ID: 2016-ISFT-195

Audit of Data Integrity using PDP Technique in Cloud

Radhika Patil[1], Nirmala C.R[2]

[1]Assistant Professor, [2]HOD, Dept. of CS & E
Bapuji Institute of Engineering and Technology (B.I.E.T), Shamanur Road,
Davanagere-577001 Karnataka, India
[1]ps.radhika@gmail.com

Abstract: *Cloud computing provides vast online computing resources and services on-demand over the Internet, where users can remotely store their data into the cloud. It also reduces the maintenance of data from customer by providing a comparably low-cost, scalable, location-independent platform. This new model brings new security challenges, which requires independent auditing services. However in cloud, the clients have no direct physical possession of data. It shows that client faces security risks like loss of data or changes in data. Third Party Audit services are essential to ensure the integrity and availability of outsourced data and credibility on cloud computing. There are different existing auditing services available in cloud which audit data integrity remotely in static fashion but these are not applicable whenever data is dynamically updated in cloud, since it requires secure auditing techniques. To achieve this goal, we use Provable Data Possession (PDP) technique to verify the integrity of data without accessing it at an untrusted server while performing public audit service. This technique proposes interactive auditing that also preserves the privacy of data. In this audit service, it can issue periodic verification to monitor the change of outsourced data. To realize the audit model, we only need to maintain the security of the third party auditing tool and deploy a thin client to execute the verification protocol. Hence, this technique can be easily adopted in a cloud computing environment to replace the traditional Hash-based solution. This approach greatly reduces the workload on the storage servers, while still achieves the detection of any changes with high probability.*

Keywords: *Cloud computing, Data integrity, Provable Data Possession (PDP), Third party auditor, Public Audit, Privacy preserving.*

1. INTRODUCTION

The changes in both technology and business over the recent years has led to many IT infrastructure challenges. The business application architecture has evolved from desktop-centric to client/server solutions, and now to loosely coupled web services and service-oriented architectures (SOA). Each change has brought with it new challenges and opportunities for IT and its users.

Cloud computing means delivering useful functions and resources on-demand on pay per use basis while hiding the internals details that has many distributed components such as processing, storage, data, and software resources. Buyya et al. [1] have defined it as follows: “Cloud is a parallel and distributed computing system consisting of a collection of inter-connected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements (SLA) established through negotiation between the service provider and consumer.” It denotes a model on which a computing infrastructure is viewed as a “cloud,” from which businesses and individuals access applications from anywhere in the world on demand [1]. The main principle behind this model is offering computing, storage, and software “as a service.”

Cloud data storage refers to storing data to an off-site storage system maintained by a third party. The users can be relieved from the burden of local storage and maintenance of data. As users do not have physical possession of data, it becomes challenging to protect the data integrity in cloud. Inspite of many advantages associated with cloud computing, accessing and storing data on cloud storage has some security risks. This makes the cloud information inconsistent and unreliable. For sensitive and confidential data there should be some security mechanism to provide protection for private data. To keep away from the security risks, audit services are significant to make sure about the integrity and availability of data in clouds.

Cloud Infrastructures are still susceptible to security threats both from outside and inside the cloud [2] for the benefits of their possession, there exist various motivations for cloud service providers (CSP) to behave unfaithfully toward the cloud users [3] furthermore, the dispute occasionally suffers from the lack of trust on CSP. Consequently, their behaviors may not be known by the cloud users, even if this dispute may result from users’ own improper operations [4]. Therefore, it is necessary for cloud service providers to offer an efficient audit service to check the integrity and availability of the stored data [5]. Traditional cryptographic technologies for data integrity and availability, based on hash functions and signature schemes [6-7], cannot work on the outsourced data without a local copy of data. In addition, it is not a practical solution for data validation by downloading them due to the expensive transaction, especially for large-size files. Moreover, the solutions to audit the correctness of the data in a cloud environment can be formidable and expensive for the cloud users [2]. The audit service is significantly important for data assurance in clouds. The architecture used is based on interactive cryptographic verification protocol for public auditability.

1.1 PROBLEM STATEMENT

To implement an interactive PDP (Provable Data Possession) protocol to prevent the fraudulence of proof (Soundness Property) and the leakage of verified data (zero-knowledge property).

1.2 OBJECTIVES

The objective is to build a security service which will be provided with a trusted 3rd party audit, and would lead to providing only security services and wouldn't store any data in its system.

To securely introduce an effective third party auditor (TPA), the following two fundamental requirements have to be met:

- TPA should be able to efficiently audit the cloud data storage without demanding the local copy of data, and introduce no additional on-line burden to the cloud user;
- The third party auditing process should bring in no new vulnerabilities towards user data privacy.
- Detailing it further:
- To construct a Web service that would provide data integrity verification of the client data.
- Defining access list for sharing data securely with specific band of individuals.
- To construct thin client application which would call this web service before uploading/downloading the data to and from cloud.

1.3 EXISTING SYSTEM

To check the integrity of stored data without download, some researchers have proposed two basic approaches called provable data possession (PDP) [8] and proofs of retrievability (POR) [9]. Ateniese et al. [8] are the first to consider public auditability in their “provable data possession” (PDP) model for ensuring possession of data files on untrusted storages. Their approach was based on a probabilistic proof RSA-based technique for the static case to prove that clients’ data remain intact without downloading the stored data, which is called “verification without downloading”. When used directly, their protocol is not provably privacy preserving, and thus may leak user data information to the external auditor. In [10], the authors proposed a dynamic auditing protocol that can support the dynamic operations of the data on the cloud servers, but this method may leak the data content to the auditor because it requires the server to send the linear combinations of data blocks to the auditor. Juels et al. [9] describe a “proof of retrievability” (PoR) model, where spot-checking and error-correcting codes are used to ensure both “possession” and “retrievability” of data files on remote archive service systems. The authors complete their dynamic auditing system to be privacy preserving and it support the batch auditing for multiple owners However, auditing protocols will cause storage overhead on the server due to large number of tags [11]. Thus, new frameworks or models are desirable to enable the security of public verification protocol in cloud audit services.

1.4 PROPOSED SYSTEM

To avoid the security risks, audit services are important to ensure the integrity and availability of outsourced data on cloud computing. Provable data possession (PDP), which is a cryptographic technique for verifying the integrity of data without retrieving it at an untrusted server, can be used to realize audit services. The security is provided by an interactive PDP protocol to prevent the fraudulence of prover (soundness property) and the leakage of verified data (zero-knowledge property) [12]. The protocol provides these properties. This protocol is efficient with respect to periodic verification to reduce the audit costs per verification and implement abnormal detection timely.

2. ARCHITECTURE AND IMPLEMENTATION

The data integrity verification can be implemented by TPA (Trusted Third Party) without the help of data owner. In this architecture, the data owner needs to dynamically interact with CSP to access or update their data for various application purposes. It is based on the assumption that neither CSP is not required to guarantee the security of stored data, nor the data owner has the ability to collect the evidences of CSP’s fault after errors occur. Hence, TPA, as a trust third party, is used to ensure the storage security of their outsourced data. It is assumed that the TPA is reliable and independent:

- TPA should be able to make regular checks on the integrity of data at appropriate intervals;
- TPA should be able to take evidences of data in terms of authentic records for all data operations.
- In this audit architecture, the idea is to maintain the security of TPA. This is because it is more easy and feasible to ensure the security of one TPA than to maintain the credibility of the whole cloud. Hence, the TPA could be considered as the root of trust in clouds.
- To support this architecture, a cloud storage provider only needs to add a corresponding algorithm module to implement this audit service. The audit process could

be considered as an interactive implementation. This implementation is designed as a server daemon to respond audit requests of TPA through cloud interfaces. This daemon is just a simple lightweight service as it does not need to transfer the verified data to the TPA (audit-without-downloading property). Hence, this daemon can be easily appended into various cloud computing environments.

The system has 3 sub-systems:

- *User:* This module will implement the functionality of generating the key for encrypting the file and encrypting the hash generated for file blocks. Encrypted file is sent to cloud server and the signed hash content is sent to TPA.
- *Cloud Service Provider (CSP):* It stores the files in the cloud & responds to integrity challenge request from the TPA.
- *Third Party Auditor (TPA):* TPA verifies the integrity by posing challenge request to the cloud server & then checks the validity. It raises alert to the data owner if the integrity has failed.

In the auditing construction, the auditing protocol involves two-way communication: *challenge* and *proof.* During the confirmation auditing phase, the owner requires the auditor to check whether the owner's data is correctly stored on the server. The auditor conducts the confirmation auditing phase as:

1. The auditor runs the challenge algorithm to generate the challenge for all the data blocks in the data component and sends the challenge to the server.
2. After receiving the challenge from the auditor, the server runs the prove algorithm to generate the proof and sends it back to the auditor
3. When the auditor receives the proof from the server, it runs the verification algorithm to check the correctness of proof and extract the auditing result.

The auditor then sends the auditing result to the owner. If the result is true, the owner is convinced that its data is correctly stored on the server and it may choose to delete the local version of the data.

The audit scheme shows the communication between the entities in the fig 1.

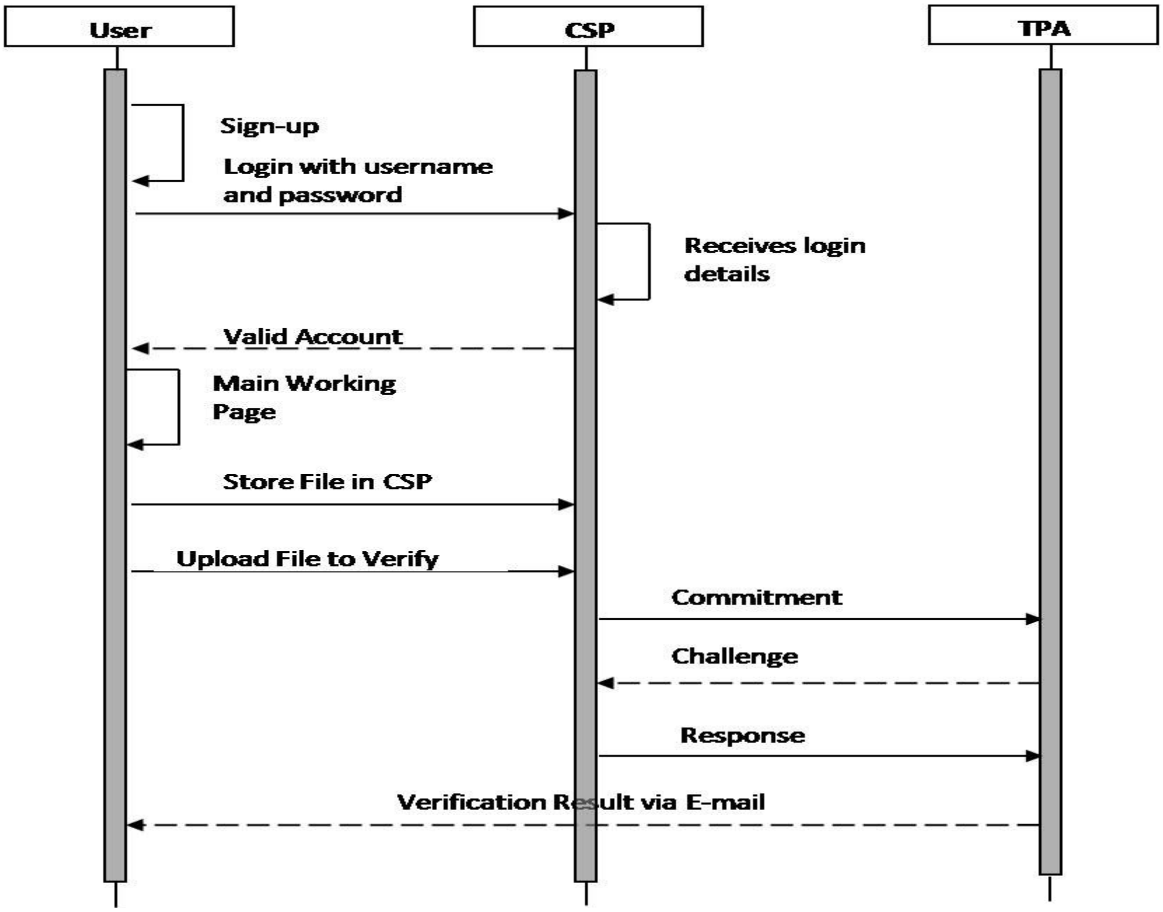


Fig. 1. Audit Scheme

2.1 FRAMEWORK OF INTERACTIVE AUDIT SCHEME

This audit scheme involves three algorithms:

1. Key Generation
2. Tag Generation
    i. Commitment
    ii. Challenge
    iii. Response
3. Verification Protocol

The algorithms used for the implementation are as follows:

1. KeyGen Algorithm

```
//Class for generating secret key and public key Bilinear pairing

public class KeyGen

{ void init() {
// creates bilinear pairing for the generation of secret and public key
  }
public static List<Integer>generatePrimes(int max)
{

   // finds all prime numbers up to max
}
public int [ ] getSecretKey()
{
   // it returns the secret key
}
int [ ] getPublicKey()
{
  // it returns the public key
}
}
```

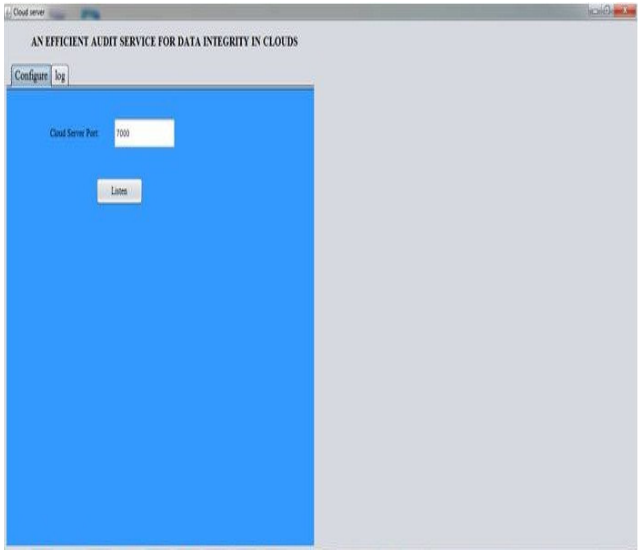
2. TagGen Algorithm

```
// Class to create unique pair of tags for each input file
public class TagGen {
void generateTags(
 {
//Takes filename, secret key and public key as input to generate cryptographic hash of file to convert it into array of 10 byte sectors and stores in a table
 }
}
```

3. Verifier Algorithm

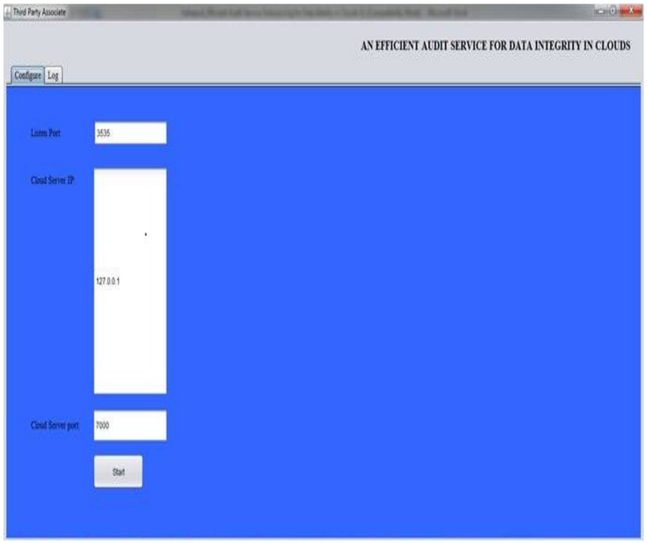
```
//Class to verify the integrity of file
public class Verifier {
void generateCommitment(int S,int [ ] publickeys)
 {
  // User sends request for verification of file by uploading the file.
  // CSP then commits to TPA
 }
Vector<Long>generateChallenge(int S, int [] publickeys,longapi,long ah)
{
  // Third Party Auditor (TPA) challenges the CSP
}
Vector<Long>generateChallengeres(Vector<Long>coeff,Stringfilename,int[] publickeys)
{
  // CSP provides the challenge response to TTP.
  // TPA sends the verification result to the user.

}
```

The snapshots of the implementation of TPA that can be used in any cloud computing environment are shown in the figures below:



Cloud Server Port 7000

Fig. 2. Cloud Server

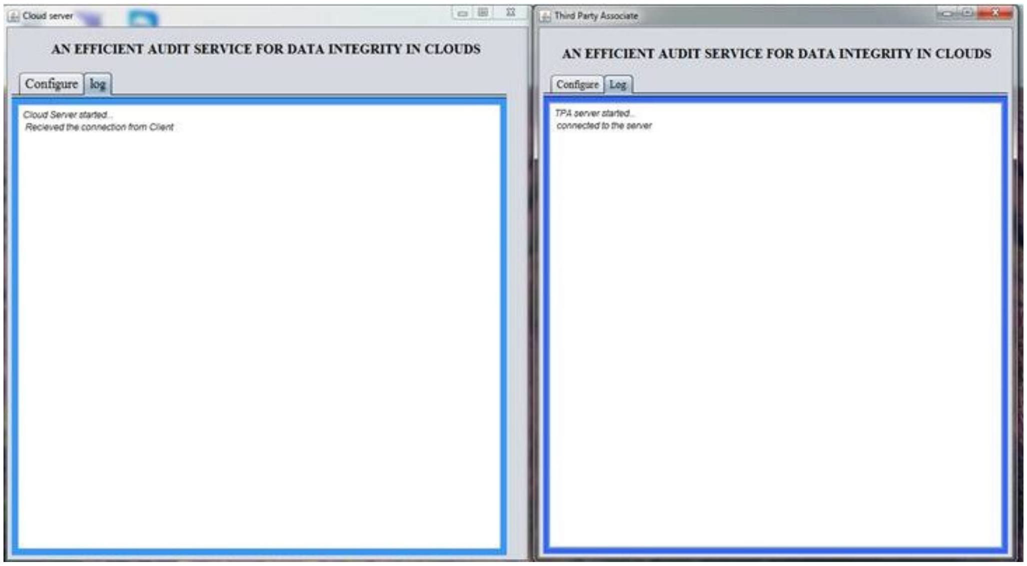


Listen Port 3535

Cloud Server IP 127.0.0.1

Cloud Server port 7000

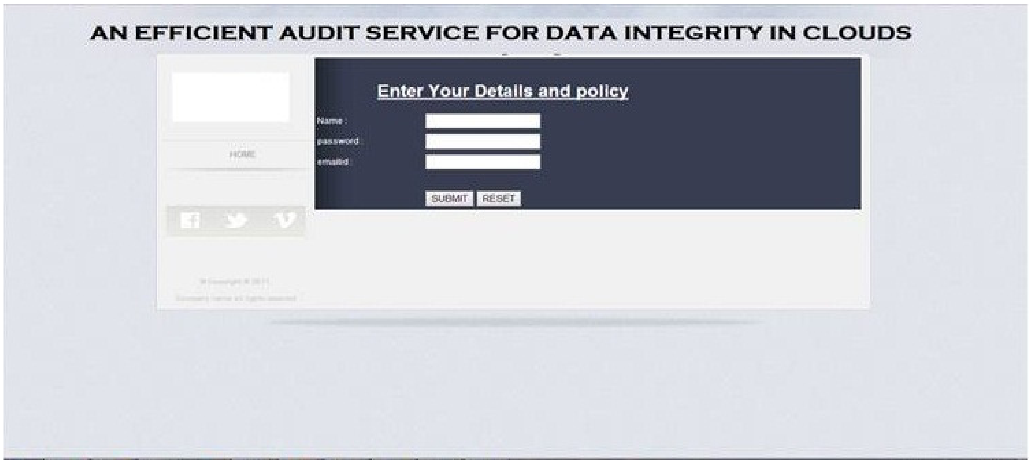
Fig. 3. TPA (Third Party Auditor) Server


Cloud server
AN EFFICIENT AUDIT SERVICE FOR DATA INTEGRITY IN CLOUDS
Configure
log
Third Party Associate
AN EFFICIENT AUDIT SERVICE FOR DATA INTEGRITY IN CLOUDS
Configure
Log

```
Cloud Server started...
 Recieved the connection from Client
```

```
TPA server started...
 connected to the server
```

Fig. 4. Cloud Server and TPA started



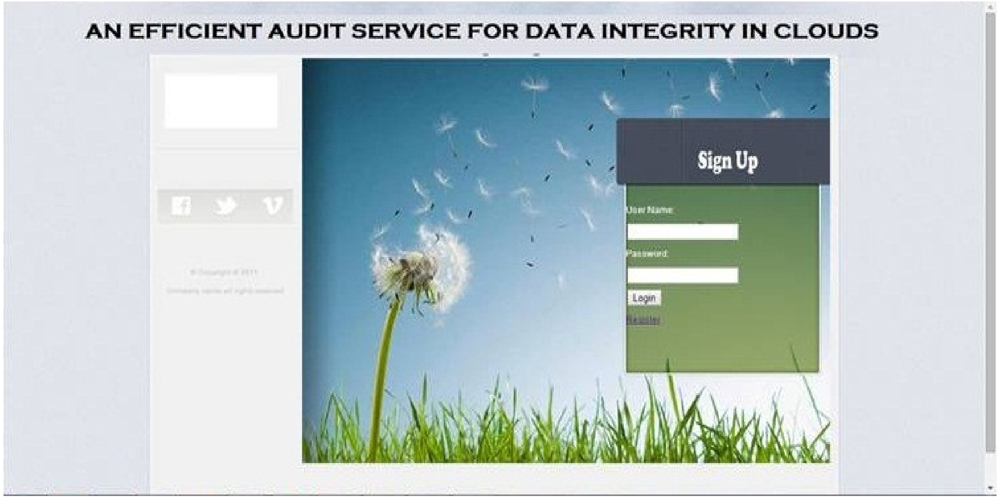
Enter Your Details and policy

Name :

password :

emailid :

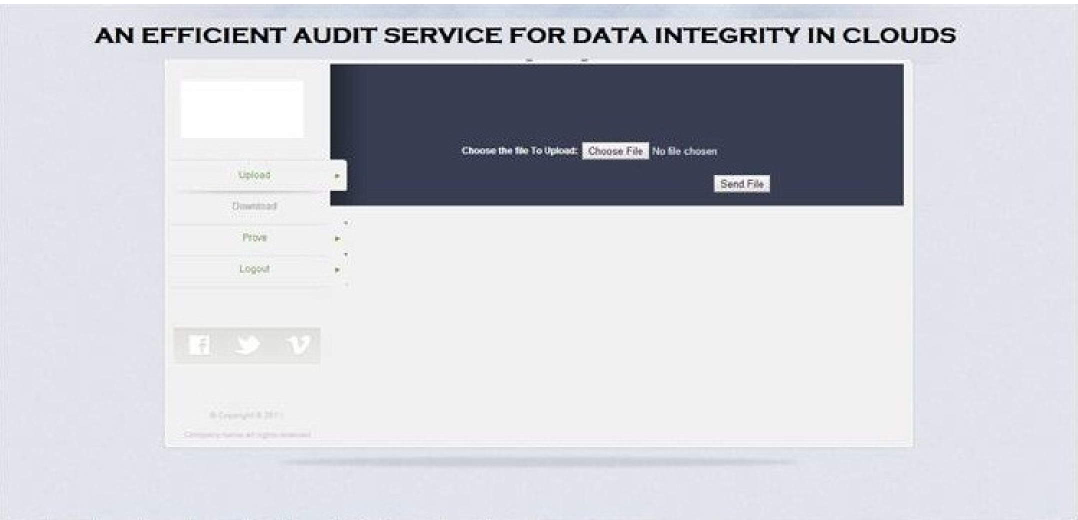
Fig. 5. Sign Up



User Name:

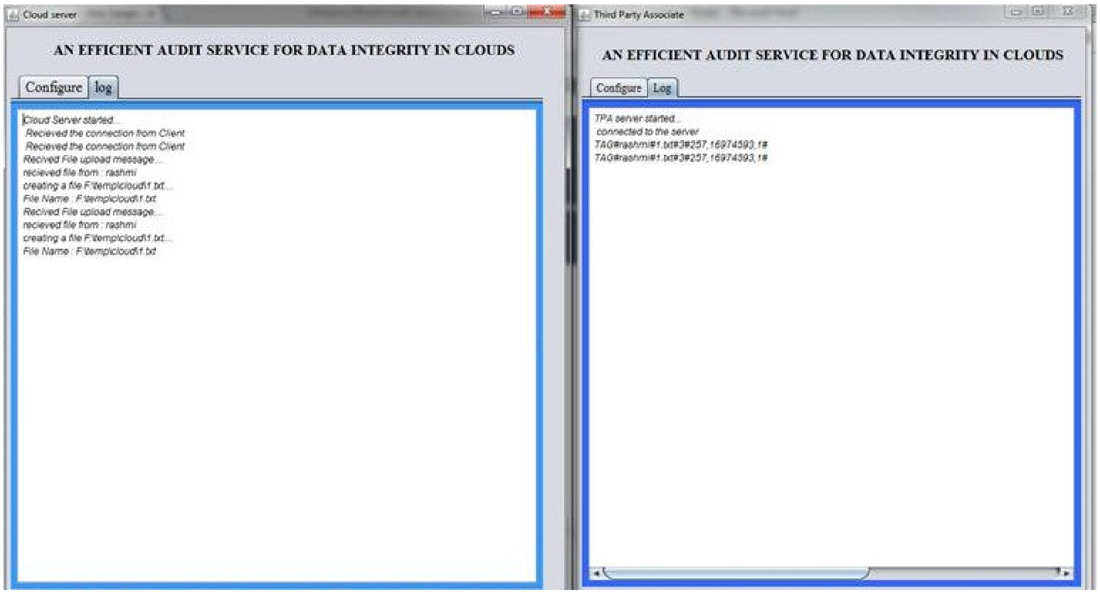
Password:

Fig. 6. Login



Choose the file To Upload: No file chosen

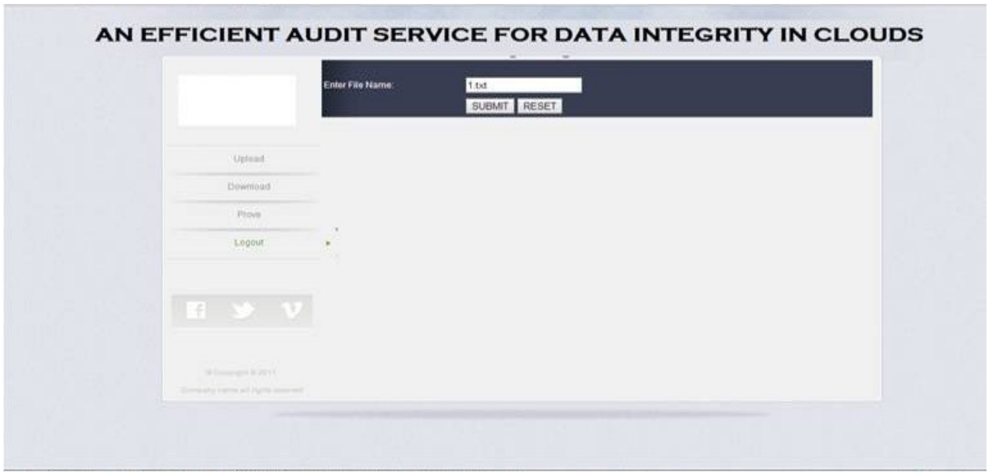
Fig. 7. File Upload


Cloud server
AN EFFICIENT AUDIT SERVICE FOR DATA INTEGRITY IN CLOUDS
Configure
log
Third Party Associate
AN EFFICIENT AUDIT SERVICE FOR DATA INTEGRITY IN CLOUDS
Configure
Log

```
Cloud Server started...
 Recieved the connection from Client
 Recieved the connection from Client
Recived File upload message...
recieved file from : rashmi
creating a file F:\temp\cloud\1.txt...
File Name : F:\temp\cloud\1.txt
Recived File upload message...
recieved file from : rashmi
creating a file F:\temp\cloud\1.txt...
File Name : F:\temp\cloud\1.txt
```

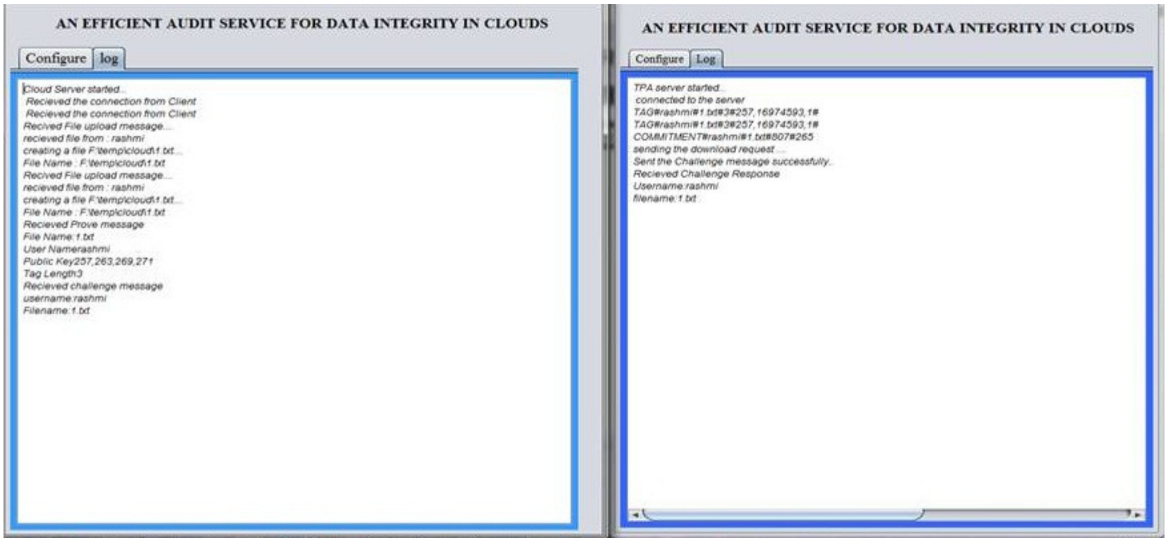
```
TPA server started...
 connected to the server
TAG#rashmi#1.txt#3#257,16974393,1#
TAG#rashmi#1.txt#3#257,16974393,1#
```

Fig. 8. Successful File Upload



Enter File Name: 1.txt

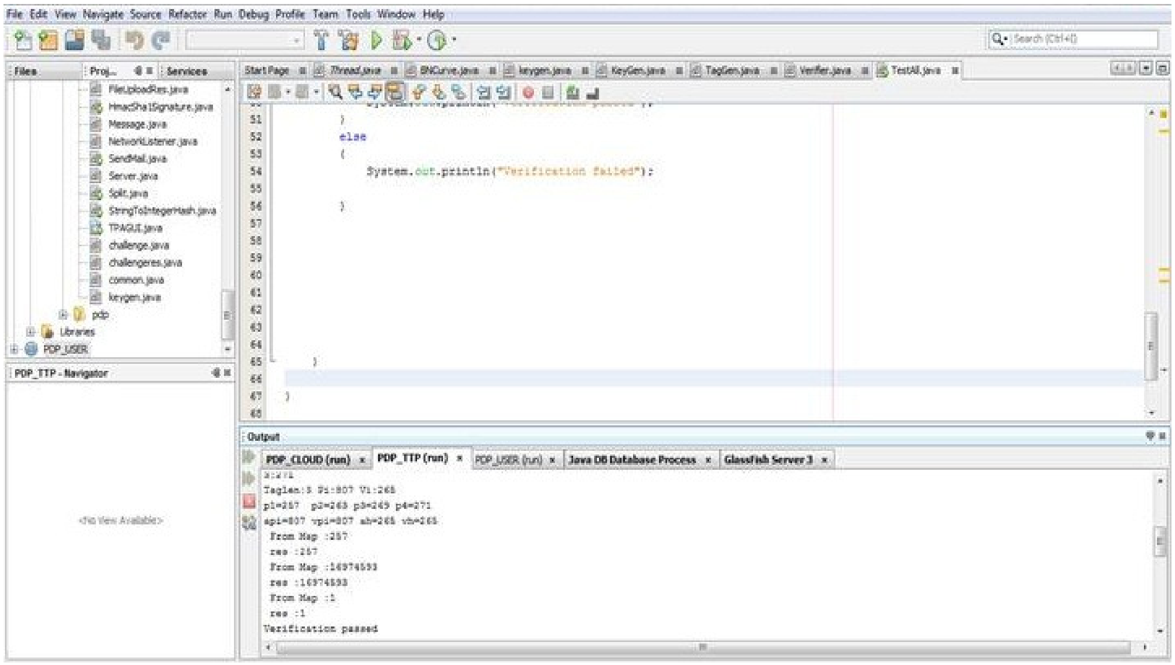
Fig. 9. Submit File for Verification


AN EFFICIENT AUDIT SERVICE FOR DATA INTEGRITY IN CLOUDS
Configure
log
AN EFFICIENT AUDIT SERVICE FOR DATA INTEGRITY IN CLOUDS
Configure
Log

```
Cloud Server started...
 Recieved the connection from Client
 Recieved the connection from Client
Recived File upload message...
recieved file from : rashmi
creating a file F:\temp\cloud\1.txt...
File Name : F:\temp\cloud\1.txt
Recived File upload message...
recieved file from : rashmi
creating a file F:\temp\cloud\1.txt...
File Name : F:\temp\cloud\1.txt
Recieved Prove message
File Name:1.txt
User Namerashmi
Public Key257,263,269,271
Tag Length3
Recieved challenge message
username:rashmi
Filename:1.txt
```

```
TPA server started...
 connected to the server
TAG#rashmi#1.txt#3#257,16974593,1#
TAG#rashmi#1.txt#3#257,16974593,1#
COMMITMENT#rashmi#1.txt#807#265
sending the download request ....
Sent the Challenge message successfully..
Recieved Challenge Response
Username:rashmi
filename:1.txt
```

Fig. 10. Challenge/Response dialogue between CSP and TPA



```
            }
            else
            {
                System.out.println("Verification failed");

            }








        }

    }
```

```
[illegible]:271
Taglen:3 Pi:807 Vi:265
p1=257  p2=263 p3=269 p4=271
api=807 vpi=807 ah=265 vh=265
 From Map :257
 res :257
 From Map :16974593
 res :16974593
 From Map :1
 res :1
Verification passed
```

Fig. 11. Verification Result sent to User via E-mail

3. CONCLUSIONS

The audit scheme addresses the construction of an audit service to verify the integrity of data in clouds. It proposes an interactive audit scheme to implement the audit service based on a third party auditor. In this audit service, the third party auditor, known as an agent of data owners, can issue a periodic verification to monitor any change in data. This implementation requires only maintaining the security of the third party auditor and deploying a lightweight daemon to execute the verification of data. Hence, this technology can be easily adopted in a cloud computing environment to replace the traditional Hash-based solution. This approach greatly reduces the workload on the storage servers, while still achieves the detection of servers' misbehaviour with a high probability.The auditing protocol can also be implemented to support the batch auditing for multiple owners and multiple clouds.

REFERENCES

[1] Buyya, C.S.Y.; Venugopal, S.; Broberg, J. and Brandic, I. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems, 25:599616,2009.

[2] Armbrust, M; Fox, A.; Griffith, R.; Joseph, A.D.; Katz, R.H.; Konwinski, A.; Lee, G.; Patterson, D.A.;

Rabkin, A.; Stoica, I; Zaharia, M., 2010. A view of cloud computing. Commun. ACM 53 (4), 50– 58.3.

[3] Tchifilionova, V., 2011. Security and privacy implications of cloud computing c lost in the cloud. In: Camenisch, J.; Kisimov, V.; Dubovitskaya, M. (Eds.), Open Research Problems in Network Security. Vol. 6555 of Lecture Notes in Computer Science. Springer, Berlin/Heidelberg, pp. 149–158.

[4] Ko, R.K.L.; Lee, B.S.; Pearson, S. 2011. Towards achieving accountability, auditability and trust in cloud computing. In: Abraham, A., Mauri, J.L., Buford, J.F., Suzuki, J., Thampi, S.M. (Eds.), Advances in Computing and Communications. Vol. 193 of Communications in Computer and Information Science. Springer, Berlin/Heidelberg, pp. 432–444.

[5] Yavuz, A.A.; Ning, P. 2009. Baf: An efficient publicly verifiable secure audit logging scheme for distributed systems. In: ACSAC, pp. 219–228.

[6] Hsiao, H.-C.; Lin, Y.-H.; Studer, A.; Studer, C.; Wang, K.-H.; Kikuchi, H.; Perrig, A.; Sun, H.-M.; Yang, B.-Y.; 2009. A study of user-friendly hash comparison schemes. In: ACSAC, pp. 105–114.

[7] Yumerefendi, A.R.; Chase, J.S.; 2007. Strong accountability for network storage. ACM Trans. Storage (TOS) 3 (3).

[8] Ateniese, G.; Burns, R.C.; Curtmola, R.; Herring, J.; Kissner, L.; Peterson, Z.N.J.; Song, D.X.; 2007. Provable data possession at untrusted stores. In: Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, pp. 598–609.

[9] Juels Jr.; A.; Kaliski; B.S.; 2007. Pors: proofs of retrievability for large files. In: Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, pp. 584–597.

[10] Ateniese, G.; Pietro, R.D.; Mancini, L.V.; Tsudik, G.; 2008. Scalable and efficient provable data possession. In: Proceedings of the 4th International Conference on Security and Privacy in Communication Networks, SecureComm, pp. 1–10

[11] Wang, Q.; Wang, C.; Li, J.; Ren, K.; Lou, W. 2009. Enabling public verifiability and data dynamics for storage security in cloud computing. In: Proceedings of the 14th European Symposium on Research in Computer Security, ESORICS 2009, pp. 355–370.

[12] Zhua Y.; Hu H.; Ahn G-J.; Yau S.S. 2012. Efficient audit service outsourcing for data integrity in clouds, The Journal of Systems and Software 85 (2012) 1083–1095.