ISSN 2278-3091

**International Journal of Advanced Trends in Computer Science and Engineering**, Vol.2 , No.6, Pages : 42-46 (2013)
*Special Issue of ICETEM 2013 - Held on 29-30 November, 2013 in Sree Visvesvaraya Institute of Technology and Science, Mahabubnagar – 204, AP, India*

# Outsourcing and Discovering Storage Inconsistencies in Cloud Through TPA

**Sumathi Karanam[1], GL Varaprasad[2]**

Student, Department of CSE, QIS College of Engineering and Technology, Ongole, AndhraPradesh,India [1]
Assistant Professor, Department of IT, QIS College of Engineering and Technology, Ongole,AndhraPradesh,India [2]
sumathi.karanam@gmail.com
glv.prasad19@gmail.com

**Abstract--**Cloud computing has changed the way computing takes place. It is the technology that enables outsourcing of computing and storage to a public cloud maintained by cloud service providers. Cloud users can use cloud storage and other facilities without capital investment in pay as you use fashion. As the data is stored in remote server in the data center of cloud service provider, there is security concern among the cloud users. Wang et al. studied this problem and ensured data integrity in cloud storage by proposing third party auditing concept. The third party auditor is responsible to verify the integrity of data on behalf of cloud data owners. The auditing mechanism monitors the data dynamics. The solution makes use of bilinear aggregate signature for simultaneous auditing and Merkle Hash Tree for secure block level authentication. In this paper we implement a prototype, Java custom simulator, which implements the proof of concept proposed by Wang et al. The empirical results revealed that the prototype is effective to demonstrate the efficiency of auditing mechanism to ensure data integrity.

**Index Terms –** Cloud computing, outsourcing data, cloud storage security, public auditability

## I. Introduction

Cloud computing has finally been made a reality which enables people of all walks of life to gain access to cloud services through Internet. It is a new computing model which takes place through Internet. It provides three service models such as Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS). With virtualization technology the could computing became a viable solution. High quality of services can be made possible with cloud computing as the cloud has required computing resources that can be used by public without the need for investment. However, there are many issues pertaining to cloud storage. Security of cloud storage is very important concern. Data integrity is the important feature expected by cloud users. To ensure data integrity when the data stored in cloud, there were many security models proposed in the literature. They include [1], [2], [3], [4], [5], [6], and [7]. These solutions focused on data integrity of cloud storage in one way or other. Each solution may differ technically but the purpose of the all solutions remains the data integrity of cloud storage only. Auditing mechanism may be of two types namely private and public. The former provides auditing service to private users while the latter provides the services to all in general. The clients also can delegate auditing services to a third party auditor.

There is another problem in the previous designs. They could not provide robust data dynamics along with security mechanisms. The proposed system takes care of data dynamics too along with auditing services. The proposed system also supports block insertion which is the main drawback of earlier systems. The system also provides scalable public auditing for cloud storage security. Batch auditing is also a feature of the proposed system in which third party auditing works in parallel. We built a prototype, a custom Java simulator, to demonstrate the proof of concept. The empirical results reveal that the prototype is effective in public auditing. The remainder of the document is structured as follows. Section II provides review of literature. Section III presents problem description. Section IV describes security mechanisms. Section V presents proposed solution. Section VI provides experimental results while section VII concludes the paper.

## II. Related Work

As the cloud computing is becoming popular more people are interested in outsourcing their data to cloud. This has caused many security concerns. Thus it has become an open problem for researchers. Many came with their solutions as explored in [8], [9], [10], [1], [2], [4], [6], and [11]. First of all a model by name “Provable Data Possession” is explored in [2] to ensure the integrity of outsourced data. They proposed an RSA-based solution which makes use of homomorphic tags for auditing data. They have achieved public auditability. However, there was some problem in data dynamics in the proposed solutions. Afterwards Wang et al. [10] proposed a secure model which is supports auditing of outsourced data besides supporting data dynamics. The solution also can determine data correctness and also the source of errors if any. Proof of irretrievability is another solution for cloud storage security proposed by Juels and Kaliski [7] which makes use of error-correcting codes for data possession. All the existing system strives to protect cloud storage in one way or other. However, it still needs improvement.

## III. Problem Description

The proposed system has three parties involved. They are namely clients or data owners, cloud service providers and third party auditors. The problem scenario is presented in figure 1. The cloud server provider maintains required storage space for outsourced data. The clients are responsible to store and retrieve data as and when required while the third party auditor is responsible to verify the integrity of data which is being flown between data owner and service provider. The third party auditor is a trusted entity that is responsible to audit data being flown for verification of integrity.

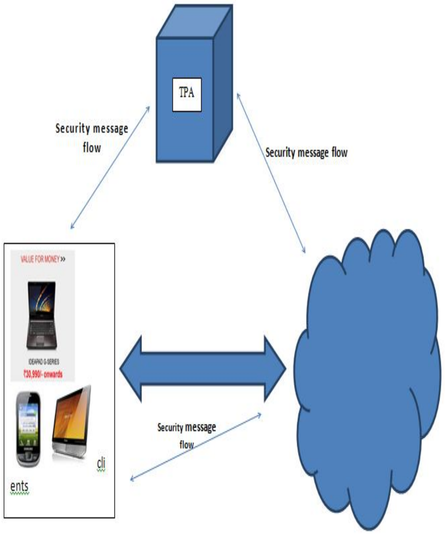


**Fig. 1** shows proposed architecture

Provided the architecture in fig. 1, the following assumptions are made in this paper.

- The third party auditor is trusted who does not create any security problems.
- There might be latent storage inconsistencies that are not disclosed by cloud service providers.
- Cloud service providers may delete some data of data owner for monetary gains or other reasons.

### Proposed Solution

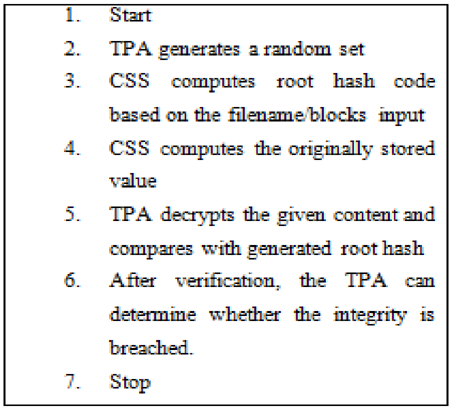
This section provides proposed solution by introducing a new scheme for cloud storage security. The proposed system supports both data dynamics and public auditing.

### Setup

This phase is used to generate security keys like private key and public key. It is achieved by invoking KeyGen(). In the pre-processing, it makes use of homomorphic authenticators and Meta data. The methods needs two arguments namely file and key. The content of file is divided into multiple blocks. Hash code is computed for each block. The hash code of two blocks is merged. Then this merged key is merged with other key made up of two merged keys. This process continues until all leaf nodes are found in the Merkle hash tree. Then the root element sent to cloud server.

### Data Integrity Verification

The data integrity verification is done by third party auditor. The TPA challenges server for block level data verification at regular intervals by sending file name and block randomly. On challenge, the root hash code is computed by server and returns the same along with original hash code to TPA. The TPA provides security keys and decrypted the content in order to compare the integrity of date. The procedure is as shown below.

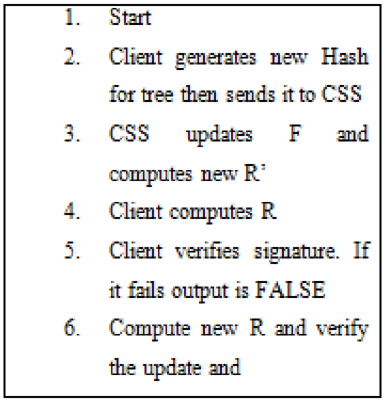


1. Start
2. TPA generates a random set
3. CSS computes root hash code based on the filename/blocks input
4. CSS computes the originally stored value
5. TPA decrypts the given content and compares with generated root hash
6. After verification, the TPA can determine whether the integrity is breached.
7. Stop

**Table 1**- Data integrity verification

### Data Modification and Data Insertion

Cloud users need to perform modifications to their data online frequently. These are known as data dynamics. Unlike some of the existing systems, the proposed system supports data dynamics. Data modification and data insertion are the two important operations required by the system. The former modifies existing data while the latter inserts new data. The procedure for data dynamics is as presented in table 2.



1. Start
2. Client generates new Hash for tree then sends it to CSS
3. CSS updates F and computes new R'
4. Client computes R
5. Client verifies signature. If it fails output is FALSE
6. Compute new R and verify the update and

**Table 2** – Algorithms for data dynamics

### Batch Auditing for Concurrency

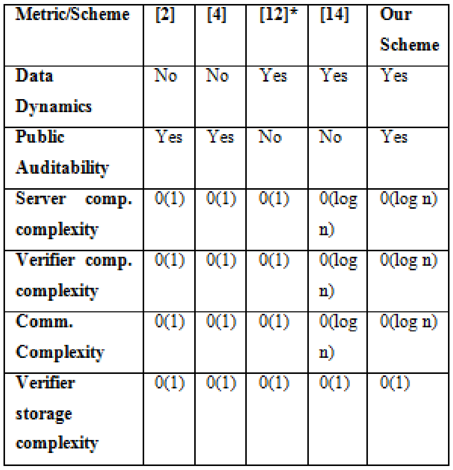
Cloud users need concurrent access to their resources. It does mean that the cloud servers have to support concurrent access. In the same fashion, the auditing mechanism also should work simultaneously. For this reason the proposed scheme supports concurrently using a concept named “Bilinear aggregate signature scheme” [12].

### Design Considerations

Public auditing and data dynamics are two considerations while designing this solution. The solution is made up of RSA based signatures and the BLS. They are 1024 and 160 bits based solutions respectively. With BLS queries and responses are shortest. Variable sized blocks are supported by RSA. The secure storage and block level verification is achieved by using Merkle Hash Tree. For achieving data dynamics the existing schemes such as PoR or PDP has to be extended. These schemes have security problems as they can be broken by adversaries. The changes are allowed in the existing blocks while insertions can be done at any point file. Static files are stored in PDP without error correction features. However, the proposed system is designed to be stateless and block less verification of data. This is a significant feature required by the system as the TPA does not need the actual content of files. Therefore no data is shown to third party auditor. Distributed cloud storage security is another design consideration. The data sent by clients might be stored in multiple servers and then retrieved without any problems. It does mean the file (F) is split into number of pieces and stored in multiple cloud storage servers.

## IV. Experimental Results

Table 3 presents many security schemes and the comparison is made with our approach. In [9] CPDP scheme is presented which is actually an extension to the scheme proposed in [11]. This enhancement enables supporting data dynamics with complete security. The proposed scheme is known as DPPP scheme. It is based on the RSA-based algorithm and BLS. The testbed includes Core 2 dual processor, 2GB RAM, 2.4 GHz processor. The table 3 shows the results of various data integrity tools.



| Metric/Scheme | [2] | [4] | [12]* | [14] | Our Scheme |
|---|---|---|---|---|---|
| Data Dynamics | No | No | Yes | Yes | Yes |
| Public Auditability | Yes | Yes | No | No | Yes |
| Server comp. complexity | 0(1) | 0(1) | 0(1) | 0(log n) | 0(log n) |
| Verifier comp. complexity | 0(1) | 0(1) | 0(1) | 0(log n) | 0(log n) |
| Comm. Complexity | 0(1) | 0(1) | 0(1) | 0(log n) | 0(log n) |
| Verifier storage complexity | 0(1) | 0(1) | 0(1) | 0(1) | 0(1) |

**Table 3**- Summary of literature compared with our scheme

As shown in table 3, the data presented reveal the fact that data dynamics are supported by our scheme along with public auditability. The other schemes do not support both as they support only either of them. Therefore the proposed system is better than the existing one.

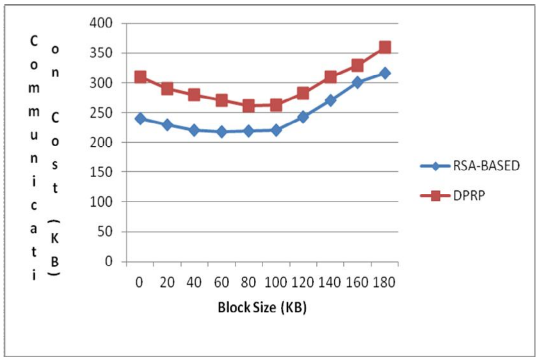


**Fig. 2** – Communication cost vs. Block Size

As shown in fig. 2, the communication cost of DPRP is higher than the proposed RSA based scheme. Higher performance is associated with the RSA based approach.

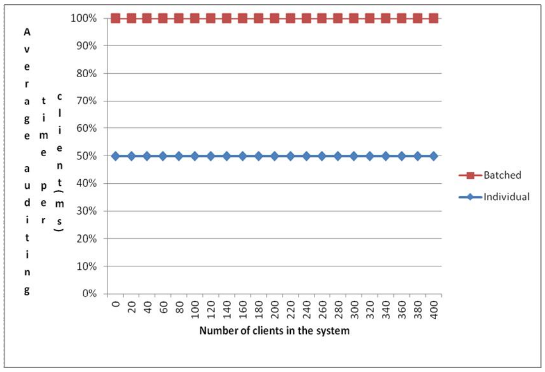


**Fig. 3** – Illustrates average auditing time per client

As shown in fig. 3, the horizontal axis represents number of clients in the system while the vertical axis represents average auditing time per client.

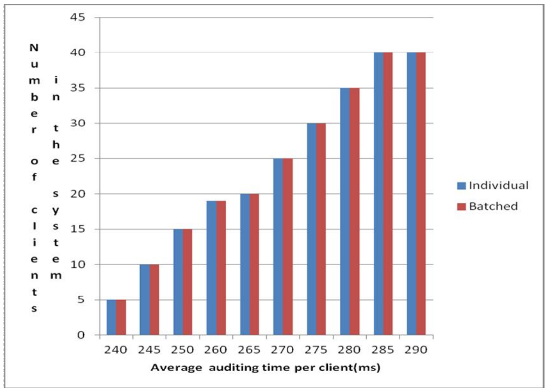


**Fig. 4** – Average auditing time per client

As seen in fig. 4, the average auditing time per client is presented in the graph. Better performance is shown by individual approach when compared to batch approach.

## V. Conclusion

In this paper we implement the cloud storage security mechanism proposed by Wang et al. [13]. This will consider cloud architecture with three parties namely cloud service provider, third party auditor and cloud client. The clients can perform data dynamics besides able to delegate the auditing task to third party auditors. The security mechanisms allow public auditability. It does mean that the verification services can be used by all users. The third party auditor is responsible to monitor all transactions for verification of integrity. There are two challenges resolved in this solution. The ability to support multiple verifications at a time and the ability to support on – demand block verification for integrity. These are achieved using bilinear aggregate signature and Merkle Hash Tree respectively. Provide all security features the third party auditor is capable of proving third party auditing services to public. We built a prototype, a custom Java simulator that demonstrates the proof of concept. The experimental results are encouraging.

### References

[1] A. Oprea, M.K. Reiter, and K. Yang, “Space-Efficient Block Storage Integrity,” Proc. 12th Ann. Network and Distributed System Security Symp. (NDSS ’05), 2005.

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

[3] E.-C. Chang and J. Xu, “Remote Integrity Check with DishonestStorage Server,” Proc. 13th European Symp. Research in ComputerSecurity (ESORICS ’08), pp. 223-237, 2008.

[4] M. Naor and G.N. Rothblum, “The Complexity of Online MemoryChecking,” Proc. 46th Ann. IEEE Symp. Foundations of ComputerScience (FOCS ’05), pp. 573-584, 2005.

[5] K.D. Bowers, A. Juels, and A. Oprea, “Proofs of Retrievability:Theory and Implementation,” Report 2008/175, Cryptology ePrintArchive, 2008.

[6] H. Shacham and B. Waters, “Compact Proofs of Retrievability,”Proc. 14th Int’l Conf. Theory and Application of Cryptology andInformation Security: Advances in Cryptology (ASIACRYPT ’08), pp. 90-107, 2008.

[7] A. Juels and B.S. Kaliski Jr., “Pors: Proofs of Retrievability forLarge Files,” Proc. 14th ACM Conf. Computer and Comm. Security (CCS ’07), pp. 584-597, 2007.

[8] K.D. Bowers, A. Juels, and A. Oprea, “Hail: A High-Availabilityand Integrity Layer for Cloud Storage,” Proc. 16th ACM Conf.Computer and Comm. Security (CCS ’09), pp. 187-198, 2009.

[9] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia,“Dynamic Provable Data Possession,” Proc. 16th ACM Conf.Computer and Comm. Security (CCS ’09), 2009.

[10] C. Wang, Q. Wang, K. Ren, and W. Lou, “Ensuring Data StorageSecurity in Cloud Computing,” Proc. 17th Int’l Workshop Quality of Service (IWQoS ’09), 2009.

[11] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z.Peterson, and D. Song, “Provable Data Possession at UnstructuredStores, “ Proc.14th ACM Conf. Computer and Comm., Security(CCS’07). Pp.598-609, 2007.

[12] D.Bonesh, C.Gentry, B.Lynn, and H.Shacham, “Aggregate andVerifiably Encrypted Signatures from Bilinear Maps,” Proc 22ndInt’l Conf. Theory and Applications of Cryptographic techniques (Eurocrypt’ 03), pp.416-432, 2003.

[13] Qian Wang, Cong Wang, Kui Ren, Member, Wenjing Lou and Jin Li, “Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing”. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 22, NO. 5, MAY 2011.

(

## AUTHOR

**Sumathi Karanam is** student of QIS College of Engineering and Technology, Ongole, AP, INDIA. She has received PG Degree in Computer Applicant and M.Tech Degree in Computer Science. Her main research interest includes Cloud Computing and Data Mining.

## CO-AUTHOR

**GLV Varaprasad** is working as an Assistant Professor in QIS College of Engineering and Technology, Ongole, and Andhra Pradesh, India. He has completed M.Tech from JNTUH. His main research interest includes Cloud Computing and Data Mining.