ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

Vol. 3, Issue 11, November 2015

# A Survey on Cloud Storage Security Based on Trusted Third Party Auditing

Priyanka S. Nangare[1], Prof. Manjusha Jagtap[2]

M.E. Student, Dept. of Computer Engineering, DPCOE, Pune, India[1]

Assistant Professor, Dept. of Computer Engineering, DPCOE, Pune, India[2]

**ABSTRACT**: Protect outsider data stored in the cloud against fraud, adding fault tolerance to cloud storage together with data accuracy and consistency checking and failure reformation becomes critical. Recently, regenerating codes have gained popularity because of their lower repair bandwidth while working properly in case of failure. Existing remote checking methods for regenerating-coded data provide private auditing, it must require data owners to always stay online and handle auditing and repairing also, which is sometimes impossible. In this paper, we propose a public auditing scheme for the regenerating-code-based cloud storage. To solve the reconstruction problem of failed authenticators when the data owner is not present, we introduce a proxy, which is authorized to reconstruct the authenticators, into the traditional public auditing system model. Moreover, we design a innovative public verifiable authenticator, which is generated by a couple of keys and can be reconstructed using limited keys. Thus, our scheme can thoroughly release data owners from staying online. In addition, we randomize the encode coefficients with a pseudorandom function to preserve data privacy. Extensive security analysis shows that our scheme is proved secure under random oracle model and also experimental evaluation shows that our scheme is highly efficient and can be conceivably integrated into the regenerating- code-based cloud storage.

**KEYWORDS**: Cloud storage, regenerating codes, public audit, privacy preserving, authenticator reconstruction, proxy, and authorized, provable secure.

## I. INTRODUCTION

Cloud storage is becomes popular because it provide a flexible on-demand data outsourcing service with appealing benefits: ease of the burden for storage management, universal data access with location independence, and avoidance of chief cost on hardware, software, and personal maintenances, etc., Nevertheless, this new paradigm of data introducing service also brings new security risk toward users data, thus making individuals or enterprisers still feel unsure. It is noted that data owners loss his control over their outsourced data; thus, maintain the correctness, availability and purity of the data become risky. On the one hand, the cloud service is usually faced with a broad range of internal/external attackers, who would maliciously delete or corrupt users' data; on the other hand, the cloud service providers may act maliciously, attempting to hide data loss or corruption and claiming that the files are still stored correctly for position or financial reasons. Thus it important for users to implement efficient conventions to perform serially verifications of their outsourced data to make sure that the cloud indeed maintains their data correctly.

## II. RELATED WORK

### [1] "ABOVE THE CLOUDS: A BERKELEY VIEW OF CLOUD COMPUTING,"

**From This Paper we Referred-**

The IT organizations have expresses worry about critical issues (such as security) that exist with the broad implementation of cloud computing. These types of worry originate from the fact that data is stored remotely from the customer's location; in fact, it can be stored at any location. Security is one of the most argued-about issues in the cloud computing field; certain enterprises look towords cloud computing warily due to projected security risks.

DOI: 10.15680/IJIRCCE.2015. 0311174

**[2] "Provable data possession at untrusted stores,"**
**From This Paper we Referred-**

This keynote paper: In Cloud Computing the application software and databases moves to the centralized large data centers, where the management of the data and services may not be faithfull. This unique paradigm brings about many new security challenges, which have not been understood well. This paper mark the problem of to make sure the integrity of data storage in Cloud Computing.

**[3] PORs: Proofs of Retrievability for large files**
**From This Paper we Referred-**

The distributed storage systems apply excess coding techniques to stored data. One form of repetition is based on regenerating codes, which can minimize the amount of data transferred when repairing a failed storage node. Existing regenerating codes mainly require extant storage nodes encode data during repair.

**[4] Multiple-replica provable data possession**
**From This Paper we Referred-**

In this approach, cloud computing is to account all the resources at one place which is in the form a cluster and to perform the resource allocation by different users. They defined the user request in the form of query. Cloud Computing devices being able to exchange data like text files and business information using internet. Technically, it is completely different from an infrared. Using new models Iaas, Paas and Saas.

**[5] HAIL: A high-availability and integrity layer for cloud storage**
**From This Paper we Referred-**

In this paper to provide fault tolerance for cloud storage to spread data across multiple cloud vendors. However, if a cloud go through a permanent failure and loses all its data, it is necessary to repair this lost data with the help of the other residual clouds to save data redundancy. This paper presented a proxy-based storage system for fault-tolerant multiple-cloud storage like NCCloud, which achieves cost-effective repair for a permanent single-cloud failure.

## III. SCOPE OF RESEARCH

1. We design a natural homomorphic authenticator based on BLS signature, which can be generated by a couple of secret keys and verified publicly. Utilizing the linear subspace of the regenerating codes, the authenticators can be computed effectively. Also, it can be adapted for data owners equipped with low end computation devices (e.g. Tablet PC etc.) in which they need to sign the original blocks.

2. To the best of our knowledge, our scheme is the first to allow privacy-preserving public auditing for regenerating code- based cloud storage. The coefficients are masked by a PRF(Pseudorandom Function) at the time of Setup phase to avoid leakage of the original data. This method is lightweight and not introduce any computational overhead to the cloud servers or TPA.

3. Our scheme completely gives relief to data owners from online burden for the reconstruction of blocks and authenticators at faulty servers and it provides the privilege to a proxy for the reparation.

4. Optimization measures are taken to improve the flexibility and efficiency of our auditing scheme; hense, the storage overhead of servers, the computational overhead of the data owner and communication overhead during the audit phase effectively reduced.

DOI: 10.15680/IJIRCCE.2015. 0311174

### *Applications:*

1. To protect outsourced data in cloud storage.
2. Our scheme can completely release data owners from online burden.

## IV. PROPOSED METHODOLOGY AND DISCUSSION

In this paper, we focus on the integrity verification problem in regenerating-code-based cloud storage, especially with the functional repair strategy. Similar studies have been performed by Bo Chen et al. and H. Chen el al. separately and independently. Extend the single-server CPOR scheme (private version in) to the regenerating- code-scenario; designed and implemented a data integrity protection (DIP) scheme for FMSR based cloud storage and the scheme is adapted to the thin-cloud setting1. However, both of these scheme are designed for private audit, only the data owner is allowed to verify the integrity and repair the servers which are faulty. Considering the large size of the outsourced data and the user's constrained resource capability, the tasks of auditing and reparation in the cloud can be dangerous and expensive for the users.

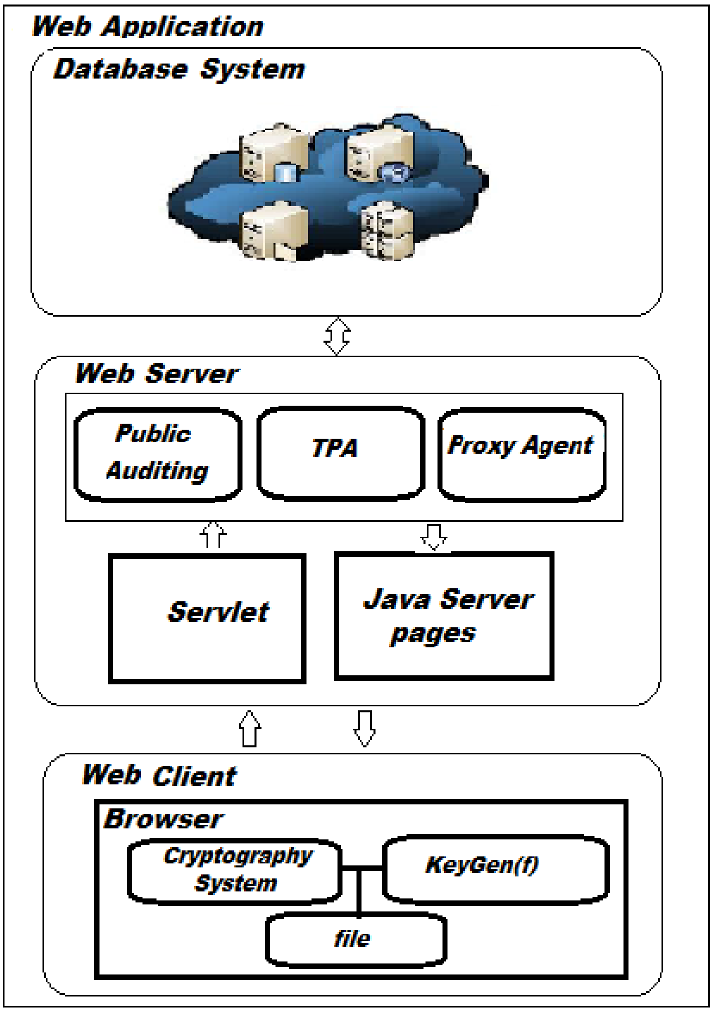

**Fig. 1: System Architecture**

**Advantages of Proposed System:**

1. Public Auditability: to allow TPA to verify the intactness of the data in the cloud on demand without giving additional online burden to the data owner.
2. Storage Soundness: to ensure that the cloud server can never pass the auditing procedure except that it indeed manages the owner's data intact.
3. Privacy Preserving: to ensure that neither the auditor as well as nor the proxy can derive users' data content from the auditing and reparation process.
4. Authenticator Regeneration: the authenticator of the re- paired blocks can be correctly regenerated in the absence of the data owner.
5. Error Location: to ensure that the wrong server can be quickly indicated when data corruption is detected.

**Methodology Used:**

1. **Setup**: The data owner maintains this procedure to initialize the auditing scheme.

**KeyGen(1κ) → (pk, sk):** This polynomial-time algorithm is run by the data owner to initialize its public as well as secret parameters by taking a security parameter κ as input for that .
**Degelation(sk) → (x):** This algorithm represents the interaction between the data owner as well as proxy. The data owner delivers partial secret key x to the proxy through a secure approach.
**SigAndBlockGen(sk, F):** This polynomial time algorithm is run by the data owner and takes the secret parameter sk as well as the original file F as input, and then outputs a coded block set, an authenticator set and a file tag t.

2. **Audit**: The cloud servers and TPA both interact with one another to take a random sample on the blocks and check the data intactness in this procedure.

**Challenge(Finfo) → (C):** This algorithm is performed by the TPA with the information of the file Finfo as input and a challenge C as output.
**ProofGen→ (P):** This algorithm is run by each cloud server with input challenge C, coded block set and authenticator set, then it outputs a proof P.
**V erify(P, pk, C) → (0, 1):** This algorithm is run by TPA immediately after a proof is received. Taking the proof P, public parameter pk and the corresponding challenge C as input, it output is 1 and if the verification passed and 0 otherwise.

3. **Repair:** In the absence of the data owner, the proxy interacts with the cloud servers to repair the wrong server which detected by the auditing process.

**ClaimForRep(Finfo) → (Cr):** This algorithm is similar with the Challenge() algorithm in the Audit phase, but outputs a claim for repair Cr.
**GenForRep→ (BA):**The cloud servers run this algorithm upon receiving the Cr and finally output the block and authenticators set BA with another two inputs.
**BlockAndSigReGen(Cr,BA):** The proxy implements this algorithm with the claim Cr and responses BA from each server as input, and outputs a new coded block set and authenticator set if successful, outputting ⊥ if otherwise.

## V. CONCLUSION

In this paper, we propose a privacy-preserving public auditing system for data storage security in Cloud Computing.
We use the homomorphic linear authenticator and random masking to guarantee that the TPA would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process, which not only eliminates the burden of cloud user from the endless and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage. Considering TPA may concurrently handle multiple audit sessions from different users for their data files which are outsourced, we further extend our privacy preserving public auditing agreement into a multi-user setting, where the TPA can perform the multiple auditing tasks in a batch manner for better efficiency. Extensive analysis shows that our schemes are provably secure and highly efficient.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

Vol. 3, Issue 11, November 2015

## REFERENCES

[1] M. Armbrust et al., "Above the clouds: A Berkeley view of cloud computing," Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, CA, USA, Tech. Rep. UCB/EECS-2009-28, 2009.
[2] G. Ateniese et al., "Provable data possession at untrusted stores," in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), New York, NY, USA, 2007, pp. 598–609.
[3] A. Juels and B. S. Kaliski, Jr., "PORs: Proofs of retrievability for large files," in Proc. 14th ACM Conf. Comput. Commun. Secur., 2007, pp. 584–597.
[4] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PDP: Multiple-replica provable data possession," in Proc. 28th Int. Conf. Distrib. Comput. Syst. (ICDCS), Jun. 2008, pp. 411–420.
[5] K. D. Bowers, A. Juels, and A. Oprea, "HAIL: A high-availability and integrity layer for cloud storage," in Proc. 16th ACM Conf. Comput. Commun. Secur., 2009, pp. 187–198.