

## Data Protection Based On Dynamic Encryption for Secure Cloud Computing

Akash Jain<sup>1</sup>, Prasad Mancharkar<sup>2</sup>, Govinda Mahajan<sup>3</sup>

<sup>1</sup>(Computer Department, NMIET/ University of Pune, India)

<sup>2</sup>(Computer Department, NMIET/ University of Pune, India)

<sup>3</sup>(Computer Department, NMIET/ University of Pune, India)

---

**Abstract :** Cloud Computing is the long dreamed vision of computing as a utility, where users can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. By data outsourcing, users can be relieved from the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the possibly large size of outsourced data makes the data integrity protection in Cloud Computing a very challenging and potentially formidable task, especially for users with constrained computing resources and capabilities. Thus, enabling public auditability for cloud data storage security is of critical importance so that users can resort to an external audit party to check the integrity of outsourced data when needed. To securely introduce an effective third party auditor (TPA), the following two fundamental requirements have to be met: 1) TPA should be able to efficiently audit the cloud data storage without demanding the local copy of data, and introduce no additional on-line burden to the cloud user; 2) the third party auditing process should bring in no new vulnerabilities towards user data privacy. In this paper, we utilize and uniquely combine the public key based homomorphic authenticator with random masking to achieve the privacy-preserving public cloud data auditing system, which meets all above requirements. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend our main result into a multi-user setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis shows the proposed schemes are provably secure and highly efficient.

**Keywords:** Cloud computing, Dynamic Encryption, Third Party Auditor (TPA), Threats.

---

### I. INTRODUCTION

The focus of this thesis to add a new dimension to Internet-Cloud Computing has been envisioned as the next generation architecture of IT enterprise, due to its long list of unprecedented advantages in the IT history: on-demand self-service, ubiquitous network access, location independent resource pooling, rapid resource elasticity, usage-based pricing and transference of risk. As a disruptive technology with profound implications, Cloud Computing is transforming the very nature of how businesses use information technology. One fundamental aspect of this paradigm shifting is that data is being centralized or outsourced into the Cloud. From users perspective, including both individuals and IT enterprises, storing data remotely into the cloud in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with independent geographical locations, and avoidance of capital expenditure on hardware, software, and personnel maintenances, etc. While Cloud Computing makes these advantages more appealing than ever, it also brings new and challenging security threats towards users' outsourced data. Since cloud service providers (CSP) are separate administrative entities, data outsourcing is actually relinquishing user's ultimate control over the fate of their data. As a result, the correctness of the data in the cloud is being put at risk due to the following reasons. First of all, although the infrastructures under the cloud are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal and external threats for data integrity. Examples of out ages and security breaches of noteworthy cloud services appear from time to time. Secondly, for the benefits of their own, there do exist various motivations for cloud service providers to behave unfaithfully towards the cloud users regarding the status of their outsourced data. Examples include cloud service providers, for monetary reasons, reclaiming storage by discarding data that has not been or is rarely accessed or even hiding data loss incidents so as to maintain a reputation. In short, although outsourcing data into the cloud is economically attractive for the cost and complexity of long-term large-scale data storage, it does not offer any guarantee on data integrity and availability. This problem, if not properly addressed, may impede the successful deployment of the cloud architecture. As users no longer physically possess the storage of their data, traditional cryptographic primitives for the purpose of data security protection cannot be directly adopted. Thus, how to efficiently verify the correctness of outsourced cloud data without the local copy of data files becomes a big challenge for data storage security in Cloud Computing. Note that simply downloading the data for its integrity verification is not a practical solution due to the expensiveness in I/O cost

and transmitting the file across the network. Besides, it is often insufficient to detect the data corruption when accessing the data, as it might be too late to recover the data loss or damage. Considering the large size of the outsourced data and the user's constrained resource capability, the ability to audit the correctness of the data in a cloud environment can be formidable and expensive for the cloud users. Therefore, to fully ensure the data security and save the cloud users' computation resources, it is of critical importance to enable public auditability for cloud data storage so that the users may resort to a third party auditor (TPA), who has expertise and capabilities that the users do not, to audit the outsourced data when needed. Based on the audit result, TPA could release an audit report, which would not only help users to evaluate the risk of their subscribed cloud data services, but also be beneficial for the cloud service provider to improve their cloud based service platform. In a word, enabling public risk auditing protocols will play an important role for this nascent cloud economy to become fully established, where users will need ways to assess risk and gain trust in Cloud. Recently, the notion of public auditability has been proposed in the context of ensuring remotely stored data integrity under different systems and security models. Public auditability allows an external party, in addition to the user himself, to verify the correctness of remotely stored data. However, most of these schemes do not support the privacy protection of users' data against external auditors, i.e., they may potentially reveal user data information to the auditors, as will be discussed in Section III-C. This severe drawback greatly affects the security of these protocols in Cloud Computing. From the perspective of protecting data privacy, the users, who own the data and rely on TPA just for the storage security of their data, do not want this auditing process introducing new vulnerabilities of unauthorized information leakage towards their data security. Moreover, there are legal regulations, such as the US Health Insurance Portability and Accountability Act (HIPAA), further demanding the outsourced data not to be leaked to external parties. Exploiting data encryption before outsourcing is one way to mitigate this privacy concern, but it is only complementary to the privacy-preserving public auditing scheme to be proposed in this paper. Without a properly designed auditing protocol, encryption itself cannot prevent data from "flowing away" towards external parties during the auditing process. Thus, it does not completely solve the problem of protecting data privacy but just reduces it to the one of managing the encryption keys. Unauthorized data leakage still remains a problem due to the potential exposure of encryption keys. Therefore, how to enable a privacy-preserving third-party auditing protocol, independent to data encryption, is the problem we are going to tackle in this paper. Our work is among the first few ones to support privacy-preserving public auditing in Cloud Computing, with a focus on data storage. Besides, with the prevalence of Cloud Computing, a foreseeable increase of auditing tasks from different users may be delegated to TPA. As the individual auditing of these growing tasks can be tedious and cumbersome, a natural demand is then how to enable TPA to efficiently perform the multiple auditing tasks in a batch manner, i.e., simultaneously. To address these problems, our work utilizes the technique of public key based homomorphic authenticator, which enables TPA to perform the auditing without demanding the local copy of data and thus drastically reduces the communication and computation overhead as compared to the straightforward data auditing approaches. By integrating the homomorphic authenticator with random masking, our protocol guarantees that TPA could not learn any knowledge about the data content stored in the cloud server during the efficient auditing process. The aggregation and algebraic properties of the authenticator further benefit our design for the batch auditing. Specifically, our contribution in this work can be summarized as the following three aspects:

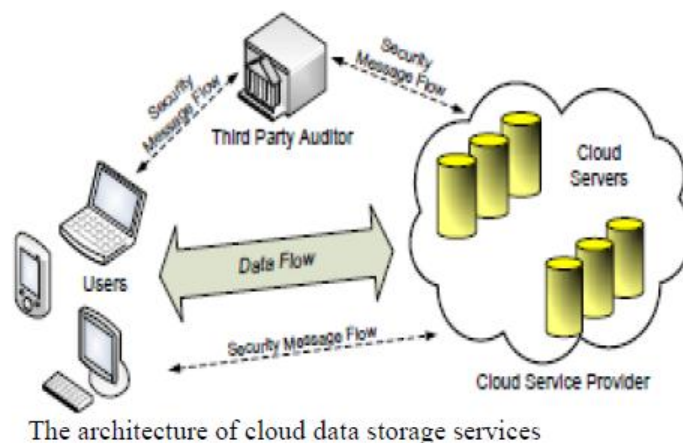
- 1) We motivate the public auditing system of data storage security in Cloud Computing and provide a privacy-preserving auditing protocol, i.e., our scheme supports an external auditor to audit user's outsourced data in the cloud without learning knowledge on the data content.
- 2) To the best of our knowledge, our scheme is the first to support scalable and efficient public auditing in the Cloud Computing. In particular, our scheme achieves batch auditing where multiple delegated auditing tasks from different users can be performed simultaneously by the TPA.
- 3) We prove the security and justify the performance of our proposed schemes through concrete experiments and comparisons with the state-of-the-art. The rest of the paper is organized as follows.

## **II. PROBLEM STATEMENT**

### **2.1: The System and Threat Model**

We consider a cloud data storage service involving three different entities, as illustrated in Fig. 1: the cloud user (U), who has large amount of data files to be stored in the cloud; the cloud server (CS), which is managed by Cloud Service Provider (CSP) to provide data storage service and has significant storage space and computation resources (we will not differentiate CS and CSP hereafter.); the third party auditor (TPA), who has expertise and capabilities that cloud users do not have and is trusted to assess the cloud storage service security on behalf of the user upon request. Users rely on the CS for cloud data storage and maintenance. They may also dynamically interact with the CS to access and update their stored data for various application purposes. The users may resort to TPA for ensuring the storage security of their outsourced data, while hoping to keep their

data private from TPA. We consider the existence of a semi-trusted CS as does. Namely, in most of time it behaves properly and does not deviate from the prescribed protocol execution. However, during providing the cloud data storage based services, for their own benefits the CS might neglect to keep or deliberately delete rarely accessed data files which belong to ordinary cloud users. Moreover, the CS may decide to hide the data corruptions caused by server hacks or Byzantine failures to maintain reputation. We assume the TPA, who is in the business of auditing, is reliable and independent, and thus has no incentive to collude with either the CS or the users during the auditing process. TPA should be able to efficiently audit the cloud data storage without local copy of data and without bringing in additional on-line burden to cloud users.



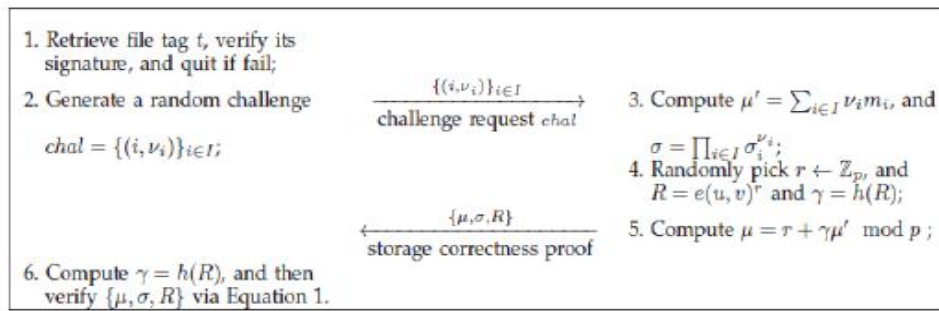
However, any possible leakage of user’s outsourced data towards TPA through the auditing protocol should be prohibited. Note that to achieve the audit delegation and authorize CS to respond to TPA’s audits, the user can sign a certificate granting audit rights to the TPA’s public key, and all audits from the TPA are authenticated against such a certificate. These authentication handshakes are omitted in the following presentation.

### III. PROPOSE SCHEMES

This section presents our public auditing scheme which provides a complete outsourcing solution of data – not only the data itself, but also its integrity checking. After introducing notations and brief preliminaries, we start from an overview of our public auditing system and discuss two straightforward schemes and their demerits. Then we present our main scheme and show how to extent our main scheme to support batch auditing for the TPA upon delegations from multiple users. Finally, we discuss how to generalize our privacy-preserving public auditing scheme and its support of data dynamics.

#### 3.1: Definitions and Framework

We follow a similar definition of previously proposed schemes in the context of remote data integrity checking and adapt the framework for our privacy-preserving public auditing system. A public auditing scheme consists of four algorithms (KeyGen, SigGen, GenProof, VerifyProof). KeyGen is a key generation algorithm that is run by the user to setup the scheme. SigGen is used by the user to generate verification metadata, which may consist of digital signatures. GenProof is run by the cloud server to generate a proof of data storage correctness, while VerifyProof is run by the TPA to audit the proof. Running a public auditing system consists of two phases, Setup and Audit. Privacy-preserving as long as it is impossible before giving our main result, we study two classes of schemes as a warm-up. The first one is a MAC-based solution which suffers from undesirable systematic demerits – bounded usage and stately verification, which may pose additional online burden to users, in a public auditing setting. This also shows that the auditing problem is still not easy to solve even if we have introduced a TPA. The second one is a system based on homomorphic linear authenticators (HLA), which covers many recent proofs of storage systems. We will pinpoint the reason why all existing HLA-based systems are not privacy preserving. The analysis of these basic schemes leads to our main result, which overcomes all these drawbacks. Our main scheme to be presented is based on a specific HLA scheme.



Privacy preserving Public auditing protocol

#### IV. CONCLUSION

In this paper, we propose a privacy-preserving public auditing system for data storage security in Cloud Computing. We utilize 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 tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage.

#### REFERENCES

- [1] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for storage security in cloud computing," in Proc. of IEEE INFOCOM'10, March 2010.
- [2] P. Mell and T. Grance, "Draft NIST working definition of cloud computing," Referenced on June. 3rd, 2009. <http://csrc.nist.gov/groups/SNS/cloudcomputing/index.html>.
- [3] 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, "Above the clouds: A Berkeley view of cloud computing," University of California, Berkeley, Tech. Rep. UCBECS-2009-28, Feb 2009.
- [4] Cloud Security Alliance, "Top threats to cloud computing,"2010, <http://www.cloudsecurityalliance.org>