Ensuring Dynamic Data Operation using Pre-computed verification tokens in Cloud Computing

#K.Lalitha[1], M.Tech Computer Science &EngineeringE-mail: lalli.konda@gmail.com
#Dr Siddharthaghosh[2], Professor, Department of CSE, E-mail: profsiddhartha@gmail.com
#Keshav memorial institute of technology, Narayanguda, Hyderabad, Telangana, India.

Abstract: Cloud storage enables users to remotely store their data and enjoy the on-demand high quality cloud applications without the burden of local hardware and software management. Though the benefits are clear, such a service is also relinquishing users' physical possession of their outsourced data, which inevitably poses new security risks towards the correctness of the data in cloud. In order to address this new problem and further achieve a secure and dependable cloud storage service, we propose in this paper a flexible distributed storage integrity auditing mechanism, utilizing the homomorphic token and distributed erasure-coded data. The proposed design allows users to audit the cloud storage with very lightweight communication and computation cost. The auditing result not only ensures strong cloud storage correctness guarantee, but also simultaneously achieves fast data error localization, i.e., the identification of misbehaving server. Considering the cloud data are dynamic in nature, the proposed design further supports secure and efficient dynamic operations on outsourced data, including block modification, deletion, and append. Analysis shows the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

Index Terms:Data integrity, Dependable distributed storage, Error localization, Data dynamics, Cloud Computing, Dynamic environment, Mutual trust, Access control.

1. INTRODUCTION

Several trends are opening up the era of CloudComputing, which is an Internet-based developmentand use of computer technology. The ever cheaper andmore powerful processors, together with the software asa service (SaaS) computing architecture, are transformingdata centers into pools of computing service on a hugescale. The increasing network bandwidth and reliable yetflexible network connections make it even possible thatusers can now subscribe high quality services from dataand software that reside solely on remote data centers.Since the data owner physically releases sensitive data to a remote CSP, there are some concerns regarding confidentiality, integrity, and access control of the data. The confidentiality feature can be guaranteed by the owner via encrypting the data before outsourcing toremote servers. For verifying data integrity over cloud servers, researchers have proposed provable data possession technique to validate the intactness of data stored on remote sites. A number of PDP protocols have been presented to efficiently validate the integrity of data. Proof of retrievability was introduced as a stronger technique than PDP in the sense that the entire data file can be reconstructed from portions of the data that are reliably stored on the servers.

Commonly, traditional access control techniques assume the existence of the data owner and the storage servers in the same trust domain. This assumption, however, no longer holds when the data is outsourced to a remote CSP, which takes the full charge of the outsourced data management, and resides outside the trust domain of the data owner. A feasible solution can be presented to enable the owner to enforce access control this solution, the data is encrypted under a certain key, which is shared only with the authorized users. The unauthorized users, including the CSP, are unable to access the data since they do not have the decryption key. This general solution has been widely incorporated into existing schemes, which aim at providing data storage security on untrusted remote servers. Another class of solutions utilizes attribute-based encryption to achieve fine-grained access control. Different approaches have been investigated that encourage the owner to outsource the data, and offer some sort of guarantee related to the confidentiality, integrity, and access control of the outsourced data. These approaches can prevent and detect malicious actions from the CSP side. On the other hand, the CSP needs to be safeguarded from a dishonest owner, who attempts to get illegal compensations by falsely claiming data corruption over cloud servers. This concern, if not properly handled, can cause the CSP to go out of business.

Our work is among the first few ones in thisfield to consider distributed data storage security inCloud Computing. Our contribution can be summarizedas the following three aspects:

1) Compared to many of its predecessors, which onlyprovide binary results about the storage status acrossthe distributed servers, the proposed scheme achievesthe integration of storage correctness insurance and dataerror localization, i.e., the identification of misbehavingserver(s).

2) Unlike most prior works for ensuring remote dataintegrity, the new scheme further supports secure andefficient dynamic operations on data blocks, including:update, delete and append.

3) The experiment results demonstrate the proposedscheme is highly efficient. Extensive security analysisshows our scheme is resilient against Byzantine failure,malicious data modification attack, and even server colludingattacks.

2. PROBLEM STATEMENT

Representative network architecture for cloud storageservice architecture is illustrated differentnetwork entities can be identified as follows:

• User: an entity, who has data to be stored in thecloud and relies on the cloud for data storage andcomputation, can be either enterprise or individualcustomers.

• Cloud Server (CS): an entity, which is managed by*cloud service provider* (CSP) to provide data storageservice and has significant storage space and computationresources (we will not differentiate CS andCSP hereafter.).

• Third Party Auditor (TPA): an optional TPA, whohas expertise and capabilities that users may nothave, is trusted to assess and expose risk of cloudstorage services on behalf of the users upon request.

The cloud computingstorage model considered in this work consists offour main components: (i) a dataowner that can be an organization generating sensitivedata to be stored in the cloud and made availablefor controlled external use; (ii) a CSP who managescloud servers and provides paid storage space on itsinfrastructure to store the owner's files and make themavailable for authorized users; (iii) authorized users a set of owner's clients who have the right to accessthe remote data; and (iv) a trusted third party (TTP), anentity who is trusted by all other system components,and has capabilities to detect/specify dishonest parties.

3. RELATED WORK

In cloud data storage, a user stores his data through aCSP into a set of cloud servers, which are running in asimultaneous, cooperated and distributed manner. Dataredundancy can be employed with technique of erasurecorrectingcode to further tolerate faults or server crashas user's data grows in size and importance. Thereafter,for application purposes, the user interacts with the3cloud servers via CSP to access or retrieve his data.In some cases, the user may need to perform blocklevel operations on his data. The most general forms ofthese operations we are considering are block update,delete, insert and append. Note that in this paper, weput more focus on the support of file-oriented cloudapplications other than non-file application data, suchas social networking data. In other words, the clouddata we are considering is not expected to be rapidlychanging in a relative short period.

The CSP is untrusted, and thus theconfidentiality and integrity of data in the cloud maybe at risk. For economic incentives and maintaining
a reputation, the CSP may hide data loss, or reclaimstorage by discarding data that has not been or is rarelyaccessed. To save the computational resources, the CSPmay totally ignore the data-update requests, or executejust a few of them. Hence, the CSP may return damagedor stale data for any access request from the authorizedusers. Furthermore, the CSP may not honor the accessrights created by the owner, and permit unauthorizedaccess for misuse of confidential data.On the other hand, a data owner and authorized usersmay collude and falsely accuse the CSP to get a certainamount of reimbursement. They may dishonestly claimthat data integrity over cloud servers has been violated,or the CSP has returned a stale file that does not matchthe most recent modifications issued by the owner.

As users no longer possess their data locally, it is ofcritical importance to ensure users that their data arebeing correctly stored and maintained. That is, usersshould be equipped with security means so that theycan make continuous correctness assurance (to enforcecloud storage service-level agreement) of their storeddata even without the existence of local copies. In casethose users do not necessarily have the time, feasibility orresources to monitor their data online, they can delegatethe data auditing tasks to an optional trusted TPA oftheir respective choices. However, to securely introducesuch a TPA, any possible leakage of user's outsourceddata towards TPA through the auditing protocol shouldbe prohibited.In our model, we assume that the point-to-point communicationchannels between each cloud server and theuser is authenticated and reliable, which can be achievedin practice with little overhead. These authenticationhandshakes are omitted in the following presentation.

4. ENSURING CLOUD DATA STORAGE

In cloud data storage system, users store their data in thecloud and no longer possess the data locally. Thus, thecorrectness and availability of the data files being storedon the distributed cloud servers must be guaranteed.One of the key issues is to effectively detect any unauthorizeddata modification and corruption, possibly dueto server compromise and/or random Byzantine failures.Besides, in the distributed case when such inconsistenciesare successfully detected, to find which server thedata error lies in is also of great significance, since it canalways be the first step to fast recover the storage errorsand/or identifying potential threats of external attacks.To address these problems, our main scheme for ensuringcloud data storage is presented in this section.

The first part of the section is devoted to a review ofbasic tools from coding theory that is needed in ourscheme for file distribution across cloud servers. Then,the homomorphic token is introduced. The token computationfunction we are considering belongs to a family of universal hash function chosen to preserve the homomorphicproperties, which can be perfectly integratedwith the verification of erasure-coded data.Subsequently, it is shown how to derive a challengeresponseprotocol for verifying the storage correctness aswell as identifying misbehaving servers. The procedurefor file retrieval and error recovery based on erasurecorrectingcode is also outlined. Finally, we describe howto extend our scheme to third party auditing with onlyslight modification of the main design.

4.1 Challenge Token Pre-computation:

In order to achieve assurance of data storage correctnessand data error localization simultaneously, our schemeentirely relies on the pre-computed verification tokens.The main idea is as follows: before file distribution theuser pre-computes a certain number of short verificationtokens on individual vector G (j) (j ∈ {1, . . . , n}), eachtoken covering a random subset of data blocks. Later,when the user wants to make sure the storage correctnessfor the data in the cloud, he challenges the cloudservers with a set of randomly generated block indices.Upon receiving challenge, each cloud server computes ashort "signature" over the specified blocks and returnsthem to the user. The values of these signatures shouldmatch the corresponding tokens pre-computed by theuser. Meanwhile, as all servers operate over the samesubset of the indices, the requested response values forintegrity check must also be a valid codeword determinedby secret matrix P.

4.2 Correctness Verification and Error Localization:

Error localization is a key prerequisite for eliminatingerrors in storage systems. It is also of critical importanceto identify potential threats from external attacks. However, many previous schemes do not explicitlyconsider the problem of data error localization, thus onlyproviding binary results for the storage verification. Ourscheme outperforms those by integrating the correctnessverification and error localization (misbehaving serveridentification) in our challenge-response protocol: theresponse values from servers for each challenge not onlydetermine the correctness of the distributed storage, butalso contain information to locate potential data error(s).

4.3 Towards Third Party Auditing

As discussed in our architecture, in case the user doesnot have the time, feasibility or resources to performthe storage correctness verification, he can optionallydelegate this task to an independent third party auditor,making the cloud storage publicly verifiable. However,as pointed out by the recent work, to securelyintroduce an effective TPA, the auditing process shouldbring in no new vulnerabilities towards user data privacy.Namely, TPA should not learn user's data contentthrough the delegated data auditing. Now we show thatwith only slight modification, our protocol can supportprivacy-preserving third party auditing.Recall thatthe reason of blinding process is for protection of thesecret matrix P against cloud servers. However, this canbe achieved either by blinding the parity vector or byblinding the data vector (we assume $k < m$). Thus, ifwe blind data vector before file distribution encoding,then the storage verification task can be successfullydelegated to third party auditing in a privacy-preservingmanner.

5. CONCLUSION

In this paper, we investigate the problem of data security in cloud data storage, which is essentially a distributed storage system. To achieve the assurances of cloud data integrity and availability and enforce the quality of dependable cloud storage service for users, we propose an effective and flexible distributed scheme with explicit dynamic data support, including block update, delete, and append. We rely on erasure-correcting code in the file distribution preparation to provide redundancy parity vectors and guarantee the data dependability. By utilizing the homomorphic token with distributed verification of erasure-coded data, our scheme achieves the integration of storage correctness insurance and data error localization, i.e., whenever data corruption has been detected during the storage correctness verification across the distributed servers. Considering the time, computation resources, and eventhe related online burden of users, we also providethe extension of the proposed main scheme to supportthird-party auditing, where users can safely delegate theintegrity checking tasks to third-party auditors and beworry-free to use the cloud storage services.

REFERENCES:

[1] Amazon.com, “Amazon Web Services (AWS),” Online at http://aws.amazon.com, 2008.
[2]F. Seb´e, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J.Quisquater, “Efficient remote data possession checking in critical information infrastructures,” IEEE Trans. on Knowl.And Data Eng., vol. 20, no. 8, 2008.
[3] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, “Scalable and efficient provable data possession,” in Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks, 2008, pp. 1–10.
[4] C. Erway, A. K¨upc¸ ¨ u, C. Papamanthou, and R. Tamassia, “Dynamic provable data possession,” in Proceedings of the 16th ACM Conference on Computer and Communications Security, 2009, pp. 213–222.
[5] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling public verifiability and data dynamics for storage security in cloud computing,”
[6] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, “MR-PDP: multiple-replica provable data possession,” in 28th IEEE ICDCS, 2008, pp. 411–420.
[7] S. Wilson, “Appengine outage,” Online at http://www.cio-weblog.com/50226711/appengine outage.php, June 2008.
[8] B. Krebs, “Payment Processor Breach May Be Largest Ever,” Onlineat http://voices.washingtonpost.com/securityfix/2009/01/payment processor breach may b.html, Jan. 2009.
[9] A. Juels and J. Burton S. Kaliski, “Pors: Proofs of retrievability forlarge files,” in *Proc. of CCS’07*, Alexandria, VA, October 2007, pp.584–597.
[10] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson,and D. Song, “Provable data possession at untrusted stores,”in *Proc. of CCS’07*, Alexandria, VA, October 2007, pp. 598–609.

K.Lalitha receivedB.Tech from JNTU University,Hyderabad. She is currentlypursuing M.Tech in Computer Science &Engineering Department,Keshav Memorial Institute of Technology,Narayanguda, Hyderabad,Telangana, India-500029

Prof Dr.Siddharthaghosh received Dr. from Osmania University Hyderabad 2011 in artificial intelligence, currently he is working as a Head of the department of computer science&engineering at keshav memorial institute of technology Narayanguda, Hyderabad ,Telangana, India-500029