International Journal of Knowledge and Innovation Studies
https://www.acadlore.com/journals/IJKIS

A Blockchain and Attribute-Based Encryption Scheme for Hazardous Materials Circulation Data Sharing

Xuewei Li[1], Jialin Ma[1*], Ashim Khadka[2]

[1] Faculty of Computer and Software Engineering, Huaiyin Institute of Technology, 223003 Huaian, China
[2] Nepal College of Information Technology, Pokhara University, 44700 Lalitpur, Nepal

[*] Correspondence: Jialin Ma (majl@hyit.edu.cn)

Received: 01-12-2024 **Revised:** 03-23-2024 **Accepted:** 03-28-2024

Citation: X. Li, J. Ma, and A. Khadka, "A blockchain and attribute-based encryption scheme for hazardous materials circulation data sharing ," *Int J. Knowl. Innov Stud.*, vol. 2, no. 1, pp. 19–28, 2024. https://doi.org/10.56578/ijkis020103.

Abstract: The regulatory system for hazardous materials is complex, with poor inter-departmental communication and low levels of data sharing, making effective regulation challenging. Blockchain technology, known for its decentralization, traceability, and secure and trustworthy information, is widely applied in data sharing. Concurrently, attribute-based encryption (ABE), a novel encryption technique, offers high security and fine-grained access control, providing technical support for secure data access and privacy protection. However, existing attribute encryption algorithms do not consider the hierarchical relationship of access structures among data files during data sharing. Moreover, the immutable nature of blockchain means that access policies stored on it cannot be altered, leading to a lack of flexibility in data sharing. To address these issues, this paper proposes a blockchain and attribute-based dynamic layered access scheme for hazardous materials circulation data sharing. By constructing a Linear Secret-Sharing Scheme (LSSS) matrix, layered access control is achieved, allowing data decryption related to the matching parts of a user's attributes with the access structure. Additionally, through the design of a policy update algorithm, the blockchain structure is organized into transaction blocks and policy blocks, storing the encrypted symmetric keys separately to enable dynamic updates of access policies. Security analysis and experimental comparisons demonstrate the scheme's effectiveness and security in hazardous materials circulation data sharing.

Keywords: Hazardous materials; Data sharing; Blockchain; Attribute-Based Encryption (ABE)

1 Introduction

With the increase in globalization and the complexity of supply chains, the circulation and management of hazardous chemicals face increasingly severe challenges. The improper use or accidental leakage of hazardous chemicals can lead to serious personal injury, environmental damage, and property loss. Therefore, ensuring effective regulation and safe management of hazardous chemicals has become crucial. However, current methods of hazardous chemical circulation data sharing and management have many problems. Traditional centralized data management systems may be prone to data tampering, information asymmetry, and security vulnerabilities. In addition, the lack of efficient data interchange and collaboration mechanisms between different institutions leads to information silos and process bottlenecks. Thus, establishing a secure, transparent, and efficient data sharing scheme for hazardous chemical circulation is imperative.

In recent years, blockchain technology has developed rapidly and is widely applied in the field of data sharing due to its decentralization, traceability [1], and secure and trustworthy information. Ge et al. [2] combined the medical system with blockchain and proposed a blockchain-based medical data secure access model to address the privacy and security issues that may be encountered in centralized storage. Franciscon et al. [3] discussed the feasibility of introducing blockchain into government organization management and data sharing from the perspectives of blockchain consensus and control. Wang et al. [4] use smart contracts and homomorphic encryption technology to protect users' privacy. Xu et al. [5] protect transaction privacy with zero-knowledge proof method. Wu and Du [6] introduced data desensitization technology and proposed a blockchain-based electronic medical record security sharing model. Cao et al. [7] established a blockchain-based steel traceability system to achieve real-time tracking and recording of product quality and other information. Zhuang et al. [8] combined blockchain with secret sharing

schemes to propose an IP privacy protection and traceable identity management scheme. As a new encryption technology, attribute encryption provides technical support for data security access control and privacy protection with its high security and fine-grained access capabilities.

Blockchain avoids many problems existing in traditional centralized sharing due to its immutable nature, providing a secure and reliable environment for cross-departmental data sharing. At the same time, blockchain is somewhat limited in handling sensitive business and privacy data due to its public and transparent characteristics. In 2005, the concept of ABE was proposed by Sahai and Waters [9], which is mainly divided into Key Policy Attribute-Based Encryption (KP-ABE) and Ciphertext Policy Attribute-Based Encryption (CP-ABE) according to the different binding positions of ciphertext, key expression form, and access policy. KP-ABE [10], where the access policy is hidden in the key, is suitable for applications in systems such as video-on-demand and pay television, while CP-ABE, where the access policy is hidden in the ciphertext, is widely used in data security sharing schemes [11]. In CP-ABE, the encryption time of plaintext is linearly proportional to the number of attributes, thus it has a certain predictability. The Shamir secret sharing scheme is the basis of traditional CP-ABE algorithms. In 2007, Bethencourt et al. [12] provided the first construction of ABE based on ciphertext policy. In 2011, Waters [13] first proved the security of CP-ABE in the standard model. In 2016, Cui et al. [14] proposed a CP-ABE scheme to implement partially hidden access structures under prime-order groups, In 2018, Belguith et al. [15] proposed a revocable outsourcing CP-ABE scheme for the application scenarios of edge computing. In 2019, Xiong et al. [16] proposed a PHOABE scheme that alleviates the strong coupling between the computational cost of decryption and the complexity and number of attributes of the access policy.

In real application scenarios, shared data often contains multiple access structures with hierarchical relationships. Traditional CP-ABE does not consider the hierarchical relationship between attributes, requiring data owners to generate multiple ciphertexts to encrypt files, leading to a significant computational overhead. In 2013, Liu et al. [17] proposed Ciphertext Policy Weighted Attribute-Based Encryption (CP-WABE) to express attributes with hierarchical relationships. However, CP-WABE usually lacks flexibility in attribute comparison and is less efficient. In 2016, Wang et al. [18] proposed a scheme using weighted attributes to represent attribute levels, reducing the hierarchy and encryption overhead of the access tree. Since this scheme used a tree-like access structure, its efficiency was still low. Another issue to consider is that blockchain is immutable and undeletable. Once the CP-ABE access control policy specified by the data owner is stored on the blockchain, it cannot be updated by modifying the block content, leading to a lack of flexibility in data sharing.

In response to these problems, this paper proposes a blockchain and attribute-based dynamic layered access scheme for hazardous chemical circulation data sharing. By designing a layered CP-ABE algorithm with an LSSS matrix storage structure, multiple access control structures of hazardous chemical circulation data files are integrated into one LSSS matrix. When the attributes of a hazardous chemical transport company or regulatory department match part of the access control structure, they can decrypt the data associated with that part, while allowing the access control policy to be updated, enabling hazardous chemical-related institutions to flexibly modify the access control policy. This scheme only requires one encryption/decryption operation to complete what traditional schemes require multiple encryption/decryption operations to achieve.

2 Related Theories

2.1 Bilinear Mapping

Let q be a large prime number, and let G_1 and G_2 be two groups of order q, with their operations defined as addition and multiplication, respectively. A bilinear mapping from G_1 and G_2 is defined as e: $G_1 \times G_1 \rightarrow G_2$, the bilinear pairings satisfy the following conditions [19]:

(1) Bilinearity: for $\forall u, v \in G_1, a, b \in Z_p, e\left(u^a, v^b\right) = e(u, v)^{ab}$.

(2) Non-degeneracy: for $\exists g \in G_1, e(g, g) \neq 1$.

(3) Computability: for $\forall u, v \in G_1$, there exists an efficient algorithm to compute $e(u, v)$.

2.2 LSSS

Scholar Beimel [20] proposed the definition of LSSS in his paper: A secret sharing scheme Π over a set of participants P is described as linear over Z_p, if:

(1) The shares of the secret for each participant form a vector over Z_p.

(2) For the secret sharing scheme Π, there exists an $l \times n$ matrix M, where the i-th row is identified by a participant $\rho(i)$, and a function ρ satisfies $\{1, 2, \ldots, l\} \rightarrow e$. Given a column vector $v = (s, r_2, \ldots r_n)$ where $s \in Z_p$ is the secret to be shared and $(r_2, \ldots r_n) \in Z_p$ are randomly chosen. $M\vec{v}$ represents m shares of the secret s, where $\lambda_i = M\vec{v}_i$ is the share belonging to participant $\rho(i)$.

Steps for using an LSSS matrix to implement hierarchical access control is as follows: Let T be an access control tree with its Boolean formula as $(AAND(BAND(CORD)))$. By applying the Lewko-Waters algorithm,

it is converted into a matrix M. During encryption, a vector $\vec{v} = (s_1, s_2, s_3)$ is randomly chosen, where s_1, s_2, s_3 are the keys assigned to a non-leaf node in the tree T, calculating $\lambda = M \cdot \vec{v}$.

According to the LSSS, it's known that $s_j = \sum_{i \in I} \omega_{i,j} \lambda_i$, $I = i : \rho(i) \epsilon S$, wherein $\rho(i)$ maps the i-th row of the matrix to an attribute, denoting the user's attribute set. When $\lambda_A = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_i \\ \vdots \\ \lambda_l \end{pmatrix}_{i \in I}$, $s_j = \omega_j^T \lambda_A$. When $M_A = \begin{pmatrix} M_1 \\ \vdots \\ M_i \\ \vdots \\ M_l \end{pmatrix}_{i \in I}$, $s_j = s_j^T = \lambda_A^T \omega_j = \left(M_A \cdot \vec{v}^T\right)^T \omega_j = \vec{v} \cdot \left(M_A^T \omega_j\right)$, let $M_A^T \omega_j = \varepsilon_j$. At this time, $s_j = \vec{v} \cdot \varepsilon_j, \varepsilon_j$ is a row vector with the j-th element as 1 and all other elements as 0. During decryption, if only attributes B and C are satisfied, meaning only a part of the access structure is met, then through $M_A^T \omega_j, \omega_2, \omega_3$ can be solved. Subsequently, by solving $s_j = \omega_j^T \lambda_A, s_2, s_3$ can be obtained. If all attributes are satisfied, meaning the entire access structure is met, then through the steps mentioned above, all shared keys can be acquired.

2.3 Hierarchical Access Process

In traditional CP-ABE schemes, a user's attributes either satisfy the access control structure to decrypt the plaintext or do not satisfy the access control structure to decrypt the plaintext. In hierarchical access control, multiple access structures with hierarchical relationships can be integrated into a single access structure. As shown in Figure 1, the access structures of $m1$ and $m2$ clearly have a hierarchical relationship, thus they can be integrated into a single access structure. When a data accessor's attributes match part of the access structure, they can decrypt data related to that part (User 2). Only when the entire access structure is satisfied can all data be decrypted (User 1).

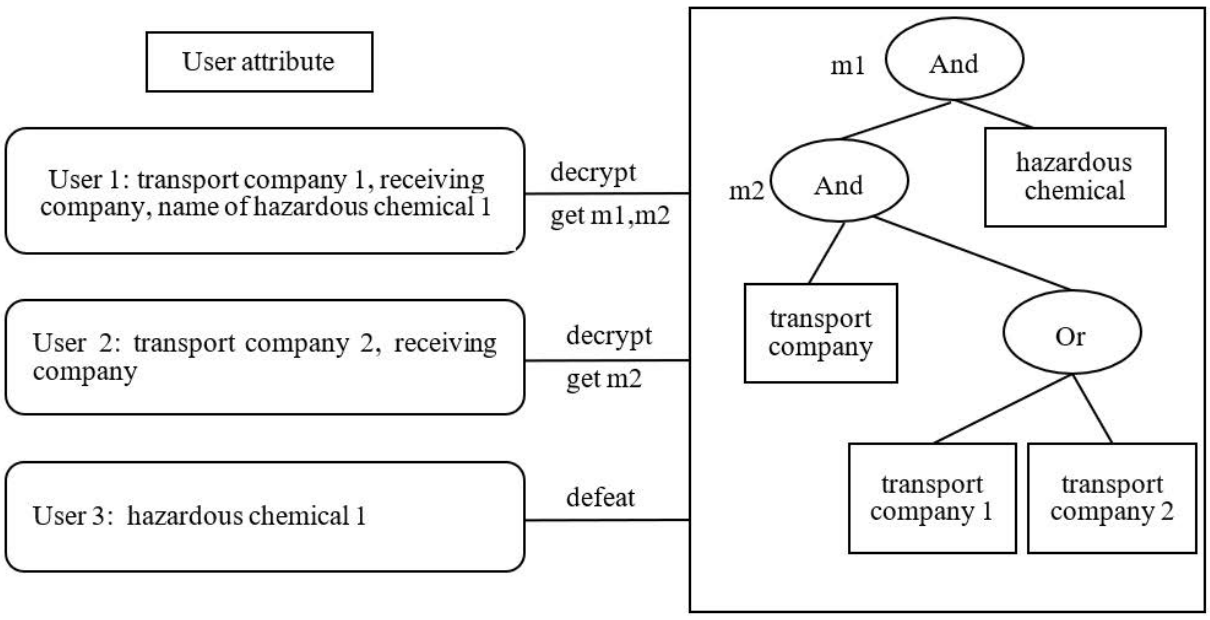

Figure 1. Hierarchical access process

3 Hazardous Materials Circulation Data Sharing Scheme

3.1 System Model

The system architecture, as shown in Figure 2, mainly includes the following four roles:

(1) Central Authorization Center: The central authorization agency can be appointed by a third party and is mainly responsible for generating system parameters, key management, and user attribute management.

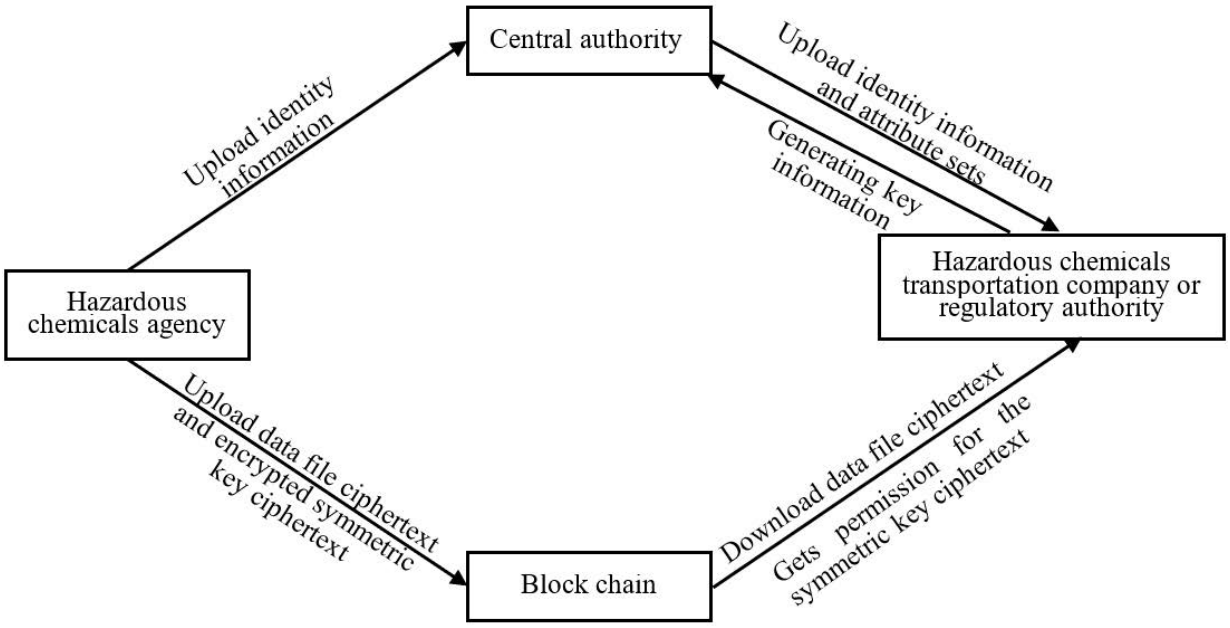

Figure 2. System architecture diagram

(2) Hazardous Materials Related Institutions: Hazardous materials related institutions select n symmetric keys to encrypt data files containing hazardous materials information, generating an encrypted data file D. They encrypt their symmetric keys using an encryption algorithm to produce encrypted symmetric key ciphertext CT. Both the encrypted symmetric key ciphertext CT and the encrypted data file D are uploaded to the blockchain for storage. The encrypted symmetric key is divided into CT_1 and CT_2, with the ciphertext CT1 acting as a transaction block TX and the ciphertext CT_2 acting as a policy block A.

(3) Blockchain: The encrypted data files and access policies are stored on the blockchain. Only data users who meet the access policy can obtain the encrypted data files from the blockchain. At the same time, the blockchain is used to prevent data tampering and allow flexible modification of access policies.

(4) Hazardous Materials Transport Companies or Regulatory Departments: Hazardous materials transport companies or regulatory departments submit their identity information and attribute sets to the central authorization agency, which then generates corresponding keys and returns them to the hazardous materials transport companies or regulatory departments. The blockchain first checks whether the attributes of the hazardous materials transport companies or regulatory departments match the partial or entire access control structure of these data files. If there is no match, the blockchain refuses the user request; otherwise, the blockchain sends the encrypted symmetric key ciphertext CT and the encrypted data file D to the user. Upon receiving the encrypted symmetric key ciphertext CT, the hazardous materials transport company or regulatory department uses the decryption algorithm to obtain the symmetric key, then uses this symmetric key to decrypt the data file containing hazardous materials information through the decryption algorithm, thereby gaining access to the hazardous materials circulation data.

3.2 Algorithm Description

The improved CP-ABE algorithm includes six types of algorithms: System Initialization, Private Key Generation, Encryption, Decryption, Ciphertext Policy Generation, and Ciphertext Policy Update. These algorithms cover scenarios where a data accessor can decrypt data associated with a part of the access control structure that matches their attributes, and when the entire access structure is satisfied, all data can be decrypted, allowing for the updating of access policies. Below are the details of these algorithms:

(1) $\text{Setup}(k, U) \rightarrow (PK, MSK)$: The system initialization algorithm takes the system's attribute set and security parameters as input and generates the system master key and the system public key. Let G_0 and G_1 be two multiplicative cyclic groups of prime order p, with g being a generator of G_0, and define a bilinear mapping $e : G_0 \times G_0 \rightarrow G_1$. Randomly select elements $h_1, \ldots, h_u \in G_0$, which are associated with each attribute in U. Randomly select two elements $\alpha, \beta \in Z_p$. The system outputs the key pair (PK, MSK).

$$PK = \left\{g, e(g,g)^{\alpha}, g^{\beta}, h_1, \ldots, h_u\right\}$$
$$MSK = g^{\alpha}$$

(2) $\mathrm{KeyGen}(PK, MSK, S) \rightarrow SK$: The private key generation algorithm outputs a private key related to the user's attributes by inputting the system public key, the system master private key, and the user's attribute set S.

Randomly select $t \in Z_p$, compute $K = g^{\alpha} g^{\beta t}, L = g^t, \forall x \in S : K_x = h_x^t$, the system outputs the user attribute private key SK as follows:

$$SK = \{K, L, \forall x \in S : K_x\}$$

(3) $\mathrm{Encrypt}\left(m_j, j \in (1,n), \sigma, PK, (M, \rho)\right) \rightarrow CT$: The encryption algorithm outputs ciphertext CT by inputting a plaintext set $\{m_j, j \in (1,n)\}$, system public key PK, policy updater identity σ, and LSSS matrix structure (M, ρ). During encryption, M is an $l \times n$ access matrix, M_i represents the i-th row of the matrix. The mapping function $\rho(i)$ assigns attributes to the rows of M.

Randomly select a vector $\vec{v} = (s_1, s_2, \ldots, s_n) \in Z_p^n$ to generate ciphertext information

$$CT_1 = \left\{C_j = m_j e(g,g)^{\alpha s_j \sigma}, C_j' = g^{s_j \sigma}, j \in (1,n)\right\},$$

compute $\lambda_i = \vec{v} M_i, i \in (1,l)$ to represent the share of $\rho(i)$, randomly generate $r_1, r_2, \ldots, r_l \in Z_p$, compute $CT_2 = \left\{C_i = g^{\beta \lambda_i \sigma} h_{\rho(i)}^{-r_i}, D_i = g^{r_i}, i \in (1,l)\right\}$, and the final generated ciphertext CT is:

$$CT = \{CT_1, CT_2\} = \left\{C_j, C_j', C_i, D_i, j \in (1,n), i \in (1,l)\right\}$$

After encryption, the attribute encryption algorithm encrypts s_1 using a secure encryption algorithm to obtain encrypted information $Enc(s_1)$ for dynamic access policy updates.

(4) $\mathrm{Decrypt}(CT, SK) \rightarrow m_j$: The decryption algorithm outputs the plaintext set m_j by inputting the ciphertext CT and the user attribute private key S.

In the decryrption algorithm, M_A is a matrix composed of a set of row vectors from M, corresponding to the attribute set S associated with the user attribute private key SK. ε_j is a row vector of length n where the j-th element is 1 and all other elements are 0. Let the index set be $I = \{i : \rho(i) \in \mathrm{S}\}$. For each $j \in (1,n)$, according to $M_A^T \omega_j = \varepsilon_j$, it's known that $\sum_{i \in I} \omega_{i,j} M_i^T = \varepsilon_j$, if $\omega_{i,j}$ can be successfully retrieved, the decryption formula is:

$$\begin{aligned}
&\frac{e\left(C_j', k\right)}{\Pi_{i \in I, j}\left(e\left(C_i, L\right) e\left(D_i, k_{p(i)}\right)\right)^{\omega_{i,j}}} \\
&= \frac{e\left(g^{s_j \sigma}, g^{\alpha} g^{\beta t}\right)}{\Pi_{i \in I, j}\left(e\left(g^{\beta \lambda_i \sigma}, g^t\right) e\left(g^{\beta \lambda_i \sigma}, h_{\rho(i)}^{-r_i}\right) e\left(g^{r_i, h_{\rho(i)}^t}\right)\right)^{\omega_{i,j}}} \\
&= \frac{e(g,g)^{\alpha s_j \sigma} e(g,g)^{\beta s_j t \sigma}}{\Pi_{i \in I, j} e(g,g)^{t \beta \lambda_i \omega_{i,j}}} = e(g,g)^{\alpha s_j \sigma}
\end{aligned}$$

Finally, the plaintext set is $m_j = \frac{C_j}{e(g,g)^{\alpha s_j \sigma}}$, if $\omega_{i,j}$ can not be retrieved, the decryption fails.

(5) $\mathrm{UpdatePolicy}\left(\mathrm{Enc}\left(s_1\right), \sigma, (M', \rho')\right) \rightarrow CT_2'$: The ciphertext policy update algorithm outputs new ciphertext CT_2' for the access policy by inputting encrypted information $\mathrm{Enc}\left(s_1\right)$, the policy updater's identity σ, and the new access policy $(M', \rho'))$. M' is an $l' \times n'$ access matrix, and M_i' represents the i-th row of the matrix. Randomly select a vector $\vec{v}' = (s_1', s_2', \ldots, s_n') \in Z_p^{n'}$, and compute $\lambda_i' = \vec{v}' M_i', i \in (1,l')$. Randomly generate $r_1', r_2', \ldots, r_l' \in Z_p$, compute $C_j'' = g^{s_j' \sigma}, C_i' = g^{\beta \lambda_i' \sigma} h_{\rho'(i)}^{-r_i'} D_i' = g^{r_i'}$.

Finally, generate new policy ciphertext information CT_2':

$$CT_2' = \left\{C_j'', (C_i', D_i')\right\}$$

(6) $\mathrm{VerifyPolicy}\left(\sigma, CT_2'\right) \rightarrow CT_2''$: The ciphertext policy verification algorithm inputs the policy updater's identity σ and the new policy ciphertext CT_2'. Blockchain consensus nodes first verify the policy updater's identity σ and then determine whether C_j' and C_j'' are consistent. If consistent, the updated ciphertext is output $CT_2'' = \left\{C_j, C_j', (C_i', D_i')\right\}$.

3.3 Scheme Construction

The hazardous materials circulation data sharing scheme proposed in this paper consists of seven operations: System Initialization, Data File Encryption, Symmetric Key Encryption, User Authorization, Symmetric Key Decryption, Data File Decryption, and Dynamic Access Policy Update. The specific steps of the scheme are as follows:

(1) System Initialization: The central authorization agency specifies an attribute set U and generates the system master key MSK and system public key PK through the system initialization algorithm.

(2) Data Encryption: Hazardous materials related institutions select n symmetric keys $\{ck_1, \ldots, ck_n\}$ and encrypt their data files $\{d_1, \ldots, d_n\}$ using a symmetric encryption algorithm, resulting in encrypted data files $D = \{D_{ck_1}(f_1), \ldots, D_{ck_n}(f_n)\}$.

(3) Symmetric Key Encryption: Hazardous materials related institutions define access trees $\{T_1, \ldots, T_n\}$ for data files $\{d_1, \ldots, d_n\}$ containing hazardous materials information and integrate them into a single access tree T. They convert the access tree T into a $LSSS$ matrix structure (M, ρ) sing a labeling method. The symmetric keys $\{ck_1, \ldots, ck_n\}$ are encrypted using the encryption algorithm, producing encrypted symmetric key ciphertext CT. Both the encrypted symmetric key ciphertext CT and the encrypted data file EF are uploaded to the blockchain for storage. In the meantime, the encrypted symmetric key ciphertext CT is divided into CT_1 and CT_2,with the ciphertext CT_1 acting as a transaction block and the ciphertext CT_2 acting as a policy block on the blockchain.

(4) User Authorization: For any hazardous materials transport company or regulatory department, the central authorization agency specifies a set of attributes S and generates corresponding user attribute private key SK using the private key generation algorithm.

(5) Symmetric Key Decryption: When a hazardous materials transport company or regulatory department wants to obtain data files with hazardous materials information from the blockchain, the blockchain first checks if the attributes of the hazardous materials transport company or regulatory department match the partial or entire access control structure of these data files. If there is no match, the blockchain refuses the user request; otherwise, the blockchain sends the encrypted symmetric key ciphertext CT to the user. Upon receiving the encrypted symmetric key ciphertext CT, the hazardous materials transport company or regulatory department uses the decryption algorithm to obtain the symmetric key. They can decrypt the symmetric key associated with this part of the access tree when their attributes match part of the access tree. Only when their attributes match the entire access control structure can all symmetric keys be obtained.

(6) Data File Decryption: After obtaining $\{D_{ck_1}(d_1), \ldots, D_{ck_n}(d_n)\}$, the hazardous materials transport company or regulatory department uses $\{ck_1, \ldots, ck_n\}$ nd the decryption algorithm to decrypt the data file containing hazardous materials information, obtaining the data files $\{d_1, \ldots, d_n\}$.

(7) Dynamic Access Policy Update: Hazardous materials related institutions generate new policy ciphertext CT_2' by inputting the policy updater's identity σ and the new access policy (M', ρ') through the access policy update algorithm. They send their identity and the new policy ciphertext to the blockchain consensus nodes, which verify the policy updater's identity σ and the new policy ciphertext CT_2' using the ciphertext policy verification algorithm. Once verified, a new policy ciphertext CT_2'', is generated and appended as a new policy block A' to the transaction block TX.

4 Security Analysis and Performance Evaluationl

4.1 Security Analysis

Theorem: If the decisional q-parallel BDHE assumption holds, then no polynomial-time adversary can break the improved CP-ABE scheme proposed in this paper by choosing a challenge access structure (M^*, ρ^*).

Proof: If there exists a polynomial-time adversary $\mathcal{A}$ who can break our scheme with a non-negligible advantage ε, then there exists another adversary $\mathcal{B}$ who can solve the q-parallel BDHE assumption with the non-negligible advantage ε. The setup of the challenger is as follows: Choose a bilinear group ^{G} of prime order p(including a generator g), randomly select $\beta, s, b_1, \ldots, b_p \in Z_p$, and publicly disclose:

$$\vec{y} = \left\{ \begin{array}{c} g, g^s, g^\beta, \ldots, g^{(\beta q)}, , g^{(\beta q+2)}, \ldots, g^{(\beta 2q)} \\ \forall_{1\le j\le q} g^{s\cdot b_j}, g^{(\beta/b_j)}, \ldots, g^{(\beta^q/b_j)}, , g^{\left(\beta^{q+2}/b_j\right)}, \ldots, g^{\left(\beta^{2q}/b_j\right)} \\ \forall_{1\le j\le q, k\le q, k\ne j} g^{\beta s b_k/b_j}, \ldots, g^{\beta^q s b_k/b_j} \end{array} \right.$$

Randomly select $\theta \in \{0, 1\}$, if $\theta = 0$, let $Z = e(g, g)^{\beta q+1_s}$, set $T = (y, Z)$. Before starting the game, the adversary $\mathcal{B}$ first gets the old (M^*, ρ^*) and the new (M^{**}, ρ^{**}) access policies that $\mathcal{A}$ wishes to challenge.

(1) Initialization: The adversary $\mathcal{B}$ randomly selects $\alpha = \alpha' + \beta^{q+1}$, making the system public key $e(g, g)^\alpha = e(g, g)^{\alpha'+\beta^{q+1}} = e\left(g^\beta, g\right)^{\beta^q} e(g, g)^{\alpha'}$. For $h_1, \ldots, h_u$, each $x \in U$ selects a random parameter $z_x \in Z_p$, setting the index set $X = \{i : \rho^*(i) = x\}$, performing the following computation: if $x \in U$, then $h_x = g^{z_x} \prod_{i\in X} g^{\beta M^*_{i,1}/b_i}, g^{\beta^2 M^*_{i,2}/b_i} \ldots g^{\beta^n M^*_{i,n}/b_i}$; otherwise $h_x = g^{z_x}$.

(2) Phase 1: The adversary $\mathcal{A}$ submits an attribute set S that does not satisfy the access policies (M', ρ') and (M^*, ρ^*).The adversary $\mathcal{B}$ generates the corresponding attribute private keys through the private key generation algorithm, as follows: Based on the linear reconstruction property, there exists a set of vectors $\omega = (\omega_1, \omega_2, \ldots, \omega_{n^*}) \in Z_p^{n^*}$ such that $\omega_1 = -1$, for any $i : \rho^*(i) \in S, \omega \cdot M_i^* = 0$, set $t = k + \omega_1\beta^q + \omega_2\beta^{q-1} + \cdots + \omega_{n^*}\beta^{q-n^*+1}, k \in Z_p$, solve $L = g^t = g^k \prod_{i=1,\ldots,n^*} \left(g^{\beta^{q-n^*+1}}\right)^{\omega_i}$

Through the settings of t, include term $g^{\beta^{q+1}}$ when constructing K, during initialization, $\alpha = \alpha' + \beta^{q+1}$ can be eliminated through exponentiation operation. The adversary $\mathcal{B}$ calculates K as follows:

$$K = g^{\alpha} g^{\beta t} = g^{\alpha' + \beta^{q+1}} g^{\beta t} = g^{\alpha'} g^{\beta k} \prod_{i=1,\ldots,n^*} \left(g^{\beta^{q+2-i}} \right)^{\omega_i}$$

For $\forall x \in S$, compute K_x. When $x \in S$, there is no i such that $\rho^*(i) \in S$, set $K_x = L^{Z_x}$, when $x \in S$, there exists i such that $\rho^*(i) \in S$, due to $\omega \cdot M_i^* = 0$, eliminate the term $g^{\beta^{q+1}}/b_i$ contained in K_x based on this property. Let $S_1 = \{x : \rho(i) \in S \cap U\}$ be the set of elements of the user-submitted attribute set that satisfy the system attribute U, and $S_2 = \{x : \rho(i) \in S, \rho(i) \notin U\}$ otherwise, the computation method is as follows:

$$K_x = h_x^t = L^{Z_x} \prod_{i \in X} \prod_{j=1,\ldots,n} \left(g^{\left(\beta^j / b_i\right) k} \prod_{\substack{m=1,\ldots,n^* \\ m \neq j}} \left(g^{\beta^{q+1+j-m}} \right)^{\omega_m} \right)^{M_{ij}^*}, x \in S_1$$

$$K_x = h_x^t = g^{Z_x t} = L^{Z_x}, x \in S_2$$

(3) Challenge: The adversary $\mathcal{A}$ sends two equal-length messages m_0 and m_1. The adversary $\mathcal{B}$ randomly chooses $c \in \{0, 1\}$, computes the components of the ciphertext of m_c using the old access policy, $C_j = m_c Z e(g, g)^{a s_j \sigma}$ and $C_j' = g^{s_j \sigma}$, randomly select $y_2', y_3', \ldots, y_{n^*}'$, and share the key through $v = s_1, s_2\beta + y_2', s_3\beta^2 + y_3', \ldots, s_n\beta^{n-1} + y_{n^*}' \in Z_p^{n^*}$, then select random numbers $r_1', r_2', \ldots, r_l' \in Z_p$, for $i = 1, \ldots, n^*$, define Q_i as the set of all p that makes $\rho^*(i) = \rho^*(p)$. The challenge ciphertext is:

$$C_i^* = h_{\rho^*(i)}^{r_i'} \left(\prod_{j=2,\ldots,n^*} \left(g^{\beta\sigma} \right)^{M_{i,j}^* y_j'} \right) \left(g^{b_i s} \right)^{-Z_{\rho^*(i)}} \left(\prod_{p \in Q_i} \prod_{j=1,\ldots,n^*} \left(g^{\beta^j s (b_i / b_p)} \right)^{M_{p,j}^*} \right)$$

$$D_i^* = g^{-r_i'} g^{-s b_i}$$

The adversary $\mathcal{B}$ performs following calculations according to the new access policy (M^{**}, ρ^{**}) provided by $\mathcal{A}$ and each encrypted shared key. Randomly select $v = s_1', s_2'\beta + y_2', s_3'\beta^2 + y_3', \ldots, s_n'\beta^{n-1} + y_{n^{**}}' \in Z_p^{n^*}$ and select random numbers $r_{1^*}', r_{2^*}', \ldots, r_{l^*}' \in Z_p$, for $i = 1, \ldots, n^{**}$, define Q_i as the set of all p that makes $\rho^{**}(i) = \rho^{**}(p)$ when $p \neq i$. Then, the new challenge ciphertext is:

$$C_i^{**} = h_{\rho^{**}(i)}^{r_i'} \left(\prod_{j=2,\ldots,n^{**}} \left(g^{\beta\sigma} \right)^{M_{i,j}^{**} y_j'} \right) \left(g^{b_i s} \right)^{-Z_{\rho^{**}(i)}} \cdot \left(\prod_{p \in Q_i} \prod_{j=1,\ldots,n^{**}} \left(g^{\beta^j s (b_i / b_p)} \right)^{M_{p,j}^{**}} \right)$$

$$D_i^{**} = g^{-r_i'} g^{-s b_i}$$

Adversary $\mathcal{B}$ sends the ciphertext $CT = \left\{ C_j, C_j', C_i^{**}, D_i^{**} \right\}$ to $\mathcal{A}$.

(4) Phase 2: Similar to Phase 1.

(5) Guess: $\mathcal{A}$ outputs a guess c' for c. If $c' = c$, then adversary $\mathcal{B}$ outputs $\theta = 0$, indicating $T\epsilon P_{\text{q- parallel BDHE}}$, at this time, the advantage of the adversary is $\Pr[c = c' \mid \theta = 0] = \frac{1}{2} + \varepsilon$; if $c' \neq c$, then output $\theta = 1$, indicating $T\epsilon P_{\text{q- parallel BDHE}}$, at this time, the advantage of the adversary is $\Pr[c = c' \mid \theta = 0] = \frac{1}{2}$. Therefore, the advantage of $\mathcal{B}$ attacking the q-parallel BDHE assumption is

$$\frac{1}{2} \Pr[c = c' \mid \theta = 0] + \frac{1}{2} \Pr[c = c' \mid \theta = 1] - \frac{1}{2} = \frac{\varepsilon}{2}$$

Thus, the advantage of any polynomial-time adversary in winning the IND-SAS-CPA game is negligible.

Proof complete.

4.2 Performance Evaluation

4.2.1 Theoretical analysis

This paper compares our scheme with the ones in the studies [8] and [10] from five aspects, as shown in Table 1. The schemes in literatures [8], [10], and our proposed scheme all support fine-grained access control and resistance to collusion attacks. Our scheme supports hierarchical access control and dynamic update of access policies, which are not supported by the schemes in literatures [8] and [10].Table 2 compares the computational overheads of private key generation, encryption, and decryption between our scheme and those in literatures [8] and [10]. Table 3 evaluates the storage overhead based on the key length and ciphertext length during the encryption process.

The computations involved in the tables are as follows: n denotes the number of attributes included in the user's private key, s represents the number of attributes contained in the access structure, c represents the hierarchy of the access structure, E represents an exponentiation operation on group G, P denotes a bilinear pair operation, M represents a multiplication operation on G, l indicates the length of group element G_0, and l_1 represents the length of elements in group G_T.

Table 1. Comparison of related features

Scheme	Literature [8]	Literature [10]	The Proposed Scheme
Fine-grained Access Control	Yes	Yes	Yes
Resistance to Collusion Attacks	Yes	Yes	Yes
Hierarchical Access Control	No	No	Yes
Dynamic Update of Access Policies	No	No	Yes

Table 2. Computational overhead analysis

Scheme	Literature [8]	Literature [10]	The Proposed Scheme
Private Key Generation	$2(n+1)E+(n+1)M$	$(n+2)E+M$	$(n+2)E+M$
Encryption	$2n(s+1)E+cM$	$(3s+2)cE+c(s+1)M$	$(3s+2c)E+(c+s)M$
Decryption	$csE+2snP+nM$	$csE+(2s+1)nP+nM$	$cE+(2s+1)P+M$

Table 3. Storage overhead analysis

Scheme	Literature [8]	Literature [10]	The Proposed Scheme
Key Length	$(2n+1)l$	$(n+2)l$	$(n+2)l$
Ciphertext Length	$(2sc+c)l+cl_1$	$(2sc+c)l+cl_1$	$(2sc+c)l+cl_1$

4.2.2 Experimental analysis

The experiments were conducted on a computer running the Windows operating system, with hardware configuration of AMD Ryzen 5 5600U with Radeon Graphics, 2.30GHz, and 16GB of RAM, using Python as the primary programming language to simulate the implementation of the CP-ABE algorithm and our proposed scheme.

Setting the number of attributes to vary from 10 to 50, the relationship between the number of attributes and the private key generation time is illustrated in the graphs below. As shown in Figure 3, with an increase in the number of attributes, our scheme shows a certain advantage in private key generation compared to the traditional CP-ABE scheme. As illustrated in Figure 4 and Figure 5, when the size of the encrypted/decrypted data file is set to 512B and the number of attributes gradually increases, our scheme outperforms the traditional CP-ABE scheme in terms of encryption/decryption time.

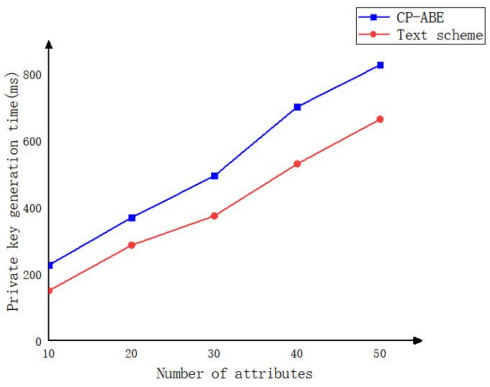

Figure 3. System architecture diagram

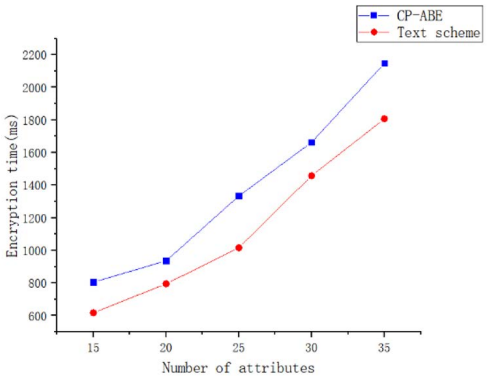

Figure 4. Encryption time

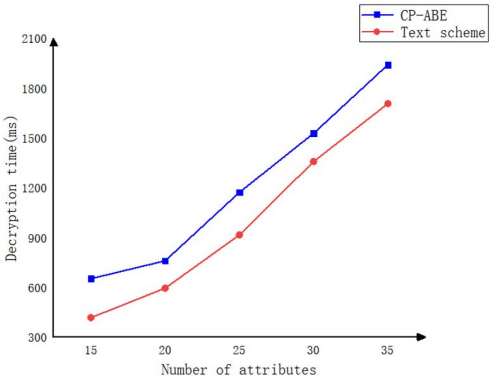

Figure 5. Decryption time

5 Conclusion

This paper proposes a hazardous materials circulation data sharing scheme based on blockchain and attribute encryption, featuring hierarchical data access control and dynamic updates. Through an integrated access structure, hazardous materials related institutions can encrypt multiple data files with hierarchical access relationships simultaneously. When the attributes of a hazardous materials transport company or regulatory department match part of the access control structure, they can obtain the data associated with that part through a single decryption. Additionally, by using blockchain to store access policies, dynamic updates to access policies after data sharing are enabled. Comparative analysis with other schemes through simulation experiments demonstrates that our scheme has lower computational and storage overhead. The model presented in this paper has a broad application prospect and can provide an effective solution for the application of blockchain and attribute encryption in the field of hazardous materials circulation data sharing.

Data Available

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] Z. Tan and C. Tang, "A survey on privacy protection techniques in blockchain system," *J. Guangzhou Univ.*, vol. 20, no. 4, pp. 1–15, 2021.

[2] C. Ge, Z. Liu, and L. Fang, "A blockchain based decentralized data security mechanism for the Internet of Things," *J. Parallel Distrib. Comput.*, vol. 141, pp. 1–9, 2020. https://doi.org/10.1016/j.jpdc.2020.03.005

[3] E. A. Franciscon, M. P. Nascimento, J. Granatyr, M. R. Weffort, O. R. Lessing, and E. E. Scalabrin, "A systematic literature review of blockchain architectures applied to public services," in *2019 IEEE 23rd International Conference on Computer Supported Cooperative Work in Design (CSCWD), Porto, Portugal*, 2019, pp. 33–38. https://doi.org/10.1109/CSCWD.2019.8791888

[4] R. Wang, W. T. Tsai, J. He, C. Liu, Q. Li, and E. Y. Deng, "A medical data sharing platform based on permissioned blockchains," in *Proceedings of the 2018 International Conference on Blockchain Technology and Application, New York, USA*, 2018, pp. 12–16. https://doi.org/10.1145/3301403.3301406

[5] L. Xu, N. Shah, L. Chen, N. Diallo, Z. Gao, Y. Lu, and W. D. Shi, "Enabling the sharing economy: Privacy respecting contract based on public blockchain," in *Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Ccontracts, Abu Dhabi, United Arab Emirates*, 2017, pp. 15–21. https://doi.org/10.1145/3055518.3055527

[6] S. Wu and J. Du, "Electronic medical record security sharing model based on blockchain," in *Proceedings of the 3rd International Conference on Cryptography, Security and Privacy, New York, USA*, 2019, pp. 13–17. https://doi.org/10.1145/3309074.3309079

[7] Y. Cao, F. Jia, and G. Manogaran, "Efficient traceability systems of steel products using blockchain-based industrial internet of things," *Trans. Ind. Inform.*, vol. 16, no. 9, pp. 6004–6012, 2019. https://doi.org/10.1109/TII.2019.2942211

[8] C. Zhuang, Q. Dai, and Y. Zhang, "BCPPT: A blockchain-based privacy-preserving and traceability identity management scheme for intellectual property," *Peer-to-Peer Netw. Appl.*, vol. 15, no. 1, pp. 724–738, 2022. https://doi.org/10.1007/s12083-021-01277-1

[9] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology–EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark*, 2005, pp. 457–473. https://doi.org/10.1007/11426639_27

[10] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security, New York, USA*, 2006, pp. 89–98. https://doi.org/10.1145/1180405.1180418

[11] S. Wang, K. Liang, J. K. Liu, Y. Jianping, and W. X. Xie, "Attribute-based data sharing scheme revisited in cloud computing," *Trans. Inf. Forensics Secur.*, vol. 11, no. 8, pp. 1661–1673, 2016. https://doi.org/10.1109/TIFS.2016.2549004

[12] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *2007 IEEE Symposium on Security and Privacy (SP'07), Berkeley, CA, USA*, 2007, pp. 321–334. https://doi.org/10.1109/SP.2007.11

[13] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *International Workshop on Public Key Cryptography*, 2011, pp. 53–70. https://doi.org/10.1007/978-3-642-19379-8_4

[14] H. Cui, R. H. Deng, J. Lai, X. Yi, and S. Nepal, "An efficient and expressive ciphertext-policy attribute-based encryption scheme with partially hidden access structures," *Comput. Netw.*, vol. 133, pp. 157–165, 2018. https://doi.org/10.1016/j.comnet.2018.01.034

[15] S. Belguith, N. Kaaniche, M. Laurent, A. Jemai, and R. Attia, "Phoabe: Securely outsourcing multi-authority attribute based encryption with policy hidden for cloud assisted IoT," *Comput. Netw.*, vol. 133, pp. 141–156, 2018. https://doi.org/10.1016/j.comnet.2018.01.036

[16] H. Xiong, Y. Zhao, L. Peng, H. Zhang, and K. H. Yeh, "Partially policy-hidden attribute-based broadcast encryption with secure delegation in edge computing," *Future Gener. Comput. Syst.*, vol. 97, pp. 453–461, 2019. https://doi.org/10.1016/j.future.2019.03.008

[17] X. Liu, J. Ma, J. Xiong, Q. Li, and J. Ma, "Ciphertext-policy weighted attribute based encryption for fine-grained access control," in *2013 5th International Conference On Intelligent Networking And Collaborative Systems, Xi'an, China*, Xi'an, China, 2013, pp. 51–57. https://doi.org/10.1109/INCoS.2013.18

[18] S. Wang, J. Zhou, J. Liu, J. Yu, J. Chen, and W. Xie, "An efficient file hierarchy attribute-based encryption scheme in cloud computing," *Trans. Inf. Forensics Secur.*, vol. 11, no. 6, pp. 1265–1277, 2017. https://doi.org/10.1109/tifs.2016.2523941

[19] S. Chen and R. Huang, "Attribute-based encryption supporting conjunctive keyword," *Comput. Eng. Sci.*, vol. 43, no. 7, pp. 1219–1225, 2021.

[20] A. Beimel, "Secure schemes for secret sharing and key distribution." PhD thesis, Technion-Israel Institute of Technology, 1996.