RESEARCH ARTICLE OPEN ACCESS

Secure Cloud StorageForPrivacy-Preserving Public Audit

ShekhAhamadhusen D., Prof. Rahul Deshmukh

Abstract-
In Cloud Environment, using cloud storage service, users can remotely store their data and enjoy the on-demand high-quality applications and services from a shared pool of configurable computing resources, without the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the outsourced data makes the data integrity protection in cloud computing a formidable task, especially for users with constrained computing resources. Moreover, users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity. Thus, enabling public auditability for cloud storage is of critical importance so that users can resort to a third-party auditor (TPA) to check the integrity of outsourced data and be worry free. To securely introduce an effective TPA, the auditing process should bring in no new vulnerabilities toward user data privacy, and introduce no additional online burden to user. In this paper, we propose a secure cloud storage system supporting privacy-preserving public auditing. We further extend our result to enable the TPA to perform audits for multiple users simultaneously and efficiently. Extensive security and performance analysis show the proposed schemes are provably secure and highly efficient.

I.Introduction

Advances in networking technology and an increase in the need for computing resources haveprompted many organizations to outsource their storage and computing needs. This new economicand computing model is commonly referred to as cloud computing and includes various typesof services such as: infrastructure as a service (IaaS), where a customer makes use of a service

provider's computing, storage or networking infrastructure; platform as a service (PaaS), where acustomer leverages the provider's resources to run custom applications; and software as aservice (SaaS), where customers use software that is run on the providers infrastructure.Cloud infrastructures can be roughly categorized as either private or public. In a private cloud,the infrastructure is managed and owned by the customer and located on-premise (i.e., in thecustomers region of control). In particular, this means that access to customer data is under itscontrol and is only granted to parties it trusts. In a public cloud the infrastructure is owned andmanaged by a cloud service provider and is located o_-premise (i.e., in the service provider's regionof control). This means that customer data is outside its control and could potentially be grantedto untrusted parties.

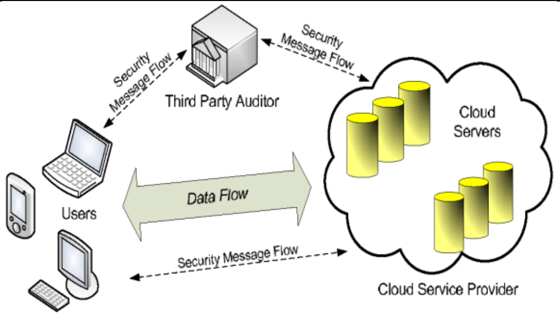

Fig. 1: The architecture of cloud data storage service

CLOUD computing has been envisioned as the next generation Information Technology (IT) architecture for enterprises, 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. From users' perspective, including both individuals and IT enterprises, storing data remotely to the cloud in a flexible on-demand manner brings appealing benefits: relief of the burden for storage management, universal data access with location independence, 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 toward users' outsourced data. Since Cloud Service Providers (CSP) is 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 outages and security breaches of noteworthy cloud services appear from time to time. Second, there do exist various motivations for CSP to behave unfaithfully toward the cloud users regarding their outsourced data status. For examples, CSP might reclaim storage for monetary reasons by discarding data that have not been or are rarely accessed, or even hide data loss incidents to maintain a reputation.

II. Literature Survey

We consider a cloud data storage service involving three different entities: the cloud user, who has large amount of data files to be stored in the cloud; the cloud server, which is managed by the cloud service provider 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, who has expertise and capabilities that cloud users do not have and is trusted to assess the cloud storage service reliability 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. As users no longer possess their data locally, it is of critical importance for users to ensure that their data are being correctly stored and maintained. In short, although outsourcing data to the cloud is economically attractive for long-term large-scale storage, it does not immediately offer any guarantee on data integrity and availability. This problem, if not properly addressed, may impede the success of cloud architecture. To save the computation resource as well as the online burden potentially brought by the periodic storage correctness verification, cloud users may resort to TPA for ensuring the storage integrity of their out sourced data. 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 consider the privacy protection of users' data against external auditors.

Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trust worthy. This unique paradigm brings about many new security challenges, which have not been well understood. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. In particular, we consider the task of allowing a Third Party Auditor (TPA), on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. The introduction of TPA eliminates the involvement of the client through the auditing of whether his data stored in the cloud is indeed intact, which can be important in achieving economies of scale for Cloud Computing. The support for data dynamics via the most general forms of data operation, such as block modification, insertion and deletion, is also a significant step toward practicality, since services in Cloud Computing are not limited to archive or backup data only. While prior works on ensuring remote data integrity often lacks the support of either public auditability or dynamic data operations, this paper achieves both. We first identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then show how to construct an elegant verification scheme for the seamless integration of these two salient features in our protocol design. In particular, to achieve efficient data dynamics, we improve the existing proof of storage models by manipulating the classic Merkle Hash Tree construction for block tag authentication. 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 show that the proposed schemes are highly efficient and provably secure[1].

We introduce a model for Provable Data Possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage systems. We present two provably-secure PDP schemes that are more efficient than previous solutions, even when compared with schemes that achieve weaker guarantees. In particular, the overhead at the server is low (or even constant), as opposed to linear in the size of the data. Experiments using our implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation[2].

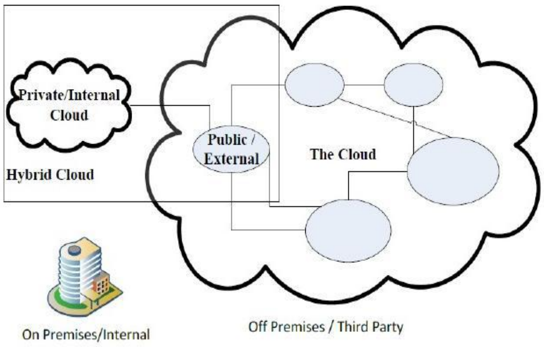

Figure 2.1.Types of cloud

Cloud computing is the long dreamed vision of computing as a utility, where data owners can remotely store their data in the cloud to enjoy on-demand high-quality applications and services from a shared pool of configurable computing resources. While data outsourcing relieves the owners of the burden of local data storage and maintenance, it also eliminates their physical control of storage dependability and security, which traditionally has been expected by both enterprises and individuals with high service-level requirements. In order to facilitate rapid deployment of cloud data storage service and regain security assurances with out sourced data dependability, efficient methods that enable on-demand data correctness verification on behalf of cloud data owners have to be designed. In this article we propose that publicly auditable cloud data storage is able to help this nascent cloud economy become fully established. With public auditability, a trusted entity with expertise and capabilities data owners do not possess can be delegated as an external audit party to assess the risk of outsourced data when needed. Such an auditing service not only helps save data owners' computation resources but also provides a transparent yet cost-effective method for data owners to gain trust in the cloud. We describe approaches and system requirements that should be brought into consideration, and outline challenges that need to be resolved for such a publicly auditable secure cloud storage service to become a reality[4].

III. Problem Statement

The cloud user, who has large amount of data files to be stored in the cloud; the cloud server, which is managed by the cloud service provider 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, who has expertise and capabilities that cloud users do not have and is trusted to assess the cloud storage service reliability 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. As users no longer possess their data locally, it is of critical importance for users to ensure that their data are being correctly stored and maintained. To save the computation resource as well as the online burden potentially brought by the periodic storage correctness verification, cloud users may resort to TPA for ensuring the storage integrity of their out sourced data, while hoping to keep their data private from TPA. Using third-party auditing service provides a cost-effective method for users to gain trust in cloud. We assume the TPA, who is in the business of auditing, is reliable and independent. However, it may harm the user if the TPA could learn the out sourced data after the audit[5]. Note that in our model, beyond users' reluctance to leak data to TPA; we also assume that a cloud server has no incentives to reveal their hosted data to external parties.

IV. The Proposed Mechanism

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. Our work also utilizes the technique of public key-based homomorphic linear authenticator or HLA 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 straight forward data auditing approaches. By integrating the HLA with random masking, our protocol guarantees that the TPA could not learn any knowledge about the data content stored in the cloud server (CS) during the efficient auditing process. The aggregation and algebraic properties of the authenticator further benefit our design for the batch auditing.

- Public audit ability -- to verify the correctness of the cloud data on demand without retrieving a copy of the whole data.
- It provides security and performance guarantees.

V. Methodology Used

Methodologies are the process of analyzing the principles or procedure for enabling secure external auditing process against data integrity with the support of homomorphic linear authenticator in public cloud environment.

MODULES NAMES

Service Provider

- Authentication
- Resource Provisioning

Third Party Auditor

- Authentication
- Auditing Process

Data Owner

- Authentication
- Key Maker
- Auditing Request

Authentication

If you are the new user going to consume the service then they have to register first by providing necessary details. After successful completion of sign up process, the user has to login into the application by providing username and exact password. The user has to provide exact username and password which was provided at the time of registration, if login success means it will take up to main page else it will remain in the login page itself..

SERVICE PROVIDER

- **Resource Provisioning**

The process of providing resources to customers or clients with accounts, the appropriate access to those accounts, all the rights associated with those accounts, and all of the resources necessary to manage the accounts. When used in reference to a client, provisioning can be thought of as a form of customer service.

THIRD PARTY AUDITOR

- **Auditing Process**

In public auditing module, the auditor perceives and recognizes the propositions before him for examination, collects evidence, evaluates the same and on this basis formulates his judgment which is communicated through his audit report.

DATA OWNER

- **Key Maker**

In this module, keys are generated according to the user setup in order to generate verification Meta data of uploaded file.

- **Auditing Request**

User can send the auditing request to external auditor along with the signatures and Meta data of the file. Then auditor will request for generated proof from service provider in order to do auditing process. Finally data owner will get the audit report.

PARTICLE FILTERING ALGORITHM

Public Key-Based Homomorphic Linear Authenticator (HLA)

HLA-based solution is to effectively support public auditability without having to retrieve the data blocks themselves, the HLA technique can be used. HLAs, like MACs, are also some unforgeable verification metadata that authenticate the integrity of a data block. The difference is that HLAs can be aggregated. It is possible to compute an aggregated HLA which authenticates a linear combination of the individual data blocks. At a high level, an HLA-based proof of storage system works as follow. The user still authenticates each element of F = $\{m_i\}$ by a set of HLAs. The TPA verifies the cloud storage by sending a random set of challenge $\{v_i\}$. The cloud server then returns $\mu = \sum_i vi * mi$ and its aggregated authenticator σcomputed from $\emptyset$.

STEP 1:Setup Phase: The cloud user runs KeyGen to generate the public and secret parameters.

STEP 2:SigGen: Given a data file F = $\{m_i\}$, the user runs SigGen to compute authenticator.

STEP 3: Audit Phase: The TPA first retrieves the file tag t. With respect to the mechanism we describe in the Setup phase, the TPA verifies the signature via secret key, and quits by emitting FALSE if the verification fails.

VI. Conclusion

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. Considering TPA may concurrently handle multiple audit sessions from different users for their outsourced data files, we further extend our privacy-preserving public auditing protocol into a multi-user setting, where the TPA can perform multiple auditing tasks in a batch manner for better efficiency. Extensive analysis shows that our schemes are provably secure and highly efficient.

References

[1] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Trans. Parallel and Distributed Systems, vol. 22, no. 5, pp. 847-859, May 2011.

[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 598-609, 2007.

[3] M.A. Shah, R. Swaminathan, and M. Baker, "Privacy-Preserving Audit and Extraction of Digital Contents," Cryptology ePrint Archive, Report 2008/186, 2008.

[4] C. Wang, K. Ren, W. Lou, and J. Li, "Towards Publicly Auditable Secure Cloud Data Storage Services," IEEE Network Magazine, vol. 24, no. 4, pp. 19-24, July/Aug. 2010.

[5] M.A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, "Auditing to Keep Online Storage Services Honest," Proc. 11th USENIX Workshop Hot Topics in Operating Systems (HotOS '07), pp. 1-6, 2007.

[6] R. Curtmola, O. Khan, and R. Burns, "Robust Remote Data Checking," Proc. Fourth ACM Int'l Workshop Storage Security and Survivability (StorageSS '08), pp. 63-68, 2008.

[7] A.L. Ferrara, M. Green, S. Hohenberger, and M. Pedersen, "Practical Short Signature Batch Verification," Proc. Cryptographers' Track at the RSA Conf. 2009 on Topics in Cryptology (CT-RSA), pp. 309-324, 2009.