IJRSET January 2016 Volume 3, Issue 1 Pages: 19-22

ISSN 2394-739X

International Journal for Research in Science Engineering and Technology

AN OUTSOURCED PROOF OF RETRIEVABILITY FOR DYNAMIC DATA OPERATION IN CLOUD

[1] C. Sarika, [2] R. Megiba Jasmine

[1] PG Scholar, [2] Assistant Professor,

[1] Ponjesly College of Engineering, [2] Ponjesly College of Engineering,

[1] Nagercoil, India, [2] Nagercoil, India.

Abstract:-

Cloud computing is popular, but has adopted because there are many security and privacy problems. A problem found in cloud storage is, when clients outsource their data to the cloud storage servers, the clients do not know that their data is not damaged. Also the computational burden is too huge. To tackle the challenge, OPoR a new cloud storage scheme involving a cloud storage server and a cloud audit server is proposed. The cloud audit server is allowed to pre-process the data before uploading to the cloud storage server instead of the cloud users. 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. OPoR outsources the heavy computation of the tag generation and eliminates the involvement of user. The proof of retrievability (PoR) model is strengthened to support dynamic data operations.

Keywords: - cloud storage, proof-of-retrievability, cloud audit server, cloud storage server.

1. INTRODUCTION

The Proof of Retrievability (PoR) is an archive that provides a concise proof that the user can retrieve the target file. PoR is an important tool for semi-trusted online archives. The users can view their file in the archive but they cannot modify the data in the file. The goal of a PoR is to accomplish these checks without users having to download the files themselves. Also in a PoR the cloud storage must prove to a verifier that is the client that it is storing all the clients' data. Although PoR provides many advantages some disadvantages are also found wit PoR. The users or the clients cannot modify their data in the file. Some security problems are also found and also the computational cost is found to be high with PoR. Also some integrity problems are also found. To tackle all the challenges faced by PoR a new scheme OPoR (Outsourced Proof of Retrievability) is used. It includes two independent servers the cloud storage server and the cloud audit server. The cloud audit server has some additional capabilities that the clients do not have and this is also responsible for preprocessing the data instead of the clients. By using OPoR dynamic data operations can be performed. And all the security concerns are avoided.

2. RELATED WORK

Dynamic data along with fairness is dealt with [2]. Existing POR solutions can only deal with static data, and actually are not secure when used to deal with dynamic data. Motivated by the need to securely deal with dynamic data, the first dynamic POR scheme is proposed for this purpose. A new property, called fairness is introduced, which is necessary to the setting of dynamic data.

The solution is based on two new tools, one is an authenticated data structure, and the other is an incremental signature scheme. The first building-block is a new authenticated data structure we call range-based 2-3 tree. The second building-block is a new incremental signature scheme called hash-compress-and-sign. The tailored incremental signature scheme is more efficient than the literature ones that operate in a more general setting because our incremental signing incurs constant hash operations rather than logarithmic many hash operations. Proof of Retrievability system s built systems that are both efficient and provably secure in [3]. 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. The first compact and provably secure proof of retrievability systems is created. The solutions allow for compact proofs with just one authenticator value in practice this can lead to proofs with as little as 40 bytes of communication. Two solutions with similar structure is presented. The first solution is privately verifiable and builds elegantly on pseudorandom functions (PRFs). The primary drawback of this first solution is that some sentinel blocks get used up with each audit. Then the second solution enumerates the problem discussed above and allows for publicly verifiable proofs. Both solutions rely on homomorphic properties to aggregate a proof into one small authenticator value.

3. EXISTING SYSTEM

According to the role of the verifier in the model, all the schemes available fall into two categories: private verifiability and public verifiability. Although achieving higher efficiency, schemes with private verifiability impose computational burden on clients. On the other hand, public verifiability alleviates clients from performing a lot of computation for ensuring the integrity of data storage. To be specific, clients are able to delegate a third party to perform the verification without devotion of their computation resources. In the cloud, the clients may crash unexpectedly or cannot afford the overload of frequent integrity checks. Another major concern among previous designs that is the support of dynamic data operation for cloud data storage applications. In cloud computing, the remotely stored electronic data might not only be accessed but also be updated by the clients, e.g., through block modification, deletion, insertion etc. to address this problems a new scheme OPoR (Outsourcing Proof of Retrievability) is proposed with two independent servers. Among the two independent servers, one server is for auditing and the other for storage of data i.e. the cloud audit server and the cloud storage server. Cloud audit server also known as the Third Party Auditor and has expertise and capabilities that clients do not have, is trusted to assess and expose risk of cloud storage services on behalf of the clients upon request. Cloud storage server is an entity, which is managed by cloud service provider has significant storage space and computation resource to maintain client’s data. The client is an entity that has large data files to be stored in the cloud and relies on the cloud for data maintenance and computation, can be either individual consumers or organizations. A proof of Retrievability (POR) is a compact proof by a file system (prover) to a client (verifier) that a target file F is intact, in the sense that the client can fully recover it. The prover refers to the cloud storage server and the verifier refers to the cloud audit server. And also in the proposed scheme multiple copies of same data is stored. This is done to recover the file when a file in a location is lost. The cloud storage server is meant to provide the proof to the cloud audit server whether it is storing the files multiple times. If the file is found in different storage

location from its original location means it is referred as reset attack. The proposed scheme is proved secure against reset attacks in the strengthened security model and support dynamic data operations simultaneously. The modules found are

- Client Module
- Cloud Storage Server Module
- Cloud Audit Server Module
- Setup Phase
- Challenge Phase

Client Module: It is an entity model. It has large data files to be stored in the cloud and relies on the cloud for data maintenance and computation. It can be either individual consumer or organization. This module is used to send a data file through to a service provider.

Cloud Storage Server Module: The cloud storage server is used to provide a storage space for the entity to store their data. The CSS is required to provide integrity proof to the clients or cloud audit server during the integrity checking phase. The main usage of the server is used to maintain the entity with required storage space and manage the data.

Cloud Audit Server Module: The cloud audit server module is used to audit the data in the storage area. The auditing is in the format of public auditing that is done by the third party auditor. A TPA, is trusted to assess and expose risk of cloud storage services on behalf of the clients upon request. This should be done before uploading to the cloud storage server. The basic goal of PoR model is to achieve Proof of Retrievability.

Setup Phase: This property ensures that if an adversary can generate valid integrity proofs of any file. It is formally defined by the following game between a challenger C and an adversary A. Where C plays the role of the audit server (the client) and A plays the role of the storage server. The challenger C generate its key pair (pk, sk), and forwards pk to the adversary A. C initiates an empty table called Rlist. A can adaptively query an upload oracle with reset capability.

Challenge Phase: Adversary can adaptively make the following two kinds of oracle queries

- Integrity Verify
- Update

Integrity Verify: When a query on a file tag t comes, C runs the integrity verification protocol Integrity Verify {A C (pk, t)} with A Update: When a query on a file tag ˆt and a data operation request “update” comes, C runs the update protocol Update{A C(sk, ˆt, update)} with A.

A. Architecture

An architecture for cloud data storage is illustrated in Fig. 1. The architecture consists of three different network entities known as

_Client. An entity that has large data files to be stored in the cloud.

_Cloud storage server. An entity, which is managed by cloud service provider.

_Cloud audit server. A TPA, which has expertise and capabilities that clients do not have.

In the cloud paradigm, the clients outsource their data to the third party auditor also known as the cloud audit server to be relieved of the burden of storage and computation. As clients no longer possess their data, they should ensure that their data are being correctly stored and maintained.

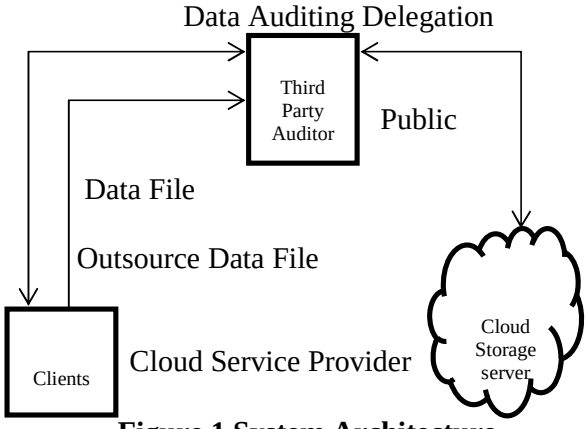

Figure.1 System Architecture

The third party auditor then upload their data to the cloud storage server which is managed by the cloud service provider.

CONCLUSION

In the proposed system OPoR, a new proof of retrievability for cloud storage is proposed. It includes two independent servers the cloud audit server and the cloud storage server. The cloud audit server is meant for auditing and the cloud storage server is for storage of dataThe cloud audit server is trustworthy and is introduced to preprocess and upload the data on behalf of the clients. The cloud audit server has expertise and capabilities that clients do not have, is trusted to assess and expose risk of cloud storage services on behalf of the clients upon request. The scheme also supports public verifiability and dynamic data operation simultaneously.

REFERENCES

[1] Jin Li, Xiao Tan, Xiaofeng Chen, Duncan S. Wong, and Fatos Xhafa , “OPoR: Enabling Proof of Retrievability in Cloud Computing with Resource-Constrained Devices” in IEEE transactions on cloud computing, vol. 3, no. 2, april / june 2015

[2] Q. Zheng and S. Xu, “Fair and dynamic proofs of retrievability,” in Proc. 1st ACM Conf. Data Appl. Security Privacy, 2011, pp. 237– 248.

[3] H. Shacham and B. Waters, “Compact proofs of retrievability,” in Proc. 14th Int. Conf. Theory Appl. Cryptol. Inf. Security, 2008, pp. 90– 107.

[4] Shingare Vidya Marshal, “Secure Audit Service by Using TPA for Data Integrity in Cloud System” in IJITEE, Volume-3, Issue-4, September 2013, ISSN: 2278-3075.

[5] K. D. Bowers, A. Juels, and A. Oprea, “Proofs of retrievability: Theory and implementation,” in Proc. ACM Workshop Cloud ComputingSecurity, 2009, pp. 43–54.

[6] Q. Zheng, and S. Xu, “Secure and efficient proof of storage with deduplication,” in Proc. ACM Conf. Data Appl. Security Privacy, 2012, pp. 1–12.

[7] A. Juels and B. S. Kaliski Jr, “Pors: Proofs of retrievability for large files,” in Proc. 14th ACM Conf. Comput. Commun. Security, 2007,pp. 584–597.