Journal of Computational Science &Engineering 38 (2019) 1003-1009

Available at http://www.asocse.org ISSN 1710-4068

Design of voting system based on blockchain

Mengxue Jin* , Pengxiang Gao

School of Data Science and Software Engineering, Qingdao University, Qingdao Shandong 266071, China

School of Data Science and Software Engineering, Qingdao University, Qingdao Shandong 266071, China

Abstract: **With the growing maturity of blockchain technology, people are looking for application solutions based on blockchain technology to solve the pain points that traditional internet cannot solve. This paper aims to apply blockchain technology to voting scenarios and design a safe and credible voting system that cannot be tampered with. This paper first introduces the basic knowledge of blockchain, including blockchain network layer, storage layer and other related knowledge. Then it focuses on the Hyperledger fabric framework, describes the overall design of the framework, and finally introduces the architecture design of the voting system. And the implementation process and method. In the system architecture design, the Hyperledger fabric is mainly used as the underlying blockchain system. The Hyperledger Composer tool realizes the interaction between the voting service network layer and the underlying blockchain system network. ReactJS serves as the Web service client and ExpressJS serves as the server. The voting system based on blockchain technology has changed the shortcomings of the previous voting system, and is committed to creating a decentralized, data-impossible, secure and credible voting system.**

Keywords: Blockchain; Voting system; Hyperledger Fabric

Received 6 September 2018, Revised 7 October 2018, Accepted 10 October 2018

*Corresponding Author: Mengxue Jin; 1183908270@qq.com

1. Introduction

The earliest electronic voting protocol was proposed by cryptographer Chaum [1] in 1981. The Internet voting scenario was standardized. This scheme does not require a universally trusted third party. The public key cryptography-based technology forms an anonymous channel. Participant information is hidden. The electronic voting system based on blind signature [2, 3] includes two voting protocols: trusted third party and non-trusted third party. Among them, the untrusted third party voting system was designed and proposed by Merrit [4] in 1982. The encryption conversion technology implements the conversion of plaintext messages to ciphertext messages, that is, manipulation of messages by password conversion. Su Yunxue et al. [5] used the ELGamal encryption system to design an electronic voting protocol without a trusted third party. The voter generates a key for encrypting the content of the ballot. In the voting session, the voter uses the knowledge to prove the validity and concealment of the ballot. The user has the advantage of being small in calculation because the number of voting rounds is independent of the number of voters. The FOO electronic voting protocol proposed by Fujioka et al. [6], the earliest large-scale voting protocol based on blind signature, is of great significance.

McCorry et al. [7] first proposed in 2017 to use the Ethereum blockchain to implement voting system self-counting and protect voter privacy. The program enforces agreements through smart contracts and consensus mechanisms, and does not rely on third-party trust institutions. The security system is guaranteed to operate securely, but the program only allows two candidates with certain defects. Dong Youkang et al. [8] designed a board voting electronic voting scheme based on the blockchain alliance chain, which is applicable to anonymous voting scenarios where the board of directors voted to meet the strict requirements of the participants in a specific small scope. The scheme adopts smart contracts instead of traditional trusted third parties, and adopts the digital certificate's identity access mechanism to ensure the identity of the participating voters. The electronic voting protocol guarantees anonymity based on elliptic curve blind signature technology.

Blockchains are divided into public chains, private chains, and alliance chains according to the degree of openness. The public chain is open to the whole network and has no user authorization mechanism. The alliance chain is also called the industry chain. It is used between different organizations. Only authorized nodes are allowed to join the network. The nodes in the network are all in the same institution and are called private chains. Among them, the private chain and the alliance chain are called the license chain, and the public chain is also called the non-licensed chain. In 2015, the Linux Foundation launched the Hyperledger Fabric [9], which is dedicated to promoting blockchain digital currency technology and transaction verification. The blockchain provides a good solution for solving enterprise-level problems. The alliance chain [10] is a blockchain category with strict admission mechanism. Only users with legal identity are allowed to join the alliance chain. The generation of each block is determined by the internally designated pre-selected nodes (recorders). The alliance chain is regarded as a distributed TTP with a specific scope. It has high security and credibility and is used to solve a new solution with certain access mechanisms such as board decision-making [11].

2. Blockchain basics

The distributed structure of blockchain provides reliable guarantee for data storage. It uses digital

2018@ The American Computational Science Society. All rights reserved.

signature algorithm and consensus mechanism to ensure data security, credibility and consistency. Timestamp technology realizes traceability of blockchain data. The verifiability of the voting system provides the basis. This design includes blockchain layer and business layer. The network layer is mainly responsible for building a distributed network of blockchains. The consensus mechanism and smart contracts ensure data security and consistency. The storage layer is responsible for data storage and information is blocked. The form of the link into a chain constitutes a blockchain, and the timestamp technology ensures the traceability of the information, making the data safe and trustworthy.

2.1. Data block

The blockchain storage uses a time-stamped chain structure to quickly verify the authenticity of the data. Each data block is divided into a header and a body. The block header contains the version number (Version), the previous block hash value, the timestamp (Timestamp), the random number (Nonce), the current block target hash value (Bits), and the Merkle root (Merkle-oot). The block body is used to store the transaction details, and the Merkle tree [12,14] is generated by the hash algorithm, and the block header is used for the storage of the Merkle root. The block structure is shown in Figure 1.

2.2. Chain structure

The blockchain has a chain structure sorted by time. The data is sequentially stored in the blockchain according to the chain structure, which ensures the traceability and verifiability of the data on the chain. It is expected that a new block must be generated through a large number of calculations. The hash value of the previous block is stored in the block header. The block header contains the hash value of the parent block, the mining difficulty, the random number, the time stamp, the Merkle root data, and the time stamp is the area. The block information increases the time dimension, which facilitates the traceability and verification of information. Figure 2 shows the structure of the block stored on the chain.

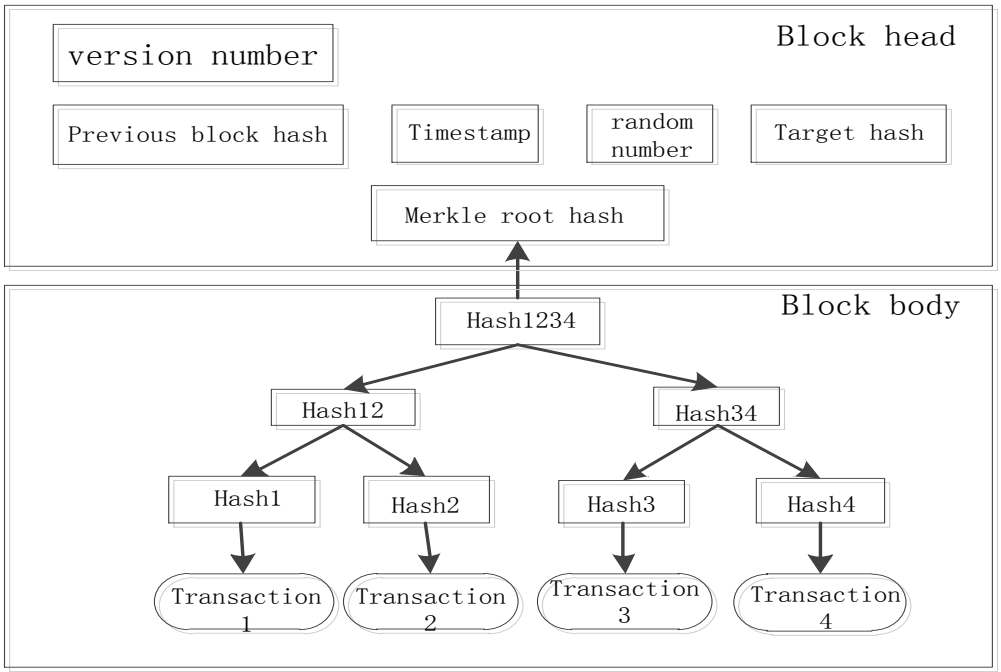


Figure 1. Block structure.

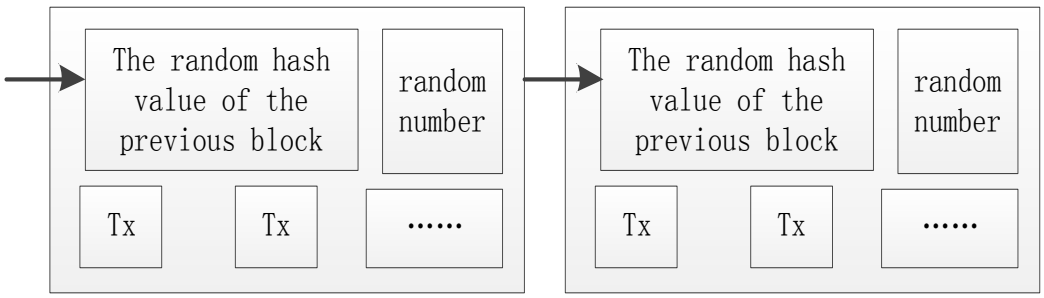


Figure 2. Chain structure diagram.

2018@ The American Computational Science Society. All rights reserved.

2.3. Cryptography

Since entering the information age, cryptography has become more and more widely used in the computer field. The advent of blockchain technology has pushed computers to the new level of cryptography. Cryptography is a very complicated subject. Let's take a brief introduction to the knowledge of cryptography used in the blockchain.

2.4. Hash function

The hash function is also called a hash function [15]. The data of any length is input, and the hash value is used to generate an output value of a specific length, which is called a hash value. The unique one-way and irreversible features of the hash function are very complicated to solve the hash function in reverse. The outside world can hardly launch the original text through the existing hash value, which effectively guarantees the security of the information. The hash value obtained by the hash algorithm operation has a fixed length, and the hash value is much smaller than the input length, and the compression guarantees that the message compression mapping of any length obtains the determined length hash value. Hash functions are highly sensitive, and the hash value of a one-byte change in input data through hashing may be completely different.

The hash function plays an extremely important role in the entire operation of the blockchain, effectively ensuring the orderly operation of data verification, digital signature, data integrity and security, and thus strengthening the data in the entire blockchain system. Safety and integrity.

2.5. Asymmetric encryption

Compared with symmetric encryption, asymmetric encryption uses the same secret key. The public key is divided into public key and private key. The public key can be generated according to the private key. Generally, the public key is processed. The user only needs to perform the private key. Personal private storage to prevent others from stealing. Compared with symmetric encryption, the advantage is that the public and private keys are separated and can be used in insecure channels. The disadvantage is that when generating the private key pair, the encryption and decryption process is often too slow, and the encryption strength is far. Not as good as symmetric encryption.

Asymmetric encryption uses a large number of classical problems such as elliptic curves, discrete logarithms, and proton decomposition to guarantee the problem. Its classical algorithms are RSA, ECC, SM2, etc. [16], in which ECC elliptic curve is one of the most used algorithms in blockchain, and a lot of ECC elliptic curve algorithms are used in Bitcoin, Ethereum and Fabric. encryption. ECC series algorithms are generally considered to have high security, but they are often laborious in the process of encryption and decryption. The RSA algorithm has been confirmed to be broken in the near future. In the choice of asymmetric encryption algorithm, it is better to use the ECC algorithm with high security. Figure 3 shows the asymmetric encryption and decryption process diagram.

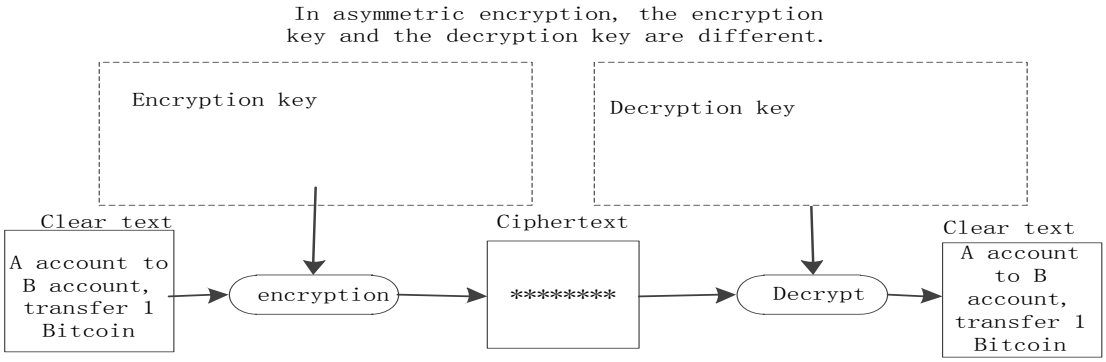


Figure 3. Asymmetric encryption and decryption process diagram.

Table 1. algorithm steps

1. Select a random number $k, k \in [1, n-1]$;
2. Calculation $kG = (x_1, y_1)$;
3. Calculate $r = x_1 \bmod n$; if $r = 0$, return to step 1;
4. Calculation $k^{-1} \bmod n$;
5. $e = SHA1(m)$;
6. Calculate $s = k^{-1}(e + xr) \bmod n$; if $s = 0$, return to step 1;
7. The signature of the message is (r, s). Sends the message m and signature (r, s) to the requesting

2018@ The American Computational Science Society. All rights reserved.

2.6. Elliptic curve digital signature algorithm

The Digital Signature Algorithm (ECDSA) [17] is a type of elliptic curve that is calculated for the correct elliptic curve selected to meet safety conditions. Among the parameters constituting ECDSA, it can be divided into public key private key parameters and global parameters. The formula is its global parameter, and the specific meanings are as follows:

(1) If based on the parameters, if based on F_r, *then* $q = 2^m$, *if based on* F_q, *then* $q = p$, *where* q *is an odd number, the finite field size is represented.*

(2) The representation of FR *is calculated by the element method in the finite field* F_r;

(3) where in the option is a bit string of at least long to $160bit$ *check whether the equation is randomly generated;*

(4) $a, b \in F_q$ *can be used to define the curve equation as follows:*

$$y^3 = x^3 + ax + b, p > 3 \quad (1)$$

$$y^2 + xy = x^3 + ax^2 + b, p = 2 \quad (2)$$

(5) $x_G, y_G \in F_q$, $G = (x_G, y_G), G \in E(F_q)$;

(6) The order in G is a prime number $n, n > 2^{160}, n > 4\sqrt{q}$;

Signature algorithm: Set the message to be signed as m; the global parameter is $D = (q, FR, a, b, G, n, h)$, the signer's public key private key pair is (Q, x).

The algorithm steps for signing are described as follows:

2.7. Hyperledger fabric architecture

Chain code: Fabric's smart contract (Chaincode) [18] responds to transactions issued by the Fabric application, executes code logic, and interacts with the ledger. After the members of the fabric agree on the business logic, the chain code can be added to the network, and each node in the system follows the contract to perform the task. The chain code is executed in an isolated Docker and interacts through the GRPC protocol. Currently supports multiple languages Go, javascipt,etc.

2.8. Blockchain service

The management of distributed ledgers in Fabric is managed by the P2P protocol on HTTP/2. Consensus agreements are pluggable and set up according to specific needs. The ledger consists mainly of two blocks: blockchain and state. Among them, the blockchain is used to record all historical transactions, and the state corresponds to the current account status. Fabric transactions are divided into two types: deployment transactions and call transactions. The deployment of the chain code is deployed to the peer node. After the deployment is successful, the chain code is deployed to other peer nodes. The calling transaction is the RESTAPI provided by the application itself through the REST API provided by the fabric itself, calling a previously deployed function to perform the transaction, writing to the database, updating the book status, and returning the result of the call to the client.

3. System architecture design

In order to achieve a fair and open blockchain voting system, the design architecture diagram is shown in Figure 4.

3.1. Application layer

The application layer adopts the design concept of B/S architecture, and the application layer consists of Client and Server. The Web Client mainly implements the rendering of the front-end interface, and sends the data to the web server, which mainly implements functions such as login, voting, and query, and returns the data obtained from the blockchain layer to the client for rendering.

The Web Client is mainly written in ReactJs. Its outstanding performance and simple code logic are very popular among front-end developers. It has become the mainstream development tool for WEB front-end. React is fast because of its unique feature: virtual DOM, as its name suggests, it is not a real DOM, he does not need the browser's DOM API support, virtual DOM is based on the DOM to create an abstraction layer, the essence of which is a JavaScript Objects, when data and state changes, are automatically and efficiently synchronized to the virtual DOM, and finally only the changed parts are synchronized to the DOM.

Web Server is mainly supported by the Node Express of Node.js. Express is based on callback to combine business logic. It has the advantage of linear logic. Through middleware subdivision and simplified business logic, the perfect combination of router and middleware is realized. The middleware passes complex business logic through a series of processing and feedback to Users, users get clear business logic from middleware.

3.2. Block chain layer

The blockchain layer is divided into a Hyperledger Composer module and a Hyperledger Fabric module. Hyperledger composer primarily allows modeling of business networks, integrating existing blockchain systems and data with programs. Hyperledger Fabric is an important alliance chain framework in the blockchain. It can be used to build the bottom layer environment of the blockchain more quickly and provide a production environment for user data uplink.

Hyperledger's positioning of this project is the upper

2018@ The American Computational Science Society. All rights reserved.

layer of Fabric. Created by Composer, Model File (.cto file), Script File (.js file), ACL (.acl file), Query File (.qry file), and finally Fabric is presented to The user is the processed business network file (.bna file). Composer provides developers with two components, the UI Playgroud and the development toolset. UI Playgroud is mainly used for the display of local fabric networks. In addition, Composer provides developers with a simple set of development tools to facilitate the development of the Hyperledger framework. Hyperledger Composer deploys to the business logic diagram of Fabric, as shown in Figure 5.

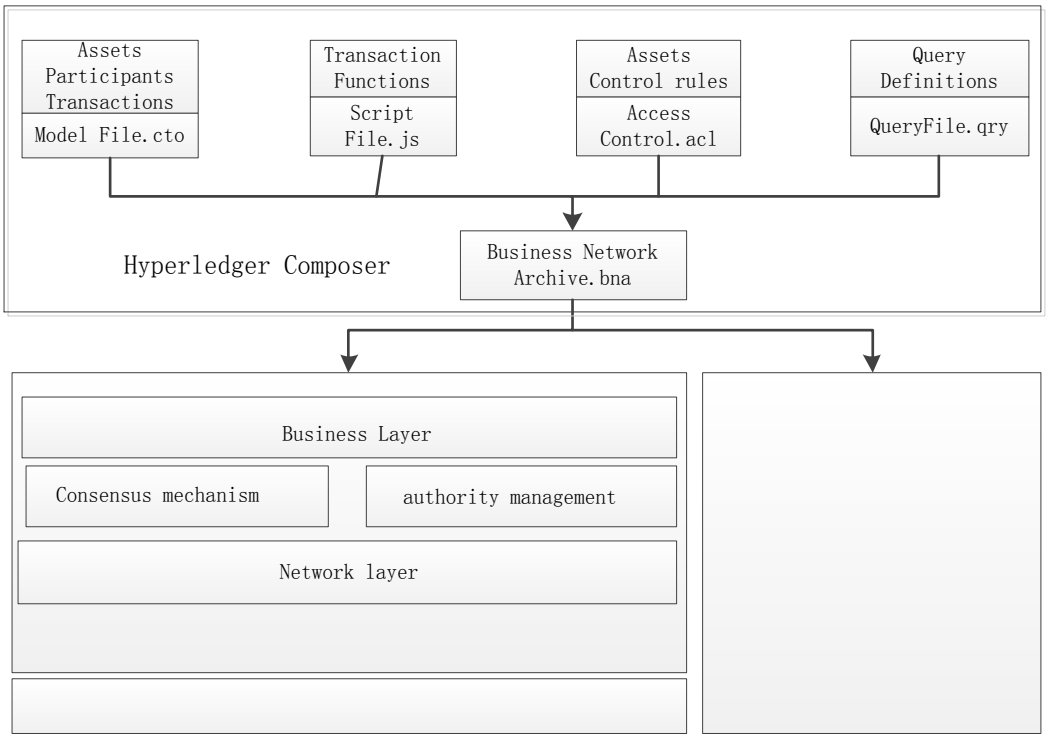
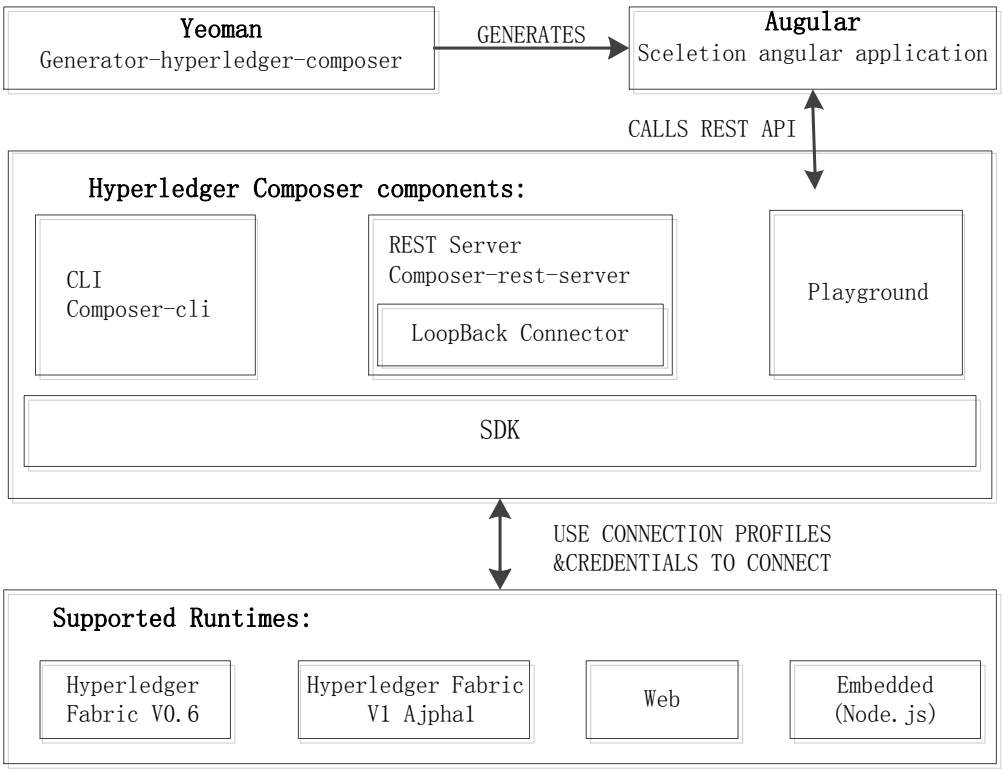


Figure 4. System architecture diagram.



2018@ The American Computational Science Society. All rights reserved.

Figure 5. Composer business logic diagram.

As can be seen from this figure, the middle layer (play grouand, etc.) is called by the front-end restAPI interface, and then the middle layer uses the SDK to interact with the back-end real fabric network. The next solution is to establish the Restful interface. The SDK will be automatically loaded in the corresponding program. The Restful interface is used in this system to use the Rest-Server tool to provide the required support.

After the interface debugging is completed, after the Fabric Client receives the information, the data initiates the transaction proposal to the endorsement node for endorsement, the endorsement result is returned to the sorting node for sorting, the sorted result is broadcasted to other nodes, and finally, the accounting node goes to the state. The database performs a write operation, returns the entire submitted transaction to the client, informs the transaction to confirm and returns the hash value of the transaction to the client user.

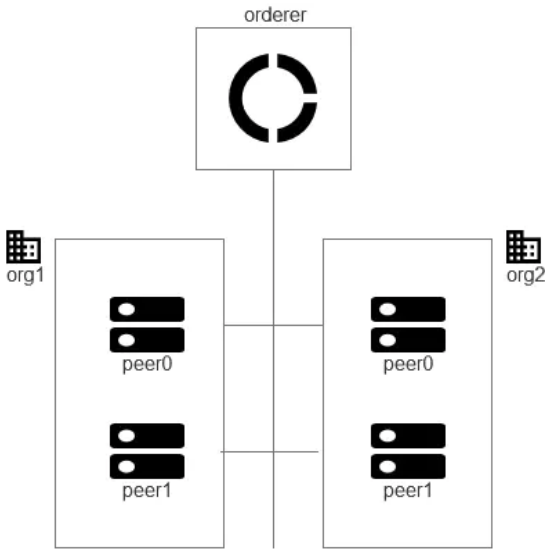


Figure 6. System node example diagram.

3.3. System architecture implementation

This system uses the blockchain as the implementation of the underlying system, because the blockchain technology can still be regarded as a relatively new "information science and technology". The application of these technologies is often not easy to understand, and the corresponding tools are relatively few. The documentation is not comprehensive enough and is still in the stage of continuous development. Before implementing this voting system, building a blockchain system is a top priority. In order to proceed from the practical application, to meet the application of the voting system, and to target the production environment, this deployment will adopt multiple nodes for deployment. In the selection of the number of nodes, four peer nodes, one orderer node, and two org modes are deployed to realize a single-node multi-node production environment. As shown in Figure 6, this is an example of the system node.

3.4. Business logic layer design

In the user voting module, the user selects an object to vote, and clicks a button to vote on the object to be voted. The voting module is one of the important modules, which will involve the blockchain layer consensus mechanism, verification mechanism, digital signature and other processes.

After the user votes, the data will be packaged and traded in the blockchain, and the hash value of the user transaction will be returned, and the hash value will be returned to the front-end interface, so that the user will generate a unique one after each voting. The ID is displayed. Because of the existence of this voting data in the blockchain, the stored procedure performs a large number of hash calculations, digital signatures and other processes, achieving the characteristics of voting that cannot be tampered with. The voting process is as follows Figure 7:

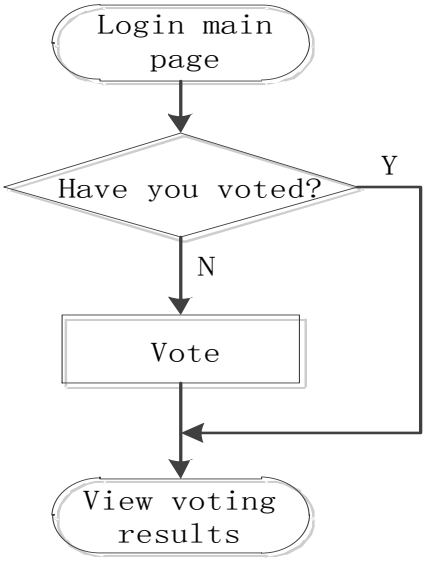


Figure 7. Voting flow chart.

4. Conclusion

This thesis aims to combine the blockchain technology with the voting system, make full use of the safe and reliable characteristics of the blockchain technology, and design a new voting system with perfect data, fair results and fair voting to ensure efficiency and credibility. The main research content of this paper is how to develop a safe and efficient electronic voting system using blockchain technology. Space and location are unrestricted, saving costs and avoiding wasting resources. Moreover, the manual counting of votes is eliminated, which not only improves efficiency and accuracy, but also allows users to know the progress of voting in time. For example, there may be problems such as tampering with data, possible hacking of the system, disclosure of user

2018@ The American Computational Science Society. All rights reserved.

information, and inability to verify the accuracy of voting information. Therefore, the blockchain technology is used to ensure the fairness and security of the voting system. Use Fabric as a framework to develop a perfect voting system.

References

[1] D. L. Chaum. Communications of the Acm, 2 (2003) 84-88.

[2] J. W. Zhang, Z. J. Li, X. Liu. Chinese Journal of Scientific and Technical Papers, 8 (2015) 980-983.

[3] Z. H. Wang, T. Z. Si, Z. B. Li. Computer Engineering and Applications, 23 (2003) 148-150.

[4] M. J. Merritt. Cryptographic protocols[C]//The Fourteenth ACM Symposium on Theory of Computing. 1982: 383-400.

[5] Y. X. Su, H. J. Yan, Y. F. Zhu. Computer Engineering, 11 (2004) 96-97.

[6] Fujioka, T.Okamoto, K. Ohta. Proc Auscrypt 92 Gold Coast Queensland Australia Dec, 1992, 718: 244-251.

[7] S. F. Pshahandashti, F. Hao. A smart contract for boardroom voting with maximumvoter privacy[C]// Financial Cryptography and Data Security, 2017.

[8] Y. K. Dong, D. W. Zhang, W. Han. Journal of Network and Information Security, 2017.

[9] Hyperledger-fabricdocs-documentation [EB/OL] http://hyperledger-fabric.readthedocs.io/en/latest/.

[10] Kiayias, H. S. Zhou, V. Zikas. Fair and robust multiparty computation using aglobal transaction ledger[C]// The Annual International Conference on the Theory and Applications of Cryptographic Techniques. 2016 705-734.

[11] X.Shen, Q. Q. Zhai, X. F. Liu. Journal of Network and Information Security, 11 (2016) 11-20.

[12] R. C. Merkle. Protocols for public key cryptosystems//IEEE Symposium on Security & Privacy DBLP[C].1980, 122-134.

[13] R. C. Merkle. Adavances in Cryptolo gycrypto[C]. 1987, 293(1):369-378.

[14] M. Szydlo. Interenation Conference on the Theory and Applications of Cryptographic TechniqueSpringer Berlin Heidelberg[C]. 2004, 541-554.

[15] National Institute of Standards and Technology (NIST), Secure hash standard.Federal Information Processing Standards Publication (FIPS PUB)180, May 1993.

[16] C Paul. Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. Springer Berlin Heidelberg, 1999, 1109:104-113.

[17] H. Q. Wang, L. J. Zhang, J. X. Zhao. Computer Engineering and Design, 7 (2015) 1819-1822.

[18] H. Watanabe, S. Fujimura, A. Nakadaira. IEEE International Conference on Consumer Electronics (ICCE) 2016:467-468.

2018@ The American Computational Science Society. All rights reserved.