International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 6 | ISSN : 2456-3307

Secure Cloud Storage Auditing Using Hash-Code Technique

Zeba Masroor , Md Ateeq Ur Rahman

[1]M.Tech Scholar, Department of Computer Science & Engineering, Shadan College of Engineering & Technology, Hyderabad, Telangana, India

[2]Professor, Department of Computer Science & Engineering, Shadan College of Engineering & Technology, Hyderabad, Telangana, India

ABSTRACT

The Main objective of the project is to provide hash-value security to the data of the data owner. Every data dynamically generates some blocks of code then every block will have hash code. This hash code or value is totally dependent on the data it contains. First data will be encrypted at client side itself afterwards it will generate hash value of that block. With this technique we are aiming to provide high level security to the data in cloud. The Major problem lies in the security of the data in cloud where a user totally trusts on a remote server to store their data securely. There is numerous amount of studies have been conducted to address the issue of block level security. Untrusted servers need high level infrastructure to overcome the issue of data security in cloud. Hash Value based security is a well formed and clearly examined technique over a period of time in cloud. It converts the actual data with a secure unreadable code called as hash value. It is a key-value pair in which elements are arranged or organized in dictionary entry object.

Keywords : Hash-Table, Hash-Value, Cryptography, Dynamic Hash-Table, Merkel Hash Tree, Cloud Servers, Third Party Auditor, Secure Auditing.

I. INTRODUCTION

Distributed storage is a vital branch of distributed computing, whose objective is to give effective and on request outsourcing information administrations for clients abusing very virtualized frameworks. Because of the minimal effort and superior of distributed storage, a developing number of associations and people are having a tendency to outsource their information storage to proficient cloud administrations suppliers (CAS), which floats the fast advancement of distributed storage and its relative methods as of late. Be that as it may, as another forefront innovation, distributed storage still faces numerous security challenges.

One of the greatest concerns is the way to decide if a distributed storage framework and its supplier meet the lawful desires of clients for information security. This is fundamentally caused by the accompanying reasons. To start with, cloud clients (information owners), who outsource their information in public, can never again confirm the trustworthiness of their information through customary procedures that are regularly utilized in neighborhood stockpiling situations.

Second, CSPs, which endure Byzantine disappointments once in a while, may decide to cover the information blunders from the information proprietors for their own particular self-intrigue. What is more extreme, CSPs may disregard to keep or even intentionally erase seldom got to information that have a place with normal clients to spare storage room. In this way, it is basic and noteworthy to create productive evaluating procedures to fortify information proprietors' trust and trust in distributed storage, of which the center is the way to viably check information uprightness remotely. Up until now, numerous arrangements have been displayed to conquer this issue, which can be by and large separated into two classifications: private evaluating and open inspecting. Private inspecting is the underlying model for remote checking of information respectability, in which the confirmation operation is performed straightforwardly between information proprietors and CSPs with generally ease. Notwithstanding, it can't give persuading reviewing comes about, since the proprietors and CSPs regularly question each other. Additionally, it isn't prudent for the clients to do the review much of the time, since it would

CSEIT172669 | Received : 05 Nov 2017 | Accepted : 15 Nov 2017 | November-December-2017 [(2)6: 173-177]

generously build the overhead that the clients may not bear.

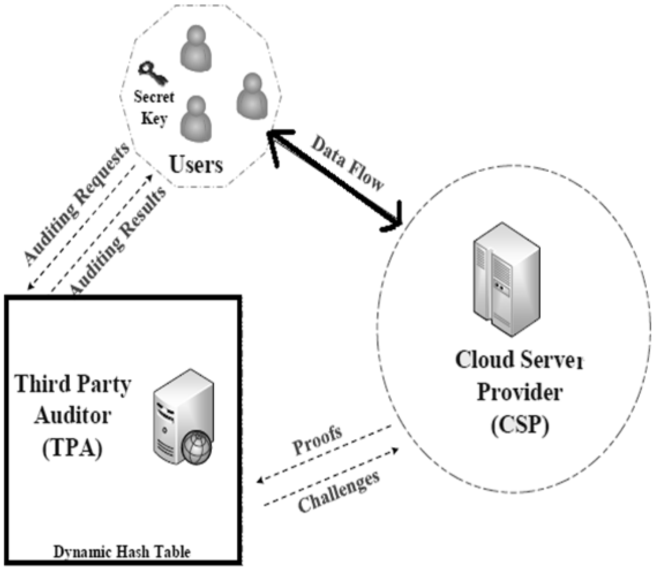


Figure 1. Proposed system framework

II. EXISTING AND PROPOSED SYSTEMS

2.1 Existing System:

In Recent year cloud auditing became the very challenging action towards data security. There are lot of studies conducted so far to address the issue of secure cloud storage auditing.

Ateniese et al. to start with displayed people in general reviewing plan, in which reliable examining comes about and fundamentally lessen clients' superfluous weight by presenting a free TPA. Accordingly, it is more normal and down to earth, and prominently accepted to be the privilege in the general population inspecting, in any case, some imperative issues as takes after stay to be tended to or additionally Gone

Privacy-saving: information security insurance has dependably been a vital subject for distributed storage. In general society inspecting, the center of this issue is the way to protect utilizations' security while presenting a TPA. In spite of the fact that abusing information encryption before outsourcing is a way to deal with alleviate the security worry in distributed storage, it can't counteract information spillage amid the check procedure. Subsequently, it is critical for the cloud inspecting to incorporate a security protecting instrument free to information encryption.

Cluster examining: to improve the productivity and empower the versatility of open reviewing, the TPA should manage different inspecting undertakings from different clients in a quick and cost-proficient way, i.e., bolster the bunching evaluating.

Dynamic inspecting: as it is outstanding that a distributed storage framework isn't only an information stockroom, the clients frequently need to refresh the information progressively spurred by different application necessities. In this manner, it is noteworthy for distributed storage examining to help information progression.

2.1.1 Disadvantages of Existing System:

- Existing schemes failed to explain the cloud auditing in real time scenario. A user needs to make a request dynamically to audit his/her data in cloud
- Some schemes any how manages to provide data security through cryptography technique
- There are many issues regarding the application performance raised as the auditing performed at server side.
- It is basic and noteworthy to create productive reviewing strategies to reinforce information owners′ trust and trust in distributed storage, of which the center is the means by which to adequately check information trustworthiness remotely.
- The existing plans would bring about substantial computational expenses of the TPA and expansive correspondence overhead amid the refreshing and check forms.

2.2 Proposed System:

This paper shows Secure Cloud Storage Auditing by using Hash-Table Technique (SCSA-HT).

A user can totally rely on untrusted server i.e., cloud server for their personal data security.

This paper is an extension of an open examining plan (DHT-PA) utilizing another information structure called dynamic hash table (DHT). Misusing the DHT, our plan can accomplish dynamic inspecting.

2.2.1 Advantages of Proposed System:

- Auditing process is very clear and transparent with the cloud server of the user’s data in cloud.
- Performance will be increased as the data in encrypted at client side for further processing of auditing.
- Security of the application will be increased to higher level as the hash-value is used for block level security of a single file.

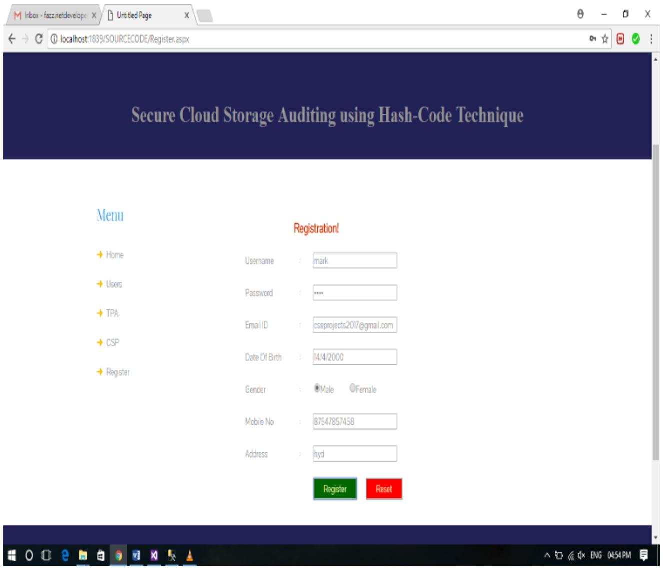
III. METHODOLOGY

The fact that DHT-PA moves the approved data from the CSP to the TPA, its computational expenses and correspondence overhead are altogether littler than the plan in view of skip list and the one in view of MHT. DHT-PA likewise beats IHTPA in refreshing, as the seasons of refreshing operations on the DHT are many less than that on the IHT.

In expansion, we stretch out DHT-PA to accomplish security safeguarding by consolidating the homomorphic authenticator in light of people in general key with irregular concealing produced by the TPA.

IV. IMPLEMENTATION & RESULTS

Below are some of the screenshots of the process.


Inbox - fazz.netdevelope
Untitled Page
localhost:1839/SOURCECODE/Register.aspx
Secure Cloud Storage Auditing using Hash-Code Technique
Menu
Home
Users
TPA
CSP
Register

Registration!

Username : mark

Password : ••••

Email ID : cseprojects2017@gmail.com

Date Of Birth : 14/4/2000

Gender : ● Male ○ Female

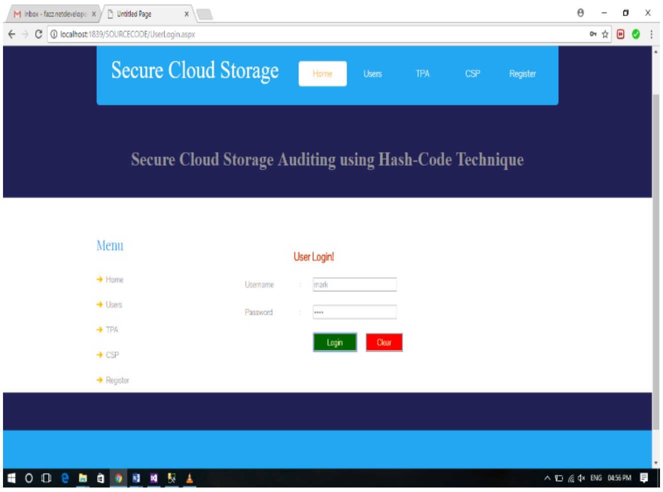
Mobile No : 87547857458

Address : hyd

Register Reset

Output Screenshot 1: Registration page

This is the Registration Page of our application. Here users' needs to provide their personal details like username, password, emailed, dob, gender, mobile no. and address in order to create an account in the cloud.



User Login!

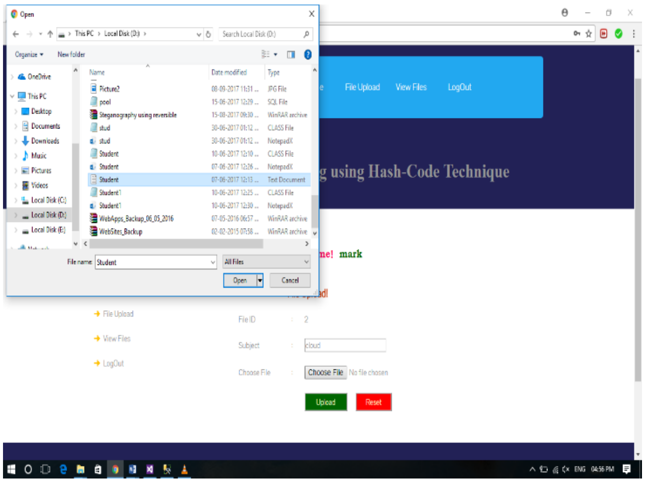
Username : mark

Password : ••••

Login Clear

Output Screenshot 2: User Login page

After Successful Registration Users can login in to the application. Users will provide credentials like username and password, this will be verified with the cloud and then users will be authenticated.



Name	Date modified	Type
Picture2	08-09-2017 11:31 ...	JPG File
pool	15-06-2017 12:29 ...	SQL File
Steganography using reversible	15-08-2017 09:30 ...	WinRAR archive
stud	30-06-2017 01:12 ...	CLASS File
stud	30-06-2017 01:12 ...	NotepadX
Student	10-06-2017 12:10 ...	CLASS File
Student	07-06-2017 12:26 ...	NotepadX
Student	07-06-2017 12:13 ...	Text Document
Student1	10-06-2017 12:25 ...	CLASS File
Student1	10-06-2017 12:30 ...	NotepadX
WebApps_Backup_06_05_2016	07-05-2016 06:57 ...	WinRAR archive
WebSites_Backup	02-02-2015 07:58 ...	WinRAR archive

File name: Student All Files

Open Cancel

File ID : 2

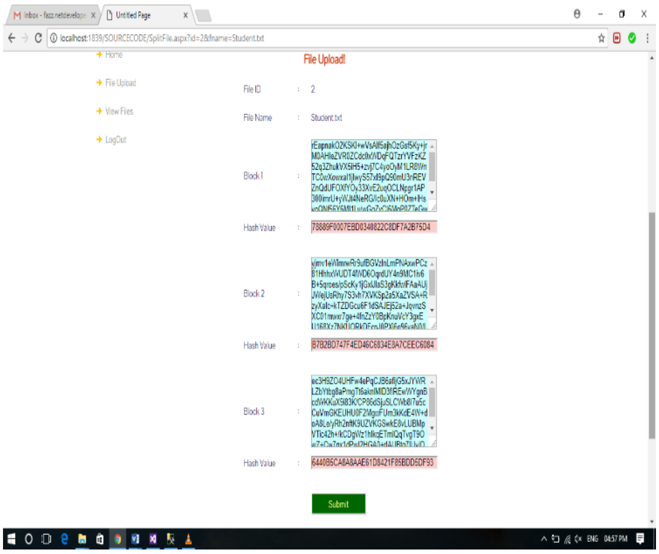
Subject : cloud

Choose File : Choose File No file chosen

Upload Reset

Output Screenshot 3: File upload page

After Successful Login Users can upload their personal data in cloud. Here an example has been given where user is trying to upload a text file with the name student.txt.



File Upload!

File ID : 2

File Name : Student.txt

Block I : rEapnakO2KSKl+wVsAlf5ajhOzGsf5Ky+jr MOAHleZVR0ZCdc0xWDqFQTzrYVFzKZ 52q3ZhukVX5lH5+zvj7C4yoOyM1LR8Wn TCOwXowxal1jlwyS57xl9pQ90mU3rREV ZnQdUFOXfYOy33XvE2uqOCLNpgr1AP 300imrU+yWJt4NeRGllc0uXN+HOm+lHs

Hash Value : 78889F0007EBD0340822C8DF7A2B75D4

Block 2 : yjmv1eWimrwRr9ufBGVzlnLmPNAxwPCz 81HhhxWUDT4fWD6OqrdUY4n9MC1iv6 B+5qroes/pScKy1jGxUlaS3gKkfwFAaAUj JWejUsRhy7S3vh7XVKSp2a5XaZVSA+R zyXalc+kTZDGcu6F1dSAJEj52a+JqvnzS XC01mwxr7ge+4fnZzY0BpKnuVcY3gxE

Hash Value : B7B2BD747F4ED46C6834E8A7CEEC6084

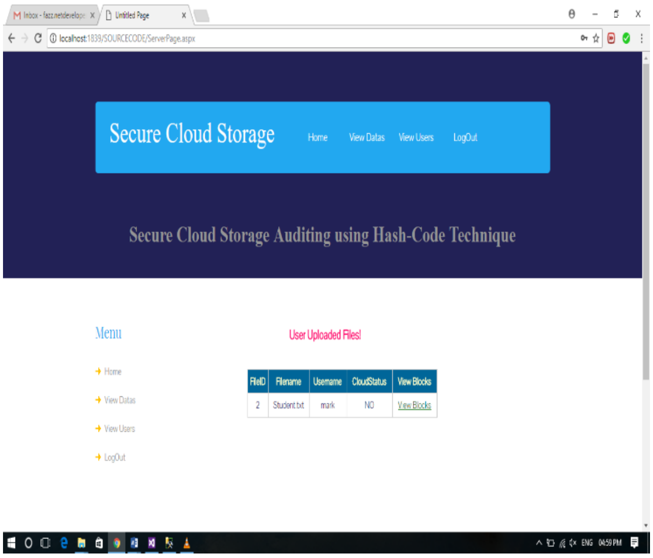
Block 3 : ec3H9ZO4UHFw4ePqCJB6afjG5xJYWR LZbYtbg8aPmgTt6aknlMlD3fREwWYgnB cdVKKuX9i83K/CP86dSjuSLCWb8l7u5c CuVmGKEUHU0F2MgwFUm3kKdE4W+d oA8LeyRh2nftK9UZVKGSwkE8vLUBMp VTic42h+/kCDgWz1hlkqETmlQqTvgT9O

Hash Value : 6440B5CA8A8AAE61D8421F85BDD5DF93

Submit

Output Screenshot 4: file hash value generation

The Uploaded file will be separated in to three different blocks or chunks for security. These blocks contain the actual data but in encrypted format. Every block will have a hash code with which it will be identified.

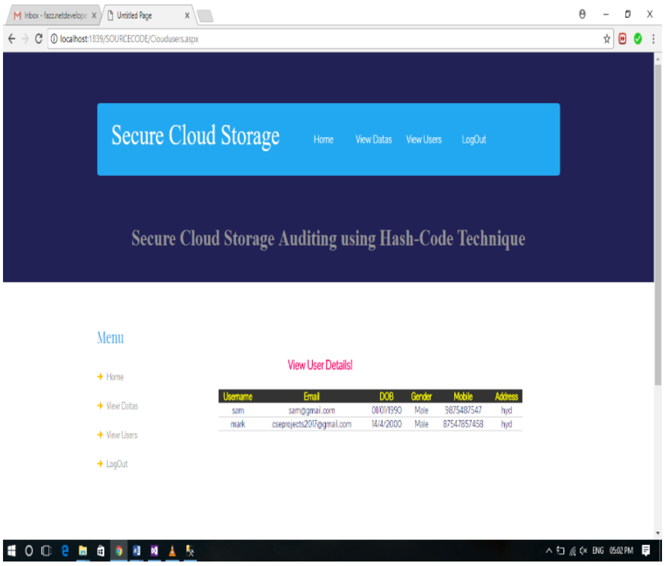

localhost:1839/SOURCECODE/ServerPage.aspx
Secure Cloud Storage
Home
View Datas
View Users
LogOut
Secure Cloud Storage Auditing using Hash-Code Technique
Menu
Home
View Datas
View Users
LogOut

User Uploaded Files!

FileID	Filename	Username	CloudStatus	View Blocks
2	Student.txt	mark	NO	View Blocks

Output Screenshot 5: Server View Files

This is the View File Details page of the Server. Here file details like filename, username, file id, cloud status and verify blocks link are displayed. When clicked on view blocks link a new page will be displayed for the server.

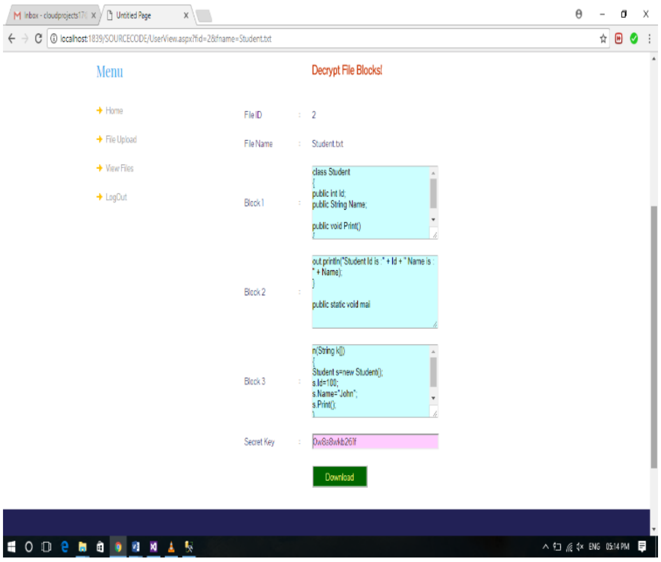

localhost:1839/SOURCECODE/Cloudusers.aspx
Secure Cloud Storage
Home
View Datas
View Users
LogOut
Secure Cloud Storage Auditing using Hash-Code Technique
Menu
Home
View Datas
View Users
LogOut

View User Details!

Username	Email	DOB	Gender	Mobile	Address
sam	sam@gmai.com	01/01/1990	Male	9875487547	hyd
mark	cseprojects2017@gmail.com	14/4/2000	Male	87547857458	hyd

Output Screenshot 6: View Users Details

This is the View Users Details page of the server. Here all the registered users from the cloud are displayed in a grid.


localhost:1839/SOURCECODE/UserView.aspx?fid=2&fname=Student.txt
Menu
Home
File Upload
View Files
LogOut
Download

Decrypt File Blocks!

File ID : 2

File Name : Student.txt

Block 1 :

```
class Student
{
public int Id;
public String Name;

public void Print()
```

Block 2 :

```
out.println("Student Id is :" + Id + " Name is :
" + Name);
}

public static void mai
```

Block 3 :

```
n(String k[])
{
Student s=new Student();
s.Id=100;
s.Name="John";
s.Print();
```

Secret Key : 0w8a8wkb26lf

Output Screenshot 7: User File Block Decryption

This is the File Decryption page. Here when provided correct secret key user file will be decrypted and they can download the file by clicking on submit button.

V. CONCLUSION

In this paper we introduce a novel scheme Secure Cloud Storage Auditing by using Hash-Table Technique (SCSA-HT) to ensure data security to a next level. These days, distributed storage, which can offer on-request outsourcing information administrations for the two associations and people, has been drawing in more consideration. In any case, a standout amongst the most genuine hindrances to its improvement is that clients may not completely believe the CSPs in that it is hard to decide if the CSPs live up to their legitimate desires for information security. Thusly, it is basic and critical to create productive evaluating strategies to fortify information proprietors' trust and trust in distributed storage. In this paper, we are propelled to introduce a novel open reviewing plan for secure distributed storage utilizing dynamic hash table (DHT), which is another two-dimensional information structure used to record the information property data for dynamic evaluating. Varying from the current works, our plan moves the evaluating metadata passage the square labels from the CSP to the TPA, and along these lines altogether decreases the computational cost and correspondence overhead. In the interim, misusing the auxiliary points of interest of the DHT, our plan can likewise accomplish preferable execution over the cutting edge conspires in the refreshing stage. Also, for security safeguarding, our plan presents an arbitrary veiling gave by the TPA into the way toward creating verification to dazzle the information data. Also, our plan additionally misuses the total BLS signature system from bilinear maps to play out numerous examining errands all the while, of which the rule is to total every one of the marks by various clients on different information obstructs into a solitary short one and check it for just a single time to lessen the correspondence cost in the confirmation procedure. We formally demonstrate the security of our plan, and assess the reviewing execution by nitty gritty analyses and examinations with the current ones.

VI. REFERENCES

[1]. H. Dewan and R. C. Hansdah. Proposed a paper "A Survey of Cloud Storage Facilities ", Proc. 7th IEEE World Congress on Services, pp. 224-231, July 2011.

[2]. C. Wang, Q. Wang, K. Ren, N. Cao and W. Lou. Proposed a paper "Toward Secure and Dependable Storage Services in Cloud Computing", IEEE Trans. Service Computing, vol. 5, no. 2, pp. 220-232, 2012.

[3]. K. Ren, C. Wang and Q. Wang. Proposed a paper "Security Challenges for the Public Cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69– 73, 2012.

[4]. J. Ryoo, S. Rizvi, W. Aiken and J. Kissell. Proposed a paper "Cloud Security Auditing: Challenges and Emerging Approaches", IEEE Security & Privacy, vol. 12, no. 6, pp. 68-74, 2014.

[5]. C. Wang, K. Ren, W. Lou and J. Li. Proposed a paper "Toward Publicly Auditable Secure Cloud Data Storage Services", IEEE network, vol. 24, no. 4, pp. 19-24, 2010.

[6]. Q. Wang, C. Wang, K. Ren, W. Lou and J. Li. Proposed a paper ‘‘Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing,’’ IEEE Trans. on Parallel and Distributed Systems, vol. 22, no. 5, pp. 847-859, 2011.