CHECKING INTEGRITY AND MAINTAIN PRIVACY OF CLOUD DATA USING BLOCKCHAIN

7th Semester Progress Report of Final Year Project

submitted in partial fulfillment of the
requirements for the award of the degree

of

Bachelor of Technology

in

COMPUTER SCIENCE & ENGINEERING

BY

Rohit Patidar (B19CS029)

Viraj Vaishnav (B19CS016)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY MEGHALAYA, INDIA

September 2022

CERTIFICATE

I hereby certify that the work which is being presented in the B.Tech. Summer internship project report titled"**Checking Integrity and Maintain Privacy of Cloud Data Using Blockchain**", in partial fulfillment of the requirements for the award of the **Bachelor of technology in computer science & engineering** and submitted to the department of computer science & engineering of National Institute of Technology Meghalaya, India is an authentic record of our own work carried out during a period from july to august 2022 under the supervision of (**Dr. Surmila Thokchom, Assistant professor**).

The matter presented in this report has not been submitted by us for the award of any other degree elsewhere.

(Signature of Candidate)
Student Name: Rohit Patidar
Roll no: B19CS029

(Signature of Candidate)
Student Name: Viraj Vaishnav
Roll no: B19CS016

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

(Signature of Supervisor)

Dr. Surmila Thokchom,
Assistant Professor
NIT Meghalaya

Head
Department of ComputerScience & Engineering,
National Institute of TechnologyMeghalaya, India

DECLARATION OF ORIGINALITY

I hereby declare that this project work titled "Checking integrity and maintain privacy of cloud data using blockchain" represents my original work carried out as student of the Department of Computer Science & Engineering of National Institute of Technology Meghalaya, India and to the best of our knowledge it contains no material previously published or written by another person unless cited. Any contribution made to this project by others, with whom we have worked at National Institute of Technology Meghalaya or elsewhere, is explicitly acknowledged.

(Signature of Candidate)

Student Name: Rohit Patidar

Roll No.: B19CS029

(Signature of Candidate)

Student Name: Viraj Vaishnav

Roll No.: B19CS016

Place: Shillong

Date: 07/09/2022

ACKNOWLEDGEMENTS

First of all I would like to thank Dr. Surmila Thokchom for giving me this opportunity to work on this project . It has been my immense pleasure to work on this project.

I would also like to thank my friends and my family who have helped me in completing this project.

Table of Contents

1. INTRODUCTION

There are lots of applications implemented in the cloud, user data is also collected centrally and handed over to the cloud. Cloud computing provides users with shared computing resources and storage resources and has multiple deployment models like private cloud, community cloud, public cloud, and hybrid cloud. For users, storing data in the cloud can bring many benefits, such as reducing hardware investment costs, reducing the local storage burden, and supporting remote access. However, while cloud storage brings convenience, it also brings corresponding challenges.

The root problem of cloud data security lies in the trust between the Cloud Service Provider (CSP) and the user. Failure of cloud devices, external attacks, or even the snooping of user data by the CSPs may result in leakage, loss, and corruption of user data. Therefore, the essence of the problem lies in the lack of trust between the two sides. Once problems occur, it is difficult for the challenged party to provide the basis agreed upon by both sides. However, the emergence of blockchain provides a new solution to this problem.

Blockchain is a type of chain structure that combines data blocks in chronological order, and it is a tamper-proof and non forgeable distributed ledger guaranteed by cryptography.

All blockchain participants maintain the blockchain's node information, so all information on the blockchain is open and transparent. Once the information is released, it is permanently retained and cannot be tampered with. With verification and tamper-proof features of the blockchain enable it to act as a reliable third party to address users' concerns in the cloud computing environment, and all results can be published to the blockchain for authentication and maintained by all users of the blockchain. Therefore, integrating the blockchain into cloud computing, using the blockchain's advantages to overcome its limitations of the cloud computing environment, can more effectively and efficiently provide users with data security.

2. OBJECTIVE

With the advent of new technologies there is an incremental growth in the scale of applications deployed on cloud storage, how to ensure cloud data integrity has become a major issue. Although many methods have been proposed earlier, they still have some drawbacks. In this project we overcome the limitations of the previous methods and propose a more efficient and reliable cloud data integrity verification scheme along with maintaining the privacy of the user with the help of an asymmetric algorithm to encrypt data.

3. METHODOLOGY

The proposed data privacy and integrity verification scheme uses blockchain, Cryptography algorithms, Cloud storage service etc. to provide data security and privacy in the cloud environment.

- Key Generation : This phase is performed by the user to generate the user's public key and private key. Using a suitable algorithm user will generate a pair of i.e. public and private.
- Signature Generation : The file to be uploaded is divided into data blocks to construct the signature set; on the other hand, Cuckoo filter is constructed based on the signature set.

- Data Encryption : The data will be divided into data chunks of fixed size i.e. 2 MB that data will be encrypted using an asymmetric encryption algorithm to maintain the privacy of the user. And only the user can decrypt the data and view its original content.

- Uploading the Data : The user encrypts the data and generates an upload request to the CSP. Upon receiving the upload request from the user, the CSP first verifies and if the request is valid then the data is uploaded to the cloud.

- Challenge : For checking the integrity of data the user sends the challenge request to the CSP through the smart contract. The user constructs an audit request and sends it to the CSP for checking integrity of data.

- Proof Generation : After the CSP receives the user's audit request for checking data integrity. The CSP first verifies the user's signature. If valid, the CSP locates the location of each file block, then calculates the corresponding signature with the public key, then constructs the proof, and sends it to the user.

- Verification of Integrity : After the user receives the proof returned by the CSP, the user first verifies the validity of the signature. Then the lookup operation of the cuckoo filter is performed to check whether all signatures exist in the cuckoo filter. If all signatures exist in the cuckoo filter, the data integrity verification is passed; otherwise, the data are compromised.

4. COMPONENTS OF VERIFICATION SCHEME

4.1 Cuckoo Filter : Cuckoo filter is a random data structure with a simple structure and high space efficiency. Compared with the bloom filter, the cuckoo filter has the advantages of good query performance, high space utilization efficiency, and support for reverse operation. It provides two possible storage locations for each keyword, dynamically relocates existing keywords, makes room for new keywords during insert operations, and quickly locates keywords during

lookup operations. the cuckoo filter's expected insertion time complexity is still O(1), although repeated relocations are required. We can calculate the two candidate buckets for a keyword through the following formula:

$$h_1(x) = \text{hash}(x) \quad (1)$$

$$h_2(x) = h_1(x) \oplus \text{hash}(x'\text{sfingerprint}) \quad (2)$$

Cuckoo filter only store fingerprint values instead of original values, so equation (1) can ensure that $h_1(x)$ can also be calculated through $h_2(x)$ and the fingerprint. So, once you know the current bucket k and its fingerprint, you can calculate another bucket by

$$k' = k \oplus \text{hash}(\text{fingerprint}) \quad (3)$$

There are three cases when inserting: the first case is that both buckets are empty, and then a vacant position is randomly selected to insert the item. The second case is that only one bucket has a vacant position so directly inserted the item into this position. The third case is that neither bucket is empty; then randomly choose a bucket, swap the item in the bucket with the item to be inserted, and then relocate the kicked item by equation (2); if the relocated bucket is still not empty, then continue to kick out the original item and relocate it, and repeat this process until all elements are inserted. The cuckoo filter also supports lookup and delete operations. The lookup operation only needs to query whether the item is in one of the corresponding two buckets. The delete operation is similar; remove the item from the corresponding bucket. The fingerprint-based insert algorithm enables the insert operation to use only the bucket's information, without reretrieving keywords. Through equations (1) and (2), the dynamically adding and deleting elements can be realized. The cuckoo filter application, which has the advantages of efficient computation and storage, can reduce the storage and calculation overhead of the verification process to the data integrity verification.

4.2 RSA Algorithm (Rivest-Shamir-Adleman) : It is a public-key encryption technique used for secure data transmission especially over the internet. Transmitting confidential and sensitive data over the internet through this

technology is safe due to its standard encryption method.It was developed by scientist Rivest, Shamir, and Adleman at RSA Data Security Inc. in 1978. In this algorithm, a code is added to the normal message for security purposes. The algorithm is based on the factorization of large number. Large numbers cannot be easily factorized, so breaking into the message for intruders is difficult.

It works on two keys:

- **Public key:** It comprises two numbers, in which one number is the result of the product of two large prime numbers. This key is provided to all the users.
- **Private key:** It is derived from the two prime numbers involved in public key and it always remains private.

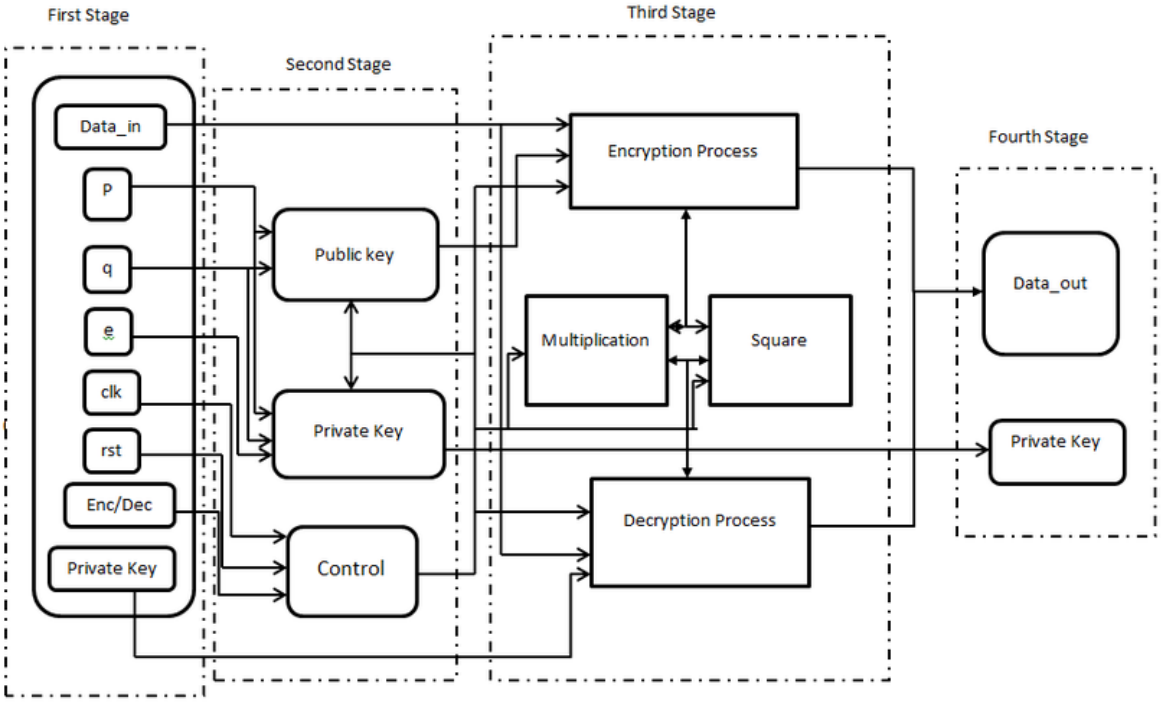

Figure 4.1 : Working of RSA Algorithm

OBJ

5. SYSTEM MODEL

Most of the integrity verification protocols use TPA to communicate the interaction between the user and the CSP, improving data integrity verification efficiency and reducing the user's computing and storage overhead. However, since TPA performs the verification, the reliability of TPA is in doubt, and there are potential threats such as conspiring with CSP to deceive users or fake proof. An ideal public audit institution should have the following characteristics: no additional computing and storage costs, no data privacy disclosure, and most importantly, fairness and justice. Therefore, we introduce blockchain as the third- party audit to replace traditional centralized audit. The system is mainly divided into three types of participants.

5.1 User : The user has the ownership of the data files and the local storage space is limited, so the user chooses to entrust the files to the CSP. For the sake of cloud data security, the user will check the integrity of the uploaded data from time to time.

5.2 CSP : The Cloud Service Provider has a large storage space and strong computing capabilities. It makes money by providing storage and computing services for various users, enabling users to upload and download data anytime and anywhere. But the CSP is only responsible for storing the data and does not ensure data security.

5.3 Blockchain : A third-party audit platform between the user and the CSP is responsible for forwarding and recording the user and the CSP interactions during the data integrity verification process. When users dispute with CSP, the blockchain's records can be submitted to the arbitration institution as valid evidence. All participants jointly maintain the blockchain network, and the behavior of users and CSP is jointly monitored to ensure the system's normal operation.

5.4 Scheme Details : The proposed scheme uses a suitable signature algorithm to sign the files on the user side, the cuckoo filter is also used to simplify the user verification process, and the blockchain network is introduced to record the interaction between the user and the CSP. The scheme mainly contains seven steps to accomplish its objective : Key Generation, Signature Generation, Data Encryption, Uploading the file, Challenging to verify integrity, Generation of proof of integrity, Proof verification.

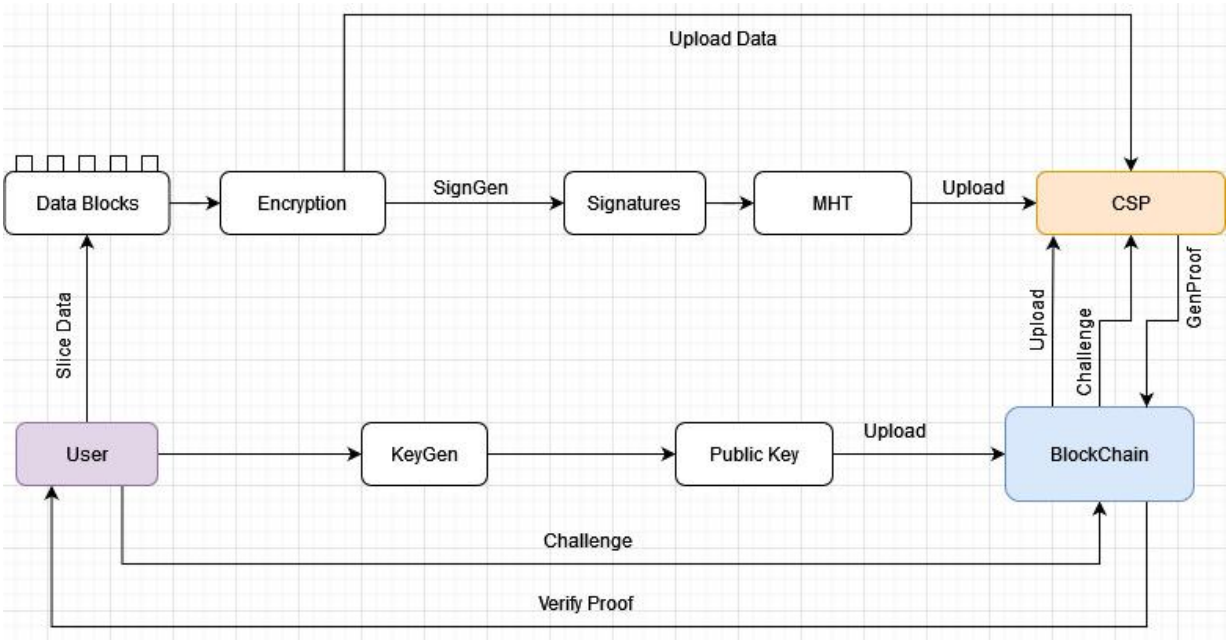

Figure 5.1 : Working of proposed scheme

6. CONCLUSION

7. REFERENCES

1. Gaopeng Xie,Yuling Liu, Guojiang Xin ,and Qiuwei Yang" Blockchain-Based Cloud Data Integrity Verification Scheme with High Efficiency" Hindawi Security and Communication Networks Volume 2021, Article ID 9921209.

2. PengCheng Wei, Dahu Wang, Yu Zhao, Sumarga Kumar Sah Tyagi, Neeraj Kumar" Blockchain data-based cloud data integrity protection mechanism".

3. Pilkington M. Blockchain Technology: Principles and Applications.Social Science Electronic Publishing, 2015.51(07):121-122.

4. S. Nepal, S. Chen, J. Yao, and D. Thilakanathan, "DIaaS: Data Integrity as a Service in the Cloud," in Proceedings of the 2011 IEEE 4th International Conference on Cloud Computing, pp. 308–315, IEEE, Washington, DC, USA, 2011.

5. X. Liang, S. S. Shetty, D. Tosh et al., "ProvChain: Blockchain Based Cloud Data Provenance," Blockchain for Distributed Systems Security, NY USA,2019.

6. P. Wei, D. Wang, Y. Zhao, S. K. S. Tyagi, and N. Kumar, “Blockchain data-based cloud data integrity protection mechanism,” Future Generation Computer Systems, vol. 102, pp. 902–911, 2020.
7. L. Chen, S. Jordan, Y. K. Liu et al., Report on Post-quantum Cryptography, National Institute of Standards and Technology, New York, NY, USA, 2016.
8. H. Zhu, Y. Yuan, Y. Chen et al., “A secure and efficient data integrity verification scheme for cloud-IoT based on short signature,” IEEE Access, vol. 7, pp. 90036–90044, 2019.
9. W. Shen, J. Qin, J. Yu, R. Hao, and J. Hu, “Enabling identitybased integrity auditing and data sharing with sensitive information hiding for secure cloud storage,” IEEE Transactions on Information Forensics and Security, vol. 14, no. 2, pp. 331–346, 2019.