

Study on Blockchain-based data sharing security

Yilun Su^{1,*}

¹School of mathematics and statistics, Hunan Normal University, Changsha, China

*Corresponding author: sumanhong@szpt.edu.cn

Abstract. Within the context of the big data age, data sharing is gradually rising with the embodiment of data value. Data value can increase through sharing, but there are security problems during the period of data sharing such as centralized deployment, malicious theft, and tampering, which greatly affect the security of data. Aiming at the common privacy leakage problem during data sharing, this research builds a data sharing platform on the chain based on blockchain technology, and combines the function encryption technology and zero-knowledge proof technology to realize the sharing of verifiable computing results, and proposes a sharing model. The purpose of this model is to enable the data owner to control the data sharing, so as to guarantee the security and privacy of the data while sharing, ensure that the original data is not leaked, and realize the availability and invisibility of the data. In addition, it is necessary to ensure the reliability of data processing results, eliminate the risk of unreliable data processing caused by original data encryption, and protect the legal right of data users.

Keywords: Blockchain; Function encryption; Zero-knowledge proof.

1. Introduction

In today's society, information technology is developing rapidly and has entered a big data era. In modern life, almost every field has realized Digitization: goods, commerce, services, etc. So a large amount of personal data is collected and stored in the cloud. Personalized services that rely on a large amount of data, such as commodity recommendations, also appear.

At the same time, people's attention to data privacy is also growing with the increase in data collection. In addition, with the introduction of data privacy protection laws and regulations, major companies and service providers are actively exploring ways to not only maintain user trust but also collect data legally and reasonably.

Although users can improve the privacy and security of data through encryption algorithms, the encryption of data will lead to the disappearance of convenient services that are widely used today, because service providers cannot analyze the original data. Therefore, the concept of privacy computing came into being. Its main goal is to realize "data is available but not visible, data is not moving and the model is moving" so that users can save data locally or upload it to the cloud after encryption, and service providers can only obtain the data processing results, thus protecting the original data.

Common privacy computing methods include secure multi-party computing, zero-knowledge proof, and a trusted execution environment. In addition, there are some encryption algorithms that can protect privacy without reducing data availability. Function encryption is one of them. Similar to homomorphic encryption technology, function encryption allows data users to directly obtain information from the ciphertext, and the data owner can accurately control the amount of information obtained by the data users from the ciphertext, which brings new possibilities for data security sharing.

The latter part of this paper is organized in the following order: Chapter 2 reviews the current research status of Blockchain-based data sharing. The third chapter describes the logical structure and operation procedure of the model. Chapter four explains the construction method of function encryption and zero-knowledge proof. The fifth chapter describes the safety analysis and experimental results. The sixth chapter summarizes the whole paper.

2. Research Status

For the past few years, the research focus has shifted to decentralized data sharing with the rise of blockchain technology.

Chen [1] and others recorded each data sharing record and relevant information in the system in the blockchain by taking advantage of the non-tamperability of the blockchain and using the smart contract as the medium for data exchange. However, considering the transparency of blockchain data, the original data is directly exposed on the chain, which makes it difficult for private data to be shared through this scheme.

Liang [2] and others created a user-centered health data sharing model by using the alliance chain. The data collected by the wearable devices and medical devices of users will be synchronized to the cloud for storage. The cloud will transfer the retrieval records and verification of the data to the blockchain for saving to guarantee the integrity of the data, and use the tree-based method to enhance the scalability and performance of the platform as well. However, its assumption that the cloud is trusted greatly weakens the security of the model.

In the blockchain-based electronic health information sharing platform proposed by Theodouli [3] and others, it is assumed that the cloud server is malicious, and comparing the hash value of the data on the chain, it ensures that the malicious cloud server does not tamper with the data. It also provides users with a perfect UI to help them quickly learn about their data access and play the role of audit and supervision. However, it does not consider the risk of data leakage by malicious servers, and it assumes that the network route is maintained by trusted entities, which makes the scheme difficult to be practical.

Azaria [4] and others proposed a decentralized electronic medical record management system. Users can customize access control policies through smart contracts and share their own electronic medical records with different institutions. In addition, the author also designed an incentive system to encourage authorities to participate actively and maintain network stability.

Through observation and comparison of existing data-sharing studies, it is not difficult to find that although there are some research results in this field, there are still many places that can be improved. Therefore, a privacy data security sharing model based on blockchain and function encryption is proposed in this paper aim to achieve more efficient and secure data sharing.

3. Design Ideas

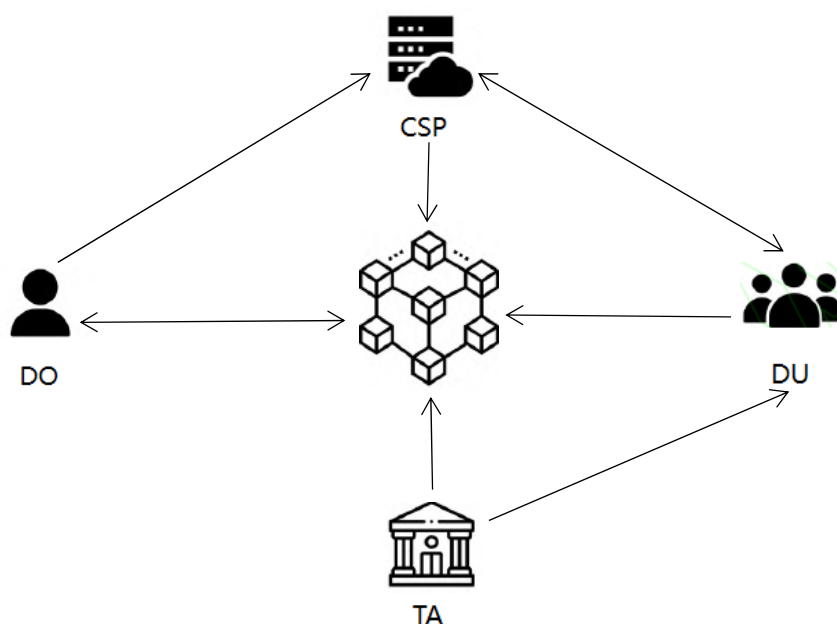


Fig. 1 Model

DO(Data Owner):

For data encryption and uploading, DO obtains the master public key from the chain and runs the function encryption algorithm to obtain the ciphertext CT.

DO uploads the encrypted data CT to the cloud service provider CSP. At the same time, the cloud service provider CSP will monitor the events on the chain to ensure that only the legitimate data user DU can get the encrypted data. Then, DO needs to specify the function that can act on the encrypted data and the corresponding price.

In addition to the DO preset functions, the DU can upload the calculation functions for the encrypted data by itself. Function types include weighted sum, average value, etc.

DU(Data User):

Some users want to use private data.

Data access: when the DU wants to obtain data, it does not directly obtain the original data, but obtains the result of the function processing running on the encrypted data.

The shared original data in the traditional data sharing is changed into the calculation result of the shared data to realize the data privacy protection and sharing.

DU can select preset functions on the chain or provide functions by itself, provide personal public keys to encrypt the function results, and finally call smart contracts through transactions.

TA(Trusted Authority):

Completely credible.Global initialization: firstly, the authority sets the global security parameters, generates the system master public key and master private key, and publishes the system master public key in the blockchain.

4. Algorithmic Knowledge

4.1 Function Encryption

Function encryption is a new public key encryption algorithm developed on the basis of IDB(Identity-Based cryptography) proposed by Shamir [5] and others and ABE(Attribute-Based cryptography) proposed by Goyal [6] and others. It was first formally proposed by Boneh [7] and others in 2011.

Specifically, in traditional public key decryption, there are usually only two results: successful decryption or failed decryption. That is to say, the decryptor either obtains all the plaintext information or cannot obtain any information. However, function encryption allows the encryptor to accurately control the amount of information disclosed to the decryptor. For example, when decrypting encrypted data using the function key that calculates the average value, only the average value of the encrypted data can be obtained, and no additional information can be obtained.

DDH(Decisional Diffie-Hellman assumption):

$$(G, p, g) \leftarrow \text{GroupGen}(l^\lambda) \quad (1)$$

GroupGen is an arbitrary probability polynomial time algorithm. Input a safety parameter l^λ to generate a triplet. G is a group of primes of order p , and g will be generated. (g, g^a, g^b, g^{ab}) and (g, g^a, g^b, g^c) is unrecognizable. $IP - FE = (\text{Setup}, \text{KeyDer}, \text{Encrypt}, \text{Decrypt})$:

(1) $\text{Setup}(l^\lambda, l^l)$ will initialize the entire system and the public key and master private key will be generated, then sets the global security parameters λ and l as parameter inputs, a key pair (mpk, msk) will be returned.

$$(mpk, msk) \leftarrow \text{Setup}(l^\lambda, l^l) \quad (2)$$

$$msk = s = (s_1, \dots, s_\zeta) \leftarrow Z_p^l \quad (3)$$

$$mpk = (h^i = g^{s_i})_{i \in [l]} \quad (4)$$

(2) $\text{Encrypt}(\text{mpk}, x)$ encrypts the data $x = x_1, \dots, x_l \in \mathbb{Z}_p^1$ with the public key to protect the privacy and security of the original data, and returns the ciphertext CT with the master public key and X as inputs.

$$Ct = (Ct_0, (ct_i)_{i \in [l]}) \quad (5)$$

$$Ct_i = h_i^r \cdot g^{x_i}, i \in [l] \quad (6)$$

Where r is a random number and g is the element generated by G , $Ct_0 = g^r$.

(3) The vector $y = (y_1, \dots, y_l)$ that $\text{KeyDer}(\text{msk}, y)$ provides to the data user will generate the corresponding SK, which will be used as the key to obtain $F(x)$, and take the master private key and y as inputs to return the key $sk_y = \langle y, s \rangle$.

(4) $\text{Decrypt}(\text{msk}, Ct, sk_y)$ will take the master public key MPK, the ciphertext CT and the key sk_y as inputs, and return the discrete logarithm based on G . In conclusion, it can be concluded that:

$$\begin{aligned} \text{Decrypt}(\text{msk}, Ct, sk_y) &= \frac{\prod_{i \in [l]} ct_i^{y_i}}{ct_0^{sk_y}} = \frac{\prod_{i \in [l]} (g^{s_i r + x_i})^{y_i}}{g^{r(\sum_{i \in [l]} y_i s_i)}} = \\ &g^{\sum_{i \in [l]} y_i s_i r + \sum_{i \in [l]} y_i x_i - r(\sum_{i \in [l]} y_i s_i)} = g^{\sum_{i \in [l]} y_i x_i} = g^{\langle x, y \rangle} \end{aligned} \quad (7)$$

4.2 Zero Knowledge Proof

Zero knowledge proof technology is a kind of encryption technology, which is used to prove an assertion without disclosing any other information. Although it appeared earlier than the blockchain, it is widely concerned because of the blockchain. Because the transparency of the blockchain allows users in the network to access all data at will, it is easy to cause problems such as data privacy disclosure. The characteristic of zero-knowledge proof makes it an important method to realize the privacy protection of blockchain. Zero knowledge proof includes the following three characteristics:

- (1) Integrity: the legal certificate generated by the honest certifier must be verified.
- (2) Reliability: the evidence is computationally forgeable.
- (3) Zero knowledge: the verifier cannot acquire any additional knowledge except the evidence itself.

This paper uses zero knowledge proof technology zk-SNARK based on Groth16J [8] scheme. The main steps are as follows:

- (1) Trust Setup will generate CRS as a public security parameter. First select $\alpha, \beta, \gamma, \delta, x \leftarrow \mathbb{Z}_p^*$, $\tau = (\alpha, \beta, \gamma, \delta, x)$ and calculate $\sigma = ([\sigma_1]_1, [\sigma_2]_2)$.

$$\sigma_1 = \left(\alpha, \beta, \delta, \{x^i\}_{i=0}^{n-1}, \left\{ \frac{\beta u_i(x) + \alpha v_i + w_i(x)}{\gamma} \right\}_{i=0}^l, \left\{ \frac{\beta u_i(x) + \alpha v_i + w_i(x)}{\delta} \right\}_{i=l+1}^m, \left\{ \frac{x^i t(x)}{\delta} \right\}_{i=0}^{n-2} \right) \quad (8)$$

$$\sigma_2 = (\beta, \gamma, \delta, \{x^i\}_{i=0}^{n-1}) \quad (9)$$

(2) Select $r, s \leftarrow Z_p$ and msk , random variables g and h with the same length as y , and hide them.

$$c_1 = g \cdot y \quad (10)$$

$$c_2 = g \cdot y + h \quad (11)$$

$$v = msk \cdot g \quad (12)$$

(3) Then calculate $\pi = ([A]_1, [C]_1, [B]_2)$.

$$A = \alpha + \sum_{i=0}^m a_i u_i(x) + r\delta \quad (13)$$

$$B = \beta + \sum_{i=0}^m a_i v_i(x) + s\delta \quad (14)$$

$$C = \frac{\sum_{i=0}^m a_i (\beta u_i(x) + \alpha v_i(x) + w_i(x) + h(x)t(x))}{\delta} + As + Br - rs\delta \quad (15)$$

(4) Finally, π is passed into the smart contract for verification.

$$[A]_1 \cdot [B]_2 = [\alpha]_1 \cdot [\beta]_2 + \sum_{i=0}^l a_i \left[\frac{\beta u_i(x) + \alpha v_i(x) + w_i(x)}{\gamma} \right] \cdot [\gamma]_1 + [C]_1 \cdot [\delta]_2 \quad (16)$$

5. Experiment and Safety Analysis

5.1 Safety Analysis

This model is based on the function encryption scheme proposed by Abdalla [9] and others and introduces blockchain and zero knowledge proof technology to realize data privacy protection and secure sharing. The following is the security proof of this model.

The cloud service provider CSP is assumed, to be honest, that is, he will execute the user's commands, but at the same time, he is curious about the user's data. In this model, the original data is first encrypted by the user and then uploaded and stored, and the key is only stored at the TA. So that the privacy and security of the data are protected. In addition, in order to prevent the CSP from providing wrong encrypted data to the data user DU, when the data is uploaded, the data is hashed and signed, and stored on the tamper-proof blockchain. Therefore, DU can easily identify whether the data is tampered with by running Hash algorithm comparison, and the integrity of the data can also be protected. For the availability of data, users can choose to store the data at multiple cloud service providers to ensure the availability of data through redundancy but the costs arising from redundancy need to be borne by users themselves.

The data owner DO can specify how the data can be processed, that is, functions, or customize whether uploading is allowed. Thereby achieving complete control over the use of data. After the DU pays the fee, the TA is responsible for generating the key and the zero-knowledge proof of the validity of the key to ensure that the DU cannot obtain any additional information except the function processing result. The smart contract automatically completes the payment after the zero-knowledge proof is verified on the chain. From the whole network, the data transaction has been completed. DU

can retrieve the key at any time and decrypt the obtained results. Therefore, this model is safe and effective.

5.2 Experimental Result

In this experiment, Ethereum is used to build a local 5-node private network, and the node hosts are all VMware virtual machines of Ubuntu 22.04. I wrote the smart contract Market in solidity [10] language, and the Market contract is deployed in the local private network through Remix. And then achieved the goals of data release on the do chain, data purchase on the Du chain, and verification on the zero-knowledge proof chain. Using rust language to realize the calculation under the chain has high complexity. It is computationally intensive and is not suitable for the algorithmic part of the blockchain, such as function encryption and generation of zero-knowledge proof evidence. The off-chain data is transmitted to the chain through the form of transactions. An on-chain transaction is a transaction event defined in the smart contract, that is, the corresponding action will trigger a pre-designed event to notify the off-chain cloud service provider and other related parties.

In the initial stage of the experiment, after waiting for the network to be stable, the Market contract was deployed in the local private network through Remix. The Gas cost required by the deployment contract and each function are measured, and in Table 1, the results are illustrated.

Table 1. Contract deployed, Gas cost and time cost of each function in the contract

Name	Gas Required	Time Required(s)
Market(Deployed)	2031134	0.45
SetFunctionAndPrice	40103	0.02
ChooseCiphertext	49221	0.02
UploadFunction	930943	0.21

It can be seen that if the DU uses a function preset by the data owner, its gas consumption is only 5% of the user-defined function, and the change of Gas consumed by uploading the user-defined function with the length of the variable is shown in Fig. 2.

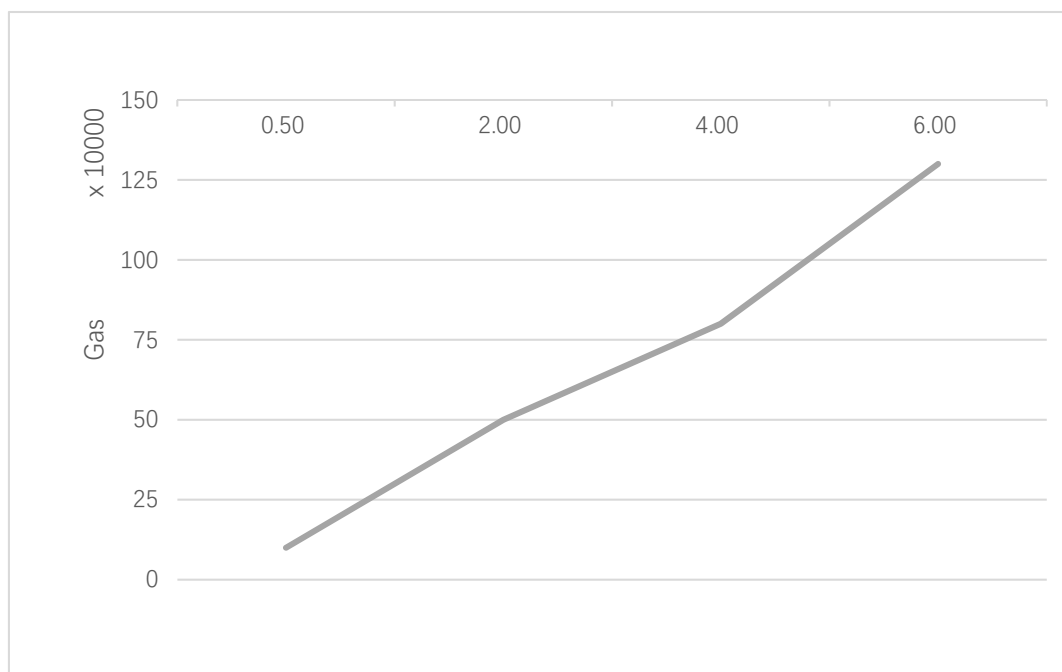


Fig. 2 Upload the graph of Gas required by the custom calculation function changing with the length of the required variable

The main reason for using the custom function with high Gas is that as the length of the requested variable increases, the size of the parameters passed in by calling UploadFn will also increase. This function will write parameters to the chain and notify the cloud service provider through events. In Ethereum, the storage cost is much higher than the calculation cost. Therefore, the DU can select a predefined function as much as possible to decrease the consumption of using data.

6. Conclusion

The privacy data sharing security model based on blockchain and function encryption which is mentioned in this paper realizes the secure sharing of data by using blockchain and function encryption, enabling users to control the amount of information obtained by data users independently by virtue of the characteristics of function encryption, and avoiding the problem of the uncontrollable amount of shared data when using traditional public key encryption. Moreover, the trusted proof of relevant calculation is generated by zero-knowledge proof, and the automatic payment on the chain is realized. It ensures the privacy of data and reduces security problems during data sharing. However, there are also some shortcomings in this scheme. Although the data owner can set the function in advance, as the length of the required variable increases, the resources consumed by uploading the user-defined function will also increase.

References

- [1] Chen J, Xue Y. Bootstrapping a blockchain based ecosystem for big data exchange[C]//2017 IEEE international congress on big data (bigdata congress). IEEE, 2017: 460-463.
- [2] Liang X, Zhao J, Shetty S, et al. Integrating blockchain for data sharing and collaboration in mobile healthcare applications[C]//2017 IEEE 28th annual international symposium on personal, indoor, and mobile radio communications (PIMRC). IEEE, 2017: 1-5.
- [3] Theodouli A, Arakliotis S, Moschou K, et al. On the design of a blockchain-based system to facilitate healthcare data sharing[C]//2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). IEEE, 2018: 1374-1379.
- [4] Azaria A, Ekblaw A, Vieira T, et al. Medrec: Using blockchain for medical data access and permission management[C]//2016 2nd international conference on open and big data (OBD). IEEE, 2016: 25-30.
- [5] Shamir A . Identity-Based Cryptosystems and Signature Schemes[J]. Crypto, 1984.
- [6] Goyal V , Pandey O , Sahai A , et al. Attribute-based encryption for fine-grained access control of encrypted data[J]. ACM, 2006.
- [7] Dan B , Sahai A , Waters B . Functional Encryption: Definitions and Challenges[C]// Theory of Cryptography Conference. Springer, Berlin, Heidelberg, 2011.
- [8] Groth J. On the size of pairing-based non-interactive arguments[C]//Annual international conference on the theory and applications of cryptographic techniques. Springer, Berlin, Heidelberg, 2016: 305-326.
- [9] Abdalla M, Bourse F, Caro A D, et al. Simple functional encryption schemes for inner products[C]//IACR International Workshop on Public Key Cryptography. Springer, Berlin, Heidelberg, 2015: 733-751.
- [10] Page, T. J., Thorson, E., & Heide, M. P. (1990). The memory impact of commercials varying in emotional appeal and product involvement. *Emotion in advertising*, 255-268.