

## Data Integrity Proofs on Cloud Computing Using Blockchain

Ms.Khushi Jadhav<sup>1</sup>, Ms.Tejashri Jadhav<sup>2</sup>, Mr.Manas Jadhav<sup>3</sup>, Mr.Prajwal Aher<sup>4</sup>

<sup>1</sup>UG Student, Dept. of IT, MVP Samaj's Karmaveer Baburao Thakare College of Engineering, Nashik.

<sup>2</sup> UG Student, Dept. of IT, MVP Samaj's Karmaveer Baburao Thakare College of Engineering, Nashik.

<sup>3</sup> UG Student, Dept. of IT, MVP Samaj's Karmaveer Baburao Thakare College of Engineering, Nashik.

<sup>4</sup> UG Student, Dept. of IT, MVP Samaj's Karmaveer Baburao Thakare College of Engineering, Nashik.

\*\*\*

### Abstract -

Cloud data deduplication can save cloud storage resources and network communication bandwidth, and improve cloud storage efficiency. In our project we develop the series of deduplication technologies that are proposed to achieve cloud data deduplication. In past, the traditional duplication technologies and scheme mainly implements data deduplication without providing data integrity verification, so it cannot be verified whether the cloud service provider correct stores user data. In our project we proposes a data integrity verification scheme of deduplication for cloud ciphertexts, including cloud ciphertext deduplication and cloud data integrity verification. In our project Data Integrity Proofs on Cloud Computing using Blockchain we adopted the signature method. In addition, our project ensures the privacy of cloud data, and satisfies the user data is secure and our result shows the higher efficiency.

**Key Words:** Data Deduplication, Cloud Storage, Data Integrity, Blockchain.

### 1. INTRODUCTION

In recent years, more and more individuals and companies choose to store data in the cloud servers instead of local storage systems, which not only alleviates the cost pressure caused by local data storage and maintenance, but also eliminates the need for complex networks and maintenance of software and hardware systems. In the cloud storage system, the user can use and process the data on the cloud server at any time, but the user

loses control of the data stored on the remote node. On the one hand, no matter what highly reliable measures the cloud service provider takes, data loss may occur. On the other hand, for their own economic interests, cloud service provider (CSP) may deliberately discard some unaccessed or rarely accessed data to save storage space, and claim that the data is still stored correctly in the cloud. Therefore, CSP cannot be fully trusted by users. This scheme allows any third party to challenge the cloud server to verify the possession of the data, which greatly expands the application area of the PDP. Due to the development of big data and cloud computing technology, CSP is also facing the problem of efficiently managing cloud resources and avoiding the waste of cloud resources. In cloud storage, the same data file may be uploaded to the cloud server by different users, resulting in duplicate file copies. Such duplicate data greatly wastes cloud storage space and complicates data management. Therefore, for the cloud storage and processing of big data, it is crucial to solve the problem of duplicate data storage.

Our project proposed a random client-side data deduplication scheme, which realized multi-user ownership management and data sharing by means of a dynamic encryption key tree. But the scheme does not support user revocation in this project. Client-side secure deduplication scheme for ciphertext data in cloud storage, but did not achieve data integrity verification. Our project proposed a cloud data audit scheme that supports encrypted data deduplication, and does not support multi-user operations; proposed an integrity audit scheme that supports key update and ciphertext data deduplication, but requires users to participate in

key update online, and does not support multiple users operating.

In response to the above problems, we propose a data integrity verification scheme for ciphertext data security and deduplication. In this scheme, users store data in a cloud server in the form of ciphertext, which can solve the problem of data privacy. The scheme uses block tags to verify the ownership of user data to achieve cloud data ciphertext deduplication. In data integrity verification, a proxy re-signature method is used to ensure that users do not have to participate in the audit process online all the time, which also reduces the amount of calculations on the cloud server. By comparing similar schemes, our scheme has better efficiency.

## 2. RELATED WORK

These are several applications that detect duplicate data but no application match the accuracy provided by the blockchain technology. In [2], Qian Wang, Cong Wang, Ensure the integrity of data storage in Cloud Computing. Third party auditor verify the integrity of the dynamic data stored in the cloud. In [3], Mark W. Storer, Darrell D. E. Long, Kevin Greenan, Ethan L. Miller, They developed a solution that provides both data security and space efficiency in single-server storage and distributed storage systems. Encryption keys are generated in a consistent manner from the chunk data; thus, identical chunks will always encrypt to the same ciphertext. In [4], C. Sasikala<sup>1</sup>, C. Shoba Bindu<sup>2</sup>, Certificateless RDIC protocol is secure and it provides the privacy against the verifier, and performance analysis guarantees that it makes the less computation overhead over the public verifier. In [6], S. Meena, V. Krithika, TY-JOUR, They use blockchain technology in file transfer system. Since blockchain provides only the authentication, we intend to provide confidentiality to the data by encrypting it with the encryption algorithm, AES before hashing. Thereby, they can ensure the security of data and can make it trustworthy for the users. In [9] Balamurugan N<sup>1</sup>, Bhuvanesh R<sup>1</sup>, Lathapriya K M, Sharmasth Vali Y<sup>2</sup>, Shakkeera L<sup>3</sup>, They implement the blockchain technology, the user's data will be split into blocks and stored in different servers which makes the data more secure.

## 3. DATA INTEGRITY VERIFICATION SCHEME OF DEPLICATION FOR CLOUD CIPHERTEXTS

### Scheme description

1) Key generation - The first uploading user divides the file M into n blocks, calculates the convergence key for each data block, and performs privacy protection processing on each data block. We add the encryption decryption to the data to secure the users files.

2) File initialization - The user uses the encryption key to calculate the ciphertext and file label of the data block, and encrypts the encryption key to ensure the confidentiality of the encryption key.

3) File storage - When the CSP detects that the file label if it does not exist, it means that the user who uploaded the file is the first uploading user, the user needs to initialize and upload the file to the CSP.

4) File Upload - The user uploads all the data blocks of the file M, all the ciphertexts and the secret keys. CSP verifies whether first time file is uploading if it is true, it means that the ciphertext and the tag uploaded by the user match CSP verifies ID represents the collection of the identity information of all users who own the file M; otherwise, CSP returns an error message.

5) Signature value - When users again upload the same file on cloud storage then this signature value gets check.

6) Integrity Verification - Third party auditor verifies the integrity of the data stored in the cloud service provider on behalf of users.

7) File deduplication - When detecting the existence of a file tag T<sub>t</sub>, CSP performs file deduplication.

Table 1. Calculate Overhead Comparision  
Of  
File Storage And Dedupliocation

File Storage	Data block intialization
File Storage	File Upload
File Deduplication	File Proof Generation
File Deduplication	File Proof Verification

Table 2. Calculate Overhead Comparision  
Of  
Integrity Verification Stage

Integrity Verification	Proof Generation
Integrity Verification	Proof Verify

#### 4. ALGORITHM STYLE AND OTHER TERMINALOGIES

**SHA-1** - Hash-based data deduplication methods use a hashing algorithm to distinguish chunks of data individually. The mainley used algorithm is SHA-1. As a hashing algorithm processes data, a hash is generated that represents the data and detects the duplicate ones via certain forms of the comparison process.

In our System, Firstly, the traditional algorithm compares the hash values of the file, if they are the same, then the algorithm will compare the two files byte to byte get checked that it contains duplicate data or not, if they are identical it does not get saved on cloud and get detected as deduplicate data.

**FILES** - we can upload the data in the form of document, word, text, video, audio, picture, pdf, format document, etc. When we upload the data through our system it gets stored on firebase cloud, the use of Cloud Storage for Firebase is to upload and share user generated content, such as images and video, which allows users to build rich media content into our system. Users data is stored in a Google Cloud Storage bucket an exabyte scale object storage solution with high availability and global redundancy.

#### 5. SOFTWARE REQUIREMENTS

##### a. Programming language - C#.NET

C# is a programming language developed by Microsoft as part of the .NET framework. C# is an object-oriented language that offers a combination of strong typing, garbage collection, and scalability. Visual Studio is used as Integrated development environment (IDE). It is supported on Windows, macOS, Linux, iOS, Android, and more. It provides a rich set of features and libraries that make development efficient and productive.

##### b. Firebase Cloud

Firebase Cloud is a cloud-based platform developed by Google that provides a variety of services and tools to help developers build and scale their applications. It offers a wide range of features, including real-time database, authentication, hosting, storage, cloud functions. Cloud Storage for Firebase lets users securely upload these files directly from mobile devices and web browsers, handling spotty networks with ease. When we upload the data on firebase cloud the data get converted into n number of blocks, and each block having its own hash value, then signature value is generated.

## 6. PROPOSED SYSTEM

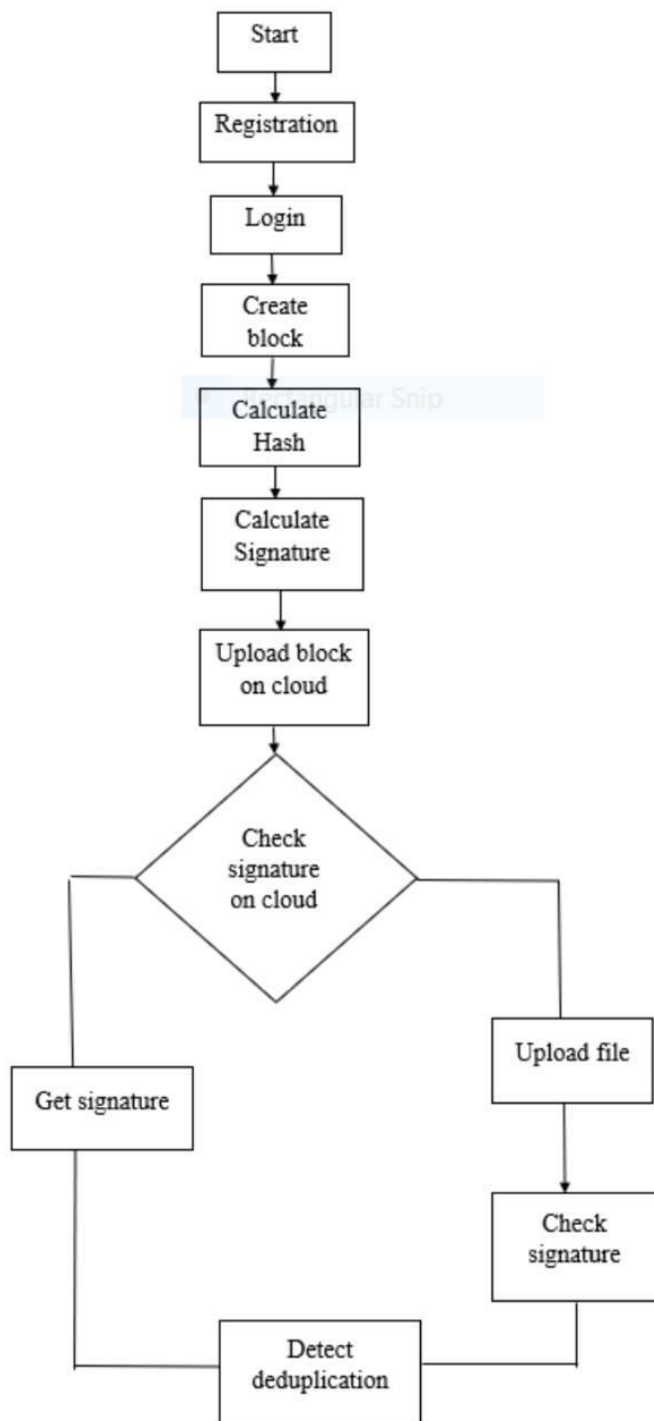


Figure 1. Proposed System

- 1) Firstly, with the help of GUI we try to register the new patient and new doctor.
- 2) Once patient and doctor is registered, data is uploaded on the firebase cloud.
- 3) Through doctor login, doctors can fill the details of the patients and upload their medical history details file on cloud.
- 4) There is one more login where admin get access to patient details and treatment

details.

- 5) If any one change any information about a patient on a cloud admin get a message file is corrupted.
- 6) As doctor uploaded medical history details file on cloud already. If doctor wish to upload the same file again on a cloud with the same name or by changing the file name it is not a possible.
- 7) When same file uploaded again and again the our system it get detected and show the message “Data deduplication detected file can’t save.”

## 7. IMPLEMENTED SYSTEM

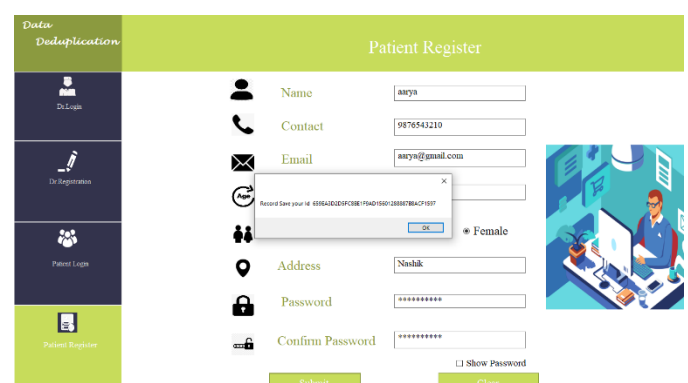
Here we implement a system for a Hospital Management System.

### • Home Page

Home page consist of three main component that is Patient login, Doctor login and Admin login . After clicking any one of the buttons we get navigate to their respective page.

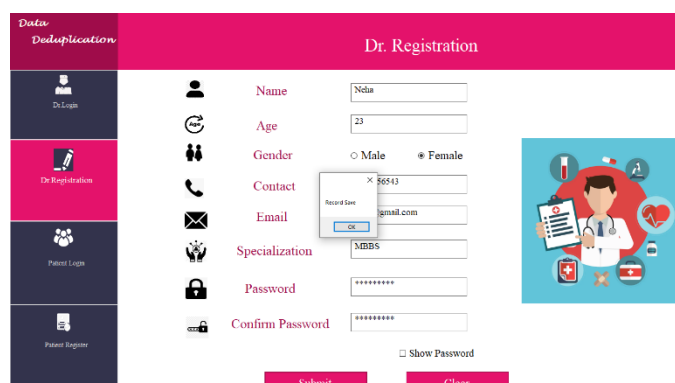
### • Registration Form

Here we implement a system for a hospital. Through which we save the data on cloud system securely without duplicate data. So here we created registration form for patient. Where they can enter personal information like their name, contact, Email, age, gender, address and password they can register themselves on our system.



The screenshot shows a web application interface with a sidebar on the left containing navigation links: 'Data Deduplication', 'Dr Login', 'Dr Registration', 'Patient Login', and 'Patient Register'. The main content area is titled 'Patient Register' and contains a form with the following fields: Name (filled with 'Anya'), Contact (filled with '9876543210'), Email (filled with 'anya@gmail.com'), Address (filled with 'Nashik'), Password (filled with '\*\*\*\*\*'), and Confirm Password (filled with '\*\*\*\*\*'). There are 'Submit' and 'Clear' buttons at the bottom. A small notification box is visible over the form, stating 'Record Save your id: 605642025FC0BE1940155012888786AC7F937'.

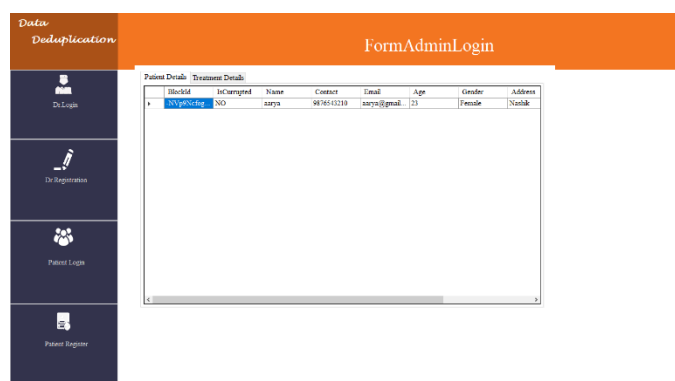
Alike this we create registration page for Doctor. Where they can enter personal information like their name, contact, Email, age, gender, specialization and password they can register themselves on our system.



## • Login Form

By using registration details i.e., contact and password patient and doctor can login to the system. If patient or a doctor are not register earlier, he cannot login.

Their also one admin login form where all the details of patient are saved. If any one change their data on cloud or try misbehave with it then at that time admin can see that file is corrupted.

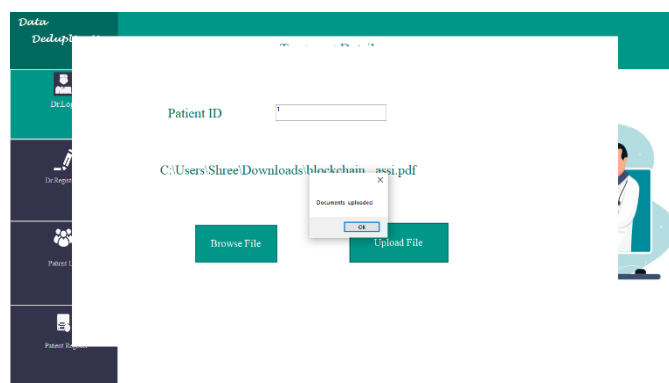


## • Treatment Details

We created treatment details form for patient. Where doctor can enter information like their patient id, symptoms, disease, medicine and

quality they can save their details on our system.

After that by using patient id doctor can upload the patient file on a cloud.



## 8. MATHEMATICAL MODULE

$I = \{ I_1, I_2, I_3, \dots, I_n \}$

$I_1$  = patient login

$f_1$  = create block

$f_2$  = calculate hash

$f_3$  = calculate signature

$f_4$  = upload file

$f_5$  = get signature

$f_6$  = check signature

$f_7$  = upload block on cloud

$O = \{ O_1, O_2, O_3, \dots, O_n \}$

$O_1$  = data store on firebase

$O_2$  = detect deduplication

## 9. ADVANTAGES & LIMITATIONS OF THE SYSTEM

### Advantages of the system

- Our system detects the duplicate file size of 1 tb within just a second this is the most important advantage of our system.
- The cryptographic nature of blockchain ensures the security and confidentiality of data integrity proofs



- Data integrity proofs on the blockchain can facilitate the detection of data corruption or unauthorized modifications.

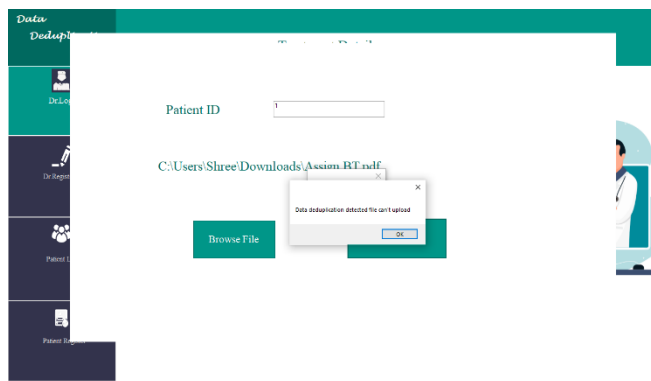
### Limitations of the system

- More computer processing required
- Good internet connectivity required

## 10. RESULT

### Data Deduplication Check

When doctor try to upload the same file again and again the our system detect it and show the message “Data deduplication detected file can’t save.”



## 11. FUTURE SCOPE

It provides a significant contribution to the society, and the potential benefits of this method are clear. With the development of new technologies and continuous research, the reliability and accuracy of this method could be improved, making it a practical solution for detect the deduplicate data in various aspect. The potential benefits of this method in various field where public data nativeness and security is most important , and further studies and improvements are required to validate its accuracy and reliability.

## 12. CONCLUSION

The project data integrity proofs on cloud computing using blockchain designs a data integrity verification scheme of deduplication for cloud ciphertexts, in order to achieve cloud ciphertext data deduplication and data integrity verification. In the scheme, the user and third party audit jointly generate the encryption key of the data block, which realizes the secondary encryption of the convergence key, making the data encryption more secure. The scheme uses block signatures to verify user’s ownership of the data, which achieve deduplication of the cloud data. In the integrity verification process, a public audit and proxy re-signature methods is used to verify data integrity. By comparing similar scheme, our scheme has higher efficiency in data deduplication and integrity verification.

## ACKNOWLEDGEMENT

The project has been a lot of work, but we couldn’t have done it without the support and guidance from some very important people. We want to thank Mr. Dhiraj Birari Sir our project guide for all their help they provided us essential information that was needed to complete our task successfully.

Thank you also goes out to our parents and friends who were there every step of the way.

## REFERENCES

- X. Yang, Y. Li, J. Wang, et al. “Revocable identity-based proxy resiganture scheme in the standard model,” Journal on Communications, vol. 40, pp. 153-162, 2019.
- [2] Q. Wang, C. Wang, J. Li, et al. Enabling public verifiability and data dynamics for storage security in cloud computing,” European Symposium on Research in Computer Security, pp. 355-370, 2019.
- [3] Mark W. Storer, Darrell D. E. Long, Kevin Greenan, Ethan L. Miller, Secure data deduplication, Conference: Proceedings of the 2008 ACM Workshop On Storage Security And Survivability, StorageSS 2008, Alexandria, VA, USA, October 31, 2008.

[4] C.Sasikala<sup>1</sup>,C.Shoba Bindu<sup>2</sup>, Certificateless remote data integrity checking using lattices in cloud storage, Received: 28 March 2018 / Accepted: 11 May 2018 / Published online: 6 June 2018. The Natural Computing Applications Forum 2018.

[5] V.Kher, Y.Kim, “Securing distributed storage: challenges, techniques, and systems,” Proceedings of the 2005 ACM workshop on Storage Security and Survivability, pp. 9-25, 2005.

[6] S.Meena, V.Krithika, TY-JOUR, Data Security using Blockchain Technology, [https://www.researchgate.net/publication/364385200\\_Data\\_Security\\_using\\_Blockchain\\_Technology](https://www.researchgate.net/publication/364385200_Data_Security_using_Blockchain_Technology), October 2022

[7] B. Schroeder and G. A. Gibson, “Disk failures in the real world: What does an mttf of 1, 000, 000 hours mean to you.” Proceedings of FAST USENIX, pp. 1-16, 2007.

[8] X. Lu, Z. Pan, H. Xian. An integrity verification scheme of cloud storage for internet-of-things mobile terminal devices,” Computers Security, vol.92, pp.101-116, 2019.29

[9] Balamurugan N1, Bhuvanesh R1, Lathapriya K M, Sharmasth Vali Y2, Shakkeera L3, Building Secure Clouds by Perpetual Auditing using Blockchain Technology, April 2020.

[10] Y. Fan, X. Lin, W. Liang, et al. “A secure privacy preserving deduplication scheme for cloud computing,” Future Generation Computer Systems, vol. 101, pp. 127-135, 2019.

[11] L. Wang, B. Wang, W. Song, et al. “A key-sharing based secure deduplication scheme in cloud storage,” Information Sciences, vol. 504, pp. 48-60, 2019.

[12] X. Zheng, Y. Zhou, Y. Ye, et al. “A cloud data deduplication scheme based on certificateless proxy re-encryption,” Journal of Systems Architecture, vol. 102, pp. 106-116, 2020.