Simulation of Secure Data Storage Using Third Party Auditor (TPA)

GRADUATE PROJECT

Submitted to the Faculty of
The School of Engineering & Computing Sciences
Texas A&M University-Corpus Christi
Corpus Christi, TX

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Science

by

Naga Sandeep Thatipally
Spring 2014

Committee Members

Dr. Mario Garcia
Committee Chairperson

Dr. Long-zhuang Li
Committee Member

Dr. David Thomas
Committee Member
ABSTRACT

Cloud computing enables the sharing of services over a network. Many Internet users store their data in the cloud. Despite cyberspace users having no physical retention of huge amounts of their outsourced data, data integrity security in cloud computing is a difficult task, especially for users who have only partial computing capabilities and resources. Cyberspace users should be able to use network storage as if it is in the local system, without worrying about the verification of integrity. Hence, allowing public analysis for cloud storage is important so that users can use an alternative to a third-party auditor (TPA) to check the integrity of the outsourced data. To securely initiate the services of an efficient TPA, the auditing process must not introduce new vulnerabilities to users’ data privacy or further burden users’ online load. In this research, data storage in the cloud system that is secure and also supports privacy is recommended. The implemented system performs audits for multiple users concurrently and in an efficient manner.
# TABLE OF CONTENTS

Abstract .......................................................................................................................... ii

Table of Contents ........................................................................................................... iii

List of Figures ................................................................................................................ vi

List of Tables .................................................................................................................. vii

1. Background and Rationale ....................................................................................... 1
   1.1 Cloud Computing ................................................................................................. 1
       1.1.1 Types of Cloud Computing ........................................................................ 2
       1.1.2 Services of Cloud Computing ..................................................................... 3
       1.1.3 Advantages of Cloud Computing ............................................................... 4
   1.2 Motivation ............................................................................................................. 6

2. Existing System and Proposed System ..................................................................... 7
   2.1 Existing System .................................................................................................... 7
   2.2 Proposed System .................................................................................................. 8
   2.3 Current Works ..................................................................................................... 9

3. Architecture and Design .......................................................................................... 10
   3.1 Architecture ........................................................................................................ 10
   3.2 Design ................................................................................................................ 11
   3.3 System Design .................................................................................................... 12
       3.3.1 Use Case Diagram ....................................................................................... 12
       3.3.2 Dataflow Diagram ....................................................................................... 13
       3.3.3 Sequence Diagram ...................................................................................... 16
       3.3.4 Activity Diagram ......................................................................................... 17
3.4 Modules.........................................................................................................................19
3.5 Software Requirements.................................................................................................20

4. Implementation and Testing ..........................................................................................21

4.1 Implementation of the Design .....................................................................................21
  4.1.1 Layout .......................................................................................................................21
  4.1.2 Admin .......................................................................................................................21
  4.1.3 Auditor ......................................................................................................................23
  4.1.4 User ..........................................................................................................................24
    4.1.4.1 User Signup .........................................................................................................24
    4.1.4.2 User Login ...........................................................................................................25

4.2 Expected Results ..........................................................................................................28

4.3 Testing ..........................................................................................................................28
  4.3.1 Test Cases ................................................................................................................28
    4.3.1.1 Sign-up Phase .......................................................................................................28
    4.3.1.2 Login Phase .........................................................................................................30
    4.3.1.3 File Downloading ...............................................................................................32
    4.3.1.3 Verifying Correctness of Data ..............................................................................34

5. Conclusion and Future Work ........................................................................................36

Bibliography and References ............................................................................................37
LIST OF FIGURES

Figure 1 Traditional Stack vs Virtualized Stack ......................................................1
Figure 2 Types of Cloud Computing ........................................................................2
Figure 3 Cloud Computing Services ........................................................................3
Figure 4 Advantages of Cloud Computing ...............................................................5
Figure 5 Architecture .............................................................................................10
Figure 6 Design Procedure ....................................................................................12
Figure 7 Proposed System’s Use-Case Diagram .....................................................13
Figure 8 Administrator Dataflow Diagram .............................................................14
Figure 9 User’s Dataflow Diagram .........................................................................15
Figure 10 Auditor's Dataflow Diagram ...................................................................16
Figure 11 Proposed System’s Sequence Diagram ..................................................17
Figure 12 Proposed System’s Activity Diagram .....................................................18
Figure 13 Layout of the Home Screen ....................................................................21
Figure 14 Login Page of Admin ............................................................................22
Figure 15 Admin Generating GenProof ..................................................................22
Figure 16 Login Page of Auditor ..........................................................................23
Figure 17 Auditor Sending File Details to Admin ....................................................24
Figure 18 Users Signup Page ...............................................................................24
Figure 19 Login Page of User ...............................................................................25
Figure 20 Public Auditability Key Request ..............................................................26
Figure 21 Users Uploading the File ........................................................................26
Figure 22 Packet Sending ......................................................................................27
Figure 23 Providing Secret Key to the File .................................................................27
Figure 24 Successful Signup by the User ..................................................................29
Figure 25 Unsuccessful Signup .................................................................................30
Figure 26 Fraudulent Login .......................................................................................31
Figure 27 Fraudulent Login Invalid Public Key ..........................................................32
Figure 28 Server Sending Email to the User When the Server Logs in .................32
Figure 29 Successful File Download .........................................................................33
Figure 30 File Download Not Allowed .......................................................................34
Figure 31 Hash Function Verifying Correctness of the Data ..................................35
LIST OF TABLES

Table 1 Software Requirements ........................................................................................................... 20
1. BACKGROUND AND RATIONALE

Cloud storage has become a popular class for online service to achieve backup and sharing data. It helps in synchronizing multiple devices and is envisioned as the future of primary data storage because it offers a long list of advantages. With cloud storage space, users store their data distantly and on demand can enjoy high-quality applications. Cloud computer offers a shared collection of computing resources for services, vastly reducing the burden of data storage on local systems and system maintenance.

1.1 Cloud Computing

Cloud computing is a communication network that involves many computers connected together. This network has the ability to allow users to perform an application or a program at the same time in all the connected systems. The difference in working methods, of traditional stack and virtualized stack computing are readily apparent (see figure 1).

![Traditional Stack vs Virtualized Stack](image)

Figure 1 Traditional stack vs Virtualized stack
1.1.1 Types of Cloud Computing

Cloud is an approach for building IT services rather than a technology; the cloud is an approach that tackles the power of servers. It also makes multiple virtual machines by dividing single servers. For implementing cloud technology, we primarily have three models (see figure 2).

- **Public Cloud**: In the public cloud computing services, like applications and storage are made available over the Internet for general use. They are offered as purchasing models on a pay-per-usage basis.

- **Private Cloud**: In private cloud computing, the data center will be operated within the firewall. They are managed privately.

- **Hybrid Cloud**: The combination of both public and private cloud is hybrid cloud.

![Figure 2 Types of Cloud Computing](image-url)
1.1.2 Services of Cloud Computing

After the establishment of the cloud and depending on users’ needs, deployed services can follow the requirements of users’ business models. There are many types of cloud computing services (see figure 3).

![Diagram of Cloud Computing Services]

Figure 3 Cloud Computing Services

Services available in the cloud include the following:

- **Software as a Service (SaaS):** this model is referred as “on-demand software.” With SaaS, the application will be hosted as a service via the Internet.

- **Infrastructure as Service (IaaS):** consumers of IaaS do not control the cloud infrastructure themselves; rather they manage and control the system in terms of network connectivity, operating systems, storage, and applications.

- **Platform as a Service (PaaS):** users of PaaS deploy their own software in the cloud by purchasing access to the platform. Users can also run their own applications in the cloud.
• Storage as a Service (STaaS): with STaaS, the individual or smaller company rents space from the storage infrastructure of large companies. Offsite backup challenges are often solved using this model.

• Security as Service (SECaaS): large providers integrate their security services on the subscription basis into a corporate infrastructure.

• Data as Service (DaaS): provides data on demand in a manner similar to how SaaS provides software on demand.

• Test Environment as a Service (TEaaS): also known as “on-demand test environment,” helps in providing test environment when needed.

• Desktop as a Service (DaaS): a virtual desktop or desktop virtualization. To the third-party service provider, DaaS offers outsourcing of virtual desktop infrastructure (VDI).

• API as a Service (APIaaS): creating and hosting of application programming interfaces (APIs) can be enabled by the user in this model.

1.1.3 Advantages of Cloud Computing

Nowadays, most business applications use the cloud, representing a large shift from traditional software applications. There are many advantages of cloud computing, several of which are shown in figure 4.
Advantages of cloud computing have made it a popular option. Among these are:

- **Cost effectiveness**: the cost of hardware and software can be saved; because most issues are solved in the back end, there are also cost savings for tech support.
- **Accessibility**: if Internet service is available, data from the cloud can be accessed using an Internet-enabled device.
- **Storage**: there is a vast area for data storage in the cloud, obviating concern for storage needs.
- **Backup**: in the event of any breakdown of the system, backup data are available in the cloud, adding to savings of backup data costs.
- **Software integration**: software integration costs are avoided because individual purchases and installations are avoided. Software is installed, stored, and accessible in the cloud.
- **Flexibility**: cloud operation and adjustment is flexible and can be done in less time.
1.2 Motivation

Cloud computing is gaining more attention recently from both academic and commercial environments. Many market research firms are recognizing the potential of cloud computing and trying to make use of the cloud to its greatest extent. Accuracy and security of data are fundamental concerns. Security in the cloud network is achieved by inscribing the data block before sending the data to the cyber network.

The objective of this research is to implement an application that helps store the data in a secure way by allowing the auditor to check the integrity of the file; on download, there should not be any loss of data.
2. EXISTING SYSTEM AND PROPOSED SYSTEM

2.1 Existing System

There are many existing schemas that aim to provide integrity verification for data storage systems. Most schemas have not addressed the problems related to public auditability. For cloud computing, integrating security and efficient design remains a challenge. The main drawback of the existing system is the need to support block insertion.

A redundant array of cloud storage (RACS) is proposed by authors in [Abu-Libdeh 2010], RACS is a cloud storage proxy. RACS’s use of optimal erasure codes are used to disperse data across multiple cloud providers. The major concern of RACS is to check the availability of data, but it is not concerned with data security and improving privacy. The key-value store (KVS) is proposed by authors in [Basescu 2012], KVS was developed by commercial cloud storage services to access data in the cloud with multiple reader and multiple writer settings. KVS does not do a great deal to address the problem of data security and data privacy.

High-Availability and Integrity Layer (HAIL) is proposed by authors in [Bowers 2009]. It deals with the concept of distributed cloud storage by providing the ability to unify the use of file redundancy across independent cloud providers. The author’s main concern was to provide integrity and consistency, but the author has not given much priority to data privacy and file authenticity. CS2 is proposed by authors in [Kamara 2011], CS2 is a semantic cloud storage system, and it integrates search authenticators with searchable encryption. CS2 mainly concentrates on confidentiality and integrity but is not consistent.
2.2 Proposed System

This research proposes a technique that supports the concept of block insertion. For supporting that objective in a secure manner, we use three components: client, cloud storage server, and TPA server.

The Client can upload and send the file to the admin (who is the in charge of the cloud storage server) by providing the three ways secret key to the file, and once the file is stored, after the verification has been done by the auditor, it is then ready for download. The file can be downloaded by all the authorized users who know the secret keys. TPA is the third-party auditor. He or she sends the request to the admin asking for the key, which is generated by running the GenProof algorithm as proof for the data storage in the server. Once the auditor receives the key, he or she then verifies and sends the acknowledgement to the user on the file status. The cloud storage server is managed by the admin. Here, the admin takes care of the data storage in the cloud. He or she sends the key generated by the GenProof on the file to the auditor on request as proof that the correct file is stored in the cloud. In this project, we are using the local database as the storage server of size 500GB (see figure 5), as both perform the same function of storing the data, but when it comes to the cloud; it is much larger than the local database is.
2.3 Current Research

Many approaches are being used for improving the security of stored data in the cloud. A few approaches are mentioned below.

The paper by [Rani 2014] gives a complete picture of various challenges regarding the privacy in the cloud storage space, and also brings in a new scheme based on encryption that is used to handle different challenges. Key Insertion and Splay Tree encryption is the new approach proposed that is based on an encryption algorithm.

The paper by [Lunawat 2014] presents the architecture for outsourcing the data, which is secure, and the calculation for the application where a large amount of work is required. This approach also uses an RSA algorithm for high security with the use of Garbled Circuits.

In [Chiwande 2014], a large-scale development, a load balancing algorithm is used to manage the problem of load rebalancing. This is also used in vibrant and distributed file systems in the cloud and balances the loads of various nodes to reduce the demanded cost to the maximum extent, which takes advantage of the locality of the physical network and node heterogeneity.

The paper [Jaekyung 2014] proposed a virtual or practical framework for real media cloud services. We also designed an isolated module based on Virtual Media Service to solve the issues related to security in the media services that are in an environment of cloud computing.

In [Poorvadevi 2013], while trying to present the efficient user identification technique with the help of mining analysis, the fuzzy-based user authentication module is used to provide satisfactory security for the access of cloud service, which in turn also reduces the difficulty that is involved in the exchanges of key.
3. ARCHITECTURE AND DESIGN

3.1 Architecture

The architecture design is shown in figure 5. Notice the flow of data between the client (user), cloud storage server (CCS), and the TPA. First, the user sends the data to be stored in cloud to the server, and then the TPA checks the data integrity. After that data blocks are stored in the cloud.

Figure 5 Architecture

The User can upload and send the file to the administrator by providing the secret key to the file, the file can be downloaded by all the authorized users who know the secret keys. TPA is the third-party auditor. He or she sends the request to the administrator asking for the key, as proof for the data storage in the server. Once the auditor receives the key, he or she then verifies and sends the acknowledgement to the user on the file status. The cloud storage server is managed by the admin. Here, the admin takes care of the data storage in the cloud. He or she sends the key generated by the GenProof on the file to the auditor on request as proof that the correct file is stored in the cloud.
3.2 Design

Along with the user, the TPA can also allow the external party to verify remotely stored data. Four algorithms are used in the TPA schema. They are as follows

- **KeyGen**: a user runs this key generation algorithm to set up the schema. KeyGen is generated using RSA algorithm, key length is set such that it does not exceed 5.
- **SigGen**: a user generates verification metadata using this algorithm to obtain information useful for auditing. SigGen creates the blocks by converting the given password to binary and add string value to it to create the metadata.
- **GenProof**: a cloud server runs this algorithm to generate the proof of correctness for data storage. Here correctness is to check whether the correct is file uploaded or not. GenProof is the combination of the string and metadata.
- **VerifyProof**: audits the cloud server, by comparing the file id, file name, user name and file size of the file.

The TPA can be run in two phases. The two phases are shown in figure 6. They are the set-up phase and the audit phase.

- **Setup Phase**: By executing KeyGen, user initializes public and secret parameters. Then the data file F is pre-processed by using SigGen to generate verification metadata. User deletes the local copy of the file, once the metadata and the data file F are stored in the cloud. Data file F can be altered by the user at the time of pre-processing by including additional metadata for storing it in the server.
- **Audit Phase**: An audit message or a challenge is issued by the user to cloud server to check whether the data file F is retained and stored correctly by the cloud server. By executing GenProof the cloud server will derive a response message through the function
of stored data file F and verification metadata. Then VerifyProof is used by the TPA to verify the response.

![Diagram of Setup and Audit processes]

**Figure 6 Design Procedure**

### 3.3 System Design

This section will discuss about the proposed application design with the help of various UML diagrams.

#### 3.3.1 Use Case Diagram

Graphical representation of the system’s functionality with regard to actors and their aims is the purpose of use case diagram. Figure 7 shows the use case diagram for the proposed system.
3.3.2 Dataflow Diagram

The dataflow diagram explains about the flow of data in a system. With the help of figures 8, 9 and 10 the dataflow of the proposed project is presented.
As is shown in figure 8, the details of the data flow in the administrator (CSP – Cloud Service Provider) phase are shown graphically. If the admin is authorized, he or she would audit the file and send the alert message along with the key to the auditor. If he or she is not authorized, he or she cannot enter the system.

![Administrator Dataflow Diagram](image)

Figure 8 Administrator Dataflow Diagram

As is shown in figure 9, the details of the data flow in the user phase are shown graphically. If the user is authorized, he or she will upload the file and wait for the audit status of the file. If the file has been uploaded successfully, then with the help of a one-time password, which will be received via e-mail, he or she can download the file when needed. If he or she is not authorized, the entrance to the system will be denied.
As is shown in Figure 10, the details of the data flow in the auditor phase are shown graphically. If the auditor is authorized, he or she enters the system and can view all user data and sends the data to admin to wait for the alert message from admin. If the key is correct, he or she accepts the data and intimates the user that the data has been successfully stored in the cloud and is ready for download. If the key is not correct, the auditor rejects the data. If he is not authorized, he cannot enter the system.
3.3.3 Sequence Diagram

Display of the operations performed with each other and the order they are displayed in is the purpose of sequence diagram. Figure 11 shows the sequence diagram for the proposed system.
3.3.4 Activity Diagram

The activity diagram explains the work flow of stepwise activities and actions in a system. The activity diagram concerning the proposed project is shown in figure 12.
Figure 12 Proposed System’s Activity Diagram
3.4 Modules

Six modules perform operations within the cloud:

- Public audit ability to assure storage correctness,
- Dynamic data operation support,
- Blockless verification,
- Dynamic data operation with integrity assurance, and
- Data modification.
- Batch auditing for multi-client data

Public audit ability allows the users to ensure their data have been stored correctly. It also does verification on the correctness of the on-demand stored data. Dynamic data operation support executes block-level operations on the stored data simultaneously, maintaining the correctness of the data on the same level. This operation should be capable to support the seamless integration of public auditability and dynamic data operation support. The blockless verifier should not be able to challenge blocked files during the verification process. Dynamic data operation with integrity assurance is engaged after the data modification and data insertion operations and data deletion operations in the cloud storage have been successfully executed. File F and the signature have already been generated and properly stored on the server. The technique used is called Root Metadata R, which is signed by the client and is stored in the cloud server. The main functionality of dynamic data operation with integrity assurance is that anyone the with client’s public key can question the data storage correctness.

Data modification operations in cloud data storage are used frequently. Data modification entails replacing new blocks with specified blocks. To start the process, corresponding signatures are generated by the client on the new block. R’ is the new root meta data. The client signs R’ to
get signk(H(R')) and the server is updated. Finally, the default integrity verification protocol is executed by the client. A duplicate file is generated if the output is true after deleting signk(H(R')). Batch auditing for multi-client data are used to handle multiple verification sessions i.e. from K clients if we have K distinct data files then there will be K signatures, instead by aggregating all the signatures into one single shot one and verifying it will be more advantageous. To achieve this goal, this scheme is extended to allow for provable data verification and updates in a multi-client system. Signature scheme is made to allow the creation of signatures on arbitrary distinct messages. It also supports the aggregation of multiple signatures by distinct signers on distinct messages into a single short signature, and thus helps in reducing the communication cost while providing efficient verification for the authenticity of all messages.

3.5 Software Requirements

The software technologies used for the development of application are listed in Table 1 as follows:

<table>
<thead>
<tr>
<th>Category</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Windows 7</td>
</tr>
<tr>
<td>Application Server</td>
<td>Tomcat 6.0</td>
</tr>
<tr>
<td>Design Pattern</td>
<td>J2EE</td>
</tr>
<tr>
<td>Scripts</td>
<td>JSP</td>
</tr>
<tr>
<td>Database</td>
<td>MySQL</td>
</tr>
</tbody>
</table>
4. IMPLEMENTATION AND TESTING

4.1 Implementation of the Design

4.1.1 Layout

Layouts are the user interface components that user can view and use to interact with the application. The Layouts that are used in this application are as follows:

Figure 13 shows the layout, which is the home screen of the application. Top of the home page contains the navigation buttons to the Admin, TPA (auditor), and user and sign up pages, which takes the user to his or her specific screens on a single click.

![Figure 13 Layout of the Home Screen](image)

4.1.2 Admin

Clicking on Admin from the home screen allows the user to navigate to figure 14, i.e., the login page of the admin. After successful login, it navigates to admin’s homepage. Here, admin checks for pending key request messages from the auditor. In case of any pending requests it generates key using GenProof for the file is shown in figure 15 and sends the key to the auditor.
Figure 14 Login Page of Admin

Figure 15 Admin Generating GenProof
4.1.3 Auditor

Clicking on TPA (auditor) from the home screen allows the user to navigate to figure 16, i.e., the login page of the auditor. After successful login, it navigates to the auditor’s homepage.

![Figure 16 Login Page of Auditor](image)

After logging in, admin checks for alert messages. If there any, he or she sends them to admin (see figure 17) for verification regarding the status of the file and waits for the message with the key from admin. On receiving the message, he verifies and sends the acknowledgement to the user.
4.1.4 User

4.1.4.1 User Sign Up

Clicking on SIGNUP from the home screen allows the user to navigate to figure 18, i.e., the sign-up page for the user.

After successful signup, the user can login and can perform operations on the file.
4.1.4.2 User Login

Clicking on USER from the home screen navigates the user to figure 19, i.e., the login page of the user. After entering the correct details, it navigates to a new page asking the user to enter the public audiatability key, which is sent to the user mail ID (see figure 20) that he or she enters at the time of signup.

After entering the correct key, the user will be navigated to the user’s homepage.

![Figure 19 Login Page of User](image)

Figure 19 Login Page of User
Now, the user can upload the file from his local machine by giving a file name to it as shown in figure 21. The file ID will be generated automatically and will be a unique one for each file upload and cannot be changed by the user.
Once the file is uploaded on clicking packet sending figure 22 will be displayed. By clicking on CLICK as shown in figure 23 will be displayed, there the user needs to enter the secret key for providing security for the file and then sends the file for storage.

Figure 22 Packet Sending

Figure 23 Providing Secret Key to the File
Once the file is stored in the server, the auditor sends the acknowledgement to the client, and the client can view the file upon clicking download and can download the file.

4.2 Expected Results

The results contain data in cloud, which are reliable i.e., the resulted search item contains knowledgeable but not a fake link or non resource link to download or view the data.

4.3 Testing

4.3.1 Test Cases

The proposed project was tested for following test cases:

- Sign-up phase
  - Successful signup
  - Unsuccessful signup
- Login phase
  - Genuine user
  - Fraudulent user
- File downloading
  - Successful download
  - Unsuccessful download

4.3.1.1 Sign-up Phase

In the sign-up phase, users would be able to sign themselves up for uploading their files using the application. Two scenarios for a successful registration and unsuccessful registration are presented in the following screenshots.
Successful signup

Figure 24 Successful Sign Up by the User

Figure 324 shows us that, the user entered the credentials correctly and thus is successful. In figure 24 signup was successful.

Unsuccessful signup

As is shown in figure 25, the user has to take the following things for the sign-up process

- Name should be more than six (6) characters.
- User ID should be unique; i.e., the current user should be the first person using that ID.
- The password should have more than six (6) characters.
- The mobile number should be in numbers and should be only ten (10) digits.
- E-mail ID should be currently valid.

However, in figure 25, the user did not enter a valid mobile number with 10 numbers; hence, the registration was unsuccessful.
4.3.1.2 Login Phase

In this phase, the user needs to enter the user ID and the password. If either is not correct, the user cannot enter. If they are correct, the verification of the registered user will be completed by asking the user to enter the public auditability key.

Genuine user

The user once enters the proper credentials and then the correct public auditability key; he or she will be directed to the user’s home page.
As it is seen in figure 26, the user has entered the wrong credentials, so he is not able to login. But in case of figure 27 the user entered the correct credentials but has not entered the correct public key which will be sent to the actual user’s mail (see figure 28). As such, the user trying to login might not be a genuine user.
Sending email when user logs in

![Image of email]

**Figure 28 Server Sending Email to the User When the ser Logins**

### 4.3.1.3 File Downloading

In this section, the testing was done to verify whether the file downloaded by the user is the same file he or she wanted to download and has the correct data.
Successful download

User to download the file first the secret keys should be entered correctly that are used at the time the packet is sent. If the user enters the correct keys as shown in figure 29, the file will be downloaded.

Unsuccessful download

As can be seen in figure 30, the file download is not allowed, as the user has entered the wrong secret key, and it displays the message asking the user to enter the correct key for downloading the file.
4.3.1.3 Verifying Correctness of Data

Here our main focus is to verify the correctness of the data, i.e. to check whether the file downloaded by the user is same as the file which is uploaded. To verify it we use hashing functionality on the file before uploading and after downloading. If the hashing value is same for both the files, then the correctness of the data is proved. Here md5 hashing is used to check the hashing of the document.

As we can see in figure 31 two files with hashing values are displayed along with the tool which is used for getting the hashing values. One file has the data before uploading and other file has the data after downloading. As both the hashing values are same we can state that the file downloaded contains the same data as the file upload.
Implementation of an Application Using Third Party Auditor (TPA) for Secure Cloud Storage

ABSTRACT

Cloud computing enables the sharing of services over a network. Many Internet users store their data, and despite the convenience of cloud services, users are concerned about the security of their data. Cloud computing enables the sharing of services over a network. Many Internet users store their data, and despite the convenience of cloud services, users are concerned about the security of their data. To seek services of an efficient TPA, the auditing process must not introduce new vulnerabilities to users' data. To seek services of an efficient TPA, the auditing process must not introduce new vulnerabilities to users' data. Further burden users' online load. In this research, data storage is in the cloud system that is more secure with administrator(s) having full control and data access. The implemented system performs audits for multiple users in a more efficient manner.

Figure 31 Hash Function Verifying Correctness of the Data
5. CONCLUSION AND FUTURE WORK

We have proposed a public auditing system that preserves privacy for data storage space security in cloud computing. We also took measures such that the TPA would have not learned anything about the data stored on the server during the effective auditing process, thereby eliminating costly auditing tasks and workloads on the user. Even so, the process makes some users worry about outsourced data outflow. Considering the TPA might handle numerous audit sessions simultaneously from various users, their outsourced data files would be at risk. Our design extends the public auditing protocol with preserving privacy in a multi-user setting where TPA can carry on multiple auditing tasks in a group for improved efficiency. The application was tested with test cases like signup phase, login phase and file downloading which were successful.

In future, this application must be simulated with real cloud and check whether it works exactly in the same way and helps the user with its functionalities. This application should be able to upload different types of files such as images, video files, audio files, etc. to upload and allow the auditor to view the content of the file and, if possible, check it to verify it does not contain threats and to be sure about the file’s content.


Referred Sites:

http://java.sun.com
http://www.sourcefordgde.com
http://www.networkcomputing.com/
http://www.roseindia.com/
http://www.java2s.com/