Design of Safety and Integrated Disaster Prevention System Based on Big Data Technology

Yungui Chen¹*, Liwei Tian¹, Lei Yang¹, Longqing Zhang¹

¹Guangdong University of science and technology, Dongguan, 523000, Guangdong, China

*Corresponding author e-mail: teesgx@gdust.edu.cn

Abstract. With the development of Internet technology, with the continuous increase of data volume, it has become more and more difficult to maintain the traditional centralized data storage method. Data is easy to copy, difficult to share, high storage costs, and low data usage efficiency. Further trigger the demand for more efficient data storage technology. This article aims to study the application of blockchain technology in the data security storage and sharing system. On the basis of analyzing the problems of data sharing and cryptography, the functional modules of the data security storage and sharing system are designed. Encryption uses public key encryption algorithm to ensure encryption performance. The simulation experiment results show that the system is effective for file sharing, and the average generation time of the algorithm in this paper is within the controllable range.

Key words: Blockchain Technology, Data Security Storage, Data Sharing System, Data Encryption

1. Introduction

With the rapid development of Internet technologies and the continuous improvement of social knowledge, various application software and electronic devices have generated a large amount of data, and data is playing an increasingly important role [1-2]. Therefore, the demand for data storage by application software is very common. When user data is stored privately in application software, data holders and data managers are separated, which is prone to user data tampering and data leakage problems. The emergence of blockchain technology provides opportunities for existing data storage and sharing problems, and provides reliable solutions to data privacy problems [3-4].

Foreign research on data storage has been carried out relatively early, and a company officially introduced Hadoop in its Nutch project. After years of development, it has gradually matured, and the Hadoop platform has become one of the most widely used distributed databases. Implement a distributed file system called HDFS. For external clients, HDFS can manage files and is highly fault-tolerant, much like a traditional hierarchical file system. It can also be developed on inexpensive hardware while providing high-performance data access. However, HDFS consumes a lot of memory when storing a large number of small files, and the file processing efficiency is not high [5-6]. Ceph is a distributed storage project developed by designers using the C++ language. The project has released the first Argonaut LTS version under the LGPL agreement. Compared with HDFS, Ceph is more
scalable, non-faulty, and performs better when dealing with small files. It is a general storage system [7-8]. Domestic research on distributed storage has achieved results in all aspects. Some researchers have designed a Clover file system that can use directory-based name management methods to improve the scalability of metadata and increase metadata performance by adding a single server [9-10]. In response to the storage efficiency problems caused by the use of Ceph full-copy fault-tolerant technology, some researchers have proposed a combination of full fault-tolerance and error-correcting codes. On the basis of improving the file level, they chose a suitable storage solution [11-12]. Researchers need to think about and continue to study how to ensure secure data storage and provide reliable traceability of access files.

Based on consulting a large number of references on "blockchain technology", "data storage" and "data sharing", this paper designs a data sharing system, which includes three functional modules, namely user registration and login module, and data upload module and data decryption module, and designed for data security storage.

2. Research on the Application of Blockchain Technology in Data Security Storage and Sharing System

2.1. Problems with Data Sharing

(1) The data is concentrated on a third-party organization or server. Some traditional database sharing solutions use third-party organizations or servers to write the entire data sharing process. This part of the organization controls the entire data source, where the data uploaded by the data sharer is stored. Excessive data collection makes data security a factor that needs to be considered in the data sharing process. When these institutions are audited, the data received from the sharer loses security.

(2) The data source is difficult to trace. Some data sharing platforms allow users to exchange data offline. In such a system, users will inevitably upload malicious shared data and attack the security of the entire system. Due to the low traceability of data sources, it is difficult to assume responsibility once a malicious situation occurs.

(3) The lack of a public platform for data sharing. In the past, these data transmission methods were interdependent using e-mail. It is very difficult to ensure the reliability, confidentiality and feasibility of big data. It is not a single standard, because the data recording mode between different data holders is completely different. Statistical methods vary from season to season and cannot be shared. Data-opening countries lack the necessary scientific and technological support and talent guarantee, and cannot strictly comply with national technical standards and regulations.

2.2. Cryptography

(1) Digital signature

In encryption, the blockchain also uses the principle of digital signature to protect transactions from rejection. Among the keys, encryption keys are divided into two types: symmetric keys and asymmetric keys. A symmetric key refers to a key that performs encryption and decryption at the same time, while an asymmetric key refers to a completely different key used in encryption and decryption. In both the message and the conversation process, symmetric ciphers are used in these two parts, that is, a negotiated key is required to encrypt and send the message. When using asymmetric keys, usually follow the "public key password, private key password". It uses the other party's public key encryption to keep the message confidential, and after the other party accepts the ciphertext, it decodes the message through the other party's private key.

(2) Agency signcryption agreement

Strict legal requirements and privacy regulations must ensure that the data in the electronic information system is confidential and unacceptable. Signature as a basic encryption method can effectively solve these problems. Signature is a fairly mature research field. Both digital signature and
public key information encryption can be performed simultaneously in accordance with logical steps, and the cost of calculation and communication is also significantly less than the traditional signature first and then password.

(3) Hash function
Hash is usually displayed as a blockchain indicator, such as transactions and blocks. The hash function mixes inputs of various lengths to create a fixed-length string of letters and numbers. Hash functions also have two important characteristics. The first is determinism, different hash values have different initial inputs; the second is one-way, that is, the hash value can be easily calculated from the initial input. It is not possible to reverse the original input directly from the hash value.

2.3. Data Encryption Algorithm
The elliptic curve cryptosystem is based on the discrete logarithm problem, which can be represented by the addition and multiplication of points on the elliptic curve. The curve equation is as follows:

\[
pxy + py \mod 7 \mod 32 + 1 = 0
\]

Where \( p \) is a very large prime number.

The elliptic curve digital signature algorithm uses an elliptic curve analog digital signature algorithm. The process of the signer using the private key in the elliptic curve cryptosystem to sign the message \( m \) is as follows:

1. Generate a random integer \( k \).
2. Calculate:
   \[
e = H(m), s = k^{-1}(e + dr) \mod n
\]

The calculated \( (r, s) \) is the signer's signature on the message \( m \).

3. Experiment

3.1. Functional Module Design

3.1.1. User Registration and Login Module

1. User registration
When registering, the user only needs to enter the user name, set the password, and enter the password in the confirmation password. The account table does not have the same user name data, so the new user can register successfully. Registering a new user is divided into three steps. The first step is to enter the username so that users and system administrators can easily identify it. The second step is to set a password. The password should be set to 6 to 16 digits to protect the user account. Short passwords are insecure and hard to remember. The third step is to verify the password and verify the compatibility. When new users sign up, if one of the above three steps does not meet the requirements, the reliability of the system will be ignored and guarantees will not be added to the data. Only the required user information is successfully entered into the system and stored in the connection system.

2. User login
Successfully registered users only need to log in to access the main interface of the system. The user first enters their username in the username field. Since the account is limited during registration, the account is also limited at this time. Enter the correct account and notify the user to enter the account that matches the user name. Finally, click the login button and you need to do it. The form submits the account and password in the background to complete the editing. Find the password in the "Database Account" panel through your account. If the user name does not exist in the table, or the password does not match the entered password, the user registration will also be invalid. Once you have the corresponding account and password, that is, the user has successfully registered, the page will enter the main page of the system.
3.1.2. Data Upload Module
Java classes involved in the implementation of the data entry model are mostly classes in Up File Servlet and Multicast Controller. The Upload File Servlet class is primarily responsible for processing data sent from the front of the main system and processing the username of the file currently downloaded from the system session. Excess boundaries include user selection file data and descriptive information for that file. The file information is recorded in the file load class. The Multicast Controller class is primarily responsible for encrypting and installing and editing File Load objects and restoring the results in the next section.

3.1.3. Data Decryption Module
When a data requester decrypts the CT ciphertext received from the network, it must first request a private key from the CA through its own attributes, and then the CA obtains the private key corresponding to the user attribute, and then sends it to it. After the data requester receives the message, it executes the ABE decryption algorithm. Decrypt to obtain the H hash value of the shared data, dataPlace data storage address and access token. The data storage address is based on the data and the access mark, and the data requester retrieves the data from the data storage facility, uses a hash function to slice the data, and decrypts and checks the integrity of the data.

3.2 Data Storage Design

3.1.4. Block Structure Design
The recordable transaction information in Hyperledger is the only blockchain that uses the Bitcoin system for the first time. As we all know, because the blockchain of the Bitcoin system records a large amount of transaction information, it usually refers to the information recorded in the exchange block. Hyperledger Fabric is called Hyperledger because it is formed by connecting the previous and subsequent hash values between each block that records transaction information.

3.1.5. Database Design
Retrieval is a more difficult problem on the blockchain. In order to avoid full-chain retrieval, Fabric maintains a file system for data storage, and maintains a state database, historical database, and index database for each node responsible for recording data. The state database is responsible for recording the final quantity value. The state database can support CouchDB and can query data. Some information about the location block is recorded in the index database. According to the index database, a complete chain scan can be eliminated. The data history database records data change information, and the changed data can be quickly retrieved from the data history database.

4. Discussion
In order to verify the feasibility of the algorithm in this paper, this paper will build a simulation experiment environment. The simulation system simulates users uploading files to the cloud, and grants file access permissions to the requested data through the blockchain. There are 2 servers running public and private chain services, simulating 100 data requesters.

| Number of nodes | Time/s |
|-----------------|--------|
| 7.5             | 7.43   |
| 10              | 7.57   |
| 12.5            | 6.5    |
| 15              | 11.25  |
| 17.5            | 7.83   |
| 20              | 7.86   |
| 22.5            | 7      |
| 25              | 7.65   |
Table 1 and Figure 1 show the time curve from a node requesting public chain data to obtaining data. As the number of requested nodes increases, the node synchronization time is almost unchanged, indicating that the system is effective for file sharing.

Regarding the block generation efficiency, as shown in Figure 2, the simulation data shows that due to the increase in the number of blocks, the output time of a single block has greatly increased. This is because the blockchain consensus algorithm must be synchronized between each member on the chain, and the increase in the total number of blocks will also increase the update time. The calculations in this article use security modeling and data sharing algorithms, so that the average generation and calculation time is within a controllable range.

**Figure 1.** Data request node access performance curve

**Figure 2.** Block generation efficiency graph
5. Conclusions
In recent years, with the rapid development of a new generation of information technology, the value
of data has become higher and higher, which makes it inevitable to open and share data. However, due
to the current security issues of data sharing, the degree of global data sharing is much lower than
expected. In addition to common data security issues, there are two types of data sharing issues that
are difficult to solve with current technology. One is leaking secrets by internal employees, and the
other is leaking secrets by third-party partners. The decentralized application of blockchain technology
allows nodes that do not trust each other to trust each other's data, providing a solution to the above
problems.

Acknowledgments
1. Special project for key fields of colleges and universities in Guangdong Province (new generation
information technology): Research on cloud storage mechanism based on blockchain
technology(2021ZDZX1075);
2. General Social Development Project in Dongguan City in 2020(NO.2020507154401);
3. Cultivation Plan for Major Scientific Research Achievements of Universities: Research on the
Blockchain Fragmentation Mechanism Based on Merkel Tree (GKY-2019CQYJ-13).

References
[1] Mengelkamp E, Notheisen B, Beer C, et al. A blockchain-based smart grid: towards
sustainable local energy markets[J]. Computer Science - Research and Development, 2018, 33(1-2):207-214.
[2] Claudia P, Tudor C, Marcel A, et al. Blockchain Based Decentralized Management of
Demand Response Programs in Smart Energy Grids[J]. Sensors, 2018, 18(2):162.
[3] Mser M, Soska K, Heilman E, et al. An Empirical Analysis of Traceability in the Monero
Blockchain[J]. Proceedings on Privacy Enhancing Technologies, 2018, 2018(3):143-163.
[4] Esposito C, Santis A D, Tortora G, et al. Blockchain: A Panacea for Healthcare Cloud-Based
Data Security and Privacy?[J]. IEEE Cloud Computing, 2018, 5(1):31-37.
[5] Reyna A, C Martin, Chen J, et al. On blockchain and its integration with IoT. Challenges and
opportunities[J]. Future Generation Computer Systems, 2018, 88(NOV.):173-190.
[6] Jang H, Lee J. An Empirical Study on Modeling and Prediction of Bitcoin Prices With
Bayesian Neural Networks Based on Blockchain Information[J]. IEEE Access, 2018, 99:1-1.
[7] Miraz M H, Ali M. Applications of Blockchain Technology beyond Cryptocurrency[J]. Annals
of Emerging Technologies in Computing, 2018, 2(1):1-6.
[8] Ishikawa N, Sugiura S, Hanzo L. 50 Years of Permutation, Spatial and Index Modulation:
From Classic RF to Visible Light Communications and Data Storage[J]. IEEE
Communications Surveys & Tutorials, 2018, PP(3):1-1.
[9] Chen X, Zhou Y, Roy V, et al. Evolutionary Metal Oxide Clusters for Novel Applications:
Toward High-Density Data Storage. 30(3):1703950.1-1703950.9.
[10] Nguyen H H, Park J, Hwang S, et al. On-Chip Fluorescence Switching System for
Constructing a Rewritable Random Access Data Storage Device[J]. entific Reports, 2018,
8(1):337.
[11] Singh A, Garg S, Kaur K, et al. Fuzzy-Folded Bloom Filter-as-a-Service for Big Data Storage
in the Cloud[J]. IEEE Transactions on Industrial Informatics, 2019, 15(4):2338-2348.
[12] Esposito C. Interoperable, dynamic and privacy-preserving access control for cloud data
storage when integrating heterogeneous organizations[J]. Journal of Network and Computer
Applications, 2018, 108(APR.):124-136.