Volume 4, Issue 10, October 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

An Implementation of Public Auditing Mechanism for Secure Cloud Storage

Anantha Lakshmi
(M.Tech) PG Scholar
Department of Computer Science& Engineering,
G. Pulla Reddy Engineering College,
Kurnool, Andhra Pradesh, India

Dr. D. Kavitha
Professor
Department of Computer Science& Engineering,
G. Pulla Reddy Engineering College,
Kurnool, Andhra Pradesh, India

Abstract*— By providing information services and software, shared resources with other devices and computers as a utility on the network, cloud computing is the delivery products, computing more like. As well known the users no longer have physical possession of outsources data makes data integrity protection cloud computing a challenging task, especial for user with contain constrained computing resources. The users will worrying about the integrity constrain in cloud storage. Thus, enabling public auditability for cloud storage is of critical importance so that users can resort to an independent auditor to check the integrity of external data and be carefree. In order to introduce the TPA effective safely, the audit process should not introduce an additional fee for online users and carry-in, there is no new vulnerabilities to the privacy of user data. The proposed approach is a secure cloud storage mechanism as public auditing mechanism for secure cloud storage. At the same time this approach extension of the TPA performance to audit multiple users efficiently. While showing high efficiency and provable security and performance analysis a wide range of security, the proposed scheme.

Keywords*— Cloud Data Storage, Privacy Preserving, Public Auditability, Cryptographic Protocols, Cloud Computing.

I. INTRODUCTION

In a nutshell, cloud computing means that access to data or programs over the Internet instead of the hard drive of the computer and stored. Cloud is a metaphor of the Internet just. It is puffy, it means nothing, white cumulonimbus clouds, accept the connection, and represents a huge server farm infrastructure of the Internet as Doling the information such that it floats, dates back to the time of presentation and flow chart.

About hard drive has not been what cloud is computing. If you store the data of the above - has been called the computing and local storage, a program to be executed or, from the hard drive. It means that (for other people or computer, one on the local network) access to your data is fast and easy all you need, it is physically close to you have. The work from the hard drive, some, it, the computer industry is in how they function for decades, I have argued the boss still cloud computing for reasons you will soon there. Cloud, there is no information about that you have a dedicated hardware server in a house. When you store the data on the network of your home or office, it does not count as utilizing the cloud.

To be considered "cloud computing", at least, it has to be synchronized with the other information on the net data very necessary to have access to programs and data over the Internet, or certain. In large companies, you might know of everything there is to know about what is on the other side of the connection. As an individual user, you may not have any idea what kind of data processing large what is happening at the other end. The final result is the same.

A. Models of Cloud Computing

The cloud computing have three models, those are 1) Infrastructure as a Service; 2) Platform as a Service; 3) Software as a Service.

1) *Infrastructure as a Service :* As with all cloud computing services providing access to computing resources in a virtualized environment, "the cloud" via a public connection, usually the Internet. If IaaS infrastructure provided welding process, i.e., particularly, it is the virtual hardware. The definition, products such as load balancer virtual server space, network connection, bandwidth, and IP address is included. Physically, typically, hardware resource group, drawn from a large number of network and distributed servers in multiple data centers all of is responsible for cloud providers to maintain. The client, in turn, gives access to virtualized components to build their own platforms.
   Examples: Windows Azure, Amazon EC2, Google Compute Engine, Rackspace.
2) *Platform as a Service :* In many cases, the developers to be able to create applications and services via the Internet, provides an environment and platform, platform and service that is a category of cloud computing simply as PaaS. PaaS services are hosted in the cloud, the user to access only through a Web browser.
   Examples: cloudbees.com, Heroku, Apache Stratos, AWS Elastic Beanstalk, Windows Azure, Force.com, Google App Engine,.

3) *Software as a Service :* SaaS describes any cloud service, where consumers can access software applications over the Internet. To both organizations and individuals, the application can be used for a wide range of tasks and are hosted in the "cloud". The Flickr and Twitter and Facebook, Google is an example of all of the SaaS and can access the service through the Internet-enabled devices of any user. Business users can use the applications to a wide range of needs, including accounting and billing, sales tracking, planning, performance monitoring and communication.
Examples: Microsoft Office 365, Google Apps.

B. Deployment Models

1) *Private Clouds :* A private cloud is a particular model of cloud computing that involves a different and based on secure cloud in which only the specified client can operate environment. As with other models of clouds, private clouds provide computing power as a service within a virtualized environment with an underlying set of physical computing resources. However, under the private cloud model, the cloud can only be accessed by a single organization providing the organization with greater control and privacy.
2) *Public Clouds :* The cloud services public network such as the Internet, the most famous model Cloud computing for many consumers, built using the shared physical resources provided clustered in a virtualized environment then, it is a public cloud model under which is accessible. To some extent can be defined in contrast to private clouds that ring-fence the underlying set of computing resources, creating a different cloud platform to which an organization has access only. Public clouds, however, serve multiple clients using the same shared infrastructure. Examples of public clouds include Amazon Elastic Compute Cloud (EC2), IBM's Blue Cloud, Sun Cloud, Google AppEngine, Windows Azure Services Platform and Cloudbees.com.
3) *Hybrid Clouds :* A hybrid cloud is a cloud service using the integrated private and public clouds for different functions within the same organization. All cloud computing services must offer certain efficiencies in different degrees, but public cloud services are likely to be more profitable and scalable private clouds. Thus, an organization can maximize its efficiency by using public cloud services for all non-sensitive operations, only relying on a private cloud, where required and ensure that all platforms are integrated seamlessly.

C. Cloud Services

1) *Amazon Elastic Calculation Cloud (EC2)* is a center cloud computing platform of Amazon.com, Amazon Web Services (AWS). EC2 users to run their own computer applications which allows you to rent virtual computers. Amazon EC2 user containing any software desired a "sample" to create a virtual machine is called an Amazon Machine Image boots through which a Web service offering allows scalable deployment of applications. A user, hence the term "elastic" means to create, launch, and terminate server instances as needed to, you can pay by the hour for active servers. EC2 provides latency optimization and high level of redundancy gives you control over the geographical location of samples.
2) *Google App Engine* is a platform as a service (PaaS) cloud computing platform for developing and hosting web applications in Google-managed centers data. Applications are sandbox and run across multiple servers. For applications the number of requests increases, App Engine is, provides automatic scaling of Web applications App Engine to allocate more resources for the Web application to handle the demand of added automatically will.
3) *CloudBees* provides the platform services that build, run, manage, and Web applications, such as the (PaaS). Sacha Labourey, founded the company in early 2010.CloudBees PaaS The production was the first PaaS to support the entire application lifecycle from development to deployment.

D. Privacy Preserving

Organizations use cloud in a variety of different service models and deployment models (private, public and hybrid). There are a number of problems / security issues related to cloud computing, but these problems can be classified into two broad categories. , Security issues and security problems cloud provider customers face to face. Responsibility, however, goes both ways: The provider must ensure that the user is taking the appropriate security measures for the provider to protect user information, the user does not take measures, to use the authentication measures and a strong password that you need infrastructure and they are safe, make sure that the data and applications are protected customers.
The following Table1 gives clear description about exiting approaches those approaches description briefly.

Table1: Description of various existing approaches

APPROACH	DESCRIPTION
Proofs of Retrievability	1. They have developed a new encryption building blocks known as proof of retrievability (PoR). 2. Makes it possible to determine a POR is an archive file.
Public verifiability and the supporting of data dynamics for	1. They are achieved at the same time, the proposed public verifiability for cloud data storage that block both less and stateless and verification, the general formal model PoR. 2. In particular order to support the block insert missing, they are equipped with

cloud data storage	PoR Construction proposed the ability to assist for data manipulation fully dynamic in existing schemes in most cases. 3. And prove the safety of the construction work that has been proposed, they justify the performance of the method through comparison with the concrete implementation of state-of-the-art.
Effective and flexible distributed scheme	1. In comparison with many of the predecessor to provide binary results about the state of the storage distributed between servers, the only challenge in this work response protocol provides the localization of data errors. 2. Delete, add, and update: In contrast to the work of most conventional to ensure the integrity of the remote data, the new scheme supports operations on data blocks, including a dynamic safe and efficient. 3. Byzantine failure, malicious data modification attacks, and even if the conspiracy attack, performance analysis and a wide range of security, the proposed scheme shows that it is resilient and very efficient for the server.
Definitional framework and efficient constructions for dynamic provable data possession	1. They had been using the rank information to organize a dictionary entry. Therefore, it can be used as an insert which is such authentication, supports authentication operation efficiently, and delete files at the block level. 2. They are using the standard assumptions, to prove the safety of the proposed structure. 3. They also show how to extend the work that has been proposed for not only data possession guarantee the hierarchical file system, to support the file data itself.

E. Third Party Auditor

The safety of the back further in the case, including the framework that is used in the tool education, practice, and authentication, professional audit, is necessary for the transition of security from the virtualization infrastructure and data center prior art it has been exceeded. However, implementation audit requirements, technology is important for as a service as an application service, to maintain the level or audit, of the trust of virtual computing and even starts to get space, support and critical thinking.

II. PROBLEM STATEMENT

A. The System and Threat Model

The cloud data storage majorly containing three different entities, as given in fig.1: the cloud user, has lot of metadata to store in the cloud; the cloud server, which manage by cloud service provider to provide the data spaces and commuting resources; the third party auditor, who has trustable behalf of cloud users while accessing or retrieval from cloud storage requests. Cloud users will trust data storage and maintains on the cloud server. Cloud users may also interact the cloud service dynamically to access and update their stored data of various applications. To save the burden of computation resources and online burden, cloud user utilize services of TPA for ensuing storage integrity of their outsourced data, while trusted to keep their private data from TPA.

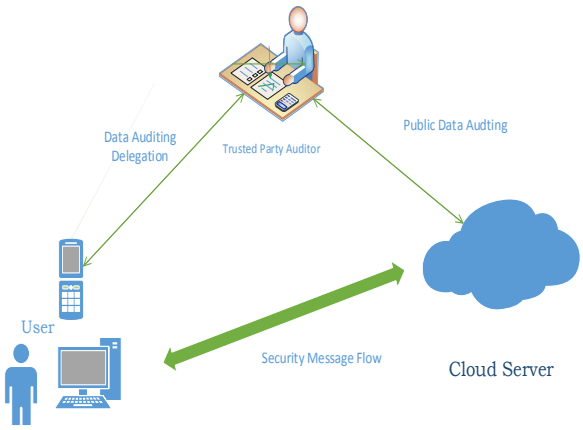

Fig. 1 The Architecture of clod data storage service

We believe that the existence of a partial trust as CS [16] does. That is, in the majority of time it behaves properly and does not deviate from the prescribed protocol execution. However, for its own benefit the CS could deliberately neglecting to maintain or delete data files are rarely accessed users belonging to the ordinary cloud. We assume the TPA, which is in the business of auditing, is reliable and independent, and therefore has no incentive to conspire with either the CS or the users during the audit process. However, damage to the user if the TPA could learn the data externalized after the audit. CS authorize the delegate to respond to audit the TPA, the user can sign a certificate of audit of concession rights to the public key of the TPA, and all audits of TPA is authenticated with a certificate of this type. These authentication handshakes are omitted in the following presentation.

B. Design Goals

To enable privacy preserving public auditing mechanism for data storage cloud under the previous model, the design of our protocol should achieve the following security guarantees and performance.

1) *Public auditability:* To allow TPA to verify the accuracy of the data cloud on demand without having to retrieve a copy of all information and adoption, online additional burden on users of the cloud.
2) *Storage correctness:* To ensure that there is no cloud server cheating that can pass the audit of the TPA without storing user data intact indeed.
3) *Privacy-preserving***:** To ensure that the TPA cannot derive the content of user data from information gathered during the audit process.
4) *Batch auditing:* TPA capacity to allow safe and efficient audit to address multiple audit delegations possibly large number of different users simultaneously.
5) *Lightweight:* To allow TPA to audit with minimum communication and computation overhead.

III. PUBLIC AUDITING MECHANISM

A public audit scheme consists of four algorithms (*KeyGen*, *SigGen*, *GenProof* and *VerifyProof*). ***KeyGen*** is a key generation algorithm that is executed by the user to the system configuration. ***SigGen*** is used by the user to generate verification metadata, which may consist of MAC, signatures, or any other related information to be used for the audit. ***GenProof*** is managed by the cloud server to generate a proof of correctness of data storage while ***VerifyProof*** is run by the TPA to audit the test from the cloud server. Running a public audit system consists of two phases, Setup and auditing:

- *Setup*: The user initializes the public parameters and secret system by running *KeyGen* and pre-processes the data file F using *SigGen* to generate verification metadata. The user then saves the file data and metadata F check in cloud server, and removes your working copy. As part of the pre-processing, the user can modify the data file F by expanding or including additional metadata to be stored on the server.
- *Audit:* The TPA, in order to ensure that it holds the data file F properly at the time of the audit, cloud server issues a challenge to the cloud server or audit messages. Cloud server to derive the response message from the verification of data and metadata to run the *GenProof*, stored in the file function F. TPA to verify the response through *VerifyProof* then.

Our framework assumes the TPA has been, which is a desirable property managed by our proposed solution. It is easy to extend the above framework to capture a complete audit system state, essentially by splitting metadata verification into two parts that are stored by the TPA and cloud server respectively.
Our design assumes no additional property in the data file. If the user wants more resilient error, he / she can always redundantly encoding the first data file, and then use our system with the data file that has the integrated correction codes errors.

A. KeyGen Process:

The keygen process will execute between user and cloud server, the user will register to cloud server the user should provide a user secure key that will be represent *sk* and additionally to that key the cloud server will add a public key to user key *pk*. By using both secure key and public key cloud server a generates a user security key by applying as *KeyGen (sk, pk)*.

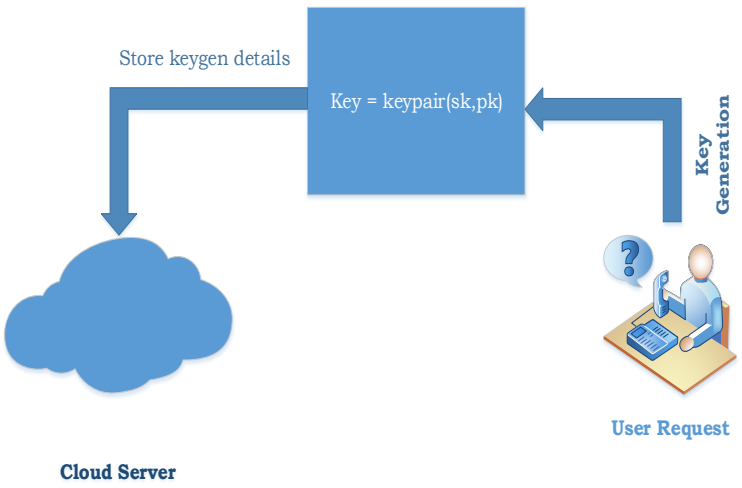

Fig. 2 KeyGen Process

B. SigGen Process

The signature generation process will generate between user and cloud server, when a user upload his data to cloud server, the cloud server will give signature of the data by using the data file *names* as attribute, it will apply *SSG(names)*.

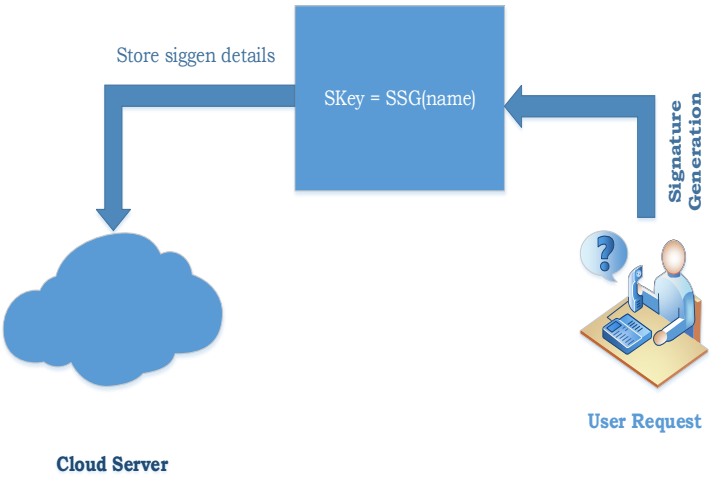

Fig. 3 SigGen Process

C. GenProof process

The generation proof is applied by cloud server, the cloud server will collect user data from trusted party auditor, then the cloud server will generate generation proof to that user data by applying challenge *chal(fk,vk)* the fk will represents a file data key and the vk will attach by the trusted party auditor key to that file this will be useful to trusted party auditor at the time of verification proof.

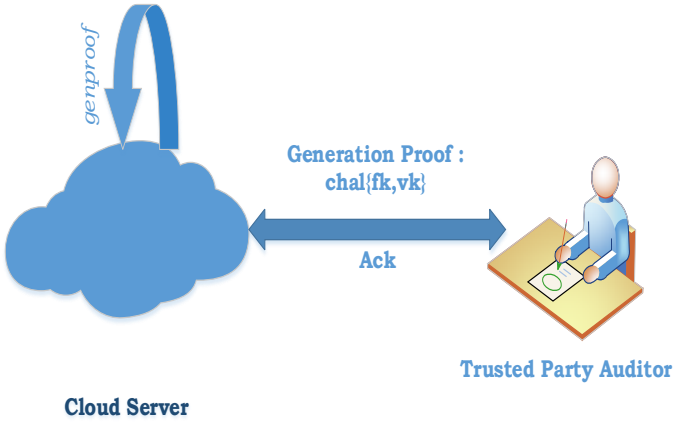

Fig. 4 GenProof Process

D. VerifyProof process

The verification proof is generated by trusted party auditor, the auditor will collect data of user that should be generated a generation proof by cloud server. The *GenProof* generated data verify by TPA with matching of *vk* matching of the data.

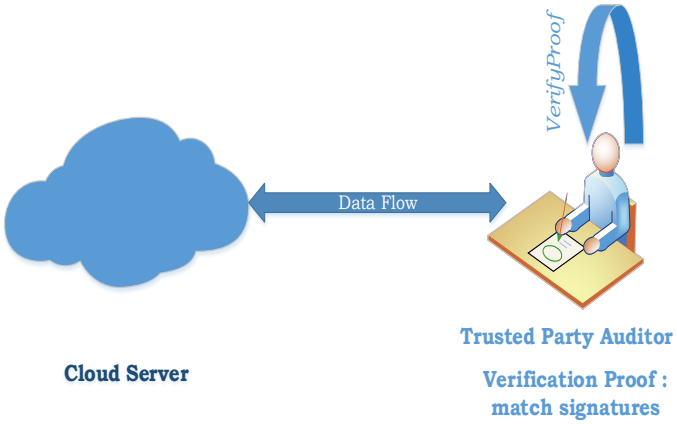

Fig. 5 VerifyProof Process

IV. EVALUATION

To evaluate our proposed public audit mechanism, we utilize the CLOUDBEES service to host our cloud application and performing the multiple users to upload data and we performing auditing service using the TPA and maintain generation proof from cloud service provider. The following figures gives the description about how our application working procedure.

The following fig 6, gives the general registration process to any user. To valid the user we use two level credentials system as we follow figure 7 and 8 gives the two user login process, the login process first one is general user login system and second one is the user needs to provide a One Time Password. KeyGen algorithm describes about user key generation based on the system configuration.

The valid user upload data files to cloud server, to upload file the user need to give some details of file and give the relative path of the file to upload file to cloud server.

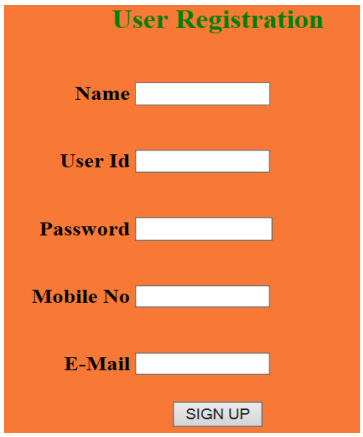

Fig. 6 User Registration form

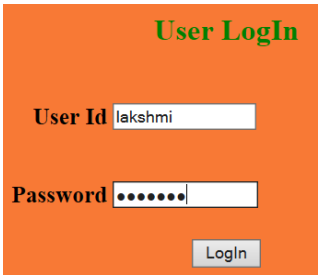

Fig. 7 KeyGen Stage One

Fig. 8 KeyGen Stage Two

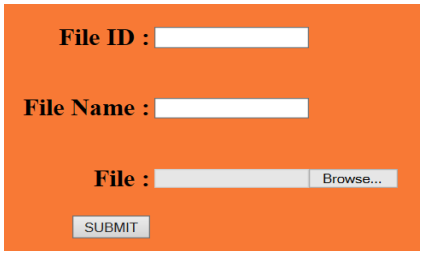

Fig. 9 User File Upload Process

After completion of file uploading the user needs to generate the signature of that file, it should we processing as follows in figure 10. *SigGen* is used to generate a verification of user information metadata by consisting of metadata based on the user information. This will help to improve audit the information.

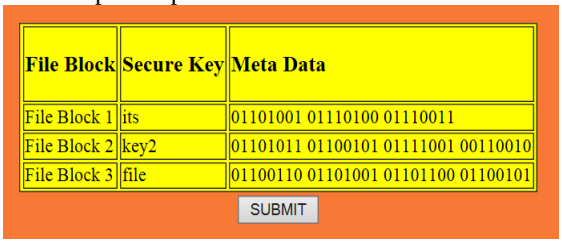

Fig. 10 File *SigGen* Process

GenProof is managed by the cloud server to generate a proof of correctness of data storage while *VerifyProof* is run by the TPA to audit the test from the cloud server.

The cloud server *GenProof* completion it will send to TPA after that TPA is verify the cloud server response and sending *VerifyProof* to client.

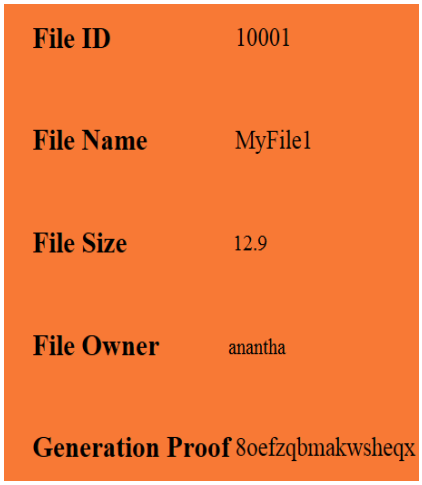

Fig. 11 GenProof from Cloud Server

Fig. 12 VerifyProof from TPA

V. RELATED WORK

Ateniese et al. [1] are the first to consider public auditability in their defined "possession of verifiable data" (PDP) model to secure possession of the data files in the tanks are not trusted. Their scheme uses RSA-based homomorphic linear authenticators for auditing outsourced data and suggests a random sampling of a few blocks of the file. However, the ability of public audit in their scheme requires linear combination of the blocks in the sample exposed to the external auditor. When used directly, the protocol is not provable preserve privacy, so information leaks user data to the auditor may occur. Juels et al.[4] In order to ensure both the data file service system as "owned", the remote file in the "search ability", are used for error correction code and random checking of "evidence recovery potential" model (it describes a POR). However, the number of audit questions that can be performed by the user, which is fixed a priori, it is not supported in its main outline. Auditability of public, for the POR of public described this approach is the creation of a simple Merkle trees can operate in only the data that has been encrypted.

Dodis et al. [11] give a survey of the different variants of PoR with private auditability. Shacham et al. [5] to design an improved PoR scheme built from BLS signature [7] with complete safety tests on the security model defined in [4]. As in the construction in [1], using the publicly verifiable homomorphic linear authenticators that are constructed from the BLS signature provably secure. Based on the BLS elegant construction, compact and public verifiable scheme is obtained. Again, their approach does not support preserving privacy audit for the same reason. [1] Shah et al. [2], [6] proposed that allows TPA to keep honest online storage by encrypting the data first and then sending a number of symmetrical hashes of pre-calculated on the encrypted data to body auditor. The auditor verifies both the integrity of the data file server and possession of a decryption key previously committed. This scheme only works for encrypted files, and it suffers from the fullness of the state auditor and the limited use that can potentially lead to charge online users when keyed hashes are exhausted.

In other related work, Ateniese et al. [19] propose a partially dynamic PDP schema version before, using only symmetric key cryptography, but with a limited number of audits. In [9], Wang et al. consider similar support for storing partial data in a distributed dynamic with the added feature of location data error scenario. In a later work, Wang et al. [3] propose combining HLA based on BLS with MHT to support both public auditability and data dynamics complete.

Almost simultaneously, Erway et al. [10] developed a scheme of jump lists to allow possession of demonstrable data supported by the dynamics of right from. However, the verification in these two protocols require the linear combination of the sample block as [1], [5], and therefore does not support the preservation of privacy audit. And to provide an efficient method for the guarantee provable and audit the accuracy of the data stored remotely scheme all of the above, but none of them, for privacy protection the public audit of cloud computing it does not meet all the requirements. More importantly, neither consider the audit batches can reduce the cost of computing the TPA when facing audit team number significantly these methods.

VI. CONCLUSION

In this paper, we have a tendency to propose a public auditing mechanism for knowledge storage security in cloud computing. We have a tendency to utilize the liner appraiser and random masking to audit the information by TPA, the audit can minimize the burden on the cloud server to store record within the cloud storage and conjointly resolve the matter of information outflow. Whereas TPA will at the same time handle multiple sessions of various users to audit their external knowledge files, additional expand our public audit protocol protective privacy during a multi-user setting wherever TPA will perform multiple auditing tasks during an approach by batch for higher potency. Intensive analysis shows that our schemes square measure demonstrably secure and extremely economical.

REFERENCES

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," October 2007.
[2] M. A. Shah, R. Swaminathan, and M. Baker, "Privacy preserving audit and extraction of digital contents," 2008.
[3] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," Sep. 2009.
[4] A. Juels and J. Burton S. Kaliski, "PORS: Proofs of retrievability for large files," October 2007.
[5] H. Shacham and B. Waters, "Compact proofs of retrievability," Dec 2008.
[6] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan, "Auditing to keep online storage services honest," 2007.
[7] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," 2004.
[8] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," 2008.
[9] C.Wang, Q.Wang, K. Ren, and W. Lou, "Ensuring data storage security in cloud computing," July 2009.
[10] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," 2009.
[11] Y. Dodis, S. P. Vadhan, and D. Wichs, "Proofs of Retrievability via hardness amplification," 2009.

