Review on Privacy Preserving Public Auditing for Data Sharing on Cloud

Dipali R Sakhare
PG student
Department of computer Engineering
Dr. D. Y. Patil School of Engineering & Technology
Savitribai Phule Pune University
Pune, India

Arti Mohanpurkar
Assistant Professor Department of computer Engineering
Dr. D. Y. Patil School of Engineering & Technology
Savitribai Phule Pune University
Pune, India

ABSTRACT

There numerous data integrity checking techniques have been projected in purpose of privacy preserving cloud data. Most of mechanisms assume that only the data owner can modify data stored on cloud. There are various techniques have been projected intended for data integrity auditing which focuses on various practical features to have user's confidence of the integrity of their cloud shares data. To check auditing various features considered like the dynamic data support, public auditing, low communication or computational audit cost, low storage overhead. In recent times a very few attempts consider extra realistic scenarios through allowing several cloud users to modify data with integrity assurance. On the other hand, these attempts are still away from practical owing to the tremendous estimation on cloud users. This paper intend a new integrity auditing technique intended for cloud data sharing services represented by multi-user updating, public integrity auditing, high error detection probability, efficient user revocation as well as Unauthorised user access detection. Also this paper addresses review on different techniques use for integrity check and privacy preservation.

Keywords

Cloud computing, integrity auditing, privacy preserving, public verification, batch verification

1. INTRODUCTION

With the use of cloud storage, users able to remotely store their private data and can take pleasure in the on-demand high-quality service from a shared pool of configurable computing resources, with no the load of local data storage and preservation. On the other hand, the reality is that users no longer have physical control of the outsourced data that makes the difficult task for privacy preservation of data integrity in cloud computing, particularly for users with constrained computing resources.

The constant growth of techniques used for cloud computing has boost up a various open source cloud storage applications. Especially, in much more cloud storage applications collaboration platforms are being used, where data are endure in cloud for storage and expose to regular updating from numerous users. There are various real time instances based on cloud data storage synchronization platforms for example Drop box and Google Drive, Version Control Systems (VCS) such as Subversion and Concurrent Versions System [1], which enable people to easily work together as a group with sharing data by means of each other. These data storage and sharing services enables various team members to work in concur by getting and updating same files which are stored on cloud servers anywhere and anytime. On behalf of accurate process of these type of collaborative applications, there arise a problem is that of assurance of data integrity, which means that, Every modification operation on data must sure that that operation is performed by an authoritative group member and the owner's data remains perfect and up-to-date after that. This issue gives the fact that cloud storage platforms possibly will experience hardware and software breakdown, human mistake and outdoor malicious. Additionally, previous observation shows that there have been numerous inconsistencies among the various data corruption events reported by users and those acknowledged by cloud service providers like drop-box-forum as a result users are doubtful on whether or not their data on cloud are really intact.

In most of existing systems solution on share data privacy is that only the data owner can modify the data, since data owner holds secret keys and all other users who are responsible for share data with the data owner only have permission to read that owners data If above solutions are insignificantly extended to support multiple users with data integrity guarantee, then there must raise some conditions on data owners. The conditions are 1) The data owner has to stay online 2) Data owner has to collecting updated data from other users and regenerating authentication tags for them. Clearly, this type of unimportant extension will commence a terrific workload to the data owner, mainly in scenarios where users are more or a frequency of data modification operations is high. Thus from above observation this paper addressed some issues and solutions on that issues regarding public auditing.

To form a capable public integrity auditing technique which supports multi-user data modification and adequate user revocation concurrently, system requires overcoming major challenges stated as follows: Firstly, Aggregation of independently generated verification tags. Particularly, every user with read and write privilege should be capable of data modification and new authenticating tags generation without the assist of the data owner. Yet, this type of result is restricted by second challenge i.e., an efficient and protected user revocation which is a challenging subject in most security systems and typically contains disabling user secret keys. And third challenge is public auditing. In realistic schemes, data integrity auditing can be performed not just by data owners or other group member but also via a Third Party Auditor (TPA) or any common user who has public keys of the system.

To project these types of complications, this paper work utilizes the efficient public integrity auditing mechanism for cloud data sharing business described by multi-user data updating, public auditing, Batch auditing, error finding probability, professional user revocation with practical computational conversation auditing achievements. This mechanism can defend against user impersonation attack, which is not treated in previous techniques that help into multi-user modification. Batch auditing of multiple tasks is also efficiently supported in scheme by means of TPA observation on user's data on cloud.

In section II, Literature survey, problem definition and problem statement is addressed in section III, section IV introduces basic schemes and in section VI conclusion is define.

2. LITERATURE REVIEW

In this section survey on privacy preservation on cloud shared data is addressed.

G. Ateniese et al. [2] introduced a provable data possession (PDP) model that allows a user who has stored data at an un-trusted server to authenticate that the server maintains the actual data without retrieving it. The PDP model generates probabilistic evidences of possession by sampling random sets of file blocks from the server, which radically reduces input/output costs.

To hold dynamic functions in authentication, G. Ateniese et al. [3] constructed a highly capable and secure PDP system which is entirely based on symmetric key cryptography, whereas this not required any bulk encryption. Additionally, the PDP technique allows of dynamic data outsourcing, i.e., it capably supports operations such as block updating, deleting and uploading. They allowed verifying data control without having access to the authentic data file.

The efficiency of POR system was later enhanced by H. Shacham and B. Waters give in [4] who gave the first proof-of-irretrievability mechanism with complete proofs of security beside arbitrary attackers in the strongest model, that of Juels and Kaliski [2]. Their first system is built from BLS (Boneh-Lynn-Shacham) signatures and protected in the random oracle model. The main features of proof-of-retrievability protocols in which the client's request query and server's response result are both very short. This mechanism was allowing public verifiability i.e. anybody be able to act as a verifier, not only the data owner. The second scheme called as pseudo-random functions (PRFs) which protected in the standard model and grants only private authentication. The main features of pseudo-random functions mechanism in which the client's request query are long and server's response result are even shorter server's response than first mechanism. Both systems were relying on homomorphic properties to collect a proof into one minute verifier value.

C. Wang et al. [5] investigated the difficulty of data privacy protection in cloud data storage, which is fundamentally a dispersed storage scheme. To make sure the accuracy of user's cloud data, they proposed a useful and elastic distributed technique with open dynamic data support, together with block upload, delete, and update.

Q. Wang [6] et al. explored the challenge of providing instantaneous public verifiability and data dynamics for distant data reliability ensure in Cloud Computing. The structure is intentionally planned to get together these two significant goals though efficiency being kept strictly in mind. They extended the PoR model by means of a graceful Merkle hash tree construction to accomplish completely dynamic data procedure.

C. Wang, Q. Wang et al. [7] proposed a secure cloud storage scheme allowing privacy preserving public auditing. Then they further expand the outcomes to allow the third party auditor (TPA) to carry out audits for various users at the same time and efficiently. For this technique they use the homomorphic linear authenticator (HLA) and random masking to assurance that the TPA would not learn and discover any information about the user's data content which stored on the cloud server through the efficient auditing procedure, which removes the workload of cloud user from the boring and maybe expensive auditing task as well as mitigates the user's fear about their data leakage.

Y. Zhuet al. [8] presented a development of dynamic verify services for untrusted and outsourced storage. Additionally they presented an efficient technique for random sampling audit to reduce the calculation costs of TPA and storage service providers i.e. cloud service provider (CSP).

Wang et al. [9] proposed a public integrity auditing mechanism using ring signature-based homomorphic authenticators. However, the scalability of this system is limited by the user group size and data size as the auditing cost grows. In their system, user revocation is not considered.

With the purpose of enhance prior work; one more effort was prepared by Wang et al, [10]. They enhanced their prior public integrity auditing system with the help of user revocation. Though, the cloud side responsible for tag modification is negotiated throughout user revocation procedure, attackers able to find out the secret keys of every other authorise users. Additionally, in this authentication cost of the TPA or user is considerably influenced with the error detection probability obligation and is as well linear to the numeral data modifier. Batch auditing is not supported in their system design. Hence, this system is restricted in its scalability.

In paper [12], authors present jPBC a Java port of the Pairing Based Cryptography (PBC) library written in C which provides complete ecosystem of interfaces and classes to make simpler the use of the bilinear maps even for system not contain cryptography.

In paper [13], J. Yuan and S. Yu proposed a scheme which is based on methods including polynomial-based authentication tags and homomorphic linear authenticators (HLA). This approach allows the deduplication of both files and their equivalent authentication tags.

Paper [14] introduced a short signature scheme with length that of half of DSA signature based on the Computational Diffie-Hellman assumption on certain elliptic and hyper-elliptic curves.

Paper [15] introduces a secure cloud storage technique supporting privacy-preserving public auditing which provides an overall outsourcing solution not only the user data itself, but also its integrity checking using Third Party Auditor (TPA). Similarly, to verify user data efficiently and securely, J. Yuan et al. [16] proposed a proof-of-retrievability (POR) system which uniquely modifying the polynomial commitment scheme and designing a new authentication tag.

3. PROBLEM STATEMENT

3.1 Problem Definition

There are some challenges observes in efficient public auditing in data sharing. Some major challenges: 1) aggregation of separately generated authentication tag values.

Particularly every user having read and write privilege should be capable of data modification and should capable to generate novel authentication tags without the data owner's help. In this circumstance, the challenge problem is how to aggregate tags from different users, since the tags are assigned with independent user's secret keys which are different from each other. Exclusive of aggregation, a user data integrity auditor has to process tags from distinct modifiers individually for an auditing task, and hence restrictive the scalability of system. A simple solution to this difficulty is to allow all users contribute to the equal secret key, so that all verification tags are in the same format in addition tags can be simply aggregated. On the other hand, this type of solution is restricted by a further challenge: 2) Efficient and safe user revocation. User revocation is a difficult problem in the majority security systems and typically involves disabling user private keys. A head revocation of user, all verification tags generated by the revoked user should be modified, and this important task is generally assigned to the cloud by discovering part of secrets to it. Once the cloud server node colludes with a revoked user, this technique can lead to revelation of secret keys of authorized users. 3) Public auditing- In realistic schemes, data integrity auditing can be performed not just by data owners or other group member but also via a Third Party Auditor (TPA) or any common user who has public keys of the system.

The accuracy of user data can be violated due to a wide range of both internal and external hazard and CSP can hide user's data loss or damage to keep a reputation. Major security problems associated with cloud user and CSP are illustrated as follows:

3.1.1 Cloud Service Provider (CSP)

Big Organization or enterprises give different services to cloud users. Since, privacy and integrity of cloud data should be maintained by cloud service provider. The service provider should guarantee that user's data and application stored on cloud are safe. CSP possibly will not disclose the information or no one modifies or access user's data content without access permission. The attacker is capable of log into network communication.

3.1.2 Cloud Server (CS)

The CS is entity that represents the cloud server where data being uploaded and accessed by cloud data owner or users.

3.1.3 Cloud User

Attackers (users) are able to access core information such as username and password. Therefore, user's Key management is main problem in Cryptography. This Data dynamic problems need to be inspected by CSP.

3.2 Adversary Model

In the Adversary model of cloud system composed of three major entities: the group of users, Cloud Server (CS), and the third party auditor (TPA). The CS is the entity that provides data storage services to group users. Group members consist of a numeral common users and a master user. The Master user is the shared data owner and handles the membership of other group members. All group members can access and update data. The TPA indicates entity that audits the integrity of data being stored on the cloud. As the projected system permits public integrity auditing, the TPA can truly be any cloud user as long as user has access to the general public keys. If the TPA catches a data fraud during the auditing process, user will report the error to group members. In this architecture, data or file can be uploaded by either the master user or other group members. The assumption is that the data are stored in form of files that files are further split into a various blocks. For integrity auditing of file block, every block is identified with a verification tag that is initially generated by the master user. Whenever a user or group member uploads or updates a block, user regenerates the corresponding verification tag with his/her own secret key and update it without contacting the master user.

Representative architecture for cloud shared data storage is shown in Fig. 1. Three different network entities user/group member, CSS and TPA can be identified as follows:

- Users: It is an entity, which can be independent consumers or enterprises and consist of set of data files to be stored on the cloud and keep trust on the cloud for data preservation and computation. In this architecture group users consist of a number of general users and a master user, who is the shared data owner and administrate the membership of other group members.
- Cloud Storage Server (CSS): It is an entity, which handled and maintained by Cloud Service Provider (CSP) and it has substantial storage space and computation cost to maintain the user's data
- Third Party Auditor (TPA): It is an entity, which has knowledge and efficiency that users do not have. TPA is supposed to be trusted to assess and depict threat of cloud storage services for the user's request.

4. THE BASIC SCHEMES

The structure supports to system stated as follows:

4.1 User System Setup Phase

To setup the system, the master user primary runs the Key Generation section and generates public keys (PK), secret (Private) keys (SK) and master keys (MK) of the system and of users.

Thereafter file is processed to split into blocks. And Authentication tag is generated for each file block as identification of block.

4.2 User

In Second stage, to audit integrity of data, the TPA generates the challenge message (CM) by running the some steps. The TPA then sends the challenging message CM to the cloud.

1. Randomly choose data blocks.
2. Assume that the blocks are modified by set of users or group members. With this assumption generates challenge message (CM).

4.3 CSP

On receiving the challenging message CM from TPA, the cloud will perform the Prove algorithm to generate the proof information Prf, which demonstrates that it originally store the challenged data file accurately.

4.4 TPA

Based on the proof information Prf of cloud, the TPA verifies the integrity of file by performing the Verify steps.

1. Compute verify request
2. Verify file
3. If verification result is true

4. Then verification successful
5. Else Fail

4.5 User Revocation-Basic

Whenever there is a user to be revoked, the master user and the cloud performs the User Revocation Basic. Particularly, every verification tags generated by revoked users are modified in order that the revoked users' secret keys are separate from the tags. In User Revocation algorithm, use a single cloud entity to modify the verification tag last updated by the revoked users.

4.6 User Revocation Advance

If the cloud node answerable for tag modernizes is negotiated due to inside errors or outside attacks, the revoked user will be capable to create valid verification tags once more. The major problem that causes such negotiation attack is the attacker can access file data that is utilized to modify verification tags when it negotiate the cloud node. Consequently, to protect this type of compromisation and improve the reliability of system uniquely incorporate a (U, N) -Shamir Secret Sharing [11] method into this system design and allocate and the verification tag update procedure to multiple cloud nodes.

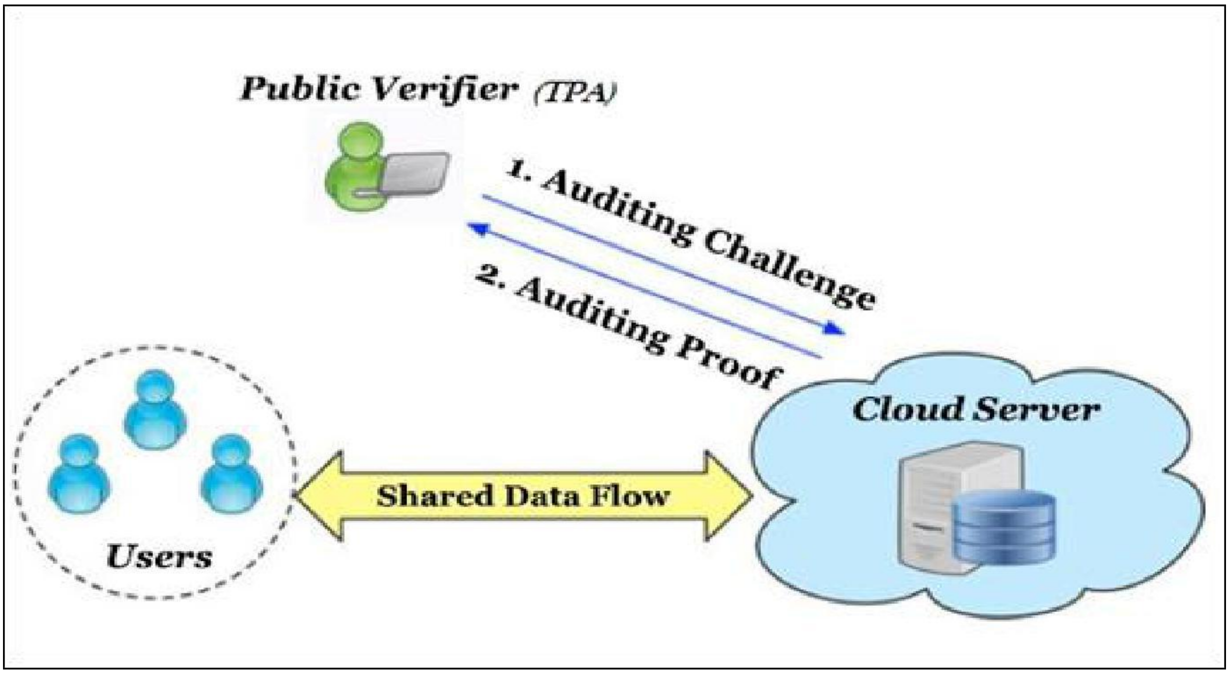

Fig 1: Architecture Diagram

5. CONCLUSION

In this paper, novel data integrity auditing system is projected that helps multiple users to share their data on cloud. The projected system is featured by most important properties of public integrity auditing and stable computational expenditure on the user side. This attains throughout innovative design on polynomial-based verification tags which permits different data blocks tags aggregation. Intended for scheme scalability, further authorize the cloud with the capability to aggregate verification tags from multiple users into one when sending the integrity proof information to the verifier. Furthermore, new system design permits secure allocation of user revocation functions to the cloud with an efficient essential system design and an advanced design with enhanced reliability. The projected system permits aggregation of integrity auditing processes for several tasks or files all the way through batch integrity auditing method.

6. ACKNOWLEDGMENTS

The authors gratefully acknowledge the contributions of [1] J. Yuan and S. Yu for their work on the Data sharing with multi-user modification of this document.

7. REFERENCES

[1] J. Yuan and S. Yu, 2015. Public Integrity Auditing for Dynamic Data Sharing with Multi-User Modification. IEEE Transactions on Information Forensics and Security Volume: 10.

[2] A. Juels and B. S. Kaliski, Jr., 2007. Pors: Proofs of retrievability for large files. In Proceedings of the 14th ACM Conference on Computer and Communications Security, ser. CCS '07. Alexandria, Virginia, USA: ACM.

[3] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song. 2007 Provable data possession at untrusted stores. In Proceedings of the 14th ACM Conference on Computer and Communications Security, ser. CCS '07. Alexandria, Virginia, USA:ACM.

[4] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik. 2008 Scalable and efficient provable data possession. In Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, ser. SecureComm '08. New York, NY, USA: ACM.

[5] H. Shacham and B. Waters. 2008 Compact proofs of retrievability. In Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology.

[6] C. Wang, Q. Wang, K. Ren, and W. Lou 2009 Ensuring data storage security in cloud computing. In Proceedings of the 17th IEEE International Workshop on Quality of Service.

[7] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou. 2009 Enabling public verifiability and data dynamics for storage security in cloud computing. In Proceedings of

the 14th European conference on Researching computer security, Saint-Malo, France.

[8] C. Wang, Q. Wang, K. Ren, and W. Lou 2010 Privacy-preserving public auditing for data storage security in cloud computing. In Proceedings of the 29th IEEE International Conference on Computer Communications, California, USA.

[9] Y. Zhu, H.Wang, Z. Hu, G. J. Ahn, H. Hu, and S. S. Yau. 2011 Dynamic audit services for integrity verification of outsourced storages in clouds. In Proceedings of the 2011 ACM Symposium on Applied Computing, ser. SAC '11. New York, NY, USA: ACM.

[10] B. Wang, B. Li, and H. Li 2012 Oruta: Privacy-preserving public auditing for shared data in the cloud. In Proceedings of the IEEE Fifth International Conference on Cloud Computing, ser. CLOUD'12, Washington, DC, USA.

[11] J. Yuan and S. Yu. 2013 Proofs of Retrievability with public verifiability and constant communication cost in cloud. In Proceedings of the International Workshop on Security in Cloud Computing, ser. Cloud Computing '13. Hangzhou, China: ACM.

[12] A. De Caro and V. Iovino. 2011 jpbc: Java pairing based cryptography. In Proceedings of the 16th IEEE Symposium on Computers and Communications.

[13] J. Yuan and S. Yu. 2013 Secure and constant cost public cloud storage auditing with deduplication. In Proceedings of the 1st IEEE Conference on Communications and Network Security, ser. CNS'13, Washington, USA.

[14] D. Boneh, B. Lynn, and H. Shacham. 2001 Short signatures from the weil pairing. In Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology.

[15] C. Wang, S. S. Chow, Q. Wang, K. Ren, and W. Lou 2013 Privacy preserving public auditing for secure cloud storage," IEEE Transactions on Computers.

[16] J. Yuan and S. Yu. 2013 Proofs of retrievability with public verifiability and constant communication cost in cloud. In Proceedings of the International Workshop on Security in Cloud Computing, ser. Cloud Computing '13. Hangzhou, China: ACM.