IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727
PP 80-85
www.iosrjournals.org

Secure Cloud Storage for Privacy Preserving Using Public Auditing

[1]S. Sahana, [2]A. Kanmani

[1,2]Department of computer science and engineering Syed ammal engineering college Ramanathapuram, India

Abstract: *Users can remotely store their data in the cloud storage and can use the applications and services from a collective pool of configurable computing resources, without the trouble of local data storage and protection. Additionally, users should be able to just use the cloud storage as if it is local, without worrying about the need to authenticate its integrity. Hence, enabling public auditability for cloud storage is of critical importance so that users can resort to a Third Party Auditor (TPA) to prove the reliability of outsourced data and be burden free. To strongly initiate a valuable TPA, the auditing process should bring in no new vulnerabilities toward user data confidentiality, and bring in no other online trouble to user. In this paper, we propose a protected cloud storage system supporting privacy-preserving public auditing for active data. To allow the TPA to carry out audits for multiple users simultaneously and also prove the accuracy of remotely stored data efficiently. The audit outcome would also be useful for the cloud service providers to progress the cloud service platform. Extensive protection and performance analysis show the proposed schemes are provably protected and highly efficient. The test accomplished on Amazon EC2 instance further demonstrates the fast performance of the design.*

Index Terms: *Data Storage, privacy-preserving, public auditability, cloud computing, batch verification.*

I. Introduction

Cloud computing has been envisioned as the next foundation information technology (IT) architecture for enterprises, due to its extended list of unique advantages in the IT history: on-demand self-service, everywhere network access, location free resource pooling, fast resource flexibility, usage-based pricing and transference of risk. As a troublesome technology with reflective implications, cloud computing is converting the very nature of how businesses use information technology. One primary aspect of this paradigm changing is that data are being centralized or outsourced to the cloud. From users' point of view, including both individuals and IT enterprises, storing remote data to the cloud in a flexible on-demand manner brings attractive benefits: relief of the load for storage management, worldwide data access with location independence, and avoiding capital costs on hardware, software, etc., [3].

First of all, although the infrastructures under the cloud are much more forceful and consistent than personal devices, they are still facing the wide range of both internal and external threats for data integrity [4]. Examples of outages and security breaches of important cloud services appear from time to time [5], [6], [7]. Second, there do exist different motivations for CSP to act falsely toward the cloud users concerning their outsourced data status. Some examples, CSP might reclaim storage for economic reasons by removing data that have not been or are rarely accessed, or even hide data loss incidents to preserve a reputation [8], [9], [10]. In short, although outsourcing data to the cloud is reasonably attractive for long-term large-scale storage, it does not instantly present any assurance on data integrity and availability. This problem, if not correctly addressed, may slow down the achievement of cloud architecture. As users no longer physically own the storage of their data, some traditional cryptographic primitives for the reason of data security protection cannot be directly adopted [11]. In particular, simply downloading all the data for its integrity certification is not a useful solution due to the expensiveness in I/O and transmission cost across the network. Besides, it is often inadequate to identify the data corruption only when accessing the data, as it does not give users correctness guarantee for those unaccessed data and might be too late to recover the data loss or damage.

Considering the huge size of the outsourced data and the user's controlled resource facility, the tasks of auditing the data exactness in a cloud environment can be difficult and costly for the cloud users [12], [8]. Moreover, the overhead of using cloud storage should be minimized as much as possible, such that a user does not need to execute too many operations to use the data. And, users may not want to go through the difficulty in verifying the data integrity. There may be more than one user accesses the same cloud storage, say in an enterprise setting. To manage easily, it is desirable that cloud only entertains authentication request from a single selected party.

To fully ensure the data reliability and keep the cloud users' computation resources as well as online burden, critical importance to enable public auditing service for cloud data storage, so that users may route to an independent third-party auditor (TPA) to audit the outsourced data when required. The TPA, who has expertise and capabilities that users do not, can occasionally check the reliability of all the data stored in the cloud on behalf of the customers, which presents a much more easier and reasonable way for the users to ensure their storage exactness in the cloud. Moreover, in addition to help users to estimate the risk of their subscribed cloud data services, the result from TPA would also be useful for the cloud service providers to improve their cloud-based service platform, also serve for independent arbitration purposes [10]. In a word, enabling public auditing services will play an important task for this nascent cloud economy to become fully established, here users will need ways to assess risk and gain trust in the cloud.

However, most of these schemes [9], [13], [8] do not consider the privacy protection of users' data beside external auditors. Indeed, they may potentially expose user's data to auditors. Therefore, how to enable a privacy-preserving third party auditing protocol, which is independent to data encryption, it is the problem we are going to tackle in this paper. By integrating the Homomorphic Linear Authenticator with random masking, our protocol assurances that the TPA could not learn any information about the data content stored in the cloud server (CS) during the efficient auditing process. Authenticator properties of aggregation and algebraic added benefit our design for the batch auditing.

II. Literature Survey

A. Scalable and Efficient Provable Data Possession

Ateniese et al. [14] is the first to propose a partially dynamic version of the prior PDP scheme, symmetric key cryptography only used, but with a restricted number of audits. Storage outsourcing is a growing trend which prompts a number of motivating security issues, which have been extensively investigated in the past. However, Provable Data Possession (PDP) is a topic that has only recently appeared in the research literature. The main issue is how to regularly, powerfully and securely prove that a storage server is truly storing its client's (potentially very large) outsourced data. The server is assumed to be untrusted in terms of both security and reliability. (It might maliciously or unintentionally remove hosted data; it might also demote it to slow or off-line storage.) Problem is exacerbated by the client being a small computing device with restricted resources. Previous effort has addressed this problem using either public key cryptography or requiring the client to outsource its data in encrypted form.

Make a highly efficient and provably secure PDP technique based entirely on symmetric key cryptography, does not requiring any bulk encryption. PDP technique allows outsourcing of active data, i.e., it efficiently supports operations, such as block alteration, deletion and append. Doesn't consider data privacy problem. Main scheme doesn't support auditability.

B. PORs: Proofs of Retrievability for Large Files

Juels et al. [11] clarify a model of "PORs". Error-correcting codes and spot-checking are used to guarantee both "possession" and "retrievability" of data files on remote archive service systems. In this paper, define and explore Proofs of Retrievability (PORs). A POR scheme enables an archive or back-up service (prover) to produce a short verification that a user (verifier) can retrieve a target file F, that is, that the archive retains and consistently transmits file data adequate for the user to recover F in its entirety. A POR may be viewed as a type of cryptographic proof of knowledge (POK), but one especially designed to handle a large file (or bit string) F. Explore POR protocols here in which the communication costs, the number of memory accesses for the prover, and storage requirements of the user (verifier) are small parameters essentially independent of the length of F. Practical POR constructions, explore performance considerations and optimizations that bear on previously explored, associated schemes.

In a POR, unlike a POK, neither the prover nor the verifier require actually have awareness of F. PORs give rise to a new and unusual security definition whose formulation is another contribution of our work. View PORs as an important tool for semi-trusted online archives. The existing cryptographic techniques help users guarantee the privacy and integrity of files they retrieve. And it is natural, however, for users to want to authenticate that archives do not delete or modify files prior to retrieval.

The goal of a POR is to achieve these checks without users having to download the files themselves. A POR can also offer quality-of-service guarantees. (Show that a file is retrievable within a certain time bound.) However, the number of audit challenges a user can perform is fixed a priori, public auditability is not supported in their main scheme.

C. Dynamic Provable Data Possession

Erway et al. [1] expand a skip list based scheme to also enable provable data possession with full dynamics support. As storage-outsourcing services and resource-sharing networks have become popular, the problem of powerfully proving the integrity of data stored at untrusted servers has received increased attention. In the provable data possession (PDP) model, the client preprocesses the data and then sends it to an untrusted server for storage, while maintaining a small amount of meta-data. The client later asks the server to show that the stored data has not been tampered with or deleted (without downloading the original data). And the original PDP scheme applies only to static (or append-only) files.

Present a definitional framework and efficient constructions for dynamic provable data possession (DPDP), which extends the PDP model to carry provable updates to stored data. Using a new version of authenticated dictionaries based on rank information. And the price of dynamic updates are a performance change from O(1) to O(log n) for a file consisting of n blocks, while maintaining the same (or better, respectively) probability of misbehavior detection. However DPDP scheme is efficient, Practical for use in distributed applications and thus does not support privacy-preserving auditing.

D. Compact Proofs of Retrievability

Shacham and Waters [13] design a better POR scheme with proofs of security in the security Model. In a proof-of-retrievability system, storage center convinces a verifier that he is actually storing all of a client's data. The fundamental challenge is to construct systems that are both efficient and provably secure, that is, it should be probable to take out the client's data from any prover that passes a verification check.

In this paper, give the first proof-of-retrievability schemes with full proofs of security against arbitrary adversaries in the strongest model. The first scheme that is built from BLS signatures and secure in the random oracle model, and has the shortest query, response of any proof-of-retrievability with public verifiability. Second scheme, which builds elegantly on pseudorandom functions (PRFs) and is secure in the standard model, and has the shortest response of any proof-of-retrievability scheme with private verifiability (but a longer query). Here both schemes rely on Homomorphic properties to aggregate a proof into one small authenticator value. However this scheme efficient and provably secure and thus doesn't consider the data privacy problem.

III. Existing System

Cloud data storage contains two entities as cloud user and cloud service provider/ cloud server in cloud computing. Cloud user is a person who stores large amount of data on cloud server which is controlled by the cloud service provider. User can upload their data on cloud without worrying about storage and maintenance. Services are provided by the cloud service provider to cloud user. Correctness and integrity of data stored on the cloud are the major issue in cloud data storage. Some form of mechanism are provided by Cloud Service Provider (CSP) through which user will get the confirmation that cloud data is secure or is stored as it is. No data loss or modification is done. The correctness of data can be violated due to a broad range of both internal and external threats and CSP also may hide data loss or damage from users to maintain a reputation. Major security issues associated with cloud user and CSP are as follows.

A. Cloud Service Provider

Services are provided by organization or enterprises to cloud users. CSP maintains confidentiality and integrity of cloud data. CSP should ensure that user's data and application are secured on a cloud. CSP may not leak the information or else cannot modify or access user's content. The attacker can log into network communication.

B. Cloud Server

Data being stored in the cloud server and accessed by cloud data owner or users. Data should not be accessed by unauthorized users, no data modification or no loss of data.

C. Cloud User

Attackers can access basic information like username and password. Key management is major issue in encryption techniques. Data dynamic issues need to be considered by CSP.

Drawbacks of Existing System

- In the early stages of security in cloud computing, the privacy preservation is done for user level and the data level. Whereas the each registered users will get the dynamic code for security.

- Then and there the user has to use that token to enter into the process. There is no user blocking, no IP address blocking, only listing unauthorized users.

IV. Proposed System

In this System, extending the previous system for privacy preserving public auditing for data storage security in cloud computing. Keeping all above requirements in mind, to store remote data securely and allowing public auditing system for cloud data storage security. Extensive protection and performance analysis shows the proposed schemes are provably safe and highly efficient. The proposed system incorporates the previous system advantages and extends to find the unauthorized user, to avoid the unauthorized data access for preserving data integrity. The proposed system observes the user requests according the user specified parameters and it checks the parameters for the new and existing users. The system accepts existing authorized user, and prompts for the new users for the parameter to match requirement specified during user creation for new users. If the new user prompts parameter matches with cloud server, it gives rights to access the audit protocol otherwise the system automatically blocks the audit protocol for specific user.

Advantages of Proposed System

- It allows 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.
- It ensures that there exists no cheating cloud server that can pass the TPA's audit without indeed storing users' data intact.
- It ensures that the TPA cannot derive users' data content from the information collected during the auditing process.
- It enables TPA with secure and efficient auditing capability to cope with multiple auditing delegations from possibly large number of different users simultaneously.
- It allows TPA to perform auditing with minimum communication and computation overhead.

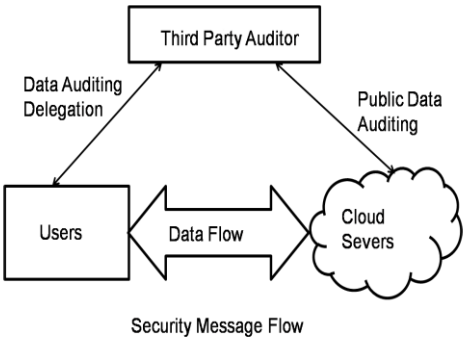

Fig.1. Architecture

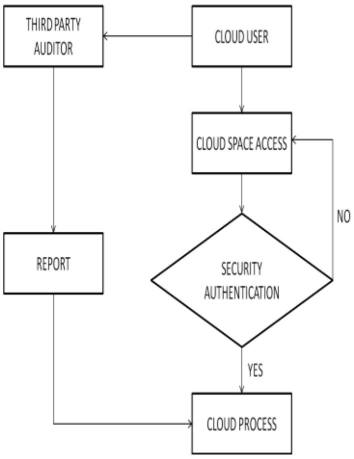

Fig.2. Flow diagram

To improve security and allowing public auditing, the following components are required.

A. Storage Correctness

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. To ensure that there exists no cheating cloud server that can pass the audit from TPA without indeed storing user's data intact.

B. File Upload and Download

In this Process we are doing the cloud process for uploading and downloading the data between cloud user and the cloud server based on the authentication information entered by the users for improve the security.

C. Protocol Blocker

Once the user initializes the parameters, all the specified parameters are checked by the system and validates the protocol for proper users, it blocks the unauthorized users, if the user newly access the cloud servers, the system prompts for security parameters, previously assigned by the system during the user creation.

D. 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 on-line burden to the cloud users. The TPA issues an audit message or challenge to the cloud server to make sure that the cloud server has retained the data file properly at the time of the audit. Then the cloud server will derive a response message from a function of the stored data file F by executing Gen Proof. The TPA verifies the response via Verify Proof using the verification metadata.

E. Privacy-Preserving

To ensure that there exists no way for TPA to derive users' data content from the information collected during the auditing process. In this process we are using the Homomorphic Linear Authenticator with random masking technique to prevent the data losses.

F. Report Generation

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. In this process we are generate the report for analysis the users who are all entered successfully and incorrectly then calculate the timeout users also.

V. Conclusion and Future Works

In this System we propose a privacy-preserving public auditing system for data storage security. We designed the simulation by considering the single user. In cloud computing, where TPA can perform the storage auditing without demanding the local copy of data. TPA does not learn any knowledge about the data content stored on the cloud server during the auditing process, which not only reduces the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the user's fear of their outsourced data leakage. Multiple audit sessions from different users for their outsourced data files are handled by the TPA concurrently and extend our privacy-preserving public auditing protocol into a multi-user setting, where multiple auditing tasks are performed by TPA in a batch manner, i.e., simultaneously. Extensive protection and performance analysis show the proposed schemes are provably protected and highly efficient. The test accomplished on Amazon EC2 instance further demonstrates the fast performance of the design.

In future work, as per the requirements stated by the user, the entire system has been developed and deployed, and as per the testing standards that is implemented, the system to be bug free. In the coming versions, any specification-untraced errors will be determined and planned to be developed in near future. At Present, the system does not take care of lower level check constraints in contacting the cloud, in future the three dimensional password for user level security in cloud computing are implemented.

References

[1] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, “Dynamic Provable Data Possession,” Proc. ACM Conf. Computer and Comm. Security (CCS ’09), pp. 213-222, 2009.

[2] Y. Dodis, S.P. Vadhan, and D. Wichs, “Proofs of Retrievability via Hardness Amplification,” Proc. Theory of Cryptography Conf. Theory of Cryptography (TCC), pp. 109-127, 2009.

[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,” Technical Report UCB-EECS 2009-28, Univ. of California, Berkeley, Feb. 2009.

[4] Cloud Security Alliance, “Top Threats to Cloud Computing,” http://www.cloudsecurityalliance.org, 2010.

[5] M. Arrington, “Gmail Disaster: Reports of Mass Email Deletions,”http://www.techcrunch.com/2006/12/28/gmail-disasterreportsof-mass-email-deletions/, 2006.

[6] J. Kincaid, “MediaMax/TheLinkup Closes Its Doors,” http://www.techcrunch.com/2008/07/10/mediamaxthelinkup-closesits-doors/, July 2008.

[7] Amazon.com, “Amazon s3 Availability Event: July 20, 2008,”http://status.aws.amazon.com/s3-20080720.html, July 2008.

[8] 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.

[9] 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.

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

[11] A. Juels and J. Burton, S. Kaliski, “PORs: Proofs of Retrievability for Large Files,” Proc. ACM Conf. Computer and Comm. Security (CCS ’07), pp. 584-597, Oct. 2007.

[12] Cloud Security Alliance, “Security Guidance for Critical Areas of Focus in Cloud Computing,” http://www.cloudsecurityalliance.org, 2009.

[13] H. Shacham and B. Waters, “Compact Proofs of Retrievability,” Proc. Int’l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (Asiacrypt), vol. 5350, pp. 90-107, Dec. 2008.

[14] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, “Scalable and Efficient Provable Data Possession,” Proc. Int’l Conf. Security and Privacy in Comm. Networks (SecureComm ’08), pp. 1-10, 2008.