# Public Batch Auditing For Secured Cloud Storage

Tushar R. Puranik1, Avani B. Pagar2, Mayur B. Bagad3, Amruta J.Dhanmeher 4.
1 Student at Brahmavalley College Of Engineering
2 Student at Brahmavalley College Of Engineering
3 Student at Brahmavalley College Of Engineering
4 Student at Brahmavalley College Of Engineering

upuranik11@gmail.com
avanipagar57@gmail.com
mayurbagad99@rediffmail.com
amrutadhanmeher@gmail.com

**Abstract—** Cloud computing is a great invention in computer science by using which user can remotely access data from cloud. But storing data on cloud has no security to the data for providing security to that data people developed the term TPV. TPV stands for " Third Party Verifier", TPV is the third party tool which provide special type of scheduler, scheduler provides a watch on that data after a specified time. In day to day life it is very difficult to secure your data on cloud any intruder can access your data easily so prevent this attack on cloud we are developing system "Public Batch Auditing For Secured Cloud Storage". This system basically provides a security to those users who storing data on cloud. In this paper we are extending the previous system by using automatic blocker for privacy preserving public auditing for data storage security in cloud computing. We utilize the public key based homomorphism authenticator and uniquely integrate it with random mask technique and automatic blocker to achieve a privacy-preserving public auditing system for cloud data storage security while keeping all above requirements in mind. Extensive security and performance analysis shows the proposed schemes are provably secure and highly efficient.

**Keywords—** TPV, Preserving, Security, Authenticator, Protocol Blocker



## I. INTRODUCTION

Cloud Computing has been envisioned as the next-generation architecture, due to its long list of advantages:on-demand network access, usage-based pricing, location independent, self-service, resource elasticity. While these advantages of using clouds are unarguable,due to the opaqueness of the Cloud. As there are separate administrative entities, the operation details i.e internal operation of Cloud Service Providers (CSP) may not be known by cloud users. The outsourcing of data is also relinquishing user's ultimate control over the fate of their data.

As we have studied in previous papers correctness of the data in the cloud is being put at risk due to the following reasons. First of all, They are still facing the broad range of both internal and external threats for data integrity. The infrastructures under the cloud are much more reliable and powerful than personal computing devices. Examples include hiding data loss incidents so as to maintain a reputation, cloud service providers, reclaiming storage by discarding data that has not been or is rarely accessed. Although outsourcing data into the cloud is economically attractive for the cost and complexity of long term large scale data storage, it does not give any guarantee on data availability and integrity.

Thus, Unauthorized data leakage still remains a problem due to the exposure of encryption keys. It does not completely solve the problem of protecting data privacy but just reduces it to the one of managing the encryption keys.Our work is among the first few ones to support privacy preserving public auditing in Cloud Computing, with a focus on data storage. We are going to tackle the problem that how to enable a privacy-preserving third-party auditing protocol, independent of data encryption. To address these problems, our work uses the technique of public key. Which based homomorphic authenticator and which enables TPV to perform the auditing without demanding the local copy of data and thus reduces the communication and computation overhead as compared to the data auditing. By integrating the homomorphic authenticator with random mask technique,Our protocol guarantees that TPV could not learn any knowledge about the data content stored in the cloud server during the efficient auditing process.

## II. LITERATURE SURVEY

1. Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing

Cloud Computing has been envisioned as the next-generation architecture. It moves the application software and databases to the centralized large centers of data. Whereas the management of the services and data may not be fully trusted.Paradigms from this brings recently added security challenges, which are not well understood. This studies the problem of ensuring the integrity of data storage in Cloud Computing. Particularly we consider the task of allowing a Third Party Verifier (TPV), on behalf of the cloud client and to verify the integrity of the dynamic data stored in the cloud. The introduction of TPV eliminates the involvement of client

through the auditing of users data stored in the cloud is intact, which can be important in achieving economies of scale for Cloud Computing. Main support for data dynamics via the most general forms of data operation, such as insertion,deletion and block modification,is also a significant step. Services in Cloud Computing are not limited to backup data or archive only. The prior works on ensuring remote data integrity often lacks the support of either public verifiability or dynamic data operation,our paper achieves both. We first identify the potential security problems and security of direct extensions. Particularly,to achieve efficient data dynamics. Extensive security and performance analysis show that the proposed scheme is highly efficient and provably secure [10].

2. Privacy-Preserving Audit and Extraction of Digital Contents

By studying that the growing number of online services, such as Yahoo!,Google and Amazon, are starting to charge users for their storage. Customers use these services to store valuable data such as important files, family photos and videos,emails, and disk backups. A customer must entirely trust such external services to maintain the integrity of hosted data and return it intact. There is no service is accurate than this. To make storage services responsible for data loss, we presents our protocols that allow a third-party auditor to verify the data stored by a service and return the data to the customer. Our protocols are privacy preserving, in that they never acknowledge the data contents to the auditor. The solution which we have given removes the burden of verification from the customer, infact both the customer's and storage service's fear of data leakage, and provides a method for independent arbitration of data retention contracts [9].

3. Provable Data Possession at untrusted Stores

We are introducing a model for provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates possible proofs of possession by sampling random sets of blocks from the server, which reduces input-output costs. The client maintains a constant amount of metadata to verify/check the proof. The challenge protocol transmits a small and constant amount of data, which minimises network communication. Thus, the Provable Data Possesion model for remote data checking supports large data sets in widely-distributed storage systems. We are presenting two provably-secure PDP schemes that are more efficient than previous, even when compared with schemes that achieve weaker guarantees of security. The overhead at the server is low, as op-posed to linear in the size of the data. The 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[8].

## III. EXISTING SYSTEM

We consider a cloud data storage service involving three different entities, as illustrated in fig.1: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 Verifier(TPV), who has expertise one 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.

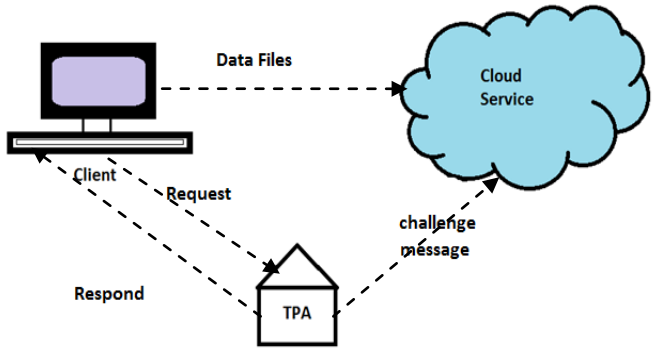


Fig 1

We have followed similar definition of previously proposed schemes in the context of remote data integrity checking [7][11][12] and adapt the framework for our privacy-preserving public auditing system.

A public auditing scheme consists of five algorithms (KeyGen,SigGen,GenProof, VerifyProof,protocol verifier).

**KeyGen** is a key generation algorithm that is run by the user to setup the scheme.

**SignGen** is used by the user to generate verification metadata, which may consist signatures, or other related information that will be used for auditing.

**GenProof** is run by the cloud server to generate a proof of data storage correctness, while VerifyProof is run by the TPV to audit the proof from the cloud server, protocol verifier is used by the cloud server .

Our public auditing system can be constructed from the above auditing scheme in three phases, Setup, Audit,Pblock

**Setup**: The user initializes the public and secret parameters of the system by executing KeyGen, and pre-processes the data file F by using SigGen to generate the verification metadata. The user then stores the data file F at the cloud server, delete its local copy, and publish the verification metadata to TPV for later audit. 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 TPV 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 by executing GenProof. Using the verification metadata, the TPV verifies the response via VerifyProof.

**PBlocker**: Once the user initializes the parameters the system checks the all the specified parameters and validates the protocol.

## IV. PROPOSED SYSTEM

In the introduction we bring the public audit ability with achieving economies of scale for cloud computing. This syatem presents our public auditing scheme for cloud data storage security. We start from the overview of our public auditing system and discuss two schemes and their demerits. Then we present our main result for privacy-preserving public auditing to achieve the mentioned design goals. We also show how to extent our main scheme to support batch auditing for TPV upon delegations from multi-users. Finally we adopt the automatic blocker at the cloud server, whenever a unauthorized user access the users data from cloud storage, the system runs an tiny application to monitor the user inputs, it matches to give access otherwise does not give user access by blocking the protocols. Recently, the notion of public audit ability has been proposed in the context of ensuring remotely stored data integrity under different systems and security models [7][9][11][12]. Public audit ability allows an external party, in addition to the user himself, to verify the correctness of remotely stored data. However, most of these schemes [7][9][11] do not support the privacy protection of users' data against external auditors,i.e.they may potentially reveal user data information to the auditors. This 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 TPV just for the storage security of their data, do not want this auditing process introducing new vulnerabilities of unauthorized information leakage towards their data security. enable a privacy-preserving third-party auditing protocol, independent to data encryption, is the problem we are going to tackle in this paper. Our work is among the first few ones to support privacy-preserving public auditing in Cloud Computing, with a focus on data storage. The System and Threat Model: We consider a cloud data storage service involving three different entities, as illustrated in fig. 1: the cloud user (U), who has large amount of data files to be stored.

International Journal of Computer Science and technology 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 (we will not differentiate CS and CSP hereafter.); the Third Party Verifier (TPV), 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.

## V. ALGORITHM USED

1.**KeyGen Algorithm**
KeyGen is a algorithm that is run by the user to setup the scheme.

2.**SignGen Algorithm**
SigGen is used/run by the user to generate verification metadata, which may consist of MAC, and other related information that will be used for auditing.

3.**GenProof Algorithm**
GenProof is algorithm run by the cloud server to generate a proof of data storage correctness.

4.**VerifyProof Algorithm**
VerifyProof is algorithm run by the TPV to audit the proof from the cloud server

5.**Protocol Verifier Algorithm**
Protocol verifier is algorithm used by the cloud server.

## VI. MATH MODULE

Here S is the set contains E and P.
E is the envoirment & P is the phase and
Envoirement is divided as follows:

E1,E2,E3 are the enviorments.
E = {E1, E2, E3}
E1 = user environment
E2 = Cloud Environment
E3 = Third Party Environment

P1,P2,P3,P4,P5,P6 are the phases of module:
P = {P1, P2,P3 ,P4,P5,P6}

P1:
The first module consist of login part as users id and password.
If user name having any of these charecters [A-Z a-z] and password containing any of these [A-Z a-z 0-9] charecters.
The user name and password should have valid length.
If above condition is satisfied then it is valid id and password otherwise login fails.

**P1:**
If x = login
F(x) =login successful
If U Є [A-Z a-z] and
P Є [A-Z a-z 0-9] and
Length (P) >= 6 and U ≠ NULL and p!=null and U=u1 and P=q1
Else
Login fails
Where,
P: Password
U: Username
P1=Predefinded Password
u1=Predefinded Username

P2:
It is second module which consist of Pseudo Random Key Generator where public key and secret key are generated using two random numbers.

**P2**
P2(a,b)
Call PKG(a,b)
Return Pk,Sk

Where,
PKG= Pseudo Random Key Generator
Pk = Public key
Sk= Secret key
a,b = two random number

P3:

In this module the user uploads his file on the cloud and with that cyphertext using RSA in the form of encryption is given.SHA1 is used for generating metadata at both ends user and TPV.

σ is Tag of a File which is used for recognizing file.And Fid is file identifier.

**P3:**

P3(F, PK)
If F = User's file
Then
Encrypt message as Cp using RSA technique
Compute σ & with the help SHA1 Technique
Send X ={σ , Fid} to TPV
Send Cp to cloud
Where,
F=User File
σ:Tag of a File F
Fid : file identifier
X : metadata of File F
Cp = Cypher text Message

P4:

In this module,if the user is authorised that is having licence key then it will retrieve chal message
It computes metadata using file identifier it will retrieve challenge message otherwise rejects it.

**P4:**

If AZ(Lk)
Then
Retrieve(chal)
Compute X using Fid
Return X to TPV
Else
Reject(chal)
Where,
Pk = Public key
Lk= License key
AZ= Authorization
Fid= File identifier
X= Metadata
Chal= Challenge message

P5:If X=Ox that means if metadata of both ends are are same then third party tool will notify user that data is safe otherwise data is manipulated.

**P5:**

Get X
If X = Ox then
Verify = 1
Else
Verify = 0
Notifyuser()
Where
X = Proof generated from cloud
Ox = proof stored on TPV

P6:

In this module user having correct id and password can get access to his files.
In this module if user is entering wrong id or password or both for fixed attempt then
He will barred for accessing filles.

**P6:**

Define P
Get Request R from user Ui
IF Ui is Valid Then
IF R Є P Then
Give File Access
ELSE
Notify-> Not Enough Permission
End If
ELSE
Notify-> Permission Denied
M = Check Attempt
IF M==N
Block USer
End IF

Where,
M=Max attempts.
N=Number Of valid attempts.
P=Permissions.
Ui=Request by user.
R=Request .

## Conclusion

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. We utilize the homomorphic authenticator and random mask technique to guarantee that TPA would not learn any knowledge about the data content stored on the 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 TPA can perform the multiple auditing tasks in a batch manner i.e simultaneously.

## ACKNOWLEDGMENT

It is my pleasure to express my knowledge to my respected sir Mr. D.B. Bagul, Computer Engineering,and our HOD Mr.H.D.Sonawane BVCOE&RI, Nashik for their valuable guidance, inspiration and continues support. This paper could not be success without apps analysis done which help to understand the necessity for this paper.

## REFERENCES

[1] IEEE INFOCOM 2010, San Diego, CA, March 2010.
[2] P. Mell, T. Grance (2009),"Draft NIST working definition of cloud computing", [Online]Available: http://www.csrc.nist.gov/groups/SNS/cloud
[3] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A.Rabkin,I.Stoica, M. Zaharia,"Above the clouds: A berkeley view of cloud computing",University of California,Berkeley,Tech.Rep.UCB-EECS-2009-28,Feb 2009
[4]N.Gohring(2008),"Amazon'ss3 down for several hours", [Online] Available: http://www.pcworld.com/businescenter/ article/142549/amazons s3 down for several hours.html
[5]Amazon.com (2008),"Amazon s3 availability event: [Online] Available: http://wwwstatus.aws.amazon. com/s3-20080720. html
[6]S.Wilson (2008),"Appengine outage", [Online] Available: http://www.cio-weblog.com/50226711/ appengine outage. php.
[7]B.Krebs,"Payment Processor Breach May Be Largest Ever",[Online] Available:http://www.voices.washingtonpost.com /securityfix/2009/01/payment processor breach mayb. html, Jan. 2009.
[8]G.Ateniese,R.Burns,R.Curtmola,J.Herring,L.Kissner,Z. Peterson, D.Song,"Provable data possession at untrusted stores",Cryptologye Print Archive, Report2007/202, 2007, [Online] Available: http://www.eprint.iacr.org/.
[9]M.A.Shah,R.Swaminathan,M.Baker,"Privacy-preserving audit and extraction of digital contents",Cryptology ePrint Archive, Report2008/186,2008,[Online] Available: http:// www.eprint.iacr.org/.
[10]Q.Wang,C.Wang,J.Li,K.Ren,W.Lou(2009),"Enabling public verifiability and data dynamics for storage security in cloud computing",in Proc.of ESORICS'09, Saint Malo, France.
[11] Cloud Security Alliance (2009),"Security guidance for critical areas of focus in cloud computing", [Online] A vailable: http:// www.cloudsecurityalliance.org.
[12]H.Shacham,B.Waters,"Compact Proofs of retrievability", in Proc. of Asiacrypt 2008, Vol. 5350, Dec 2008, pp. 90–10.

**TusharR.Puranik:-** Student Scholor at Brahmavalley College of engineering(Pune University)

**Avani B.Pagar:-** Student Scholor at Brahmavalley College of engineering (Pune University)

**Mayur B.Bagad:-**Student Scholor at Brahmavalley College of engineering (Pune University)

**Amruta J.Dhanmeher:-**Student Scholor at Brahmavalley College of engineering(Pune University)