Computers, Materials & Continua
DOI:10.32604/cmc.2020.012146
Article

Tech Science Press

Blockchain Data Privacy Access Control Based on Searchable Attribute Encryption

Tao Feng[1,*], Hongmei Pei[1], Rong Ma[1], Youliang Tian[2] and Xiaoqin Feng[3]

[1]Lanzhou University of Technology, Lanzhou, 730050, China
[2]College of Computer Science and Technology, Guizhou University, Guizhou, China
[3]School of Computer, Xiandian University, Xi'an, China
[*]Corresponding Author: Tao Feng. Email: fengt@lut.cn

Received: 16 June 2020; Accepted: 28 August 2020

Abstract: Data privacy is important to the security of our society, and enabling authorized users to query this data efficiently is facing more challenge. Recently, blockchain has gained extensive attention with its prominent characteristics as public, distributed, decentration and chronological characteristics. However, the transaction information on the blockchain is open to all nodes, the transaction information update operation is even more transparent. And the leakage of transaction information will cause huge losses to the transaction party. In response to these problems, this paper combines hierarchical attribute encryption with linear secret sharing, and proposes a blockchain data privacy protection control scheme based on searchable attribute encryption, which solves the privacy exposure problem in traditional blockchain transactions. The user's access control is implemented by the verification nodes, which avoids the security risks of submitting private keys and access structures to the blockchain network. Associating the private key component with the random identity of the user node in the blockchain can solve the collusion problem. In addition, authorized users can quickly search and supervise transaction information through searchable encryption. The improved algorithm ensures the security of keywords. Finally, based on the DBDH hypothesis, the security of the scheme is proved in the random prediction model.

Keywords: Blockchain; privacy protection; attribute encryption; access control; searchable-encryption

1 Introduction

Blockchain technology, as an undisturbed, chronologically verifiable chain-like storage architecture, provides a new method for data security and privacy protection, and its application in power balance trading platforms has also received extensive attention. For example, Xia et al. [1] analyzed the trading mechanism of the electricity surplus market. In order to make information symmetrical and fair, they designed smart contracts for multi-party bidding of power resources based on blockchain technology, and realized decentralized transaction decisions. Literature [2] summarizes the related applications and research of existing blockchain technology, and focuses on the application of blockchain traceability technology in various fields. According to the above research and investigation, blockchain will help improve solutions in multiple areas such as the Internet of Things, smart cities and supply chains. However, the ledger that records and stores transaction information in blockchain technology is open to any node that joins the blockchain network. Through mathematical analysis of transaction records in the global ledger, attackers may pose a threat to users' transaction privacy and identity privacy [3]. Transaction privacy threats refer to certain threats that contain detailed information about transactions. For example, an attacker can obtain some valuable information through in-depth analysis of a series of transactions of a specific account, transaction details, related accounts, and capital flows. Identity privacy threat mainly refers to the potential threat of the identity of the trader. In addition, on the basis of analyzing the transaction data, the attacker can obtain the identity information of the trader by combining some background knowledge. Jordan [4] uses cluster analysis to analyze transaction data in the blockchain and determine different addresses belonging to the same user. This is a good example of the use of analyzing transaction data to obtain the identity of a trader. In addition, since all transactions performed by users are permanently recorded in the blockchain, once the transaction is implemented, all relevant transaction information will be leaked. In addition, as the blockchain is increasingly used in daily payments, attackers can use off-chain information [5] to infer the identity of the account in the blockchain.

In the traditional blockchain, the user's account transaction information is directly stored in the block without using encryption technology, so the user's account is completely open to all nodes. At the same time, when a user initiates a transaction, the transaction amount in the transaction information is completely disclosed. The verification node on the blockchain performs mathematical analysis on the user's transaction amount and account balance to verify whether the transaction is legal. Although this method realizes the decentralization of the blockchain and cannot be tampered with, the user's account privacy [6] will be completely exposed. In response to these problems, there are some blockchain privacy protection mechanisms. Therefore, Shen Tu et al. [7] proposed a more effective blind signature hybrid scheme based on elliptic curve encryption algorithm m in reference. This scheme is simple and easy to operate, and is usually suitable for various digital currencies, but it is a centralized currency scheme. After that, in the hybrid scheme, Gao et al. [8] uses cryptographic techniques such as blind signatures to protect privacy issues, but this scheme increases computational costs, and the implementation of token processing by a third party inevitably increases additional service overhead. Some scholars also use ring signatures to protect the privacy of the blockchain. For example, Noether et al. [9] proposed an improved ring-based secret transaction scheme, which can hide the amount in reference. In this scheme, a large number of ring signatures are placed in multiple layers of linkable spontaneous anonymous group signatures [10], and its solution can protect identity privacy and transaction privacy. Although ring signature provides strong anonymity, it has three limitations:

1. Its transaction event volume is huge, each transaction event is nearly several kilobytes, which increases the storage space of the entire blockchain record.
2. The inherent disadvantage of ring signatures is that the size of the signature is proportional to the number of participants. Therefore, in reality, each transaction has only a limited number of outputs (for example, by default, each transaction has 4 outputs).
3. The hidden amount increases the difficulty of review, that is, it not only verifies whether a secret cryptocurrency is generated during the transaction, but also determines the additional amount at a specific moment.

Chen et al. [11] proposed an anti-quantum proxy blind signature scheme based on lattice cryptography, which provides user anonymity and non-traceability for distributed applications of BIoT. The proposed proxy blind signature scheme can completely solve the unforgeable problem without authorization to protect user privacy. Chiesa et al. [12] proposed the Zerocash scheme, which introduced zk-SNARK [13], a non-interactive zero-knowledge proof technology in cryptography that converts into digital currency, which can ensure the unlinkability of transactions And confidentiality, while it supports any amount of money. However, the disadvantage of this scheme is that if an attacker obtains the secret data introduced at the beginning, the token may be forged. Subsequently, in order to solve the privacy problem in the public chain off-chain payment protocol, many studies have constructed offline payment protocols, such as two-way micro-payment channels [14] Lightning Network and Spirtes [15]. However, in these schemes, both parties in the transaction must use the relay node to complete the transaction, and the transaction information is public to the relay node, so the privacy of both parties in the transaction will be exposed to the relay node. For offline payment privacy issues, there are some studies, such as the Tumble Bit solution proposed by Alshenibr et al. [16]. This solution can hide payment channel information from relay nodes. In addition, Tumble Bit solutions are generally suitable for compatible Bitcoin systems. But the time and efficiency costs are relatively high. Green et al. [17] proposed the Bolt scheme, which ensures that payments under the same channel are independent of each other, and the payment time does not require block confirmation to reach the second level. If someone pays through the payment channel provided by the payee, the payee will receive a notification. There are still many problems to be solved in the security of payment protocols.

A privacy protection mechanism for blockchain retrieval based on searchable keywords is proposed [18], which realizes the private search of authorized keywords without changing the retrieval order. But the keywords in this scheme are relatively short, they cannot resist collusion, and can communicate privately between nodes. Aiming at data sharing privacy, Do et al. [19] proposed a distributed data storage system using blockchain technology and a private keyword search scheme, which provides authorization for data owners and supports dedicated keyword search for encrypted data sets. However, this program has not yet implemented document revocation and Boolean search. The data storage incentive mechanism of wireless sensor network (WSN) [20], which uses double chains, one chain is used to store the data of each node, and the other chain is used to control data access. In addition, the reserved hash function is used to compare the stored data with the new data block. New data can be stored in the node closest to the existing data, and only different sub-blocks can be stored, which can greatly save the storage space of network nodes. However, the confidentiality of the data was not discussed. Using the DCOMB method, literature [21] proposed a blockchain-based IoT data query model. This model combines the IoT data stream with the timestamp of the blockchain to improve the interoperability of data and the versatility of the IoT database system. The data query model in this solution can quickly query the public key corresponding to the data stream. The query is in a fully encrypted environment, which can ensure the privacy and security of IoT data. However, the data pre-reading process will take extra time, and MySQL query performance will cost more.

In the above scheme, the data privacy problem has not been completely resolved. Therefore, it is urgent to solve its privacy problem. The contribution of this paper is as follows:

1. In this paper, the public key searchable encryption method is used to encrypt the transaction privacy data in the blockchain, which solves the privacy leakage problem caused by the disclosure of all data in the traditional public blockchain, meanwhile, realizes the privacy of protecting the sensitive information of the blockchain transaction.
2. The use of attribute encryption technology combined with secret sharing enables fine grained access control of transaction ciphertext in the blockchain.
3. In addition, the users who have access to transaction can quickly search ciphertext, which realizes the supervision of transaction information.

The Organization of the paper is structured as follows: In Section 1, we introduce the research on the privacy of blockchain; We introduce the basic cryptographic primitives in Section 2; In Section 3, the system model in general and the security model of security requirements are proposed; In Section 4, the specific construction of the scheme is described; In Section 5, we analyze the security of the scheme in detail; We compare the related work in Section 6; In Section 7, we summarize our scheme.

2 Preliminary Knowledge

2.1 Bilinear Mapping

Definition 1: Let G_1 and G_2 are multiplicative cyclic groups whose order q is prime. For a random generator p of a group, there exists a bilinear pair mapping that satisfies the following properties:

(1) Bilinear: for $\forall(P, Q) \in G_1$ and $\forall(a, b) \in Z_q{}^*$ are true.

(2) Non-degenerate: $e(P, Q) \neq 1$.

(3) Computability: For $\forall(P, Q) \in G_1$, there is a polynomial time algorithm for calculation $e(P, Q)$.

2.2 Determining the Bilinear Diffie–Hellman Assumption (DBDH)

Select a generator $g \in G_1$ and choose $a, b, c, r \in Z_q^*$, for $g^a, g^b, g^c \in G_1$, $e(g, g)^{abc}$ and $e(g, g)^r \in G_2$, determine if the relationship between $e(g, g)^{abc}$ and $e(g, g)^r \in G_2$ is equal.

Definition 2: For the arbitrary polynomial probability time algorithm adversary A, the advantage of solving the decision bilinear Diffie–Hellman (DBDH) hypothesis [i] is defined as:

$$Adv_{\mathrm{A}}^{DBDH} = \left| \begin{array}{l} \Pr\left[A\left(g, g^a, g^b, g^c, e(g, g)^{abc}\right)\right] - \\ Pr\left[A\left(g, g^a, g^b, g^c, e(g, g)^r\right)\right] \end{array} \right| \tag{1}$$

If the determined value Adv_A^{DBDH} is negligible, then the decision bilinear Diffie–Hellman hypothesis will be established.

2.3 Lsss Linear Secret Sharing Scheme

A linear secret sharing scheme [22] will be called linear on a group of participants P if the following conditions are satisfied:

(1) The share of the participants for each party comes from a matrix Z_P above.

There is a matrix M with c rows and d columns called the shared generation matrix Π. For all $i = 1, 2, \ldots, l$, the function ρ defines the participant marking of the i-th row of M as $\rho(i)$. When we consider the row vector $v = (s, v_2, \ldots, v_c)^{\mathrm{T}} \in Z_P^n$, which s is the secret that is shared, and $v_2, \ldots, v_c \in Z_P$ is randomly chosen. Mv is the secret share according to the vector Π. The share Mv_i belongs to $\rho(i)$.

(2) Assume that Π is an LSSS for the access structure. Let $S_u \in$ A be an arbitrary subset of authorizations, and $I \subset \{1, \ldots, l\}$. If $\{\lambda_i\}$ is the valid share of any secret S according to Π, there will be a constant $\{w_i \in Z_P\}_{i \in I}$ that makes $\sum_{i \in I} w_i \lambda_i = S$ at this time. Furthermore, these constants $\{w_i\}$ can be found in the time polynomial of the shared generator matrix M.

3 System Model

The block chain data privacy access control system model based on searchable attribute encryption is shown in Fig. 1, which includes four types of participating entities: Data owner, verification nodes, user and miner node, Trading generates is shown in Fig. 2.

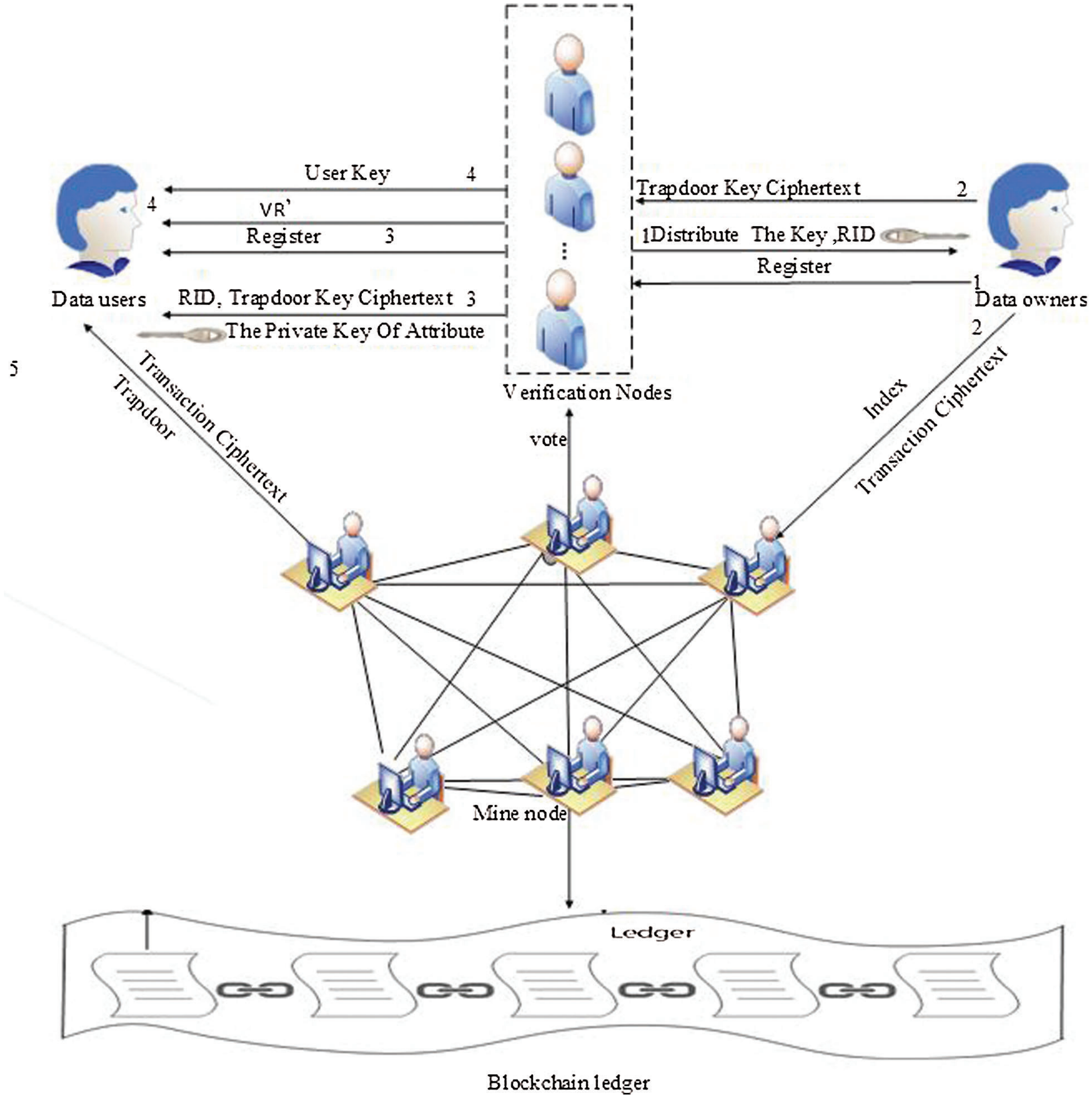

Figure 1: System model

Data Owner: Firstly, initializing generates the index key and the trapdoor key, extracts the keywords of the transaction, then uses the index key to encrypt the index and form the index ciphertext; Secondly, encrypt the trapdoor key to form the trapdoor key ciphertext and share the data. Finally, using the secure signature algorithm signs the transaction and encrypts, meanwhile, data owners appends the indexes keyword to the ciphertext file of the transaction ciphertext. Above all the data owner can be a user on the blockchain for Bitcoin transactions or a miner.

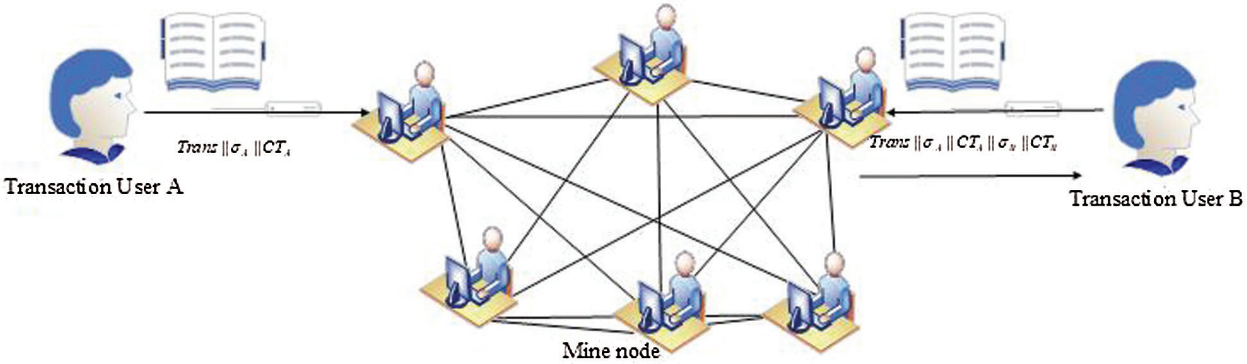
$Trans \| \sigma_A \| CT_A$
Transaction User A
$Trans \| \sigma_A \| CT_A \| \sigma_B \| CT_B$
Transaction User B
Mine node

Figure 2: Trading generates

User: The registration system generates an identity identifier RID corresponding to the real identity and a private key corresponding to the user attribute. In addition, user decrypts the trapdoor key ciphertext and gains the user key to generate a trapdoor, and sends the blockchain to request the transaction ciphertext.

Verification Nodes: Verify the correctness of the user's identity and permissions, and calculate the user's attribute and private key parameters and permission parameters in the attribute collection to distribute the trapdoor key ciphertext, and distribute the user key UK to the legitimate user.

Miner Node: The miner node broadcasts all the transaction information during this period, and each node performs verification and joins the blockchain after verification. The trapdoor and the index sent by the data owner are calculated and matched, then the transaction ciphertext will be sent to the data consumer after the matching is successful.

3.1 Threat Model

The solution proposed in this paper only the Verification Node is completely credible, the private key can be generated and distributed honestly for the user. Most miner nodes are honest but curious. In addition, users may collude to decrypt data that they do not have access to.

3.2 Security Model

The security model refers to the game between the opponent and the challenger. The game is described as follows:

IND-CPA security model

Initialization: Challenger A runs the initialization algorithm to generate the public parameters and master key, and sends the public parameters to the adversary.

Phase 1: The adversary C continually repeats the corresponding set of attributes $S_1, \ldots, S_q$, where none of the attributes satisfy the access structure.

Challenge: The enemy C picks two messages M_0, M_1 and sends them to the challenger A. The challenger A randomly picks a byte $b \in \{0, 1\}$ and encrypts the message M of the access structure, then the challenger A sends the ciphertext to the rival C.

Phase 2: Phase 1 is repeated guess: The guess of the enemy C input b, if the opponent guesses $b' = b$, the enemy C will win the game. The advantage of rival C in this game is defined as $Adv = |Pr(b = b') - 1/2|$.

Definition 1: If the advantage $Adv = |Pr(b = b') - 1/2|$ of the above game is negligible in the time of all polynomials, the proposed solution can be IND-CPA security.

IND-CKA security model

Initialization: Repeat the initialization of the above security model.

Phase 1: The adversary adaptively proposes the following polynomial query.

Hash Ask: The adversary can ask the random oracle H.

Trapdoor request: The adversary can request any keyword trapping.

Challenge: The adversary submits two keywords and gives the challenger C, the limit is that the enemy can't ask for the keyword.

Phase 2: Phase 1 is repeated.

Guess: The opponent A outputs the guess b' of b, if the opponent guesses $b' = b$, the enemy C will win the Game. The advantage of rival C in this game is defined as $Adv = |Pr(b = b') - 1/2|$.

Definition 2: If the advantage of the above game is negligible at the time of the polynomial, the proposed scheme can be IND-CKA security.

4 Specific Construction

In this part, we present the specific implementation process of the blockchain data privacy protection access control method algorithm based on searchable attribute encryption.

a) **Registration**

The user submits a registration application to the system, obtains the identity RID and the user attribute set corresponding to the real identity information, and the data owners (transaction users) register to obtain the key and the identity identifier.

b) **Initialization**

Data owner initialization: Select a group G_0 with the prime number p as the order, generate the group with the element g, select N elements in the limit field, and use the system attribute to form the system attribute set S, and the attributes in S according to the correlation between the attributes. S is divided into x trees, H_i set to the depth of the i-th tree, $H = \max\{H_i\}_{i\in[1,x]}$ defined as the maximum depth in the tree; randomly select vector $U = \left(u_{\mathrm{y}}\right)_{1\leq y\leq x}$ and $U' = \left({u'}_{y'}\right)_{1\leq y'\leq x}$, u_y represents the public parameter corresponding to the y attribute tree, data owner selects a sequence of prime p, and generates a group of G_1, $H_1 : \{0,1\}^* \rightarrow G_1$ is a hash function. Data owner chooses two random numbers η, μ, calculates the public key $PK = \{g, g^{\mu}\}$,and the private key $SK = \eta$ represents the trapdoor key.

The verification node initialization: ${Z_p}^*$ expresses a set of elements in the finite field with p-primitives, from which two random numbers α, β of different sizes are selected, the verification node calculates $PK = \left\{G_0, g, g^{\beta}, Y = e(g,g)^{\alpha}, U, U'\right\}$ and $MK = \{\alpha, \beta\}$ to define a bilinear map $e : G_0 \times G_0 = G_1$.

c) **Transaction generation and signature**

Transaction user A generates transaction information, encrypts its own identity, runs the wallet signature algorithm and signs it with the private key corresponding to the wallet address, then sends it to transaction user B. The user signature is calculated as follows:

$$\begin{aligned} &Trans||\sigma_A||CT_A \\ &Tr_{AB} = Trans||\sigma_A||CT_A||\sigma_B||CT_B \end{aligned} \tag{2}$$

d) **Index generation**

The trader extracts the keyword from the transaction plaintext information, and encrypts the keyword with the index key g^{μ} and the random number τ, μ. The keywords of the transaction information are calculated as follows:

$$I_w = (I_1, I_2) = ({g_1}^{\mu\tau}, e(H_1(w)^{\mu}, {g_1}^{\eta\tau})) \tag{3}$$

e) **Encrypt** $(M, TK, PK) \rightarrow C_M, C_{TK}, VR$ The n' user attribute $a_n{}'$ in the ciphertext policy attribute set H is located in the m' attribute tree, and its depth h', its path is $R_n{}' = (a_{n'0}, a_{n'1}, \ldots, a_{n'k}, \ldots, a_{n'h})$, where $k' \in [0, h]$, $a_{n'k'}$ is the corresponding attribute of the user attribute $a_{n'}$ in the path $R_{n'}$ the layer k'_1, and for the policy attribute $a_n{}'$, according to the mapping p selects its corresponding secret share w_i. The calculation of attribute ciphertext $C_{\mathrm{n}'}$ and policy parameters $C'_{n'}$ are as follows: $C_{n'} = \left(u'_{m'} \prod_{k'=1}^{h'} u_{k'}{}^{a_{n'\mathrm{k}'}}\right)^{w_i}$, $C'_{n'} = g^{w_i}$. The public parameter $u'_{\mathrm{m}'}$ corresponding to the m' attribute tree, $u_{k'}$ indicates the public parameter of the k' layer, the ciphertexts are as follows:

$$C_{TK} = \begin{pmatrix} lE_0 = TK \cdot Y^S, \hat{C} = g^{\frac{\alpha+r}{\beta}}, C = g^{\beta S}, \\ \left\{C_{n'}, C_{n'}{}'\right\}_{a_n \in L} \end{pmatrix} \tag{4}$$

$$C_{TM} = \left(E_1 = MY^S, C = g^{\beta S}, \hat{C} = g^{\frac{\alpha+r}{\beta}}\right) \tag{5}$$

M is the transaction plaintext information, S is a secret value, and E_1 is a partial ciphertext containing the transaction plaintext information M.

f) **Trapdoor generation** $(W', TK, UK, \lambda) \rightarrow T_{W'.}$ In this algorithm, select a random number and calculate the trapdoor:

$$T_{W'} = (T_1, T_2) = \left(\lambda \cdot UK, H_1(W')^{\lambda \cdot TK}\right) \tag{6}$$

g) **Test** $(RID, I_W, T_{W'}) \rightarrow \{C_M\}$ According to the UK execution calculation, the match $e\left(T_2, I_1{}^{UK}\right) = I_2{}^{T_1}$ submitted by the user and corresponding to this algorithm is as follows:

$$e\left(H_1(W')^{\lambda TK}, g_1^{\mu \cdot \tau \cdot UK}\right) = e\left(H_1(W)^{\mu}, g_1{}^{\eta\tau}\right)^{UK \cdot \lambda} \tag{7}$$

If the user's search keyword is the same as the search keyword contained in the index, the equation will establish. The blockchain returns the result to the user, otherwise it returns an empty set to the user.

h) **Key generation**

For the user's attribute set S_n, the n-th user attribute $a_n$is located in the i-th attribute tree, and its depth is h, its path $R_n = (a_{n0}, a_{n1}, \ldots, a_{n\mathrm{k}}, \ldots, a_{nh})$, where $k \in [0, h]$ is the corresponding attribute of the k-th layer in the user attribute a_n path R_n, and the verification node selects the random number $r \in Z^*{}_P$ used to resist the collusion attack. For the attribute a_n of the user, select the random number $r_n \in Z_P$, calculate the attribute private key d_n, the private key parameter D_n and the permission parameter set $D_n{}'$, these calculations are as follows:

$d_n = g^r \left(u_i{}' \prod_{k=1}^{h} u_k{}^{a_{n_k}}\right)^{r_n}$, $D_n = g^{r_n}$, $D_n{}' = \left\{u_{h+1}^{r_n}, u_{h+2}^{r_n}, \ldots, u_{H_i}^{r_n}\right\}$. Combining the private key component, the user can get the private key as:

$$SK_{r_n} = \{d_n, D_n = g^{r_n}\} \tag{8}$$

i) **Decryption**

In the attribute authorization set $S_u{}'$, the user attribute a_n is in the m-th attribute tree, the policy attribute $a_n{}'$ is in the m' attribute tree, and $m = m'$ is satisfied; the depth h of the user attribute a_n is satisfied with the depth h' of the policy attribute $a_n{}'$, The relationship between the path of the attribute is $h \leq h'$; The path $R_n = (a_{n0}, a_{n1}, \ldots, a_{n\mathrm{k}}, \ldots, a_{nh})$ of the user's attribute and attribute path $R_n{}' = (a_{n'0}, a_{n'1}, \ldots, a_{n'\mathrm{k}'}, \ldots, a_{n'h'})$

are satisfied: $k = k'$, $a_{nk}{=}a_{n'k'}$, $k \in [0, h]$, $k' \in [0, h']$; For the consumer attribute $a_n{}'$ of the override policy attribute, the decryption permission value $d_n{}'$ is calculated as follows:

$$d_n{}' = d_n.\left(u_{h+1}^{r_n}\right)^{a_{n'h+1}}.\left(u_{h+2}^{r_n}\right)^{a_{n'h+2}}\ldots\left(u_{h'}^{r_n}\right)^{a_{n'h'}} = g^r\left(u'_i\prod_{k'=1}^{h'}u_{k'}{}^{a_{n'k'}}\right)^{r_n} \tag{9}$$

Decrypt the bilinear map $A_{n\mathrm{i}}$ and calculate the user's permissions VR' as follows:

$$Decry(CT, SK_{\mathrm{r_n}}, A_{ni}) = \frac{e(d'_n, C'_{n'})}{e(D_n, C_{n'})}$$

$$= \frac{e\left(g^r\left(u'_i\prod_{k'=1}^{h'}u_{k'}{}^{a_{n'k'}}\right)^{r_n}, g^{w_i}\right)}{e\left(g^{r_n}, \left(u'_{i'}\prod_{k'=1}^{h'}u_{k'}{}^{a_{n'k'}}\right)^{w_i}\right)} = e(g, g)^{rw_i} \tag{10}$$

$$VR' = \prod_{i\in I}A_{ni}^{\lambda_i} = \prod_{i\in I}e(g, g)^{r\sum_{i\in I}\lambda_i w_i} = e(g, g)^{rS}$$

If the user's permissions satisfy the structure, they can decrypt the trapdoor key as:

$$\frac{E_0}{\frac{e(C, \hat{C})}{e(g,g)^{Sr}}} = \frac{TK\cdot Y^S}{\frac{e\left(g^{BS}, (g^{r+\alpha})^{-\beta}\right)}{e(g,g)^{Sr}}} = \frac{TK\cdot e(g,g)^{\alpha S}}{e(g,g)^{S\alpha}} = TK \tag{11}$$

The transaction message can be restored to:

$$M = \frac{E_1}{\frac{e(C, \hat{C})}{VR'}} = \frac{M\cdot e(g,g)^{\alpha S}}{e(g,g)^{S\alpha}} \tag{12}$$

5 Security Certificate

In this part, the proposed scheme is proved to be security under the random oracle model.

Lemma 5.1 Based on the DBDH hypothesis, our solution can resist selected plaintext attacks in the random Oracle model, then our solution is CPA-security.

Proof: Assuming a probabilistic polynomial time, an adversary can exploit advantage ∂ to attack our solution. We prove that the following DBDH games can be attacked by a advantage $\partial/2$ of enemy C. $e : G_0 \times G_0 = G_1$ is a bilinear map, where G is a cyclic group with a generator element g order of p. The challenger is randomly selected $a, b, c, z \in Z_p$, $\theta \in \{0, 1\}$ and if $\theta = 0$, set $(\mathrm{g}, A, B, C, Z) = \left(g, g^a, g^b, g^c, e(g, g)^{abc}\right)$, if $\theta = 1$, set $(\mathrm{g}, A, B, C, Z) = \left(g, g^a, g^b, g^c, e(g, g)^z\right)$.

Initialization: The adversary controls a set of authorized permissions, at least two of which are controlled by the adversary, and the remaining permissions are controlled by the challenger. The adversary affirmed the challenge of the LSSS access structure.

The challenger randomly selected $a = r, b = \alpha, c = s_0$, and $r, \alpha, s_0 \in Z_p$ were all randomly selected. Set and send $Y := e(A, B) = e(g, g)^{ab}$ public parameters to the adversary.

Stage 1: The adversary asks for many of the private keys he wants according to the attribute set, but these private keys do not satisfy the access structure. Upon receiving the private key request of A with the identity RID, the challenger randomly selects $\beta_{RID}, k \in Z_p$ and calculates the private key component for each attribute $k \in S$ as follows: $d'_j = g^r \left(u'_i \prod_{k'=1}^{h'} u_{k'}{}^{a_{j'k'}} \right)^{r_j}$, $D_j = g^{r_j}$, $d'_j = g^r \left(u'_i \prod_{k'=1}^{h'} u_{k'}{}^{a_{j'k'}} \right)^{r_j}$, $D_j = g^{r_j}$

Phase 2: Repeat phase 1.

Guess: The adversary submitted a θ' guess of ϑ, when $\theta = \theta'$, if $\theta = 0$, the simulator representing the challenger will output $\left(g, g^a, g^b, g^c, e(g,g)^{abc}\right)$, otherwise output a DBDH array$\left(g, g^a, g^b, g^c, e(g,g)^z\right)$. If $\theta = 1$, the adversary will not get useful information, and its advantage is $Pr = 1/2$.When $\theta = 0$, its advantage is $Pr = 1/2 + e$. Therefore, the probability polynomial time of the opponent in the DBDH game is $Pr(\theta = \theta') - 1/2 = 1/2(1/2 = \theta) + 1/2 \cdot 1/2 - 1/2 = \epsilon/2$. In short, if the polynomial time in our game is ϵ, the adversary has a non-negligible advantage $\epsilon/2$. Therefore, based on the DBDH assumption, the adversary has no advantage in our security game, so our solution is safe.

Lemma 5.2 If we assume DBDH is established, our scheme can resist selected keyword attacks in the random Oracle model, then our scheme is semantically CKA-security.

Initialization: The simulator received a DBDH challenge, selecting two multiplicative loop groups of order p, the generator $g \in G_0$, and a bilinear map $e : G_0 \times G_0 = G_1$. Secondly, the simulator β randomly select $x, y, z \in Z_p$, select $g^x, g^y, g^z \in G_0$, $g^{xyz} \in G_0$. The simulator randomly selects $\eta_i \in Z_{\mathrm{p}}$ so that the enemy's public key $PK_i = \{g^{\eta_i}, \mathrm{g}\}$ and private key are $SK_i{=}\eta_i$. The SK is set to represent the trapdoor key as TK, and the PK is the index key IK.

Stage 1: The adversary adaptively asks the following query:

Hash-ask: The adversary can ask the random oracle H at any time. In order to answer the inquiry, the simulator maintains the list (w_i, h_i, e_i, c_i). At first this list is empty. When the adversary asks for any keyword $w_i \in \{0,1\}^*$ on the random oracle, the simulator's answer is as follows:

1. If the query keyword w_i exists in the list $H-$, the corresponding response of the simulator can be $H(w_i) = h_i \in G_0$.

2. Otherwise, the simulator generates a random coin $c_i \in \{0,1\}$ so that $Pr[c_i = 0] = \delta$.

If $c_i = 0$, the simulator can calculate $h_i = (g^x)^{c_i} \in G_0$.

If $c_i = 1$, the simulator can calculate $h_i = g^{e_i} \in G_0$, which $e_i \in Z_p$ is randomly selected. The array (w_i, h_i, e_i, c_i) is then added to the $H-$ list and $H(w_i) = h_i$ will be returned to the adversary.

Trapdoor inquiry: When the adversary asks for any keyword $w_i \in \{0,1\}^*$ in the trapdoor, the simulator first performs an inquiry to obtain the sum corresponding to the list. The simulator answers as follows:

1. If $c_i = 0$, the simulator will announce the failure and termination and output the guess b' value of b.

2. If $c_i = 1$, $h_i = g^{e_i} \in G_0$, the simulator will select the random value λ, $\mu_i \in Z_{\mathrm{p}}$, and calculates the trapdoor, and the simulator sends the trapdoor $T_i = (T_1, T_2) = \left(\mu_i \lambda, H_1(W')^{\lambda \eta_i}\right)$ to the adversary.

Challenge: The adversary submitted a pair of keywords w_0, w_1, which w_0, w_1 are keywords that the enemy did not ask before. The keyword index generated by the adversary is as follows:

1. Firstly, simulator perform two H queries to get $h_0, h_1 \in G_0$ so that $H(w_0) = h_0$, $H(w_1) = h_1$. For $i = 0, 1$, we set the array $(w_i, h_i, e_i, c_{\mathrm{i}})$ corresponding to the list H. If $c_0 = 0$ and $c_1 = 1$, the simulator announces the failure and termination. We know that at least $c_0 = 0$ and $c_1 = 1$ one is equal to zero. The simulator selects one byte $b \in \{0,1\}$ for $c_b = 0$.

2. Simulator answers the keyword index $I_w^* = \left(I_1^*, I_2^*\right)$. The simulator then randomly selects the parameters $\tau, \epsilon \in Z_P$, and the keyword index is calculated as follows: $I_w^* = \left(I_1^*, I_2^*\right) = (g^{\epsilon\tau}, e(H_1(w)^{\epsilon}, g^{\eta\tau}))$, and the trapdoor and keyword index are sent to the adversary.

Phase 2: Phase1 is repeated.

The opponent outputs the guess b' value of the keyword b. If the adversary wins in the game, the simulator will output $b' = 0$. If $b' \neq b$, output $b' = 1$.

Possibility Analysis: The adversary A performs the most trapping q and indexing queries. The probability $Pr[\beta] = (1-\delta)^q\left(1-(1-\delta)^2\right)$ that the simulator is not aborted in the game, the probability $Pr[\beta]$ is not negligible. Under the premise that the simulator does not suspend the game, if A successfully guesses the keyword, then the value m can be known. The probability that an enemy wins the game is $Pr[b' = b] = 1/2 + \epsilon \cdot Pr[\beta]$, which $\epsilon = |Pr[b' = b] - 1/2|$ is the advantage of winning the adversary. Based on the DBDH assumption, no adversary can break our algorithm, and our solution is security.

Lemma 5.3 If the DBDH assumption is true, our scheme will be security under the random oracle model.

Proof directly from **Lemma 5.1** and **Lemma 5.2**.

6 Analysis and Comparison

6.1 Privacy Protection Analysis

Content privacy: This paper uses the ciphertext policy-based attribute encryption mechanism algorithm to encrypt trapdoor key information, which is more secure than symmetric encryption algorithms. By encrypting trapdoor key information with the LSSS linear secret sharing structure and encrypting transaction information with searchable encryption, we can ensure the privacy of both parties' content. In the process of generating the private key, the random number and the identifier RID of the user interaction are introduced. Even if different users collude with each other, they cannot obtain the private key without permission. Therefore, even if there is collusion, illegal users cannot obtain the transaction information and the secret of sharing.

Identity privacy: Use the verification node in the blockchain data privacy protection access control method based on searchable attribute encryption. The verification node stores the trapdoor key ciphertext. The transaction user A does not need to be online at any time, and randomly generates key UK and identity RID for each user. In the process of interaction, the RID sequence represents the identity of the user, which protects the identity privacy of the user.

Searching privacy: Our scheme's search mechanism is against multiple attacks. In the process of index generation, the transaction party A uses the random number μ to encrypt the indexed keyword, and the node on the blockchain cannot perform the internal keyword guessing attack by matching the candidate keyword with the trapdoor. In the trapdoor generation phase, we use random numbers to hide the search keywords, which prevents malicious nodes from performing keyword replay attacks after trapdoor cracking. Therefore, blockchain network nodes and attackers cannot obtain useful information about keywords. Therefore, our solution guarantees the privacy of the keyword without reducing the security of the previous algorithm.

Attribute privacy: The verification node implements fine-grained access control, and the verification node authorizes the user of the blockchain by verifying the *VR*, which avoids the risk of submitting the access structure to the blockchain network. This mechanism protects the attributes of the linear access structure developed by the counterparty.

6.2 Scheme Comparison

The literature [7] applied the elliptic curve encryption algorithm to propose a more efficient blind signature hybrid scheme, which protects the privacy of transaction information. In literature [9], the transaction amount is hidden by the ring signature, and a secret transaction scheme based on ring signature is proposed, thus protecting transaction privacy and identity privacy. Literature [12] presented a zero cash scheme using public key encryption to protect transaction privacy. In Literature [18], public key cryptography is used to propose a blockchain retrieval privacy protection mechanism based on searchable keywords. Literature [19] proposed a distributed data storage system using blockchain technology and a privacy keyword search scheme. It can be seen from Tab. 1 that this paper adopts the access control method of searchable hierarchical attribute encryption, which not only hides the transaction amount for the nodes without access rights on the blockchain, but also the permission of the blockchain user node can quickly and efficiently query valid information of the transactions through trapdoor keywords.

Table 1: Comparison of this article and existing privacy protection schemes

Scheme	[7]	[9]	[12]	[18]	[19]	ours
Encryption mechanism	Blind signature	Ring signature	Public key	Public key	Hash public	Attribute
Ciphertext search	—	—	—	√	√	√
Trapdoor security	—	—	—	—	—	√
Index security	—	—	—	×	×	√
Anticollusion	—	√	×	×	×	√
Unlinkability	√	√	√	×	×	√
Identity privacy	Part	√	Part	√	√	√
Content privacy	×	√	×	√	√	√
Traceability	×	√	×	√	√	√

6.3 Performance Comparison

In this section, we analyze the performance of the solution. We use E for exponential operation and P for linear operation. H represents a hash operation, and m represents the number of users. In Tab. 2, we present the performance calculations for our scheme and Reference [18–19].

Table 2: Performance calculation comparison

Scheme	Ciphertext generation	Index generation	Trapdoor generation	Search match	Space complexity
[18]	$(N-1)(H+P)+NE$	E	E	$2E+P$	$O(1)$
[19]	$2H+E+P$	$2E+H+P$	$2E+H+P$	$E+P+H$	$O(1)$
Ours	$4E$	$H+3E+P$	$2E+P$	$H+E$	$O(2m)$

7 Conclusion

Since the global ledger that uses blockchain technology to store transaction information is open to any node joining the blockchain network, it is necessary to further strengthen and improve the data privacy of the blockchain. A blockchain data privacy protection access control scheme based on searchable attribute encryption is proposed. This scheme uses attribute encryption based on ciphertext strategy to encrypt trapdoor keys, and then uses searchable encryption to encrypt transactions on the blockchain. User authorization allows authorized users to access transaction information. It not only realizes the privacy protection of transaction information on the blockchain, but also enables authorized users to efficiently access transaction information. Under the random oracle model, the safety and effectiveness of the scheme are proved.

Acknowledgement: Thanks to teacher Feng for his guidance on the scheme of my paper, thanks to teacher Youliang Tian for his proposal on the modification of my paper, and finally thanks to Xiaoqin Feng and Rong Ma for the revision of the overall grammar of my paper.

Funding Statement: This research was supported by The National Natural Science Foundation of China (No. 61462060, No. 61762060) and The Network and Information Security Innovation Team of Gansu Provincial Department of Education Lanzhou University of Technology (No. 2017C-05). Tao Feng is the corresponding author.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Z. Q. Xia, J. J. Tan, J. Wang, R. L. Zhu and H. G. Xiao, “Research on fair trading mechanism of surplus power based on blockchain,” *Journal of Universal Computer Science*, vol. 25, no. 10, pp. 1240–1260, 2019.

[2] J. Y. Zhang, S. Q. Zhong, T. Wang, H. C. Chao and J. Wang, “Blockchain-based systems and applications: A survey,” *Journal of Internet Technology*, vol. 21, no. 1, pp. 1–14, 2020.

[3] H. Y. Duan, Z. Z. Li and S. P. Qu, “Blockchain technology: Application and problems,” *Journal of Xi’an University of Posts and Telecommunications*, vol. 23, no. 1, pp. 1–13, 2018.

[4] G. Jordan, K. Levchenko, D. McCoy, S. Meiklejohn, M. Pomarole *et al.*, “A fistful of bitcoins: Characterizing payments among men with no names,” in *Proc. of on Int. Measurement Conf.*, ACM, pp. 127–140, 2013.

[5] M. Conti, E. Sandeep Kumar, C. Lal and S. Ruj, “A survey on security and privacy issues of bitcoin,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 3416–3452, 2018.

[6] Q. Feng, D. He, M. K. Khan, N. Kumar and S. Zeadally, “A survey on privacy protection in blockchain system,” *Journal of Network and Computer Applications*, vol. 126, no. 10, pp. 45–58, 2018.

[7] Q. C. Shen Tu and J. P. Yu, “A blind-mixing scheme for bitcoin based on an elliptic curve cryptography blind digital signature algorithm,” *Computer Science*, vol. 1510, no. 05833, pp. 1–17, 2015, arXiv preprint arXiv.

[8] F. Gao, M. Shen and L. Zhu, “Research review on blockchain privacy protection,” *Computer Research and Development*, vol. 54, no. 10, pp. 2170–2186, 2017.

[9] S. Noether, A. Mackenzie and Research Lab T. M., “Ring confidential transactions,” *Ledger*, vol. 1, pp. 1–18, 2016.

[10] J. K. Liu, V. K. Wei and D. S. Wong, “Linkable spontaneous anonymous group signature for *ad hoc* groups,” in *Australasian Conf. on Information Security and Privacy*, Berlin, Heidelberg: Springer, vol. 3108, pp. 325–335, 2016.

[11] Y. Chen, A. Haseeb, C. Li, J. Li and G. Xu, “A new anti-quantum proxy blind signature for blockchain-enabled internet of things,” *Computers, Materials & Continua*, vol. 61, no. 2, pp. 711–726, 2019.

[12] A. Chiesa, C. Garman and E. B. Sasson, “Zerocash: Decentralized anonymous payments from bitcoin,” in *IEEE Symp. on Security and Privacy*, 2014.

[13] A. Chiesa, D. Genkin and E. B. Sasson, “Snarks for c: Verifying program executions succinctly and in zero knowledge,” in *Annual Cryptology Conf.*, Berlin, Heidelberg: Springer, pp. 90–108, 2013.

[14] C. Decker and R. Wattenhofer, A fast and scalable payment network with bitcoin duplex micropayment channels. In: *A. Stabilization, Safety, and Security of Distributed Systems*, vol. 9212. New York: Springer Verlag, 3–18, 2015.

[15] I. Bentov, R. Kumaresa and A. Miller, “Sprites: Payment channels that go faster than lightning,” Available https://arxiv.org/abs/1702.05812.

[16] L. Alshenibr, F. Baldimtsi and E. Heilman, “TumbleBit: An untrusted bitcoin-compatible anonymous payment hub,” *Network & Distributed System Security Symposium*, 2017.

[17] M. Green and I. Miers, “Bolt: Anonymous payment channels for decentralized currencies,” in *Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security*, ACM, pp. 473–489, 2017.

[18] F. Guo, P. Jiang and K. Liang, “Searchain: Blockchain-based private keyword search in decentralized storage,” *Future Generation Computer Systems*, vol. 107, no. 2, pp. 781–792, 2020.

[19] H. G. Do and W. K. Ng, “Blockchain-based system for secure data storage with private keyword search,” *2017 IEEE World Congress on Services*, vol. 23, pp. 90–93, 2017.

[20] S. Ji, Y. Liu and Y. Ren, “Incentive mechanism of data storage based on blockchain for wireless sensor networks,” *Mobile Information Systems*, vol. 2018, pp. 6874158, 2018.

[21] Y. Ren, P. K. Sharma and F. Zhu, “Data query mechanism based on hash computing power of blockchain in internet of things,” *Sensors*, vol. 20, no. 1, pp. 207, 2020.

[22] J. Li, Z. Liu and L. Zu, “New ciphertext-policy attribute-based encryption with efficient revocation, 2014,” in *IEEE Int. Conf. on Computer and Information Technology*, Xi’an, pp. 281–287, 2014.