International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622
NATIONAL CONFERENCE on Developments, Advances & Trends in Engineering Sciences (NCDATES- 09th & 10th January 2015)

RESEARCH ARTICLE OPEN ACCESS

Privacy Preserving Public Auditing System for Data Storage Security in Cloud Computing

G. Praveen Kumar *, M. Omkar Sharma **

Abstract:
Using cloud storage, 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 audit ability 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. Our preliminary experiment conducted on Amazon EC2 instance further demonstrates the fast performance of the design.

INTRODUCTION

The user initializes the public and secret parameters of the system by executing Key Gen and preprocesses the data file F by using Sig Gen to generate the verification metadata. The user then store the data file F and the verification metadata at the cloud server, and deletes its local copy. As part of preprocessing, the user may alter the data file F by expanding it or including additional metadata to be stored at server.

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 unforgettable 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.

In cloud computing, outsourced data might not only be accessed but also updated frequently by users for various application purposes. Hence, supporting data dynamics for privacy-preserving public auditing is also of paramount importance. Now, we show how to build upon the existing work and adapt our main scheme to support data dynamics, including block level operations of modification, deletion, and insertion. We evaluate the security of the proposed scheme by analyzing its fulfillment of the security guarantee described namely, the storage correctness and privacy preserving property. We start from the single user case, where our main result is originated. Then, we show the security guarantee of batch auditing for the TPA in multiuser setting.

Dynamic data have also attracted attentions in the recent literature on efficiently providing the integrity guarantee of remotely stored data. Ateniese is the first to propose a partially dynamic version of the prior PDP scheme, using only symmetric key cryptography but with a bounded number of audits Wang et al. consider a similar support for partially dynamic data storage in a distributed scenario with additional feature of data error localization. In a subsequent work, Wang propose to combine BLS-based HLA with MHT to support fully data dynamics. Concurrently Erwayetal develop a skip list based scheme to also enable provable data possession with full dynamics support. However, the verification in both protocols requires the linear combination of sampled blocks as an input, like the designs and thus does not support privacy-preserving auditing.

EXISTING SYSTEM

In the Existing systems, the notion of public auditability has been proposed in the context of ensuring remotely stored data integrity under different system 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 consider the privacy protection of users' data against external auditors. Indeed, they may potentially reveal user's data to auditors. 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 toward their data security.

DRAWBACKS OF EXISTING SYSTEM:

- ✓ 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.
- ✓ Second, there do exist various motivations for CSP to behave unfaithfully toward the cloud users regarding their outsourced data status.
- ✓ In particular, simply downloading all the data for its integrity verification is not a practical solution due to the expensiveness in I/O and transmission cost across the network. Besides, it is often insufficient to detect the data corruption only when accessing the data, as it does not give users correctness assurance for those unaccessed data and might be too late to recover the data loss or damage.
- ✓ Encryption does not completely solve the problem of protecting data privacy against third-party auditing but just reduces it to the complex key management domain. Unauthorized data leakage still remains possible due to the potential exposure of decryption keys.

PROPOSED SYSTEM

In this paper, we utilize the public key based homomorphic authenticator and uniquely integrate it with random mask technique to achieve a privacy-preserving public auditing system for cloud data storage security while keeping all above requirements in mind. 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. We also show how to extent our main scheme to support batch auditing for TPA upon delegations from multi-users.

BENEFITS OF PROPOSED SYSTEM:

1. **Public auditability:** to allow TPA to verify the correctness of the cloud data on demand without retrieving a copy of the whole data or introducing additional online burden to the cloud users.
2. **Storage correctness:** to ensure that there no exists of cheating in cloud server that can pass the TPA's audit without indeed storing users' data intact.
3. **Privacy preserving:** to ensure that the TPA cannot derive users' data content from the information collected during the auditing process.
4. **Batch auditing:** to enable TPA with secure and efficient auditing capability to cope with multiple auditing delegations from possibly large number of different users simultaneously
5. **Lightweight:** to allow TPA to perform auditing with minimum communication and computation overhead.

IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

Modules Description:

Public audit ability for storage correctness assurance:

To allow anyone, the clients who originally stored the file on cloud servers, to have the capability to verify the correctness of the stored data on demand.

Dynamic data operation support:

To allow the clients to perform block level operations on the data files while maintaining the same level of data correctness assurance. The design should be as efficient as possible so as to ensure the seamless integration of public auditability and dynamic data operation support.

Block less verification:

No challenged file blocks should be retrieved by the verifier (*e.g.*, TPA) during verification process for efficiency concern.

Dynamic Data Operation with Integrity Assurance:

Now we show how our scheme can explicitly and efficiently handle fully dynamic data operations including data modification (M), data insertion (I) and data deletion (D) for cloud data storage. Note that in the following descriptions, we assume that the file F and the signature _ have already been generated and properly stored at server. The root metadata R has been signed by the client and stored at the cloud server, so that anyone who has the client's public key can challenge the correctness of data storage.

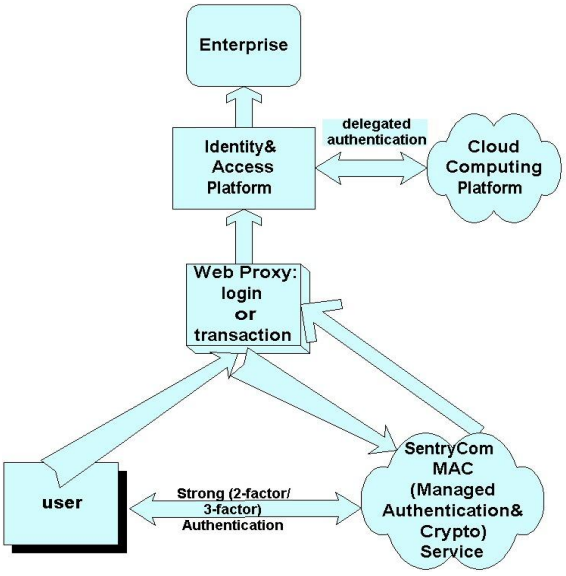
Data Modification:

We start from data modification, which is one of the most frequently used operations in cloud data storage. A basic data modification operation refers to the replacement of specified blocks with new ones. At start, based on the new block the client generates the corresponding signature. The client signs the new root metadata R′ by sigsk(H(R′)) and sends it to the server for update. Finally, the client executes the default integrity verification protocol. If the Output is *TRUE*, delete sigsk(H(R′)),and generate duplicate file.

Batch Auditing for Multi-client Data:

As cloud servers may concurrently handle multiple verification sessions from different clients, given K signatures on K distinct data files from K clients, it is more advantageous to aggregate all these signatures into a single short one and verify it at one time. To achieve this goal, we extend our scheme to allow for provable data updates and verification in a multi-client system. The signature scheme allows the creation of signatures on arbitrary distinct messages. Moreover, it supports the aggregation of multiple signatures by distinct signers on distinct messages into a single short signature, and thus greatly reduces the communication cost while providing efficient verification for the authenticity of all messages.

PROCESS DIAGRAM



Ateniese *et al.* [8] are the first to consider public auditability in their defined "provable data possession"(PDP) model for ensuring possession of data files on untrusted storages.Their scheme utilizes the RSA based homomorphic linear authenticators for auditing outsourced data and suggests randomly sampling a few blocks of the file. However, the public auditability in their scheme demands the linear combination of sampled blocks exposed to external auditor. When used directly, their protocol is not provably privacy preserving, and thus may leak user data information to the auditor. Juels *et al.*[11] describe a "proofof retrievability" (PoR) model, where spot-checkingand error-correcting codes are used to ensure both "possession" and "retrievability" of data files on remotearchive service systems. However, the number of audit challenges a user can perform is fixed apriori, and public auditability is not supported in their main scheme. Although they describe a straight forward Merkle-tree construction for public PoRs, thisapproach only works with encrypted data. Dodis *etal.* [25] give a study on different variants of PoR with private auditability. Shacham *et al.* [13] design an improved PoR scheme built from BLS signatures [17]with full proofs of security in the security model defined in [11]. Similar to the construction in [8], theyuse publicly verifiable homomorphic linear authenticators that are built from provably secure BLS signatures.Based on the elegant BLS construction, a compactand public verifiable scheme is obtained. Again,their approach does not support privacy-preserving auditing for the same reason as [8]. Shah *et al.* [9], [14]propose allowing a TPA to keep online storage honestby first encrypting the data then sending a numberof pre-

computed symmetric-keyed hashes over the encrypted data to the auditor. The auditor verifiesboth the integrity of the data file and the server'spossession of a previously committed decryption key.This scheme only works for encrypted files, and it suffers from the auditor statefulness and bounded usage, which may potentially bring in online burden to users when the keyed hashes are used up.In other related work, Ateniese *et al.* [19] propose apartially dynamic version of the prior PDP scheme, using only symmetric key cryptography but with a bounded number of audits. In [20], Wang *et al.*consider a similar support for partial dynamic data storage in a distributed scenario with additional feature of data error localization. In a subsequent work,Wang *et al.* [10] propose to combine BLS-based HLAwith MHT to support both public auditability andfull data dynamics. Almost simultaneously, Erway*et al.* [21] developed a skip lists based scheme to enable provable data possession with full dynamic ssupport. However, the verification in these two protocols requires the linear combination of sampled blocksjust as [8], [13], and thus does not support privacy preserving auditing. While all the above schemes provide methods for efficient auditing and provable assurance on the correctness of remotely stored data, none of them meet all ther equirements for privacy preserving public auditing in cloud computing. More importantly, none of these schemes consider batch auditing, which can greatly reduce the computation cost on the TPA when coping with a large number ofaudit delegations.

EXPERIMENTAL RESULTS

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

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] P. Mell and T. Grance, "Draft NIST working definition of cloud computing," Referenced on June. 3rd, 2009 Online at http://csrc.nist.gov/groups/SNS/cloud-computing/index. html, 2009.

[2] 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.

[3] M. Arrington, "Gmail disaster: Reports of mass email deletions," Online at http://www.techcrunch.com/2006/ 12/28/gmail-disasterreports-of-mass-email-deletions/, December 2006.

[4] J. Kincaid, "MediaMax/TheLinkup Closes Its Doors,"Online at http://www.techcrunch.com/2008/07/10/media maxthelinkup-closes-its-doors/, July 2008.

[5] Amazon.com, "Amazon s3 availability event: July 20, 2008," Online at

http://status.aws.amazon.com/s3-20080720.html, 2008.

[6] S. Wilson, "Appengine outage," Online at http://www. cio-weblog.com/50226711/appengine outage.php, June 2008.

[7] B. Krebs, "Payment Processor Breach May Be Largest Ever,"Online at http://voices.washingtonpost.com/securityfix/2009/01/payment processor breach may b.html, Jan. 2009.

[8] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrustedstores," in *Proc. of CCS'07*, Alexandria, VA, October 2007, pp. 598–609.

[9] M. A. Shah, R. Swaminathan, and M. Baker, "Privacypreserving audit and extraction of digital contents," Cryptology ePrint Archive, Report 2008/186, 2008.

[10] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in *Proc. of ESORICS'09, volume 5789 of LNCS*. Springer-Verlag, Sep. 2009, pp. 355–370.

[11] A. Juels and J. Burton S. Kaliski, "Pors: Proofs of retrievability for large files," in *Proc. of CCS'07*, Alexandria, VA, October 2007, pp. 584–597.

[12] Cloud Security Alliance, "Security guidance for critical areas of focus in cloud computing," 2009, http://www. cloudsecurityalliance.org.

[13] H. Shacham and B. Waters, "Compact proofs of retrievability,"in *Proc. of Asiacrypt 2008*, vol. 5350, Dec 2008, pp. 90–107.

[14] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan,"Auditing to keep online storage services honest," in *Proc. Of HotOS'07*. Berkeley, CA, USA: USENIX Association, 2007, pp. 1–6.

[15] 104th United States Congress, "Health Insurance Portability and Accountability Act of 1996 (HIPPA)," Online at http:// aspe.hhs.gov/admnsimp/pl104191.htm, 1996.

[16] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained access control in cloud computing,"in *Proc. of IEEE INFOCOM'10*, San Diego, CA, USA, March 2010.

[17] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," *J. Cryptology*, vol. 17, no. 4, pp. 297–319,2004.

[18] A. L. Ferrara, M. Greeny, S. Hohenberger, and M. Pedersen,"Practical short signature batch verification," in *Proceedings ofCT-RSA, volume 5473 of LNCS*. Springer-Verlag, 2009, pp. 309–324.

[19] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in *Proc. Of SecureComm'08*, 2008, pp. 1–10.

[20] C.Wang, Q.Wang, K. Ren, andW. Lou, "Ensuring data storage security in cloud computing," in *Proc. of IWQoS'09*, July 2009, pp. 1–9.

[21] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in *Proc. of CCS'09*, 2009, pp. 213–222.

[22] R. C.Merkle, "Protocols for public key cryptosystems," in *Proc.of IEEE Symposium on Security and Privacy*, Los Alamitos, CA, USA, 1980.

[23] G. Ateniese, S. Kamara, and J. Katz, "Proofs of storage from homomorphic identification protocols," in *ASIACRYPT*, 2009, pp. 319–333.

[24] M. Bellare and G. Neven, "Multi-signatures in the plain publickey model and a general forking lemma," in *ACM Conference on Computer and Communications Security*, 2006, pp. 390–399.

[25] Y. Dodis, S. P. Vadhan, and D. Wichs, "Proofs of retrievability via hardness amplification," in *TCC*, 2009, pp. 109–127.

****Mr.M.OmkarSharma**, working as a Assistant Professor in CMR Engineering College, Hyderabad, Telangana, india. I am having 4 years of teaching experience and 2 years of industrial experience and I was completed my M.Tech and B.Tech in Computer Science and Engineering in 2012 and 2008 respectively from JNTUH , Hyderabad, Telangana, India. I presented many papers in international and national conferences held at various places. I am interested in doing research in computer networking and also interested to study in various upcoming technologies cloud computing, mobile computing etc.

***Mr. Praveen kumar**,M (Tech) pursuing M.Tech(CSE) in CMR College of Engineering, Hydrabad, India. . I am interested in doing research in computer networking and also interested to study in various upcoming technologies