Privacy Preserved Public Auditing in Cloud Storage using Regenerating Codes
Reshma .B1, Dr.Sanjay Srivastava2
M.Tech Student1, Associate Professor2
Department of CS&E
MGM CoET, Noida, India

Abstract:
Cloud computing is a set of IT services provided by a third party provider who owns the infrastructure to the customers in need of it over the network. In still simpler terms, cloud computing refers to the delivery of computing services such as software, storage, servers, networking, analytics, databases etc. Users can store their data on cloud and access it independent of location and time. Before sending the data to the cloud the data block would be signed to provide security to user’s data. One of the primary aims of cloud computing is to ensure the user to use the cloud storage as if it is local, not worrying about data integrity. Thus arises the indispensability to introduce public auditability with the help of a Third Party Auditor (TPA) to audit the stored data on behalf of the user. In the absence of the data owner, to repair the wrong server detected by the auditing process we also introduce a proxy.

Keywords: Cloud storage, public auditing, third party auditor, regenerating codes, proxy.

I. INTRODUCTION

Cloud computing is a model designed to provide on-demand network access to shared set of computing resources which can be rapidly provided and released with minimum management effort from the user’s side and minimal service provider interaction. An organization’s approach for building an IT infrastructure has changed with the advent of cloud computing. Cloud computing promises to maximize the utilization probability using sharing of resources. Cloud computing is an emerging technology that has become part and parcel of everyone’s life today. Clouds offer us a wide range of benefits like configurable, computing resources, service flexibility and economic savings. However, to wide adoption of clouds the main obstacles are privacy and security concerns. This is what makes the user reluctant to use cloud storage. Data users lose control over their data as soon as it is outsourced which is a matter of concern for many users. On one side, there is a threat of external/internal factors which might cause damage to user’s data whereas on the other side cloud service provider himself may act dishonestly, trying to prove storage intactness even in cases of corruption. The existing methods focused on private auditability where users were demanded to stay online and handle the auditing process which is not always possible. In addition to this, users were also burdened by the repair of the corrupted files. Here, In this paper we introduce a Third Party Auditor (TPA) to check the integrity of the stored on cloud. We introduce Regenerating codes to reduce the repair bandwidth in distributed storage. Here, we completely release owner from the online burden to regenerate the corrupted blocks and authenticators at faulty servers and also we introduce a proxy to handle reparation. Various methods for public auditing in the cloud storage are also discussed.

Our contribution can be summarized as Follows:
• Based on BLS signature, we design a novel homomorphic authenticator which can be generated by a couple of secret keys and verified publicly.
• Our focus is to decrease the overhead on data owner. So, a Third Party Authenticator (TPA) is introduced to audit the data on behalf of the owner.
• We strived to completely release the online burden of users by introducing a proxy to carry out reparation.

II. FUNDAMENTALS AND PROBLEM DEFINITION
A. Preliminaries
1) Homomorphic Encryption: It is a kind of encryption that allows computations to be carried out on ciphertext, thus generating an encrypted result which when decrypted matches the result of operations performed on the plaintext.
2) Regenerating codes: These are a class of codes in distributed storage systems which deal with the bandwidth needed for repairing of a failed node with the amount of data stored per node on the network. An \([m,k,d]\) regenerating code helps to recover the data by establishing connection with any \(k\) out of \(m\) nodes in the network. Assuming cloud storage to be a collection of \(m\) servers, data file \(D\) is encoded and a copy is stored redundantly in all these servers. To retrieve data file \(D\) we can connect to any \(k\)-out-of-\(m\) storage servers. This is known as MDS2 property.

Detection of data corruption at a random server forces client to contact \(r\) healthy servers to download \(B’\) bits from each of them. This is how corrupted blocks are regenerated without the necessity to recover the entire file.
There are two types of repair strategy based on if corrupted files need to be generated exactly. After the repair, if repaired server stores an exact copy of the original file, aka replica it is called as exact repair strategy. If the corrupted blocks are
different from the newly generated blocks, it is functional repair strategy.

B. The System Model
The system model is depicted in figure 1. It comprises of four entities: Data owner, who has limited resources and wants to store enormous data of his on cloud. Cloud Service provider who owns the infrastructure and provide the same on pay per use basis. A Third Party Auditor (TPA) is introduced to carry the auditing process on behalf of the owner of the data. TPA is a trusted entity which is partial towards neither data owner nor the cloud service provider. A proxy server is another entity in our system which is delegated to carry out reparation procedure. Proxy server, a semi-trusted entity regenerates the authenticators and data blocks on failed servers on behalf of user. Data owner is assumed to be having limited resources. Also that there are more chances that he might go offline as soon as the upload procedure is finished. The proxy, which is expected to be always online is more powerful than data owner but less than the cloud service provider with respect to memory and computation capability. Since TPA is given the complete responsibility of auditing and proxy, the delegation of repairing data owners are completely relieved by the online burden.

C. Threat Model
- **Curious TPA**: Third Party Auditor is assumed to be honest but curious. The complete auditing process is performed by TPA honestly but not to forget to mention that it is curious about the user’s data stored on the cloud.
- **Semi trusted Proxy**: Proxy won’t conspire with the servers but may try to fake authenticator for invalid data blocks to pass the verification process.

D. Design Goals
In order to store data on cloud efficiently and correctly the proposed system is expected to meet the following requirements:
- **Storage Intactness**: Cloud server cannot pass the audit successfully unless it is managing the user’s data intact.
- **Public Auditability**: Third Party Authenticator is delegated to carry out the auditing procedure without demanding user to do auditing.
- **Privacy Preserving**: To ensure the non-leakage of data content to either or both the proxy and the auditor during auditing and reparation process.
- **Authenticator Regeneration**: Data owner is relieved from the regeneration process of authenticator of repaired blocks.

III. PROPOSED SYSTEM
There are so many researches that have taken place regarding public auditing in cloud storage. First, A fixed file layout does not exist in the regenerating code based cloud storage unlike cloud storage based on the ancient erasure codes or replication. If functional repair strategy is adopted repaired blocks would be different from the original blocks with high probability. Then, how to regenerate the authenticators for repaired blocks. One of the researches demanded data owner to handle regeneration, which is not practical. These issues are addressed by the system we have proposed here. First, A BLS based authenticator is constructed. It uses homomorphic property and linearity relation to generate authenticators. Second, information of the segments, blocks and server indices are stored in authenticator to make our scheme more secure. Also, batch verification is performed to reduce the bandwidth. Moreover, with the help of masking of coding coefficients using
keyed PRF, our system can be enhanced to provide privacy preserving.

IV. CONCLUSION

In the proposed paper, various techniques and methods which facilitate privacy preserving public auditing system for secured data storage have been studied. The role of a Third Party Authenticator in data integrity checking and data security on cloud has been explained. The various methods for secured public auditing are discussed. The various modules are created using web application to understand the overall system of this project based on the auditing of data files in a secure manner for regenerating codes. This paper is about public auditing scheme in cloud for regenerating codes. The owner gives privilege to the TPA to audit the owner file. To protect the data information from TPA, the encrypted data is stored by the owner. The owner while uploading the file it generates the public and secret key and shares the partial key with the proxy server for security reasons of the outsourced data. The proxy server gets privilege from the owner and recovers the coded blocks of data in different servers. This paper also can be enhanced in the future by using the batch auditing schemes for regenerating codes. This makes the system more reliable and lower the cost for large files.

V. REFERENCES

[1]. Boyang Wang, Baochun Li and Hui Li, “Public auditing for shared data with efficient user Revocation in the cloud”

[2]. KaiHe,ChuanHeHuang,KanYangandJiaoliShi,“Identity-preserving public auditing for shared cloud data,” in the 23rd IEEE International Symposium on Quality ofService (IWQOS),2015.


[21]. Guangyang Yang, HuiXia, Wenting Shen, Xiuxiu Jiangand JiaYu, “Public data auditing with constrained

