Blockchain Information Tamper-Proof Algorithm Design Based on Searchable Encryption

Pingfan Jia and Junhai Cao(✉)

Army Academy of Armored Forces, Beijing, China
caojunhai@163.com

Abstract. In view of the problem that the traditional blockchain platform data protection mechanism is not perfect, the blockchain information is easily tampered with and lost. In this study, a searchable encrypted protection scheme for blockchain information was designed to prevent information tampering. The scheme model and the specific algorithm are first introduced in detail. The complexity and security of the scheme are then analyzed. Based on the above schemes and algorithms, a data sharing system based on Fabric alliance chain platform data is constructed with blockchain searchable encryption, and the overall architecture and functional partition of the system are designed in detail. Finally, in order to test the effectiveness of the system, the test environment and the experimental environment will be built to test and encryption, and the overall architecture and functional partition of the system are designed in detail. Finally, in order to test the effectiveness of the system, the test environment and the experimental environment will be built to test and algorithms, the proposed scheme are more time efficient and significantly improved.

Keywords: Blockchain · search for encryption · privacy protection Introduction

1 Introduction

In recent years, blockchain has been widely used because of its unique traceability and non tamperability. It is a popular platform for data storage and sharing. It is a key technology in smart city, medical treatment and e-government. However, during the initial design of the current blockchain, there are some defects such as old-fashioned data organization, single information retrieval and lack of privacy protection, which are no longer suitable for the current massive data security protection and retrieval needs. Therefore, only by strengthening the search encryption technology can we meet the safe storage of data and achieve efficient retrieval. For the searchable encryption technology of blockchain data platform, scholars and experts have conducted a lot of research and achieved some research results. Fang Guoqiang [3, 4] built a blockchain based anti privileged account tampering audit system, through which various security threats and attacks were found in time, ensuring the normal operation of the business and data security of the audit department; Wang Jinmiao, Xie Yongheng and Wang Guowei [4, 9] proposed

Z. Zeng et al. (Eds.): ECIT 2022, AHE 11, pp. 681–692, 2023.
https://doi.org/10.2991/978-94-6463-005-3_70

a blockchain privacy protection and access control method based on attribute based encryption. The application of this method further protects the privacy of the blockchain and realizes the effective control of visitors; Wang Ruijin, Tang Yucheng and Zhang Weiqi [4, 10] proposed a privacy protection scheme for the Internet of vehicles based on homomorphic encryption and blockchain technology. Compared with the traditional Internet of vehicles, this scheme is better, effectively protects users' privacy and has stronger security. Based on this, combined with the experience of the above scholars and aiming at the problems existing in the traditional blockchain, this study proposes a searchable encryption scheme based on the blockchain data, and constructs a searchable encryption blockchain data sharing system based on the fabric alliance chain platform data and the corresponding searchable encryption algorithm. Through this system, the safe storage of data is improved to achieve efficient retrieval.

2 Design of Searchable Encryption Scheme for Blockchain Data

2.1 Scheme Design

The basic framework of searchable encryption designed in this study is shown in Fig. 1. The scheme architecture mainly includes three parts: data owner, data provider and search user [14]. The job of the data owner is to encrypt the data and establish a secure index, and then upload the data to the data provider.

The specific operation steps of the scheme are as follows: firstly, the data owner encrypts the data through the private key and encryption algorithm, establishes the security index at the same time, and then transmits the encrypted data and security index to the service provider; When users retrieve documents, they need to use the private key to calculate the search voucher for the search keyword, and then send the search voucher to the data provider to make a search request. Finally, they search the ciphertext data for users through the search voucher, and return the results to the requesting user.

2.2 Encryption Algorithm Design

First, two dictionaries are constructed. DC0 is the old dictionary and DCN is the new dictionary. A stage or cycle is formed through the update operation, which is expressed

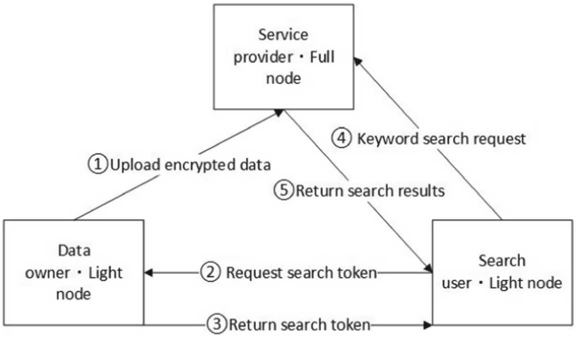

Fig. 1. Basic framework of searchable encryption.

as epoch. DLS uses F as random function and ske = (Gen, Ene, Dec) as private key encryption scheme [11].

The algorithm is constructed as DLS = (Setup, Token, Get, UToken, Put, Rebuild),and the specific algorithm is expressed as.

2.2.1 Initialization Algorithm

The basic principle of Setup algorithm is to take the security parameter k and multi mapping MM as inputs, and adopt two dictionary structures of old dictionary DC0 and new dictionary DCn [15]. The specific algorithm steps are as follows:

```
(K,st, EMM)←DLS.Setup(1k,MM)
(K1,K2)←{0,1}k
Initialization: empty set Se and four empty dictioncaries
$DC_0^{st}$,$DC_n^{st}$,DC0andDCn
foralli∈[#MM]do
return(K,st,EMM)
//among K = (K1.K2).st = (versiong.Se,$DC_0^{st}$,$DC_n^{st}$
)andEMM=(DC0,DCn)
```

2.2.2 Token Generation Algorithm

The Token algorithm takes key K, state st and a label as inputs to generate the token corresponding to the label to be searched.

```
(st’.tk)←Token(K,st,$\ell$)
parseK=(K1,K2)andst=(versiong,Se, $DC_0^{Cst}$, $DC_n^{st}$)
ifDCst n[$\ell$]$\neq\perp$then
return(st’,tk)endfor
```

2.2.3 Query Algorithm

The Get algorithm is also the Query algorithm to obtain the results of the query. The token output by the Token algorithm and the encrypted data structure EMM output by the Setup algorithm are used as inputs. The key codes are:

```
Result←Get(tk.EMM)
parsetk=(otk,ntk)
Instantiation: an empty set Result
returnResult:
```

2.2.4 Updating Algorithm

The main purpose of Utoken algorithm is to update token, which takes key K, state st, operation type op, label ℓ and value v as inputs [1]. Where counter represents the number of times the label is added to the new dictionary DCn, and the version number version is constantly updated. The key code of the algorithm is:

```
(st,utk)←UToken(K,st(op,$\ell$,v))
parseK=(K1,K2)andst=(versiong,Se, $DC_0^{st}$, $DC_n^{st}$)
else if $DC_0^{st} \neq \perp$then
Return(st',utk)
//the up dated state st'=(versiong,Se, $DC_0^{st}$, $DC_n^{st}$)and utk=(tkv,1,tkv,2),v∈ν
```

The work content of algorithm 5 is to execute the Update operation. The utk obtained by algorithm 4 and the encrypted data structure EMM generated by algorithm 1 are used as the input. The server realizes the update operation by adding the updated token to the new dictionary DCn and the output is the updated EMM.

Some key codes:

```
put(utk,EMM)
parseEMM=(DCo,DCn)and utk=(tkv,1,tkv,2),v∈ν
for all∈v do
returnEMM'
```

2.2.5 Reconstruction Algorithm

The Rebuild algorithm is divided into two parts: client and server. After merging them, C and S are used to represent the operation at the corresponding end [5]. The client input is a K and a state st, while the server input is an encrypted data structure EMM. The key codes are:

```
(st,EMM)←Rebuildc,s((K,st),EMM).
//Where C represents the Client and S represents the Server
C:parseK=(K1.K2)andst=(versiong.Se, $DC_0^{st}$, $DC_n^{st}$)andEMM=(DC0,DCn)
For all i∈[count0]do
end for
```

The above is the specific steps of the six basic algorithms of this scheme, which can resist the attack of snapshot opponents. Through the above algorithm flow and code calculation, the scheme design can be realized, and its application to the blockchain system can effectively solve the problem of searchable encryption scheme of blockchain data.

2.3 Theoretical Analysis

2.3.1 Complexity Analysis

For complex queries, this scheme is the size of the mapping set of the query label ℓ [12] when there is no reconstruction operation; O (#MM [ℓ]), when it is in dynamic update status, the query time complexity is expressed as:

$$O\{\#MM[\ell] + del(\ell, e)\} \tag{1}$$

In the above formula, del (ℓ.e) is calculated from the last reconstruction cycle. The sum of the number of recently searched labels ℓ deleted and the size of all existing labels within the space complexity is:

$$\mathrm{O}\{\sum \ell \in \mathrm{LMM}(\#\mathrm{MM}[\ell] + \mathrm{del}(\ell, \mathrm{e})\} \tag{2}$$

The spatial complexity of searching and updating tokens in this scheme is O {#mm [ℓ] + del(ℓ,e)} and O (#v) [2]. Thus, the complexity of reconstruction operation is expressed as:

$$\mathrm{O}\{\sum \ell \in \mathrm{LMM}(\#\mathrm{MM}[\ell] + \mathrm{del}(\ell, \mathrm{e})\text{-}\mathrm{up}(\ell, \mathrm{c})\} \tag{3}$$

In the above formula, #up (ℓ, e) is the number of times the label ℓ was updated in the last update operation.

2.3.2 Safety Analysis

In order to ensure the security of snapshots, the internal state of the Rebuild algorithm process should be kept confidential [6]. For all op ∈ {edit+, edit−}, the leakage function of its reconstruction process is:

$$\ell\mathrm{R}(\mathrm{MM}) = \left(\#\mathrm{del}_{\ell}^{0}\right) \quad \ell \in \mathrm{Se}$$

In the above formula, $\#\mathrm{del}_{\ell}^{0}$ is the number of index pairs of label ℓ removed from the old dictionary DC0, and Se ⊆ L is the label set searched in the current reconstruction cycle [16].

Through the above complexity and security analysis, it can be seen that the scheme proposed in this study can effectively avoid the risk of data tampering and loss, protect users' privacy and security, resist external attacks, and have certain security.

3 Design of Data Sharing System Based on Blockchain Searchable Encryption

3.1 Overall System Architecture

The research and design system is implemented based on the fabric alliance chain architecture. The system mainly includes three parts: authorization node(AN), control node and edge node [7]. The specific architecture is shown in Fig. 2.

The searchable encryption module in the system architecture designed in this study adopts pluggable mode [13]. It formulates the corresponding serialization rules and index structure for the data, abstracts it into a text structure, and then all searchable encryption schemes for the text can be optional, so it can design customized schemes according to different user needs.

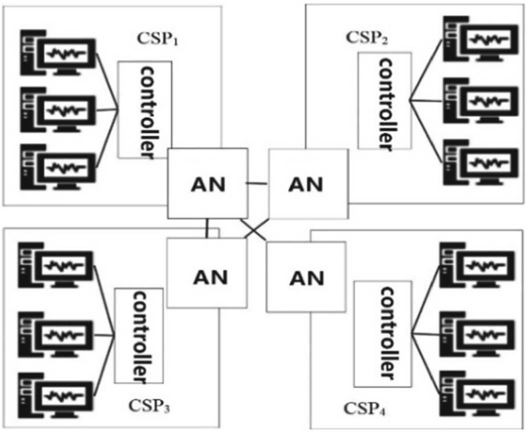

Fig. 2. Blockchain alliance chain framework.

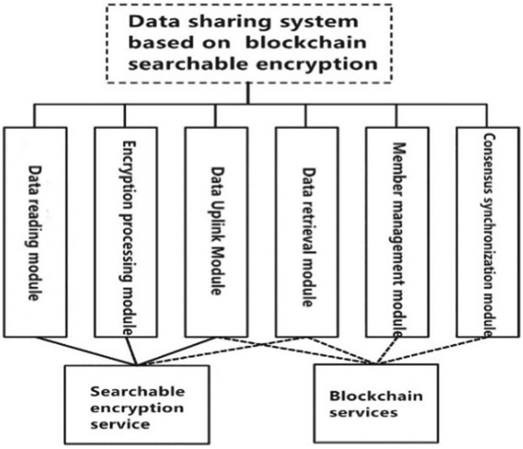
Data sharing system based on blockchain searchable encryption
Data reading module
Encryption processing module
Data Uplink Module
Data retrieval module
Member management module
Consensus synchronization module
Searchable encryption service
Blockchain services

Fig. 3. General function design drawing of the system.

3.2 System Function Design

According to the actual needs of users, this study will design the specific functions of the data sharing system based on blockchain searchable encryption. The specific functions are shown in Fig. 3.

As can be seen from Fig. 3, the system function is mainly divided into six modules, namely data reading, encryption processing, data uplink, data retrieval, member management and consensus synchronization module [8]. Among them, data reading and encryption processing can provide searchable encryption services; Data chaining and data retrieval can provide both searchable encryption services and blockchain services; The member management and consensus synchronization module can provide blockchain services.

4 System Function Test and Experimental Result Analysis

4.1 Test Environment and Experimental Environment

In order to achieve better experimental results, this study selects Java and Go language for application development; The experimental data is stored in the file and the chart is drawn by MATLAB. The experimental environment is selected in Ubuntu16.04 (64 bit) operating system, and the processor platform is Intel (R) Xeon E5-2 630 v4.2 GHz * 40, memory size 250 GB. The specific parameter settings are: -Xms 2 048m, the minimum memory is 2G, and the maximum memory of - Xmx64 000 is 64 G.

For the system function test, this study tests through the Docker container, and establishes one supervision node and three full nodes. The supervision node includes two service providers and one ordinary node, and the three full nodes include two hosted data and one non-existent data. The IP address is LAN.

4.2 Functional Testing

For the function test, this experiment selects the prototype system to design five test cases. After the prototype system automatically parses the files, it obtains the corresponding text data, then classifies the documents according to the corresponding rules, and then constructs the inverted index. The test cases are as follows:

Test case 1: The new user node requests to join the alliance chain, requests the certificate from the CA controlled by the supervision node, the CA verifies the node, and assigns the certificate after passing; The addition of new nodes and the synchronization information speed test of blockchain data;

Test case 2: The data owner makes a query request for the data hosted to the service node and needs to receive the request returned by the service node;

Test case 3: When an ordinary user requests to query the data owned by other nodes, he needs to send a query request to the data owning node, receive the authorization certificate, send a query request to the service node and receive the return result;

Test case 4: when users upload personal data, they need to encrypt the data locally, generate a security index, and upload it to the whole service node;

Test case 5: a user node needs to quit the alliance, the supervision node needs to invalidate its certificate, and the node will not be able to transmit information with other nodes;

After passing the test case, the corresponding functional test results are obtained, as shown in Table 1.

It can be seen from Table 1 that both data owners and custodians can join the alliance and upload data; Data owners and authorized users can search the data; The supervisor can effectively manage the members, which shows that the system function design is feasible and effective.

Table 1. Functional test results.

Function name	User oriented	Available
Members join the Alliance	Data owner and data keeper	Yes
Data search query	Data owner, authorized data user	Yes
Member management	Data Supervisor	Yes
Data upload	Data owner or data custodian	Yes

Table 2. Pairs logarithm of data set and number of keywords.

Pairs (10^6)	1	2	3	4	5	6	7	8	9
Keyword	46580	63920	51108	81635	103426	111651	125564	133155	125491

4.3 Data Set

In this experiment, the data set mainly selects the email data set Enron Email Dataset and the transaction data generated by simulation. The email data set is mainly used for the comparison of searchable encryption experiments under large-scale data, and the generated simulated transaction data is used to test and compare the throughput of blockchain system. Enron Email Dataset uses 500000 emails from 150 employees of a company, which are saved in TXT file format. After data cleaning, taking the mail user as the node and each mail as the edge, 36692 points and 183831 edges are obtained, and the data set is 1.4G.

4.4 Case Experiment Test

In this experiment, the experimental data will be reasonably planned, and the number of document data (w, id) pairs will be changed from 1 * 106 to 10 * 106. The indicators such as Setup time, search voucher generation time, query time, time used for dynamic update and index space will be tested respectively. Each number will be tested 10 times to obtain the average value. After dividing the dataset file, the division content is (Table 2).

After the file division is successful, this study will use this data set as the algorithm to verify the data set, and verify the efficiency of the scheme through experiments on Setup data encryption time, token generation time, query time and update time.

(1) Setup data encryption time

The Setup time is expressed as security index and data encryption. As can be seen from Fig. 4, with the increase of the amount of data, the Setup time of the proposed scheme also increases, which is in direct proportion to each other. The time consumption is small and the efficiency of the scheme is high.

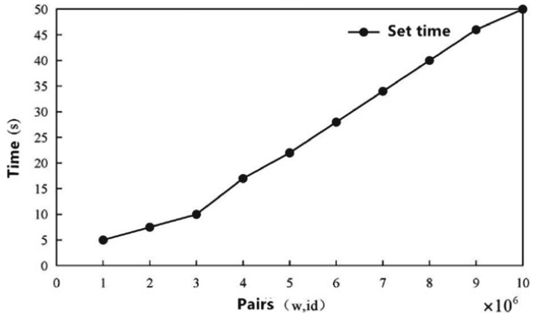

Fig. 4. Data encryption (including generating security index) time

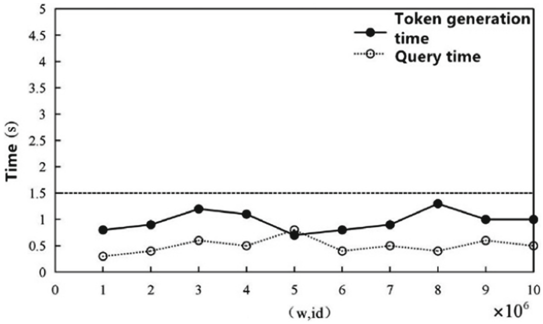

Fig. 5. Search voucher generation time and query time

(2) Search voucher generation time and query time

As can be seen from Fig. 5, the keyword search time is maintained below 1.5 ms, and the consumption time changes slightly with the continuous increase of the data set; And the generation time of search voucher is not affected by the change of data volume, which shows that the two parts consume less time and the scheme efficiency is relatively higher.

(3) Dynamic update time

In Fig. 6, the dynamic update time consumption is maintained around 100 ms, and the amount of data is less than 107, which has little impact on the system delay. With the continuous increase of data sets, the number of updated keywords has no obvious change and takes less time.

(4) Index space

As can be seen from Fig. 7, the index storage space of this scheme increases with the increase of the amount of data, and has dual security protection. In the order of 107, the storage space is maintained below 1G to meet the needs of most users.

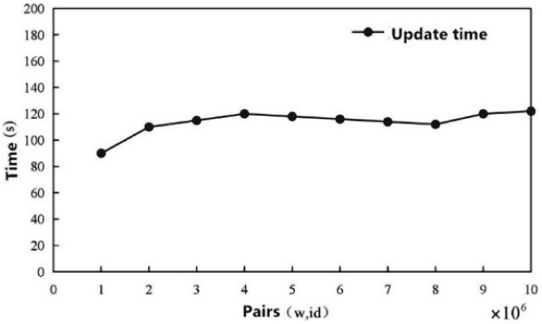

Fig. 6. Time required for dynamic update.

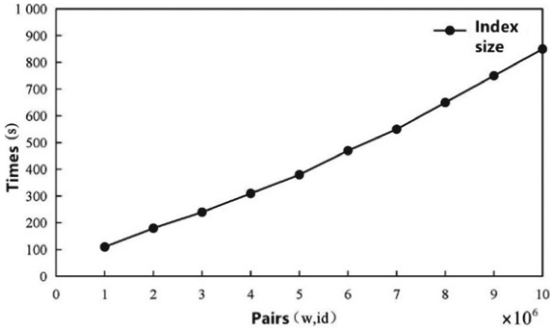

Fig. 7. Index space.

4.5 Comparative Analysis of Classical Searchable Encryption Algorithms

In order to further verify the effectiveness of the scheme proposed in this study, this study compares its Setup time with the classical searchable encryption algorithms Dynrh and Dyn2lev. The data set adopts mail data set (107 magnitude), and the experimental variables are tuple logarithm and keyword number. The comparison results are as in Fig. 8.

It can be seen from Fig. 8 that under the number of different tuple pairs, the setup time consumption of this scheme is the lowest, and this scheme is better than other schemes, indicating that this scheme has higher time efficiency.

As can be seen from Fig. 9, under different keyword numbers, this scheme has higher time efficiency in index construction and data encryption, which has been significantly improved compared with the other two schemes.

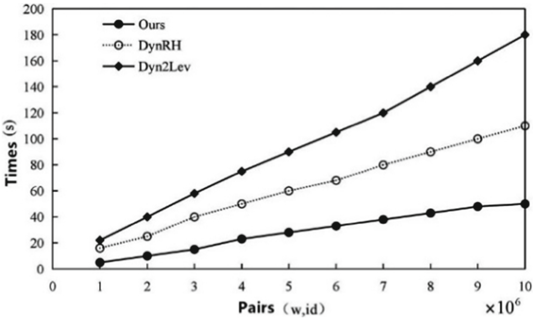

Fig. 8. Comparison of Setup time of searchable encryption schemes under different tuple pairs.

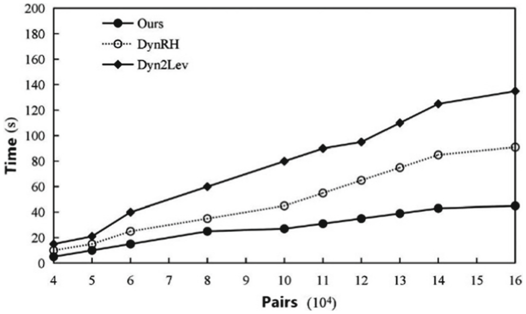

Fig. 9. Comparison of Setup time of searchable encryption schemes under different number of keywords.

5 Conclusion

To sum up, the proposed blockchain information protection scheme based on searchable encryption is feasible and effective, and can effectively prevent information from being tampered. After passing the function test, it is found that the system function design meets the searchable encryption requirements of blockchain data. The experimental results show that the proposed scheme has less time loss and high efficiency. It can be vigorously promoted in the field of blockchain data privacy protection. However, due to the limitation of conditions, there are still some deficiencies in this study, which mainly shows that the system data security query is not perfect. Therefore, in the follow-up research, we will improve this aspect, enrich the query methods of blockchain, and take it as the basis to conduct in-depth research on blockchain data security query and security protection.

References

1. Chen J, Zhang X, Shangguan P (2021) Electronic product traceability system based on blockchain and ORS. Comput Engi Des 42(02):349–355. https://doi.org/10.16208/j.issn1000-7024.2021.02.008
2. Chen L, Xiang F, Sun Z (2021) Research progress of blockchain security technology based on attribute cryptosystem. Acta Electronica Sinica 49(01):192–200
3. Fang G (2020) Blockchain based anti privileged account tamper audit system. Commun Technol 53(04):963–969
4. Huo Y (2022) Design of anti tamper algorithm for blockchain information based on searchable encryption. Autom Instrum, 48–52. https://doi.org/10.14016/j.cnki.1001-9227.2022.01.048
5. Fan J, Chen J, Shen R, Liu Z, He Q, Huang B (2021) SGX based privacy and security protection method for blockchain transactions. J Appl Sci 39(01):17–28
6. Gao H, Li L, Lin H, Li J, Deng D, Li S (2021) Research and application progress of blockchain in the field of data integrity protection. Comput Appl 41(03):745–755
7. Niu S, Xie Y, Yang P, Du X (2021) Cloud assisted attribute base searchable encryption scheme on blockchain. Comput Res Dev 58(04):811–821
8. Niu S, Yang P, Xie Y, Du X (2021) Cloud assisted ciphertext policy attribute based data sharing encryption scheme on blockchain. J Electron Inf 43(07):1864–1871
9. Wang J, Xie Y, Wang G, Li Y (2020) Blockchain privacy protection and access control method based on attribute based encryption. Inf Netw Secur 20(09):47–51
10. Wang R, Tang Y, Zhang W, Zhang F (2020) Privacy protection scheme of Internet of vehicles based on homomorphic encryption and blockchain technology. J Netw Inf Secur 6(01):46–53
11. Wang H, Liu Y, Cao S, Zhou M (2021) Medical data storage mechanism integrated with blockchain technology. Comput Sci 47(04):285–291
12. Wang N (2021) Research on fairness of searchable encryption based on blockchain. Instrum user 28(03):13–18
13. Xu Z, Wang C, Zhu X, Zhu Y, Chen R (2021) Searchable encryption scheme supporting access control based on blockchain. Radio Commun Technol 47(03):271–276
14. Yan XX, Yuan X, Tang Y, Chen Y (2020) Attribute based search encryption scheme based on blockchain and supporting authentication. J Commun 41(02):187–198
15. Zhang X, Sun LL (2020) Research on cloud storage and sharing of blockchain ciphertext re encrypted by attribute agent. J Syst Simul 32(6):1009–1020
16. Zhu P, Hu J, Lv S et al (2021) Research on social network privacy data protection method based on blockchain. Inf Sci 39(3):94–100