Concealment Perpetuate Common Auditing for Halvered Documents in the Storage

R. Rajalakshmi
M.C.A, M.E.,
Department of computer Applications,
Adhiparasakthi Engineering college
Melmaruvathur

R. Murugan
M.C.A, M.E.,
Department of computer Applications,
Adhiparasakthi Engineering college
Melmaruvathur

Abstract - Web storage services, it is common place for documents to be not only stored in the storage place, but also halvered beyond diverse users. However, common auditing for such halvered data while perpetuate identity privacy remains to be an open challenge. We introduce the first concealment perpetuate mechanism that allows common auditing on halvered data stored in the storage place. In particular, we exploit key to compute the verification information needed to audit the integrity of halvered data. With our mechanism, the identity of the signer on each block in halvered data is kept private from a auditor, who is still able to verify the integrity of halvered data without retrieving the entire file. Our experimental results demonstrate the effectiveness and efficiency of our developed mechanism when auditing halvered data.

I. INTRODUCTION

The project entitled "Concealment perpetuate common Auditing for halvered documents in the storage" for open source file developed using JAVA as front end and MYSQL server as back end. It is a web based application. The halvered is split into two parts one is free server and another one is payable. This project keeps information of the editing the open source file and manage the free server and payable. In the free server the user just view and read the file, payable one is used for download and edit the file.

The administrator can communicate with the data owner and the client, will be able to task status of open source file. The data owner is the organization maintained the information. The client open the file and will see the data owner information. The data owner is the validation login and upload file. The client select data owner and register in the site. The client add server based on specification , get validation and login. The client will able to see all the information of the data owner.

The database administrator monitors the open source file through this application. All the data about editing the open source file will be maintained in a centralized database . The administrator will be having the rights to enter the authorized user. It deals with all kinds ofsharing document, client authentication, server acquisition , data modification.

II. EXISTING SYSTEM

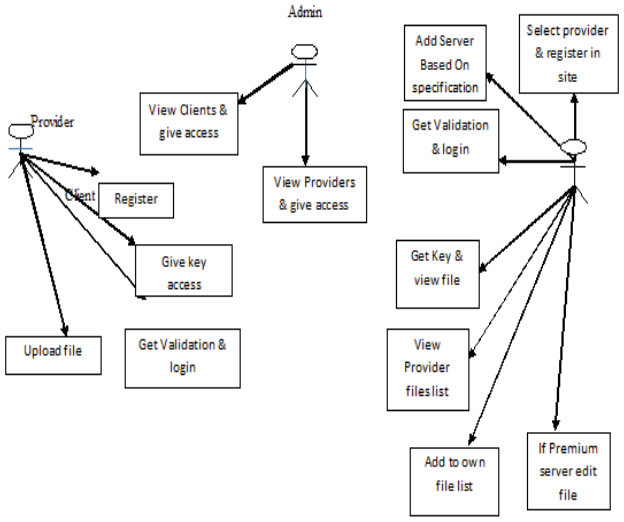
In existing system any user can login and modify the uploaded data and this cause the main disadvantage which leads to destructed data. This provides the data which is irrelevant to the users and public auditing is designed to check the correctness of data stored in an un trusted server, without retrieving the entire data. The content of private data belonging to a personal user is not disclosed to the third party auditor.

III. PROPOSED SYSTEM

In proposed system, we only consider how to audit the integrity of shared data in the cloud with static groups. It means the group is pre-defined before shared data is created in the cloud and the membership of users in the group is not changed during data sharing. The original user is responsible for deciding who is able to share her data before

outsourcing data to the cloud. Another interesting problem is how to audit the integrity of shared data in the cloud with dynamic groups a new user can be added into the group and an existing group member can be revoked during data sharing while still preserving identity privacy.

IV. SYSTEM ARCHITECTURE



V. IMPLEMENTATION

Modules:
Sharing Document
Client Authentication
Server acquisition
Data Modification

Sharing Document:
In this module an owner has to upload its files in a cloud server, he/she should register first. Then only he/she can be able to do it. For that he needs to fill the details in the registration form.

These details are maintained in a database. One user should get authentication to login into the website. User can upload the file in the website for other users benefits.

Client Authentication:
During registration the client should select the provider. Client will get the authentication from admin to login into the website.

Client can login into the website after authentication else the client will get the notification as "You are Not Authenticated by Admin".

Server acquisition:
Client can view the files from the website using the free server but cannot modify the data in the free server.

Client will get the key to view the files in the websites. Using the key only client can access the data.

Data Modification:
Client should buy a server to edit and save the data in the website. Using the server client can view and edit the data. Only newly updated files will be saved in the website.

VI. INPUT AND OUTPUT PROCESS OF THE PROJECT

Input:
Provider will register into the website and upload file into the database.Client will enter the details with the provider and the server details.

Output:
Admin will view the provider and the client details to give the login access. Client will view the uploaded files and get key to access the file. If the client using the premium server then they can edit the file. If it is a free server then client should get a key and can view the file. Provider will give the key access to the client. Client can add to the own file list.

Advantage:
In this project we construct a public auditing mechanism for cloud data, so that during public auditing, the content of private data belonging to a personal user is not disclosed to the third party

Disadvantage:
The public auditing is designed to check the correctness of data stored in an un trusted server, without retrieving the entire data. Data stored in an un trusted cloud can easily be lost or corrupted, due to hardware failures and human errors.

VII. CONCLUSION

The Concealment perpetuate common Auditing for halvered documents in the storagefor open source file developed using JAVA as front end and MYSQL server as back end.The first privacy preserving public auditing mechanism for shared data in the cloud.It is a web based application. The halvered is split into two parts one is free server and another one is payable.

This project keeps information of the editing the open source file and manage the free server and payable. In the free server the user just view and read the file, payable one is used for download and edit the file. The data stored in the dynamic form. An interesting problem in our future work is how to efficiently audit the integrity of shared data with dynamic groups while still preserving the identity of the signer on each block from the third party auditor.The database and the information can be update to the latest forthcoming versions.

REFERENCES

[1] Astels, D, Test-Driven Development – A Practical Guide, PrenticeHall, 2003.
[2] Fowler, M, Refactoring – Improving The Design of Existing Code, Addison-Wesley, 2004.
[3] Booch, G, Rumbaugh, J, and Jacobson, I, The Unified Modeling Language User Guide, Addison-Wesley, 1999.
[4] Conte, S, D, Dunsmore, H, E, and Shen, V, Y, Software Engineering Metrics, Benjamin Cummings, 1986.
[5] Fenton, N, E, Software Metrics: A Rigorous Approach, Chapman and Hall, 1991.
[6] Chikofsky, E, J, and Cross, J, H, "Reverse Engineering and Design Recovery: A Taxonomy", IEEE Software, January 1990, pp. 13-17.
[7] Henderson-Sellers, B. Object-Oriented Metrics – Measures of Complexity", Prentice Hall, 1996.
[8] Martin, R, C, Agile Software Development – Principles, Patterns, and Practices, Prentice Hall, 2002.
[9] www.mysql.com
[10] www.w3schools.com
[11] http:\\dev.mysql.com