International Journal of Engineering & Scientific Research
Vol.7 Issue 5, May 2019,
ISSN: 2347-6532 Impact Factor: 6.660
Journal Homepage: http://esrjournal.com, Email: esrjeditor@gmail.com
Double-Blind Peer Reviewed Refereed Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

DE-DUPLICATABLE EFFECTIVE PROOF OF REPOSITORY

Niraj H Patel*
Kiran Jeedi**
Parsi Sai Teja***
R Mounish Raja****

Article Received: 18 March 2019
Article Revised: 5th April 2019
Article Acepted:9th April 2019

Keywords:
Cloud Storage,
De-duplication,
Repository,
Alteration,
Cross-user.

ABSTRACT

Storage outsourcing is becoming more and more enchanting to both industry and academic due to the advantages of low cost, high accessibility, and easy sharing. As one of the storage farm out, cloud storage gains extensive attention in modern years. Many companies as Amazon, Google, and Microsoft, provide their own cloud storage services, where users can upload their files to the servers, access them from various devices, and share them with the others. Although cloud storage services are widely utilized in current days, there still remain many security issues and potential hazards .Data purity(9) is one of the most important properties when a user outsources its files to cloud storage(2). Users should be convinced that the files stored in the server are not manipulated. Traditional techniques for protecting data purity, such as digital signatures, require users to download all of the files from the cloud server for verification, which may cause a heavy communication cost. These techniques are not suitable for cloud storage services where users may check the purity frequently, such as every hour. Thus, researchers introduced Proof of Repository (PoR)(2) for checking the purity without downloading files from the cloud server. Furthermore, users may also require several impressive operations, such as alteration, insertion, and deletion, to update their files, while maintaining the capability of PoR(2).

Author correspondence:
Niraj H Patel,
B.Tech Scholars, Department of Information Technology,
Gurunanak Institutions Technical Campus, Hyderabad.
Email:nirajhpatel1@gmail.com

*****B.Tech Scholars, Department of Information Technology, Gurunanak Institutions Technical Campus, Hyderabad.
****** Assistant Professor, Department of Information Technology, Guru Nanak Institutions Technical Campus, Hyderabad.
******* B.Tech Scholars, Department of Information Technology, Gurunanak Institutions Technical Campus, Hyderabad.
******** B.Tech Scholars, Department of Information Technology, Gurunanak Institutions Technical Campus, Hyderabad

1. Introduction

Excessive increase in amount of data has determined the need to upgrade the data storage devices. The data produced by sub-sequent users, servers, personal computers, IT companies includes different forms of data such as images, videos, text files, PDFs etc. Deduplication is a well-known technique that mainly focuses to save storage space by removing repeated copies of data.

The concept of de-duplicatable effective proof of storage and propose an efficient construction called DeyPoR, to achieve effective PoR(2) and secure cross-user deduplication, simultaneously. Considering the challenges of structure diversity and private tag generation, we exploit a novel tool called Homomorphic Authenticated Tree (HAT). We prove the security of our construction, and the theoretical analysis and experimental results show that our construction is efficient in practice.

Cloud computing is used for storing and managing the data in a remote location through internet rather than the local server or a private system. Cloud computing is shared pools of configurable computer system resources and higher-level services that can be rapidly provisioned with minimal management effort, often over the Internet. Cloud computing relies on sharing of resources to achieve coherence and economies of scale, similar to a public utility.

2. Cloud Computing

Cloud computing is shared pools of configurable computer system resources and higher-level facilities that can be rapidly provisioned with minimal management effort, often over the Internet. Cloud computing relies on sharing of resources to achieve coherence and economies of scale, similar to a public utility.

DainikSanjog.com
Laptop
Network
Smartphone
Cloud Computing
Tablet computer
Server
Mobile
Database

Fig 1: Cloud Computing

3. Projected Task

3.1 Cryptographic cloud storage:

“Cryptographic(1) cloud storage” is considered as a problem of building secure cloud storage service on top of a public cloud infrastructure, where service provider is not trusted completely by the customer. They described several architectures that combine recent and non-standard cryptographic primitives in order to achieve goal. This new economic and computing model is commonly referred to as cloud computing. This includes various types of services such as, infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS).

3.1.1 A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data

More data owners are motivated to outsource their data to cloud servers for great convenience and reduced cost in data management. Sensitive data should be encrypted before outsourcing for privacy requirements. Secure(4) multi-keyword ranked search scheme over encrypted cloud data which simultaneously supports dynamic update operations like deletion and insertion of documents.

3.1.2 Secure and efficient proof of storage with deduplication:

For the success of cloud storage both security(4) and efficiency are critical. Proof of Data Possession (PDP) and Proof of Repository (POR) was proposed for detecting that the data stored in the cloud.

3.1.3 Proofs of ownership and retrievability in cloud storage:

Deduplication is a basic requirement for cloud storage as it saves storage space of cloud servers. As clients are not trusted from the perspective of the server, the concept of Proofs of Ownership (POWs) has been proposed in client-side deduplication. On the other hand, the clients cannot completely trust the server either, thus clients have to know whether their files are stored integrally in the cloud. Most of the existing system focuses on only one-way validation. A proof of Ownership and Retrievability(PoOR) is introduced in this paper. Clients can prove to the server their ownership of files and verify the retrievability(3)(10) of the files without uploading or downloading them.

4. SYSTEM DESIGN

Our system model considers two types of entities: the cloud server and users, as shown in diagram. For each file, original user is the user who uploaded the file to the cloud server, while subsequent user is the user who proved the ownership of the file but did not actually upload the file to the cloud server.

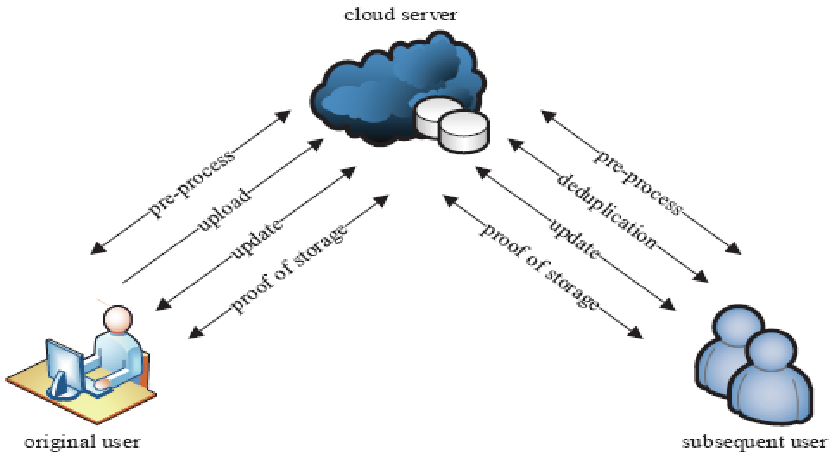

Fig 2: System Design

There are five phases in a deduplicatable dynamic PoR system: pre-process, upload, deduplication, update, and proof of repository.

a) In the pre-process phase, users try to upload their files. The cloud server decides whether these files should be uploaded. If the upload process is granted, go into the upload phase; otherwise, go into the deduplication phase.

b) In the upload phase, the files to be uploaded that are not existed in the cloud server. The users encode the local files and upload them to the cloud server.

c) In the deduplication phase, the files to be uploaded alreadyexists in the cloud server. The subsequent users possess the files locally and the cloud server stores the authenticated structures of the files. Subsequent users need to convince the cloud server that they own the files without uploading them to the cloud server.

d) In the update phase, users may modify, insert, or delete some blocks of the files.

e) In the proof of storage phase, users only possess a small constant size metadata locally and they want to check whether the files are faithfully stored in the cloud server without downloading them.

5. System Process Flow

5.1 User Interface Design

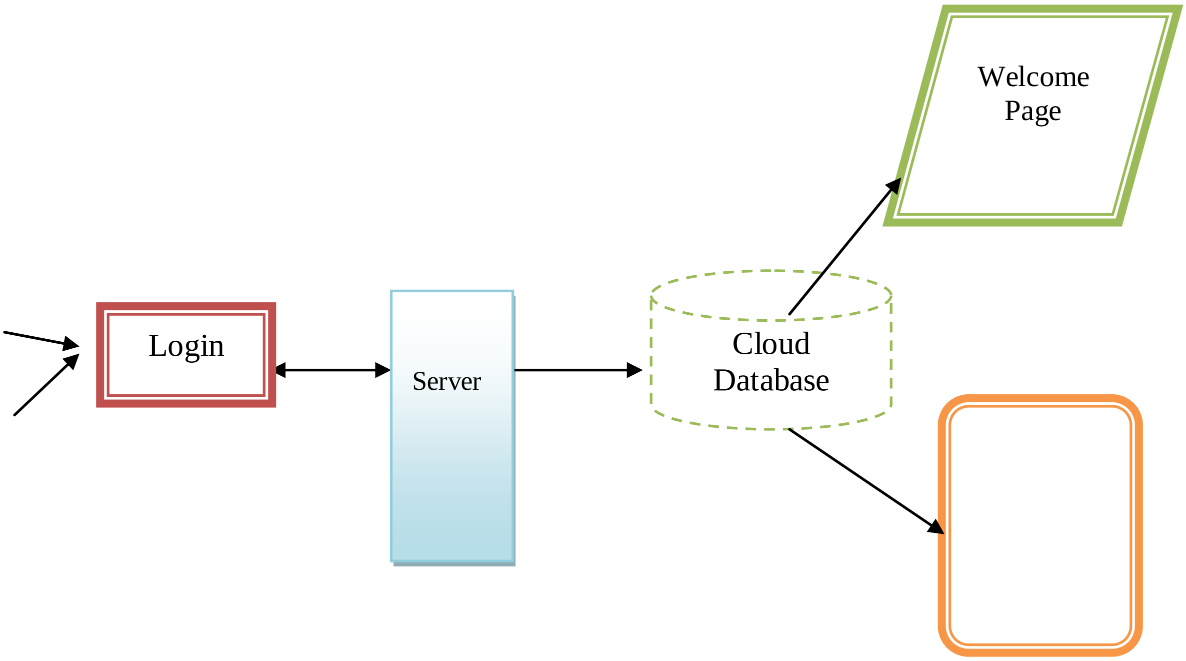

Fig 5.1:User Interface Design

Interfaces exist to enable interaction between humans and our world. They can help clarify, illuminate, enable, show relationships, bring us together, pull us apart, manage our expectations, and give us access to services. The act of designing interfaces is not Art. Interfaces are not monuments unto themselves. Interfaces do a job and their effectiveness can be measured.

5.2 Cloud User:

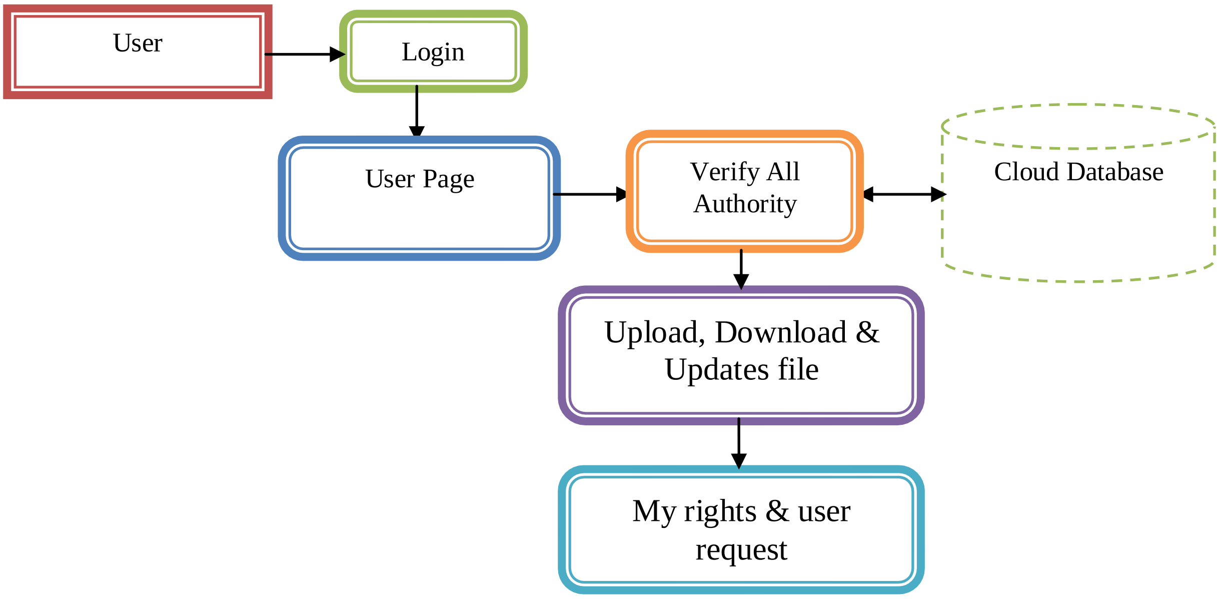

Fig 5.2:Cloud User

After the user(5) has login to the server, he has ask few permission from the cloud authority so that the user can upload or update or download the files from the server. So the requests are been sent to the cloud authority.

5.3 Cloud Authority:

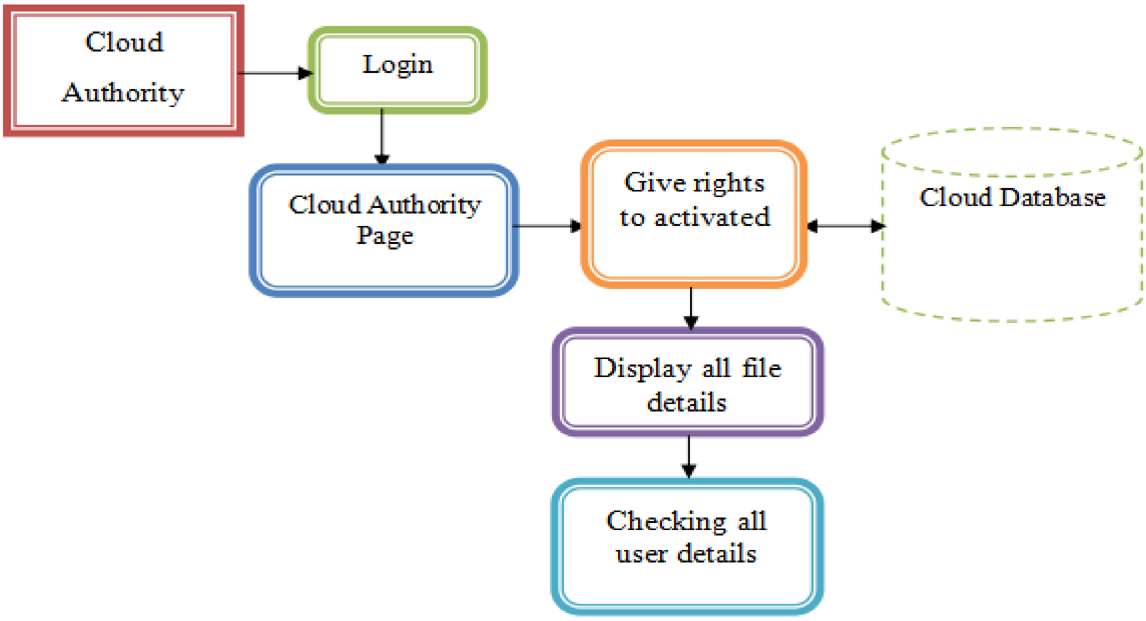

Fig 5.3:Cloud Authority

This section of the cloud authority provides guidance on different approaches to implement the cloud security(4)(7) principles. Each principle represents fundamental security aspects that you should consider when determining which cloud services meets your need.

5.4 Admin:

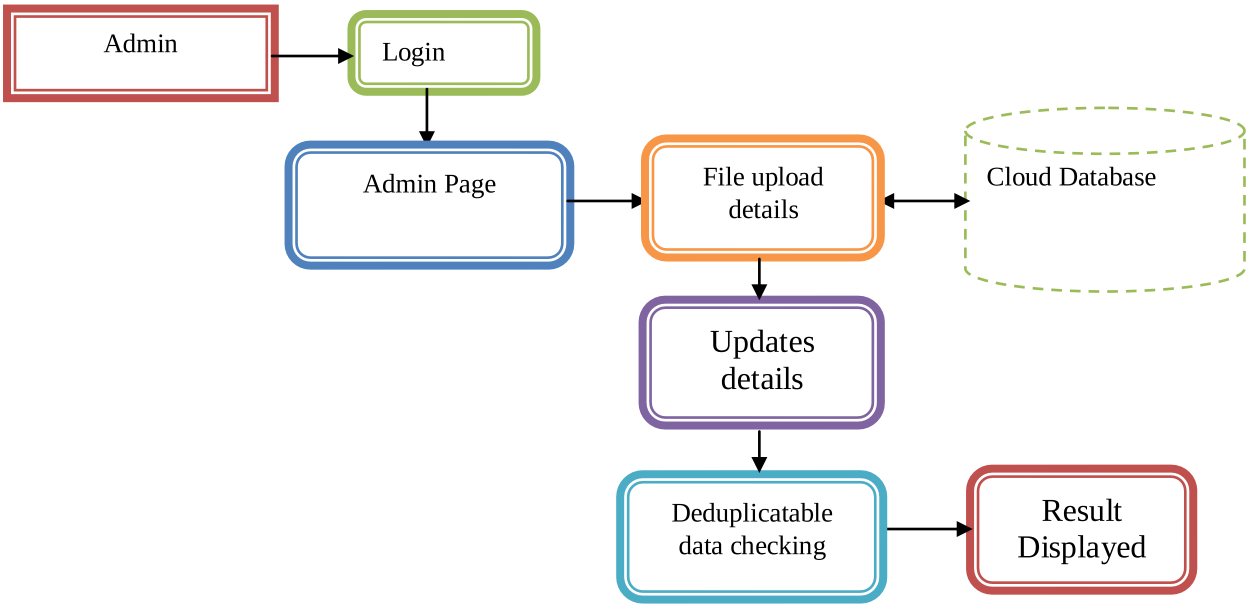

Fig 5.4:Admin

In this section the cloud admin verifies the user by sending a encryption key to the user. The cloud admin gives the operation rights to the user (i.e. upload, update and download) upon the user request.

5.5 Deduplicatable dynamic pos:

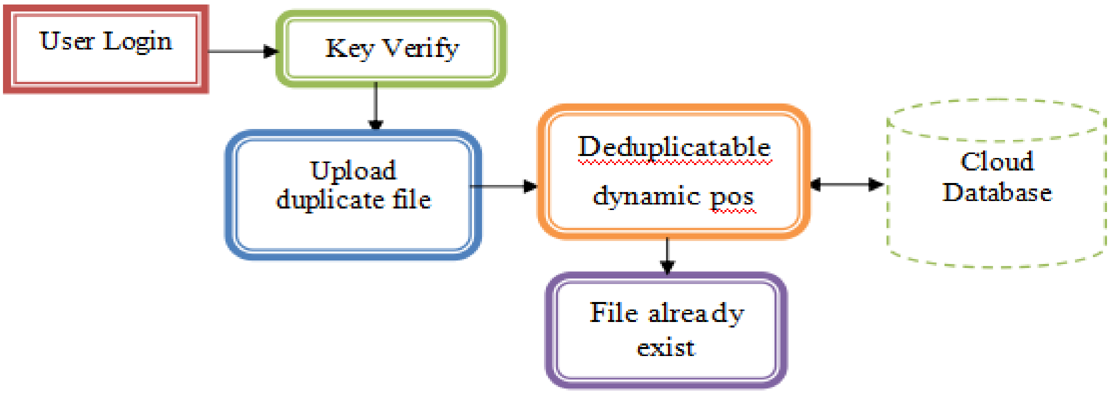

Fig 5.5:Deduplicatable dynamic pos

Here the data being uploaded by the user will be checked whether the data is not been uploaded by others or by himself. By doing this the data duplication can be controlled. So this mechanism is known as Deduplication(6).

6. Conclusion

We proposed a model where the data is been shared among all the users of the cloud. There are three different roles in this module like Cloud user, Cloud authority, and Cloud admin. Cloud authority and Cloud admin are the main roles in this module as it maintains the security of the data. The compendious requirements in multi-user cloud storage systems and the model of deduplicatable dynamic PoS is introduced.

References

1. S. Kamara and K. Lauter, “Cryptographic cloud storage,” in Proc. of FC, pp. 136–149, 2010.

2. G. Ateniese, S. Kamara, and J. Katz, “Proofs of storage from homomorphic identification protocols,” in Proc. of ASIACRYPT, pp. 319–333, 2009.

3. H. Shacham and B. Waters, “Compact proofs of retrievability,” in Proc. of ASIACRYPT, pp. 90–107, 2008.

4. C. A. Ardagna, R. Asal, E. Damiani, and Q. H. Vu, “From Security to Assurance in the Cloud: A Survey,” ACM Comput. Surv., vol. 48, no. 1, pp. 2:1–2:50, 2015.

5. S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, “Proofs of ownership in remote storage systems,” in Proc. of CCS, pp. 491– 500, 2011.

6. J. Douceur, A. Adya, W. Bolosky, P. Simon, and M. Theimer, “Reclaiming space from duplicate files in a serverless distributed file system,” in Proc. of ICDCS, pp. 617–624, 2002.

7. J. Chen, L. Zhang, K. He, R. Du, and L. Wang, “Message-locked proof of ownership and retrievability with remote reparing in cloud,” Security and Communication Networks, 2016.

8. D. Cash, A. K¨upc¸ ¨u, and D. Wichs, “Dynamic proofs of retrievability via oblivious RAM,” in Proc. of EUROCRYPT, pp. 279–295, 2013.

9. E. Stefanov, M. van Dijk, A. Juels, and A. Oprea, “Iris: A scalable cloud file system with efficient integrity checks,” in Proc. of AC- SAC, pp. 229–238, 2012.

10. M. Azraoui, K. Elkhiyaoui, R.Molva, andM. ¨Onen, “StealthGuard: Proofs of Retrievability with Hidden Watchdogs,” in Proc. Of ESORICS, pp. 239–256, 2014.