ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2014

Secure Cloud Storage Aiding Confidentiality Protecting Public Auditing

Dr. Ali Ahammed, Mukesh kumar B [2]

Associate Professor, Department of computer science, Centre for PG Studies, VTU, Bengaluru, India. [1]

Student, M.Tech, Department of computer science, Centre for PG Studies, VTU, Bengaluru, India.[2]

ABSTRACT: Cloud computing provides a services to user through network. Cloud computing allows users to use applications or functions without installation any application at any computer with internet. Since data precaution and purity is the major trouble of numerous users who place the data in cloud this article studies the problem of empowering the integrity and security of data storage in Cloud Computing. Users can store their data using cloud storage and enjoy the services through a shared pool of computing resources, without the difficulty of limited data storage and maintenance. Thus, permissive public auditability for cloud data storage security is of critical concern so that users can resort to an third party auditor to check the integrity of redistribute data when needed. Third Party Auditor should be able to efficiently audit the cloud data storage without asking the local copy of data, and should not create new liability to the user data confidentiality..

KEYWORDS: Encryption, Decryption, user, key,Third Party Auditor, cloud computing..

I.INTRODUCTION

Cloud computing is internet based computing which enables distribution of services. Many users place their data in the cloud. Cloud computing gives extensibility to users, so that users can pay as much they. The notion of public auditability has been proposed recently to ensure casually stored data integrity under different systems [8], [10], [11],[13]. However, most of the schemes [8], [10],[13] do not consider the confidentiality secure of users' data.
Thus to provide privacy protection of user data a technique called public key based homomorphic linear authenticator [8], [10],[13] is used which enables TPA to perform the auditing without asking the limited pattern of data. The effects of homomorphic linear authenticator further benefit our design for the batch auditing. Specifically, the contribution can be summarized as the following three aspects.

1) Our scheme provide a privacy-preserving auditing protocol.
2) Our scheme achieves batch auditing where multiple delegated auditing tasks from different users can be performed simultaneously by the TPA.
3) Our scheme provides the better security and justify the performance of our proposed schemes through concrete experiments.

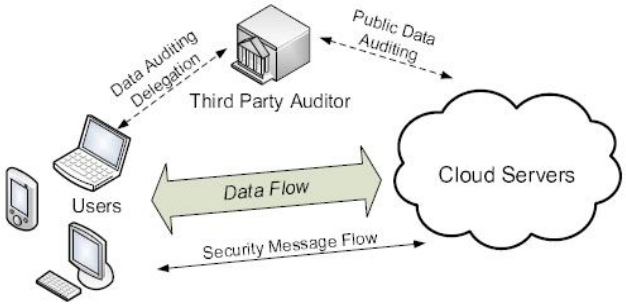

Figure.1 Architecture of cloud data storage

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2014

II. RELATED WORK

Some of the works done on privacy preserving public auditing are discussed here.

Reliable data possession at untrusted stores [1]

In this document the author describes model for provable data possession (PDP) that allows a client that has saved data at an untrusted server to authenticate that the server possesses the primary 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.

.

Proofs of retrievability for large files [2]

In this paper, author define and explore proofs of retrievability (PORs). A POR scheme enables an archive or back-up service (prover) to produce a concise proof that a user (verifier) can retrieve a target file F, that is, that the archive retains and reliably transmits file data su□cient for the user to recover F in its entirely.A POR may be viewed as a kind of cryptographic proof of knowledge (POK), but one specially designed to handle a large file (or bitstring) F. We explore POR protocols here in which the communication costs, number of memory accesses for the prover, and storage requirements of the user (verifier) are small parameters essentially independent of the strength of F. In addition to proposing new, practical POR constructions, we explore implementation considerations and optimizations that bear on previously explored, related schemes.

Proofs of retrievability via hardness amplification [3]

In a POR, unlike a POK, neither the prover nor the verifier need actually have knowledge of F. PORs give rise to a new and unusual security definition whose formulation is another contribution of our work. We view PORs as an important tool for semi-trusted online archives. Existing cryptographic techniques help users ensure the privacy and integrity of files they retrieve. It is also natural, however, for users to want to verify that archives do not delete or modify files prior to retrieval. The goal of a POR is to accomplish these checks without users having to download the files themselves. A POR can also provide quality-of-service guarantees, i.e., show that a file isretrievable within a certain time bound..

Compact Proofs of Retrievability [4]

In a proof-of-retrievability system, a data storage center must prove to a verifier that he is actually storing all of a client's data. The central challenge is to build systems that are both efficient and provably secure — that is, it should be possible to extract the client's data from any prover that passes a verification check. In this paper, we give the first proof-of-retrievability schemes with full proofs of security against arbitrary adversaries in the strongest model, that of Juels and Kaliski. Our first scheme, built from BLS signatures and secure in the random oracle model, features a proof-of-retrievability protocol in which the client's query and server's response are both extremely short. This scheme allows public verifiability: anyone can act as a verifier, not just the file owner. Our second scheme, which builds on pseudorandom functions (PRFs) and is secure in the standard model, allows only private verification. It features a proof-of-retrievability protocol with an even shorter server's response than our first scheme, but the client's query is long. Both schemes rely on homomorphic properties to aggregate a proof into one small authenticator value.

III.DESIGN

A.DESIGN GOALS

To enable privacy-preserving public auditing for cloud data storage,our protocol design should achieve the following security and performance guarantees.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2014

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 exists no cheating 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.
.

B.MODULES

Third Party Auditor:

In this module, Auditor views the all user data and verifying data .Auditor directly views all user data without key. Admin provided the permission to Auditor. After auditing data, store to the cloud.

Cryptograph:

The art of protecting information by transforming it (*encrypting* it) into an unreadable format, called cipher text. Only those who possess a secret *key* can decipher (or *decrypt*) the message into plain text. Encrypted messages can sometimes be broken by cryptanalysis, also called *code breaking*, although modern cryptography techniques are virtually unbreakable.

Cloud Computing:

Cloud computing is the provision of dynamically scalable and often virtualized resources as a services over the internet Users need not have knowledge of, expertise in, or control over the technology infrastructure in the "cloud" that supports them. Cloud computing represents a major change in how we store information and run applications. Instead of hosting apps and data on an individual desktop computer, everything is hosted in the "cloud"—an assemblage of computers and servers accessed via the Internet.

Privacy preserving:

To ensure that the TPA cannot derive users' data content from the information collected during the auditing process

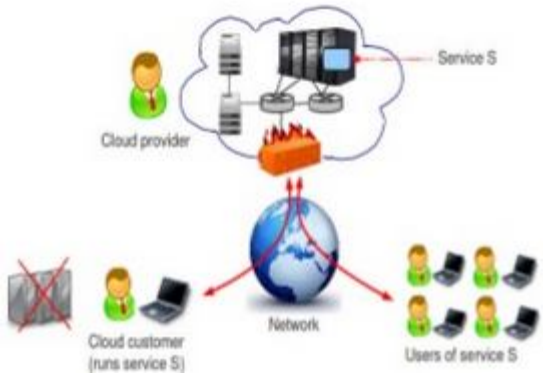

Fig. 2 The existing system

Fig 2 shows the existing system of the implementation of effective third party auditor for secure cloud computing. A cloud data storage service involving three different entities, as illustrated in Fig 2: 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 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

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2014

also dynamically interact with the CS to access and update the 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. Considering the existence of a semi- trusted CS in the sense that in most of time it behaves properly and does not deviate from the prescribed protocol execution. It's assumed that 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.

C. The Proposed Scheme

Client: an entity, which has large data files to be stored in the cloud and relies on the cloud for data maintenance and computation, can be either individual consumers or organizations.

.**Cloud Storage Server (CSS)**: an entity, which is managed by Cloud Service Provider (CSP), has significant storage space and computation resource to maintain the clients' data.

.**Third Party Auditor (TPA):** an entity, which has expertise and capabilities that clients do not have, is *trusted* to assess and expose risk of cloud storage services on behalf of the clients upon request.

Public audit ability allows an external party, in addition to the user himself, to verify the correctness of remotely stored data public key based homomorphic linear authenticator A public auditing scheme consists of four algorithms (KeyGen, Sig Gen, Gen Proof, Verify Proof). KeyGen is a key generation algorithm that is run by the user to setup the scheme. Sig Gen is used by the user to generate verification metadata, which may consist of MAC, signatures, or other related information that will be used for auditing. Gen Proof is run by the cloud server to generate a proof of data storage correctness, while Verify Proof is run by the TPA to audit the proof from the cloud server Running a public auditing system consists of two phases, Setup and Audit: • Setup: The user initializes the public and secret parameters of the system by executing KeyGen, and pre-processes the data file F by using Sig Gen to generate the verification metadata.

The user then stores the data file F and the verification metadata at the cloud server, and deletes its local copy. As part of pre-processing, the user may alter the data file F by expanding it or including additional metadata to be stored at server. • Audit: The TPA issues an audit message or challenge to the cloud server to make sure that the cloud server has retained the data file F properly at the time of the audit. The cloud server will derive a response message from a function of the stored data file F and its verification metadata by executing Gen Proof. The TPA then verifies the response via Verify Proof. 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.

D. Selection of a Programming Language

The Java programming language is a high-level language. With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2014

IV. RESULT AND DISCUSSION

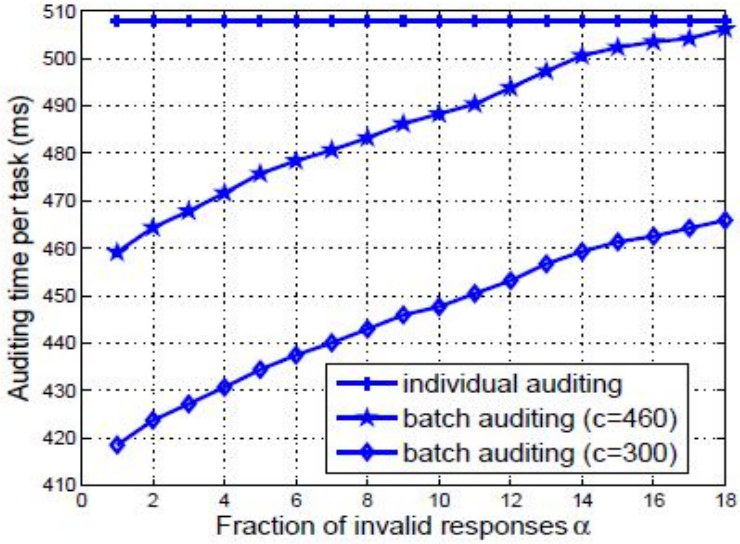

Fig. 3 Fraction of invalid response vs Auditing time per task

In the fig 3, it shows the graph of Fraction of invalid response vs Auditing time per task. Comparison on auditing time between batch and individual auditing. Per task auditing time denotes the total auditing time divided by the number of tasks. For clarity reasons, we omit the straight curve for individual auditing when c=300.

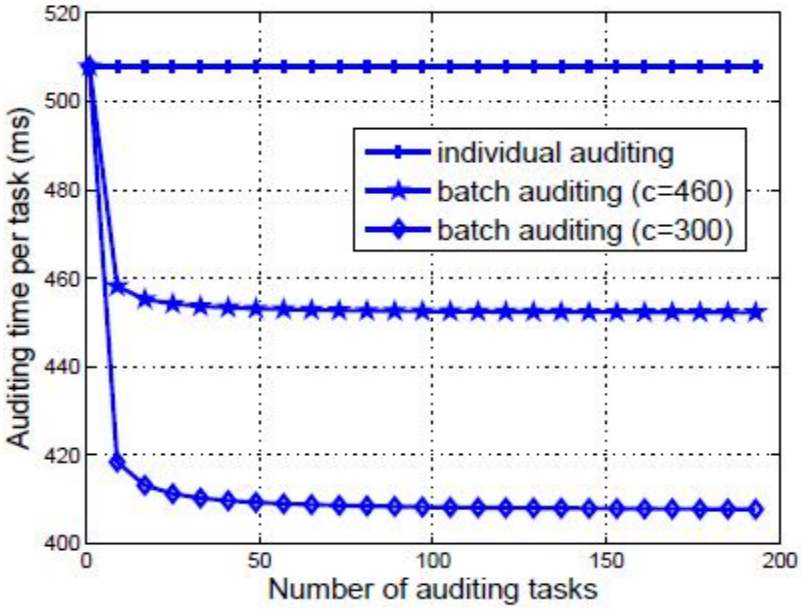

Fig. 4 Number of auditing tasks vs Auditing time per task

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2014

In the fig 4, it shows the graph of Number of auditing tasks vs Auditing time per task. Comparison on auditing time between batch and individual auditing, when _-fraction of 256 responses are invalid. Per task auditing time denotes the total auditing time divided by the number of tasks

V. CONCLUSION

This paper proposes a privacy-preserving public auditing system for data storage security in cloud computing, where the Third Party Auditor (TPA) can perform the storage auditing without demanding the local copy of data. Utilizing the homomorphic authenticator and random mask technique to guarantee that 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 server and possibly expensive auditing task. Considering TPA may concurrently handle multiple audit sessions from different users for their outsourced data files, further extending the privacy-preserving public auditing protocol into a multi-user setting, where TPA can perform the multiple auditing tasks in a batch manner, i.e., simultaneously. Extensive security and performance analysis shows that the proposed schemes are provably secure and highly efficient. It is believed that all these advantages of the proposed schemes will shed light on economies of scale for Cloud Computing. Advantages of using clouds are unarguable, due to the opaqueness of the Cloud—as separate administrative entities, the internal operation details of cloud service providers (CSP) may not be known by cloud users—data outsourcing is also 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. 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. The message size can be increased beyond 500 characters. One can further enhance this by making it closer to the real time system. This can be implemented by allowing all the users of the group to login separately. The number of users on the sender and the receiver side can be increased.

FUTURE ENHANCEMENTS

This paper presents a novel approach to securing personal and business data in the Cloud. This approach introduce a security at the time of upload process itself because to avoid the unwanted uploads in the user account. In the future more security will be provided by change the security settings frequently and based on the unauthorized access.

ACKNOWLEDGMENT

This work was supported in part by the US National Science Foundation under grant CNS-0831963, CNS-0626601, CNS-0716306, and CNS-0831628**5.**

REFERENCES

[1] www.google.com.
[2] http://en.wikipedia.org/wiki/.
[3] 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.
[4] 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
[5] M. Arrington, "Gmail disaster: Reports ofmass email deletions," Online at http://www.techcrunch.com/2006/12/28/gmail-disasterreports-of-mass-email-deletions/, December 2006
[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 untrusted stores," 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.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 6, June 2014

[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. and J. Burton S. Kaliski, "Pors: Proofs of retrievability for large files," in *Proc. of CCS'07*, Alexandria, VA, October 2007, pp. 584–597.
[11] Cloud Security Alliance, "Security guidance for critical areas of focus in cloud computing," 2009, http://www. cloudsecurityalliance.org.
[12] H. Shacham and B. Waters, "Compact proofs of retrievability," in *Proc. of Asiacrypt 2008*, vol. 5350, Dec 2008, pp. 90–107.
[13] 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.