation, meaning that the signer cannot successfully claim they did not sign a message, while also claiming their private key remains secret; further, some non-repudiation schemes offer a time stamp for the digital signature, so that even if the private key is exposed, the signature is valid.

3.3 Dynamic Operations

It allows each user in the group to modify data in the cloud and share the newest version of data with the rest of the group. Dynamic operation includes an insert, delete or update operation on a single block of data. When a user modifies a single block in shared data by performing an insert or delete operation, the indices of blocks that after the modified block are all changed and the changes of these indices require users, who are sharing the data, to re-compute the signatures of these blocks, even though the content of these blocks are not modified.

4. FIGURES/CAPTIONS

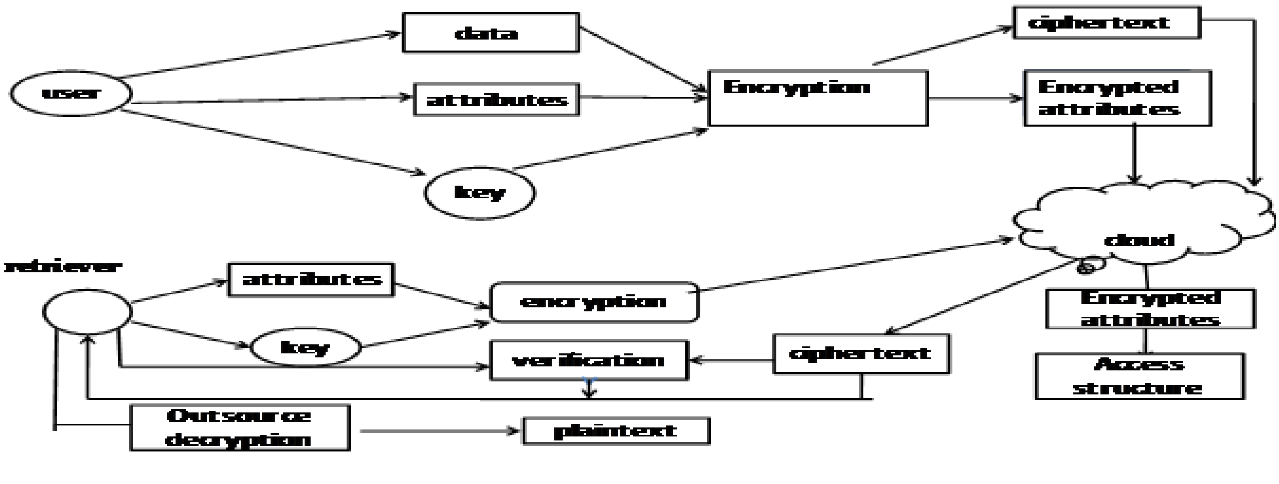

Fig 1: Architecture

5. CONCLUSION

In this paper, the first privacy-conserving public appraising mechanism is proposed for shared data in the cloud. With this mechanism, the TPA is able to efficiently audit the integrity of shared data, yet cannot differentiate who is the signer on each block, which can maintain identity privacy for users. The problem for future work is how to prove data freshness & how the admin can reveal identity of data owner.

6. ACKNOWLEDGMENTS

I am deeply indebted to my guide Prof. Sandeep Kadam whose help, stimulating suggestions and encouragement helped me in all the time of writing of this paper. I have furthermore to thank to all staff members, ME, PG Coordinator, Who gave me opportunity and encouraged me to go ahead with this topic. I want to thanks them for all their help, support, interest and valuable hints.

7. REFERENCES

[1] M. Armbrust, A. Fox, R. Griffith, A. D.Joseph, R. H.Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, Apirl 2010.

[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," in*Proc. ACM Conference on Computer and Communications Security (CCS)*, 2007, pp. 598–610.

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

[4] R. L. Rivest, A. Shamir, and Y. Tauman, "How to Leak a Secret," in*Proc. International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*. Springer-Verlag, 2001, pp. 552–565.

[5] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and Verifiably Encrypted Signatures from Bilinear Maps," in *Proc. International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT)*. Springer-Verlag, 2003, pp. 416–432.

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

[7] Y. Zhu, H.Wang, Z. Hu, G.-J.Ahn, H. Hu, and S. S.Yau, "Dynamic Audit Services for Integrity Verification of Outsourced Storage in Clouds," in *Proc. ACM Symposium on Applied Computing (SAC)*, 2011, pp. 1550–1557.

[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. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring Data Storage Security in Cloud Computing," in Proc. IEEE/ACM International Workshop on Quality of Service (IWQoS), 2009, pp. 1–9.