Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7787866, 12 pages
https://doi.org/10.1155/2022/7787866

Research Article

Privacy Protection of Task in Crowdsourcing: Policy-Hiding and Attribute Updating Attribute-Based Access Control Based on Blockchain

Kunwei Yang,[1] Bo Yang,[1] Yanwei Zhou,[1] Tao Wang,[1] and Linming Gong[2]

[1]*School of Computer Science, Shaanxi Normal University, Xi'an 710119, China*
[2]*School of Computer Science, Xi'an Polytechnic University, Xi'an 710048, China*

Correspondence should be addressed to Bo Yang; byang@snnu.edu.cn

Received 14 December 2021; Accepted 23 February 2022; Published 24 March 2022

Academic Editor: Qingqi Pei

Crowdsourcing is a new way to solve complex problems by using human intelligence. However, the tasks and user information privacy concerned in crowdsourcing have not been adequately addressed. It is necessary to design a privacy protection mechanism for tasks that need to be restricted to specific user groups. Ciphertext-policy attribute-based encryption (CP-ABE) is an efficient and feasible cryptographic tool, particularly for crowdsourcing systems. The encryptor can choose the access policy independently, which limits the scope of decryption users. At present, most CP-ABE schemes adopt a centralized management platform, which poses problems such as high trust-building costs, DDoS attacks, and single point of failure. In this paper, we propose a new access control scheme based on CP-ABE and blockchain, which has the properties of policy hiding and attribute updating. To protect the privacy of worker's attributes, we adopt a test algorithm based on a fully homomorphic cryptosystem to confidentially judge whether the worker's attribute lists match the hidden attributes policy in ciphertext or not before the decryption. Experiment results and comprehensive comparisons show that our mechanism is more flexible, private, and scalable than existing schemes.

1. Introduction

With the rapid development of science and technology, a new generation of information technology represented by 5G, Internet of Things, edge intelligence, and blockchain has emerged, which has realized a comprehensive link between people, machines, and things, and built new infrastructure, application mode, industrial ecology, and service system [1–7]. In the application of new technologies, people are most concerned about information security and privacy protection. Researchers have conducted extensive and in-depth research in many fields to promote the healthy development of related application technologies [8–13].

In recent years, crowdsourcing as a distributed problem-solving model has been widely concerned by researchers. As in Figure 1, the model consists of three parties: requesters, workers, and a crowdsourcing platform. Requester submits tasks through the crowdsourcing platform. Workers get tasks from the platform and get corresponding rewards after completing tasks. At present, many crowdsourcing applications are widely used, such as UBER and Waze [14]. These applications have been integrated into many aspects of daily life.

Due to the diversity of perceptual tasks, it is necessary to select appropriate policies for the privacy protection of specific perceptual tasks. However, the traditional crowdsourcing systems based on a centralized management platform have the following weaknesses. First, it is vulnerable to DDoS attacks, remote hijackings, and so on. Second, there is a potential danger of a single point of failure. Third, users' sensitive information and task solutions are exposed to the risks of leakage. Fourth, from the perspective of privacy

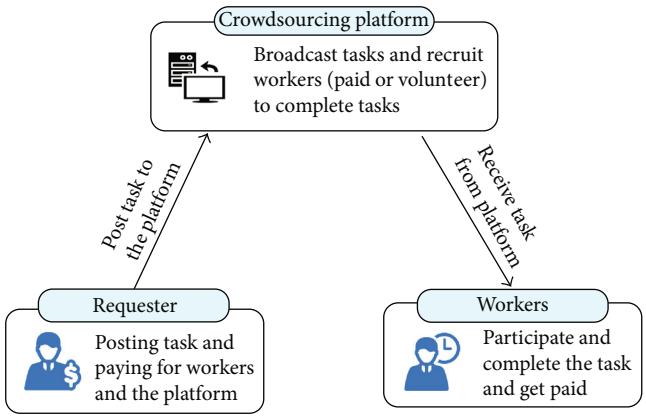

Figure 1: The model of crowdsourcing.

protection, it is impossible to implement the task hiding publishing mechanism within a limited receiver scope. The above problems seriously restrict the development of crowdsourcing system applications.

To solve these problems, we use blockchain [15] and CP-ABE [16] technologies to construct a credible task hiding access control mechanism in crowdsourcing. By using cryptography, timestamps, consensus mechanism, and incentives mechanism, blockchain enables point-to-point transaction and collaboration. Applying blockchain technology to a crowdsourcing system solves problems such as trust, incentive, and decentralization. CP-ABE is a branch of ABE mechanism [17], in which the access control policy is determined by an encryptor. The user's decryption key is associated with its own attribute set, which can be decrypted correctly only when the user attributes meet the policy requirements. Due to this special property, CP-ABE has been widely used in various scenarios.

In this paper, we integrate blockchain with CP-ABE cryptosystems to implement an access control mechanism for task privacy publishing. The main innovations of this paper are summarized as follows.

(i) We propose a new access control mechanism based on blockchain under the crowdsensing system, which has decentralized, fine-grained, flexible, and security features. It can be applied in the task distribution phase that ensuring crowdsourcing tasks can only be accessed by users who meet the requirements of the access policy. The system establishes a secure way to exercise access control between requesters and workers, which is more suitable for complex crowdsensing systems than other schemes

(ii) We propose a new CP-ABE scheme with policy hiding property. In the encryption phase, the access policy is not displayed in the ciphertext tuple, which ensures the privacy of the access policy. Different from the present CP-ABE schemes, in which decryptors need to do excessive calculations to determine whether their attributes meet the access policy in the decryption phase, the new scheme adds a test phase before the decryption. With much less computation than decryption itself, the test uses a fully homomorphic cryptosystem to ensure the privacy of policy and attributes

(iii) Our CP-ABE scheme features attribute updating. It is very flexible to add new attributes in the access policy because we can only generate the public key parameters for the new attributes and the existing public key can remain unchanged. To decrypt the ciphertext with newly added attributes in the access policy, the decryptors must obtain a new secret key including the newly added attributes again. The feature of attribute updating can greatly improve the flexibility and scalability of our scheme

The remainder of the paper is organized as follows. In Section 2, we present the related work. The preliminaries are given in Section 3, an overview of the new access control system in Section 4. Section 5 describes the proposed system, including an access control mechanism and a CP-ABE scheme. We analyze the security of our access control system in Section 6 and make comparisons and performance evaluation of our scheme in Section 7. The last section is the conclusion.

2. Related Work

In a heterogeneous and complex network environment, research on access control technology is heading towards the direction of fine granularity, which takes into consideration users, resources, operation, environment, and other factors. In this section, we would introduce some related work in the field of blockchain-based access control and CP-ABE schemes.

2.1. Blockchain-Based Access Control. Because of the low management efficiency, lack of flexibility, poor scalability, and other problems existing in the current centralized access control mechanism, researchers began to pay attention to the blockchain-based access control mechanism. [18] proposed a privacy-preserving authorization management framework for IoT by using blockchain that enables users to own and control their data. The data access control is implemented through a series of transactions that are used to grant, get, delegate, and revoke access. [19] realized the transfer of user rights through blockchain transactions and stores the transaction results on the blockchain to ensure that the executed access authorization operation cannot be denied. In view of the data privacy problem of cloud, [20] proposed a fine-grained access control scheme based on the blockchain model and attribute-based cryptosystem, which has the nature of privacy-preserving and user-controlled. In the scheme, a smart contract is used to ensure the scalability of the access control. [21] proposed an access management system for cloud federations. It allows federated organizations to enforce attribute-based access control policies on their data in a privacy-preserving fashion. By using blockchain and Intel SGX trusted hardware, the integrity of the policy evaluation process in the scheme is

ensured. To address the problem that roles are used across organizations in the network, [22] used smart contract as the trusted basis in access control and employed the challenge-response mechanism to realize the verification of user roles. [23] proposed a blockchain-based big data access control mechanism based on the attribute-based access control (ABAC) model. To ensure the tamper-resistant, auditability, and verifiability of access control information, the scheme described a transaction-based access control policy and entity attribute information management method. [24] proposed an access control model called timely CP-ABE, where the user legitimacy is verified by blockchain nodes and file sharing is based on a CP-ABE scheme that adds temporal dimension. [25] proposed an access control mechanism in a smart grid scenario based on an identity-based combined encryption, signature, and signcryption scheme. A new consensus algorithm in the power system is designed to solve the key escrow problem. [26] proposed to define an access control system that guarantee the auditability of access control policy evaluation. The core idea of the scheme is to codify access control policies as smart contracts and deploy them on a blockchain. In this way, the decision process of the policy can be executed automatically. By using CP-ABE, [27] proposed a ciphertext policy and attribute hiding access control scheme based on blockchain. To ensure the attribute privacy during authorization validation, they use the ElGamal cryptosystem [28] to secretly match user attributes and access policies. However, we point out that the ElGamal cryptosystem does not have additive homomorphism; so, there are loopholes in the access policy match phase. In addition, the scheme adopts a secure channel to transmit secret key, which increases the communication cost.

2.2. CP-ABE Schemes. The notion of ABE was first introduced by Sahai and Waters in [17], where both ciphertexts and secret keys are associated with sets of attributes. There are two variants of ABE: key-policy ABE (KP-ABE) [29] and ciphertext policy ABE (CP-ABE) [16]. In a KP-ABE scheme, the decryption key is associated with an access policy, and the ciphertext is associated with a set of attributes. In the decryption phase, the ciphertext can be decrypted if and only if the attribute set of the ciphertext satisfies the access policy of the decryption key. On the contrary, in a CP-ABE scheme, the ciphertext is associated with an access policy, and the decryption key is generated over a set of attributes. CP-ABE is perfectly suitable for fine-grained access control environments because it enables data owners to formulate and enforce access policies themselves.

Since [17] proposed the first CP-ABE scheme, researchers have proposed many specific CP-ABE schemes. In the CP-ABE mechanism, the more complex the policy, the more complicated the system is designed, and the more difficult to obtain the security proof of the mechanism. Researches on CP-ABE focus mainly on the design of access policy, which is generally classified into AND gates, access tree, and LSSS matrix. Cheung and Newport [30] presented a CP-ABE-based AND gates that is proven to be secure under the standard model. Subsequently, Nishide et al. [31] and Emura et al. [32] realized policy hiding and efficiency improvement, respectively, on the basis of the scheme [30]. Lai et al. [33] proposed a ciphertext policy hiding CP-ABE scheme, which can be expressed as AND gates on multivalued attributes with wildcards, and proved that it is fully secure. Different from the above types, there are many CP-ABE schemes [16, 34–36] with tree, linear secret sharing scheme (LSSS), and 0-1 coding as access structures, which have strong policy expression ability. For application scenarios that access policies need to be hidden and updated, many anonymous ABE schemes [27, 37–39] have been proposed. In an anonymous ABE, the access policy is hidden so that the user has no idea about the attribute content in the policy. However, most previous schemes cannot securely add new attributes in the access policy because these schemes have a common flaw that a decryptor can combine all old secret key components to reconstruct the exponent for decryption.

3. Preliminaries

In this section, we showcase the associated basic knowledge. See Table 1 for the notations used herein.

3.1. Blockchain. Blockchain was originally known as the underlying technology of Bitcoin. It was not until 2015 that blockchain became a prominent concept by researchers. Consisting of peer-to-peer (P2P) network, consensus protocol, transaction, smart contract, and a series of other technologies, blockchain can provide a trusted and distributed network environment. This new technology has solved the security risks brought by the centralization model. Applications based on blockchain technology can provide a new direction to reduce the middleman role.

3.1.1. P2P Network. Unlike the traditional client/server mode, the P2P network is a net system in which information is exchanged entirely by nodes without a central server. In a P2P network, blockchain nodes can join and exit freely, and the network system can expand and shrink freely.

3.1.2. Consensus Protocol. The consensus mechanism is the cornerstone of blockchain and an important guarantee for the security of the blockchain system. It can be used to solve the consistency problem caused by block distributed storage. The consensus mechanism is the foundation for building trust in the blockchain and contains an incentive mechanism for the effective operation of the blockchain system. The common consensus protocol includes POW [15], POS [40], BFT [41], and mixture of various mechanisms.

3.1.3. Transaction. Blockchain adopts a transaction data model composed of input, output, and digital signature to ensure that every transaction can be tracked. Merkel hash tree is used to package and aggregate all transaction information in a period of time to ensure that the transaction data will not be tampered.

3.1.4. Smart Contract. The concept of smart contract was put forward by Nick Szabo [42] in 1994. It is a computer program that executes and verifies the contract in an

Table 1: Notation description.

Notations	Descriptions
G, G_T	Two cyclic multiplicative groups
$\mathbb{U} = \{A_1, A_2, \cdots, A_n\}$	The attribute universe
$\mathbb{A}$	An attribute set
$\mathbb{P}$	An access policy
A_i	An attribute
k	A security parameter
B.PK	A public key registered in blockchain
B.SK	A private key registered in blockchain
PK	A public key in CP-ABE
SK	A secret key in CP-ABE
MSK	A master secret key in CP-ABE
M	Task information
$E(\cdot)$	A fully homomorphic cryptosystem
H	A collision-free hash function
CT	A CP-ABE ciphertext
Tx	A transaction on the blockchain

information way. The birth of blockchain provides a credible execution environment for smart contracts and accelerates the development of smart contracts. At the same time, the application of smart contracts expands blockchain technology from the earliest monetary system to a wider range of practical application scenarios.

3.2. Attribute and Access Structure. We define the attribute and access structure as provided in [43]. Let $\mathbb{U} = \{A_1, A_2, \cdots, A_n\}$ represent the attribute universe and $\mathbb{A}$ represent the user's attribute set, where $\mathbb{A} \subset \mathbb{U}$. In this paper, we use n-bit string $a_1 a_2 \cdots a_n$ to represent the user's attribute information.

$$\begin{cases} a_i = 1, A_i \in \mathbb{A}, \\ a_i = 0, A_i \notin \mathbb{A}. \end{cases} \tag{1}$$

For instance, let $n = 5$. $\mathbb{A} = 10011$ means that the attribute set consists of the attributes $\{A_1, A_4, A_5\}$.

We also represent the access policy $\mathbb{P}$ with an n-bit string $b_1 b_2 \cdots b_n$ defined as follows.

$$\begin{cases} b_i = 1, A_i \in \mathbb{P}, \\ b_i = 0, A_i \notin \mathbb{P}. \end{cases} \tag{2}$$

For instance, let $n = 5$. The $\mathbb{P} = 10111$ means the access policy $\mathbb{P}$ requires $\{A_1, A_3, A_4, A_5\}$ attributes.

In the attribute universe, each of A_i can be represented in terms of a group element $g_i \in G$, where G is a cyclic group of order N. There is a $\alpha_i \in Z_N^*$ that satisfies $g_i = g^{\alpha_i}$. Let us assume that the $\mathbb{P}$ has z elements equal to 1. We compute $\mathbb{A} \cdot \mathbb{P} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$. This implies that if an attribute set $\mathbb{A}$ fulfills the access policy $\mathbb{P}$, then $\mathbb{A} \cdot \mathbb{P}$ must be z. On the contrary, if $\mathbb{A}$ does not fulfills $\mathbb{P}$, the result of $\mathbb{A} \cdot \mathbb{P}$ must be less than z.

3.3. Fully Homomorphic Cryptosystem. The concept of fully homomorphic encryption [44] was proposed by Rivest et al. in 1970s, and it has become an important technology to solve the security problem arising in cloud service. The way to construct such schemes has been a hard problem for cryptographers. The first fully homomorphic cryptosystem based on ideal lattice was proposed by [45]. This scheme can perform any computation on the encrypted data without decrypting and does not affect the confidentiality of the data. The fully homomorphic cryptosystem means that it satisfies the properties of both additive and multiplicative homomorphisms simultaneously. It can be expressed by the following formula.

$$f(E(m_1), E(m_2), \cdots, E(m_k)) = E(f(m_1, m_2, \cdots, m_k)). \tag{3}$$

If f is an arbitrary function, it is called fully homomorphic encryption.

3.4. Composite Order Bilinear Groups. The concept of composite order bilinear groups was proposed in [46]. In this paper, we select two prime numbers of equal length as the order of two subgroups of group G.

Let $\mathcal{G}(1^\kappa)$ be a group generation algorithm and κ denote a security parameter. The algorithm outputs public parameters $\mathbb{G} = (g, p, r, G, G_T, e(\cdot, \cdot))$, such that (1) g is a generator of G, (2) G and G_T are two multiplicative cyclic groups of prime order $N = pr$, and (3) $e : G \times G \longrightarrow G_T$ denotes a computable bilinear map that satisfies the following properties:

(i) Bilinearity: $e(u^a, v^b) = e(u, v)^{ab}$, for all $a, b \longleftarrow_R \mathbb{Z}_N^*$ and $u, v \longleftarrow_R G$

(ii) Nondegeneracy: $e(g, g) \neq 1_{G_T}$, where 1_{G_T} is the generator of G_T

(iii) Computability: given the elements $v, u \in G$, $e(u, v)$ can be computed efficiently

Furthermore, let G_p, G_r denote two subgroups of G with order p, r, and $h_p \in G_p$, $h_r \in G_r$. Then, we have $e(h_p, h_r) = 1$.

3.5. Complexity Assumptions. Let the definition of $\mathbb{G} = (g, N, p, r, G, G_T, e(\cdot, \cdot))$ be the same as above, and $\mathcal{S}$ be an adversary whose purpose is to solve DBDH problem.

Definition 1. (DBDH problem). For the exponents a, b, c, z randomly selected from $\mathbb{Z}_N^*$, two tuples $\mathcal{T}_1 = (g, g^a, g^b, g^c, e(g, g)^{abc})$ and $\mathcal{T}_0 = (g, g^a, g^b, g^c, e(g, g)^z)$ are computationally indistinguishable. The advantage of adversary $\mathcal{S}$ in solving DBDH problem is defined as

$$\mathrm{Adv}_{\mathcal{S}}^{\mathrm{DBDH}}(\kappa) = |\Pr[\mathcal{S}(\mathcal{T}_1) = 1] - \Pr[\mathcal{S}(\mathcal{T}_0) = 1]|. \tag{4}$$

We say that the DBDH assumption holds if for all adversaries $\mathcal{S}$, we have

$$\mathrm{Adv}_{\mathcal{S}}^{\mathrm{DBDH}}(\kappa) \leq \mathrm{negl}(\kappa). \tag{5}$$

3.6. Definition of CP-ABE Scheme. A CP-ABE scheme consists of Setup, Encrypt, KeyGen, and Decrypt algorithms. The specific process is as follows.

Setup(1^{κ}): the setup algorithm takes the security parameter κ as input and outputs a public key PK and a master secret key MK

KeyGen(MK, $\mathbb{A}$): given the master secret key MK and an attribute list $\mathbb{A}$, the keyGen algorithm returns a secret key $SK_{\mathbb{A}}$

Encrypt(PK, M, $\mathbb{P}$).: given PK, a message M, and an access policy $\mathbb{P}$, the encrypt algorithm returns a ciphertext CT

Decrypt(CT, $\mathrm{SK}_{\mathbb{A}}$): given a ciphertext CT and a secret key $\mathrm{SK}_{\mathbb{A}}$, the decrypt algorithm returns the message M if $\mathbb{A} \vDash \mathbb{P}$. Otherwise, it returns $\perp$ with overwhelming probability

Setup(1^{κ}): the setup algorithm takes the security parameter κ as input and outputs a public key PK and a master secret key MK

3.7. Security Model for CP-ABE. In this paper, we use the security model proposed by [30]. A CP-ABE scheme is secure against chosen plaintext attacks (CPA) if no probabilistic polynomial time adversary has a nonnegligible advantage in the following game.

(i) Init: the adversary defines the target access policy $\mathbb{P}$ and sends it to the challenger

(ii) Setup: the challenger executes the setup algorithm and sends PK to the adversary

(iii) Phase 1: the adversary makes key generation queries by submitting an attribute list $\mathbb{A}$. The challenger answers with a secret key $\mathrm{SK}_{\mathbb{A}}$ if $\mathbb{A} \nvDash \mathbb{P}$. The process can be repeated adaptively

(iv) Challenge: the adversary selects two equal-length messages M_0 and M_1 and sends them to the challenger. After selecting a random bit $\mu \in \{0, 1\}$, the challenger generates a ciphertext CT by encrypting M_{μ} with $\mathbb{P}$. Then, it sends CT to the adversary

(v) Phase 2: the adversary makes key generation queries, and the challenger answers as Phase 1

(vi) Guess: finally, the adversary outputs a guess $\mu' \in \{0, 1\}$. If $\mu' = \mu$, the challenger outputs 0; otherwise, it outputs 1

4. The Specific Process of Access Control System

4.1. System Architecture. Our system architecture is shown in Figure 2, which involves four entities, namely, requesters, workers, cloud, and blockchain.

Requesters need to set an access policy first and then encrypt the task information using our CP-ABE scheme. Finally, they post the ciphertext of the task to the cloud and send the ciphertext address of the task to the blockchain via a storage transaction. When the task is completed, the requester will reward workers accordingly.

Workers are a group of users with different attributes who compete for the task to get rewards. The attributes of the workers determine whether the workers can get the plaintext of the task.

Cloud has an extremely large storage capacity to offer data storage services. In our system, the cloud is a data storage platform for storing encrypted task information.

Blockchain is a distributed platform to record relevant transaction information, which is open and transparent to requesters and workers.

4.2. Specific Process. Next, we describe the specific process of the system in detail.

Step 1. In the first step, the requester performs the setup algorithm of CP-ABE as follows.

Setup(1^{κ}): our construction takes a security parameter κ as input and runs the group generator to get $(N = p\ r, G, G_T, e)$, where $G = G_p \times G_r$. It picks the generators g_p of G_p. Let the attribute universe $\mathbb{U} = \{A_1, A_2, \cdots, A_n\}$ be the set of n attributes. For each attribute A_i where $1 \leq i \leq n$, choose random values $\{\alpha_i\}_{1 \leq i \leq n}$ from Z_N^*, set $A_i = g_p^{\alpha_i}$. Algorithm selects random elements $\omega, \beta \in Z_N^*$. Let $Y = e(g_p, g_p)^{\omega}$ and $B = g_p^{\beta}$. The public key is $\mathrm{PK} = (g_p, Y, B, \{A_i\}_{1 \leq i \leq n})$. And the master secret key is $\mathrm{MSK} = (\omega, \beta, \{\alpha_i\}_{1 \leq i \leq n})$.

Step 2. The requester sets an access policy $\mathbb{P} = \{b_1, b_2, \cdots, b_n\}$ and encrypts the task by using Encryption algorithm.

Encrypt(PK, M, $\mathbb{P}$): the requester picks up random values $r_i \in Z_N^*$ for $b_i = 1$, set $r = \sum_{b_i=1} \boxtimes r_i$. Then, the requester computes $C_0 = B^r$, $\tilde{C} = \mathrm{MY}^r$, and $\{C_{i,1}, C_{i,2}\}$ as follows:

$$\left\{C_{i,1}, C_{i,2}\right\} = \begin{cases} \left\{g_p^{r_i}, A_i^{r_i}\right\}, & \text{if } b_i = 1, \\ \left\{T_i, T_i'\right\}, & \text{if } b_i = 0, \end{cases} \tag{6}$$

where $M \in G_T$, T_i, $T_i' \in_R G_r$. The ciphertext is $CT = (C_0, \tilde{C}, \{C_{i,1}, C_{i,2}\}_{1 \leq i \leq n})$.

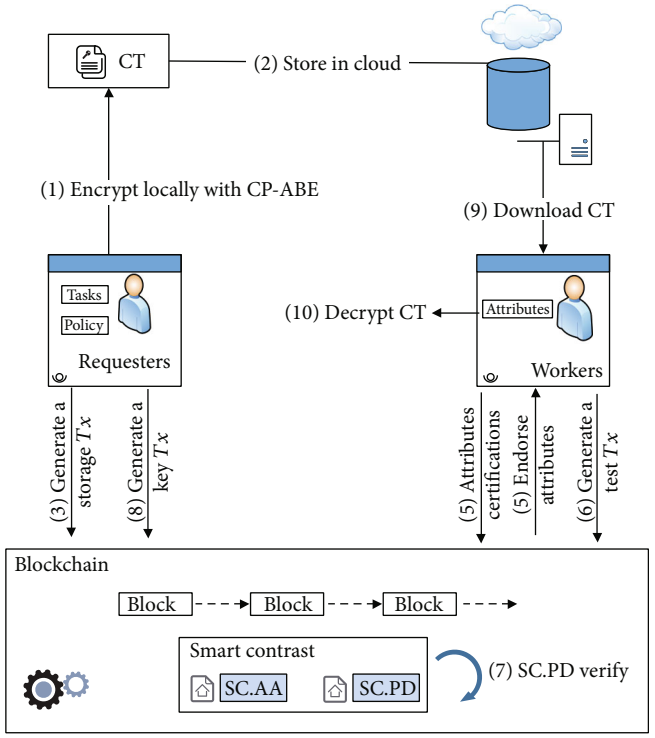

Figure 2: System architecture.

Step 3. The requester uploads the ciphertext of the task to the cloud server.

Step 4. In the storage phase, the requester only needs to publish the ciphertext address of the task to the blockchain through a storage transaction, to reduce the storage cost of the blockchain. The specific process is as follows.

The requester takes $(\mathrm{ID}_s, B_r.\mathrm{SK}, \mathrm{CT}, \mathrm{Address}_{\mathrm{CT}}, \mathbb{P}, z)$ as input, where ID_s is the identifier of the current storage transaction, $\mathrm{Address}_{\mathrm{CT}}$ is the cloud storage address of the ciphertext CT, and z is the number of attributes in the access policy. Then, the requester makes the following calculation.

(1) Calculate the hash value of the CT:

$$h_c = H(\mathrm{CT}). \tag{7}$$

(2) Calculate the ciphertext of access policy vector encrypted by the fully homomorphic cryptosystem

$$E(\mathbb{P}) = (E(b_1), E(b_2), \cdots, E(b_n). \tag{8}$$

(3) Calculate the hash value of the storage transaction:

$$h_s = H(ID_s, \mathrm{Address}_{\mathrm{CT}}, z, E(\mathbb{P}), h_c). \tag{9}$$

(4) Use the private key $B_r.\mathrm{SK}$ to sign the hash value of the storage transaction. It generates a sign (h_s).

Finally, the requester generates a transaction:

$$Tx_{\mathrm{storage}} = (ID_s, \mathrm{Address}_{\mathrm{CT}}, z, E(\mathbb{P}), h_c, \mathrm{sign}\ (h_s)) \tag{10}$$

and publishes it on the blockchain.

Step 5. Workers submit an attribute authentication transaction and get attribute certification by calling smart contract SC. AA that is used to provide attribute management services for workers. Then, workers' attributes are authenticated and endorsed by SC. AA. Meanwhile, the attributes of workers recorded on the blockchain will not change anymore.

Step 6. After getting the authentication of attributes, a worker sends the ciphertext of attributes to the blockchain via an attribute test transaction. It takes $(ID_t, B_w.\mathrm{SK}, \mathbb{A}, ID_s)$ as input and makes the following calculation.

(1) Calculate the ciphertext of the worker's attribute vector encrypted by a fully homomorphic cryptosystem

$$E(\mathbb{A}) = (E(a_1), E(a_2), \cdots, E(a_n)). \tag{11}$$

(2) Calculate the hash value of the test transaction

$$h_t = H(ID_t, E(\mathbb{A}), ID_s). \tag{12}$$

(3) Use the private key B_w.SK to sign the hash value of the test transaction. It generates sign (h_t).

Finally, the worker generates a transaction:

$$Tx_{\text{test}} = (ID_t, E(\mathbb{A}), ID_s, \text{sign}\ (h_t)) \tag{13}$$

and publishes it on the blockchain.

Step 7. It can get $E(\mathbb{A})$, $E(\mathbb{P})$, z through the identifier ID_t and ID_s. The smart contract SC.PD calculates $E(\mathbb{A}) \times E(\mathbb{P})$ by using the above fully homomorphic cryptosystem.

$$\begin{aligned} E(\mathbb{A}) \times E(\mathbb{P}) &= E(a_1) \cdot E(b_1) + E(a_2) \cdot E(b_2) + \cdots + E(a_n) \cdot E(b_n) \\ &= E(a_1 b_1 + E(a_2 b_2) + \cdots + E(a_n b_n) = E\left(z'\right). \end{aligned} \tag{14}$$

Finally, the smart contract SC. PD returns the value$E(z')$.

Step 8. The requester decrypts $E(z')$ to get z'. If $z' = z$, it can determine that the worker's attributes satisfy the CP-ABE policy. Then, the requester will run the KeyGen algorithm to generate the CP-ABE secret key $\text{SK}_{\mathbb{A}}$. Afterwards, the requester encrypts $\text{SK}_{\mathbb{A}}$ with B_w.PK and sends the ciphertext of $SK_{\mathbb{A}}$ to the blockchain.

KeyGen(MSK, $\mathbb{A}$). Firstly, the requester selects a random s from Z_N^* and let $D_0 = g_p^{\omega+s/\beta}$. Then, for every $i \in \{1, \cdots, n\}$, the requester picks up random values $\lambda_i \in Z_N^*$ and computes

$$\left\{D_{i,1}, D_{i,2}\right\} = \left\{g_p^{s+\alpha_i \lambda_i}, g_p^{\lambda_i}\right\}. \tag{15}$$

The secret key $SK_{\mathbb{A}} = (D_0, \{D_{i,1}, D_{i,2}\}_{1 \le i \le n})$.

Step 9. The worker downloads the ciphertext of the task from the cloud via CT address of Tx_{storage}. At the same time, the worker can decrypt and obtain $\text{SK}_{\mathbb{A}}$ by using B_w.SK.

Step 10. By using the private key $\text{SK}_{\mathbb{A}}$, the worker runs the decryption algorithm Decrypt to obtain the task data.

Derypt(CT, $\text{SK}_{\mathbb{A}}$): the worker tries to decrypt the CT $= (C_0, \tilde{C}, \{C_{i,1}, C_{i,2}\}_{1 \le i \le n})$ without knowing $\mathbb{P}$ by using his $\text{SK}_{\mathbb{A}} = (D_0, \{D_{i,1}, D_{i,2}\}_{1 \le i \le n})$ associated with the attribute list $\mathbb{A}$. The decryption process is as follows:

$$\begin{aligned} M &= \frac{\tilde{C} \prod_{i=1}^{n} e(C_{i,1}, D_{i,1})}{e(C_0, D_0) \prod_{i=1}^{n} \boxtimes e(C_{i,2}, D_{i,2})} \\ &= \frac{M \cdot e\left(g_p, g_p\right)^{\omega \cdot r} \prod_{b_i=1} e\left(g_p^{r_i}, g_p^{s+\alpha_i \cdot \lambda_i}\right)}{e\left(g_p^{\beta \cdot r}, g_p^{\omega+s/\beta}\right) \prod_{b_i=1} e\left(g_p^{\alpha_i \cdot r_i}, g_p^{\lambda_i}\right)} \\ &= \frac{M \cdot \prod_{b_i=1} e\left(g_p^{r_i}, g_p^{s}\right)}{e\left(g_p^{r}, g_p^{s}\right)} = M. \end{aligned} \tag{16}$$

4.3. Main Idea. Through such a CP-ABE scheme, we can achieve a policy hiding, updatable, and fine-grained access control scheme in crowdsourcing. For requesters, they only want workers who meet the policy requirements to get the task. For workers, they need to do a policy test confidentially to prove whether their attributes meet the policy requirements. The implementation of policy hiding and attribute updating is as follows.

To achieve the goal of policy hiding, it means that the ciphertext CT does not contain policy $\mathbb{P}$, and the workers can decrypt CT without knowing the access policy $\mathbb{P}$. As mentioned above, we use $n-$bit string $a_1 a_2 \cdots a_n$ and $b_1 b_2 \cdots b_n$ to represent the user's attribute set $\mathbb{A}$ and access policy $\mathbb{P}$, respectively. We chose a composite order group G with order $N = pr$. In the encrypt phase, if $b_i = 1$, let $[C_{i,1}, C_{i,2}]$ be well-formed parameters chosen from G_p. If $b_i = 0$, we set $[C_{i,1}, C_{i,2}]$ as two random elements of G_r. If an attribute A_i is required in a policy, then the worker's attribute set must also have that attribute to be validated by SC.PD. Each worker that satisfies the attribute can obtain the decryption key. Thus, if the set of attributes for the workers meet the policy requirements, it does not need to know the access policy $\mathbb{P}$ to complete the decryption.

The attribute update feature requires that it is easy to update the attribute information in the access policy, even after the setup phase is executed. The reason the previous scheme does not have this feature is that the decryptors can use their old secret key that does not contain new attributes to decrypt the new ciphertext. These schemes have a common flaw that the decryptors may combine all old secret key components to reconstruct the exponent of the secret key for decryption. In our scheme, we embed the composition factor r_i of the random number r in the ciphertext tuple instead of the private key, which forces workers to obtain the private key components corresponding to all attributes specified in the new access policy. If any new attributes were added into the access policy after the workers got their private keys, they cannot decrypt correctly until getting the new secret key components.

5. Security Analysis

Our scheme satisfies several security properties, and the specific analysis is as follows.

(1) Collusion resistance: for an attribute-based encryption scheme, it is very important to prevent collusion attacks between adversaries. In our CP-ABE scheme, to decrypt the ciphertext, adversaries have to get $e(g_p^r, g_p^s)$. When an adversary does not possess an attribute, he needs to conspire with a coconspirator who possesses the attribute. However, in the process of secret key generation, the private key of different adversaries uses different random numbers. By means of collusion, the adversary must get $\prod_{a_i \neq 0} e(g_p^{r_i}, g_p^s) \cdot e(g_p^{r_i}, g_p^{s^*})$ in the decryption phase. However, s is the random number in the adversary's secret key, and s^* is the random number in the coconspirator's secret key; so, they cannot calculate $e(g_p^r, g_p^s)$ together. That is why this scheme has the character of resisting collusion attacks

(2) Task confidentiality: this is the most basic security feature to ensure the security of the task. In our system, the task is encrypted and uploaded to the cloud server platform, which is considered to be curious. In the worst case, the cloud server platform may attempt to restore the task information; however, it either does not have a secret key or the attribute does not satisfy the access policy. Therefore, the scheme can ensure the privacy of the task

(3) Decentration: by using blockchain, we can realize an end-to-end crowdsourcing task management. In this process, requesters and workers can interact directly. It avoids DDoS attacks, single point of failure, and leakage of important data that may be encountered on a centralized management platform

(4) Policy privacy protection: in our scheme, the policy is treated as private data that need to be protected. We propose the CP-ABE with policy-hiding property. In the encryption phase, the generated ciphertext does not contain access policy information, which can protect the privacy of the policy. To authenticate the worker's attributes, a policy test algorithm that uses a fully homomorphic cryptosystem is adopted to estimate whether the attribute lists match the hidden attributes policy in ciphertext or not before the decryption. Such an approach ensures the privacy of the policy

(5) Integrity and traceability: our scheme can ensure the integrity of task data and the traceability of access control information through blockchain. Workers can compare the hash value of the task ciphertext in the cloud server platform with the information stored in the blockchain to determine whether the task has been modified. Meanwhile, all authorization records are stored as immutable access transactions in the blockchain; therefore, no one can deny their behavior

(6) Security analysis of CP-ABE: we now prove that the above CP-ABE scheme is selectively secure under the DBDH assumption

Theorem 1. *Assume that there is a probabilistic polynomial-time adversary $\mathcal{S}$ which can break out our CP-ABE scheme in a chosen plaintext attacks model with nonnegligible advantage $\varepsilon(\kappa)$, then a simulator can be constructed to distinguish the DBDH tuple $(g, g^a, g^b, g^c, e(g,g)^{abc})$ from the random tuple $(g, g^a, g^b, g^c, e(g,g)^z)$ with nonnegligible advantage $1/2\varepsilon(\kappa)$.*

Proof. We first let the Sim set the security parameter κ and run the group generator to get the public parameters $(N = pr, g_p, G, G_T, e)$, where $G = G_p \times G_r$, $g_p \in G_p$. The DBDH challenger gives a DBDH tuple $(g_p, g_p^a, g_p^b, g_p^c, Z) \in G^4 \times G_T$ to Sim, where Z is either $e(g_p, g_p)^{abc}$ or $e(g_p, g_p)^z$ with equal probability. The Sim proceeds as follows:

(i) Init: during this phase, Sim receives the challenge access policy $\mathbb{P}$ from $\mathcal{S}$

(ii) Setup: to provide a public key to $\mathcal{S}$, Sim sets Y to be $e(g_p^a, g_p^b) = e(g_p, g_p)^{ab}$. This implies $\omega = ab$. Let the attribute universe be $\mathbb{U} = \{A_1, A_2, \cdots, A_n\}$, Sim chooses random $\alpha_1, \cdots, \alpha_n, \beta \in Z_N^*$, sets $\{A_i = g_p^{\alpha_i}\}_{1 \le i \le n}$ and $B = g_p^{\beta}$. Then, Sim publishes PK as in the real scheme

(iii) Phase 1: $\mathcal{S}$ submits a set $\mathbb{A} = \{a_1, a_2, \cdots, a_n\}$, provided $\mathbb{A} \vDash \mathbb{P}$, and it means that there is at least one $k \in \{1, \cdots, n\}$ that satisfies $b_k = 1$, but $a_k = 0$. Sim answers with a secret key $SK_{\mathbb{A}}$ for $\mathbb{A}$ as follows:

Sim picks up random values $s' \in Z_N^*$. For every $i \in \{1, \cdots, n\}$, Sim chooses random $\lambda_i' \in Z_N^*$. It sets $s = \beta s' - \omega$, and the D_0 component of the secret key can be computed as $D_0 = g_p^{\omega + s/\beta} = g_p^{s'}$. Sim computes the components $\{D_{i,1}, D_{i,2}\} = \{g_p^{\beta s' + \alpha_i \lambda_i'}, g_p^{\lambda_i' + (\omega/\alpha_i)}\}$. When $\lambda_i = \lambda_i' + (\omega/\alpha_i)$, the components $\{D_{i,1}, D_{i,2}\}$ are as follows:

$$D_{i,1} = g_p^{\beta s' + \alpha_i \lambda_i'} = g_p^{\beta s' + \alpha_i \left(\lambda_i - \frac{\omega}{\alpha_i}\right)} = g_p^{s + \alpha_i \lambda_i},$$
$$D_{i,2} = g_p^{\lambda_i' + \frac{\omega}{\alpha_i}} = g_p^{\lambda_i}. \tag{17}$$

The secret key $SK_{\mathbb{A}} = (D_0, \{D_{i,1}, D_{i,2}\}_{1 \le i \le n})$.

(iv) Challenge. $\mathcal{S}$ submits two challenge messages M_0 and M_1 of equal length. Sim chooses $\mu \in \{0, 1\}$ at random and encrypts M_μ based on $\mathbb{P}$. Then, sets $C_0 = (g^c)^\beta$, $\tilde{C} = M_\mu Z$. For the policy $\mathbb{P}$, Sim continues chooses random values $r_i \in Z_N^*$ for $b_i = 1$, set $r = \sum_{b_i = 1} r_i$. Obviously, Sim can compute the ciphertext components $\{C_{i,1}, C_{i,2}\}$ easily

(v) Phase 2: same as Phase 1

(vi) Guess: $\mathcal{S}$ produces a guess μ' of μ. If $\mu' = \mu$, Sim

outputs 1 and otherwise outputs 0. If $Z = e(g_p, g_p)^{abc}$, then CT is a valid ciphertext, in which case the advantage of $\mathcal{S}$ is ε. Hence,

$$\left|\Pr\left[\mu' = \mu \mid Z = e\left(g_p, g_p\right)^{abc}\right] - \frac{1}{2}\right| = \varepsilon(k). \quad (18)$$

If $Z = e(g_p, g_p)^z$, then $\tilde{C}$ is completely random from the view of $\mathcal{S}$. Therefore,

$$\left|\Pr\left[\mu' = \mu \mid Z = e\left(g_p, g_p\right)^{z}\right] - \frac{1}{2}\right| = \text{negl}(k), \quad (19)$$

where $\text{negl}(k)$ is negligible. Hence,

$$\begin{aligned}
&\Pr\left[\mu' = \mu\right] = \Pr\left[Z = e\left(g_p, g_p\right)^{abc}\right] \\
&\Pr\left[\mu' = \mu \mid Z = e\left(g_p, g_p\right)^{abc}\right] + \Pr\left[Z = e\left(g_p, g_p\right)^{z}\right] \\
&\Pr\left[\mu' = \mu \mid Z = e\left(g_p, g_p\right)^{z}\right] = \frac{1}{2}\left(\frac{1}{2} \pm \varepsilon(k)\right) \\
&+ \frac{1}{2}\left(\frac{1}{2} \pm \text{negl}(k)\right) = \frac{1}{2} \pm \frac{1}{2}\varepsilon(k) \pm \frac{1}{2}\text{negl}(k), \text{Adv}_{\mathcal{S}}^{\text{CPA}}(k) \\
&= \left|\Pr\left[\mu' = \mu\right] - \frac{1}{2}\right| = \frac{1}{2}\left|\varepsilon(k) \pm negl(k)\right| \approx \frac{1}{2}\varepsilon(k). \quad (20)
\end{aligned}$$

From the above analysis, we can see that the Sim's advantage in the DBDH game is $1/2\varepsilon(k)$. □

6. Comparisons and Efficiency Evaluation

In this section, we first make comprehensive comparisons of our scheme with related work in terms of security, efficiency, and performance features. Then, we implement our CP-ABE scheme to analyze the efficiency of the algorithm.

6.1. Comparisons. We make a horizontal comparison with the relevant blockchain-based access control schemes in terms of important features, including policy privacy, fine granularity, attribute update, policy test, decentration, and framework. As seen in Table 2, these schemes do not consider the privacy of the policy except [27]. There is a potential problem that the attacker may infer the scope of the user group through the policy information. Centralized entities are introduced in schemes [22, 25], which results in some privacy and security concerns. Meanwhile, these three schemes can only achieve coarse-grained access control. [19, 23] support dynamic updating of policy attributes, which makes the schemes extensible. Only the scheme [27] is capable of supporting the policy test to judge whether the attribute lists match the hidden attributes policy in ciphertext or not before the decryption. However, there is an obvious mistake in the paper that the ElGamal cryptosystem does not have additive homomorphism; so, there are loopholes in the access policy match phase. Therefore, none of these blockchain-based access control schemes can support policy hiding and testing. Our scheme adopts a policy-hiding CP-ABE scheme and a fully homomorphic cryptosystem to realize policy hiding and testing. At the same time, our scheme supports attribute updating. Furthermore, a secure communication channel is not necessary anymore in our system.

Table 2: Comparisons of blockchain-based access control schemes.

Scheme	Policy privacy	Fine granularity	Attribute update	Policy test	Decentration
[22]	×	×	×	×	×
[23]	×	✓	✓	×	✓
[26]	×	✓	×	×	✓
[25]	×	×	×	×	×
[19]	×	✓	✓	×	✓
[18]	×	✓	×	×	✓
[24]	×	✓	×	×	✓
[27]	✓	✓	×	✓	✓
Our	✓	✓	✓	✓	✓

Aiming at the efficiency problem, we compare our CP-ABE scheme with some related CP-ABE schemes. In Table 3, the symbols PK, MK, SK, and CT represent the public key, the master secret key, the secret key, and the task ciphertext, respectively. We use L_G, L_{G_T}, and L_{Z_p} to denote the number of groups G, the target group G_T, and prime group Z_q, respectively. Let n, l, $|A_{\mathbb{A}}|$, $|A_{\mathbb{P}}|$ denote the number of attributes, the number of elements in an attribute category, the number of elements in the user A's attribute set, and the number of attributes in the policy set. Through the horizontal comparison, we can see that under the premise of obtaining the relevant security features, our scheme does not reduce the efficiency and even outperforms the relevant schemes in terms of ciphertext size. It is better for saving storage space on the cloud.

6.2. Efficiency Evaluation. We implement our CP-ABE scheme based on pairing-based cryptography (PBC) library on a laptop with Windows 10, Intel Core i5-8250U CPU, 2.90 GHz, and 16 GB RAM. The size of public parameters and message size is important indicators to evaluate the calculated performance of a CP-ABE scheme. In this experiment, we use type A1 pairing and let the composite N be the universe size. The composite N in our experiments consists of two prime numbers of 517 bits, which means that $|Z_N| = |G| = |G_T| = 1024$ bits.

Our main concern is how the efficiency of the scheme changes with the increase of the number of attributes. The execution result of the algorithm is shown in Figure 3. For each phase, we run the algorithm 10 times and then adopt the average value. As is illustrated, the two phase algorithm time increases as the number of attributes grows. This is due to the calculation of variables in each algorithm

Table 3: Comparison of size of keys and ciphertext in CP-ABE scheme.

Scheme	PK	MK	SK	CT
[30]	$(3n+1)L_G + L_{G_T}$	$(3n+1)L_{Z_q}$	$(2n+1)L_G$	$(n+1)L_G + L_{G_T}$
[31]	$(nl+1)L_G + L_{G_T}$	$(nl)L_G + 2L_{Z_q}$	$(2n+1)L_G$	$(nl+n+1)L_G + L_{G_T}$
[17]	$3L_G + L_{G_T}$	$L_G + L_{Z_q}$	$(2\|A_{\mathbb{A}}\|+1)L_G$	$(2\,\|\,A_{\mathbb{P}}\|+1)L_G + L_{G_T}$
[32]	$(nl+2)L_G + L_{G_T}$	$(nl+1)L_{Z_q}$	$2L_G$	$2L_G + L_{G_T}$
[47]	$(n+1)L_G + L_{G_T}$	$(n+1)L_{Z_q}$	$(\|A_{\mathbb{A}}\|+1)L_G$	$(\|A_{\mathbb{P}}\|+1)L_G + L_{G_T}$
[36]	$(n+2)L_G + L_{G_T}$	L_G	$(\|A_{\mathbb{A}}\|+2)L_G$	$(2\|A_{\mathbb{P}}\|+1)L_G + L_{G_T}$
[35]	$(n+2)L_G + L_{G_T}$	$L_G + L_{Z_q}$	$(\|A_{\mathbb{A}}\|+2)L_G$	$(2\|A_{\mathbb{P}}\|+1)L_G + L_{G_T}$
[27]	$3L_G + L_{G_T}$	$L_G + 2L_{Z_q}$	$(n+1)L_G$	$(nl+1)L_G + L_{G_T}$
Our	$(n+2)L_G + L_{G_T}$	$(n+2)L_{Z_q}$	$(2n+1)L_G$	$(2n+1)L_G + L_{G_T}$

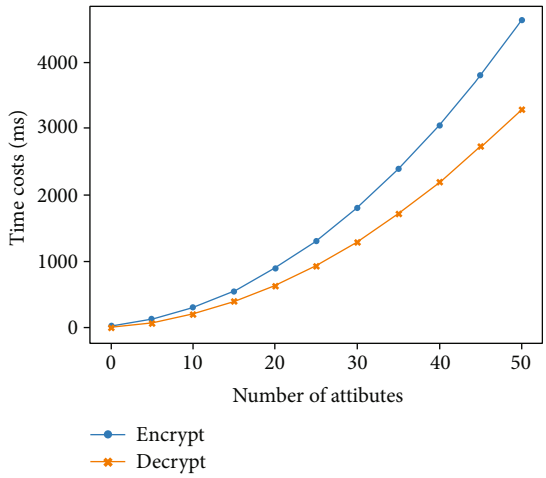

Figure 3: Performance analysis of our CP-ABE scheme.

depending on the number of attributes. Compared with other algorithms, the advantage of our algorithm is that it can support attribute update, policy hide, policy test, and other properties at the same time, while the efficiency of the algorithm does not decrease too much. Therefore, our scheme is more suitable for the crowdsourcing system with higher requirements for privacy protection.

7. Conclusion and Future Work

In this paper, we present a privacy protection mechanism for tasks in crowdsourcing, which realizes autonomous access control by adding blockchain to avoid a series of problems faced by central institutions. To solve the privacy of crowdsourcing tasks and access policies, we propose a new CP-ABE scheme with an expressive AND gate access structure that supports policy hiding and attribute updating. At the same time, we adopt a test algorithm based on a fully homomorphic cryptosystem to confidentially judge whether the worker's attribute lists match the hidden attributes policy in ciphertext or not before the decryption. Compared with previous schemes, our scheme has more advantages in flexibility, scalability, and privacy. In the future, we will consider an expressive and constant-size attribute-based access control based on blockchain.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Bo Yang, Yanwei Zhou, Tao Wang, and Linming Gong contributed equally to this work.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant nos. U2001205, 61772326, 61802241, and 61802242) and the Fundamental Research Funds for the Central Universities (Grant nos. GK202003079, GK202007033, and 2020TS087).

References

[1] L. Liu, J. Feng, Q. Pei et al., "Blockchain-enabled secure data sharing scheme in mobile-edge computing: an asynchronous advantage actor-critic learning approach," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2342–2353, 2021.

[2] K. Yu, L. Tan, S. Mumtaz et al., "Securing critical infrastructures: deep-learning-based threat detection in iiot," *IEEE Communications Magazine*, vol. 59, no. 10, pp. 76–82, 2021.

[3] K. Yu, Z. Guo, Y. Shen, W. Wang, J. C.-W. Lin, and T. Sato, "Secure artificial intelligence of things for implicit group recommendations," *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 2698–2707, 2022.

[4] F. Ding, G. Zhu, M. Alazab, X. Li, and K. Yu, "Deep-learning-empowered digital forensics for edge consumer electronics in 5g hetnets," *IEEE Consumer Electronics Magazine*, vol. 11, no. 2, pp. 42–50, 2022.

[5] L. Liu, C. Chen, Q. Pei, S. Maharjan, and Y. Zhang, "Vehicular edge computing and networking: a survey," *Mobile Networks and Applications*, vol. 26, no. 3, pp. 1145–1168, 2021.

[6] L. Zhao, J. Li, A. Y. Al-Dubai, A. Y. Zomaya, G. Min, and A. Hawbani, "Routing schemes in software-defined vehicular networks: design, open issues and challenges," *IEEE Intelligent Transportation Systems Magazine*, vol. 13, no. 4, pp. 217–226, 2021.

[7] L. Liu, M. Zhao, M. Yu, M. A. Jan, D. Lan, and A. Taherkordi, "Mobility-aware multi-hop task offloading for autonomous driving in vehicular edge computing and networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 1–14, pp. 1–14, 2022.

[8] L. Tan, K. Yu, F. Ming, X. Chen, and G. Srivastava, "Secure and resilient Artificial intelligence of things: a honeynet approach for threat detection and situational awareness," *IEEE Consumer Electronics Magazine*, p. 1, 2021.

[9] L. Tan, K. Yu, N. Shi, C. Yang, W. Wei, and H. Lu, "Towards secure and privacy-preserving data sharing for covid-19 medical records: a blockchain-empowered approach," *IEEE Transactions on Network Science and Engineering*, vol. 9, no. 1, pp. 271–281, 2022.

[10] K. Yu, M. Arifuzzaman, Z. Wen, D. Zhang, and T. Sato, "A key management scheme for secure communications of information centric advanced metering infrastructure in smart grid," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 8, pp. 2072–2085, 2015.

[11] Q. Feng, D. He, S. Zeadally, M. K. Khan, and N. Kumar, "A survey on privacy protection in blockchain system," *Journal of Network and Computer Applications*, vol. 126, pp. 45–58, 2019.

[12] J. Feng, L. Liu, Q. Pei, and K. Li, "Min-max cost optimization for Efficient hierarchical federated learning in wireless edge networks," *IEEE Transactions on Parallel and Distributed Systems*, p. 1, 2022.

[13] L. Zhao, H. Li, N. Lin, M. Lin, C. Fan, and J. Shi, "Intelligent content caching strategy in autonomous driving toward 6g," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2021.

[14] G. Wang, B. Wang, T. Wang, A. Nika, H. Zheng, and B. Y. Zhao, "Defending against sybil devices in crowdsourced mapping services," in *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys 2016*, pp. 179–191, Singapore, 2016.

[15] S. Nakamoto, *Bitcoin: A peer-to-peer electronic cash system*, Technical Report, 2008, https://bitcoin.org/bitcoin.pdf.

[16] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *2007 IEEE Symposium on Security and Privacy (S&P 2007)*, pp. 321–334, Oakland, California, USA, 2007.

[17] 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*, vol. 3494, pp. 457–473, Aarhus, Denmark, 2005.

[18] A. Ouaddah, A. A. E. Kalam, and A. A. Ouahman, "Fairaccess: a new blockchain-based access control framework for the internet of things," *Security and communication networks*, vol. 9, no. 18, 5964 pages, 2016.

[19] D. D. F. Maesa, P. Mori, and L. Ricci, "Blockchain based access control," in *Distributed applications and interoperable systems -17th IFIP WG 6.1 international conference, DAIS 2017, held as part of the 12th international federated conference on distributed computing techniques, DisCoTec 2017*, vol. 10320, pp. 206–220, Neuchatel, Switzerland, 2017.

[20] Y. Zhang, D. He, and K. R. Choo, "Bads: Blockchain-based architecture for data sharing with ABS and CP-ABE in iot," *Wireless Communications and Mobile Computing*, vol. 2018, 127836589 pages, 2018.

[21] S. Alansari, F. Paci, and V. Sassone, "A distributed access control system for cloud federations," in *37th IEEE International Conference on Distributed Computing Systems, ICDCS 2017*, pp. 2131–2136, Atlanta, GA, USA, 2017.

[22] J. P. Cruz, Y. Kaji, and N. Yanai, "RBAC-SC: role-based access control using smart contract," *IEEE Access*, vol. 6, pp. 12240–12251, 2018.

[23] A. D. Liu, X. H. Du, N. Wang, and S. Z. Li, "Blockchain-based access control mechanism for big data," *Journal of Software*, vol. 9, pp. 2636–2654, 2019.

[24] M. Jemel and A. Serhrouchni, "Decentralized access control mechanism with temporal dimension based on blockchain," in *14th IEEE International Conference on e-Business Engineering, ICEBE 2017*, vol. 4-6, pp. 177–182, Shanghai, China, 2017.

[25] Y. Zhou, Y. Guan, Z. Zhang, and F. Li, "A blockchain-based access control scheme for smart grids," in *2019 International Conference on Networking and Network Applications, NaNA 2019*, pp. 368–373, Daegu, Korea (South), 2019.

[26] D. D. F. Maesa, P. Mori, and L. Ricci, "A blockchain based approach for the definition of auditable access control systems," *Computers & Security*, vol. 84, pp. 93–119, 2019.

[27] S. Gao, G. Piao, J. Zhu, X. Ma, and J. Ma, "Trustaccess: a trustworthy secure ciphertext-policy and attribute hiding access control scheme based on blockchain," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 5784–5798, 2020.

[28] T. E. Gamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

[29] 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, CCS 2006*, pp. 89–98, Alexandria, VA, USA, 2006.

[30] L. Cheung and C. C. Newport, "Provably secure ciphertext policy ABE," in *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007*, pp. 456–465, Alexandria, Virginia, USA, 2007.

[31] T. Nishide, K. Yoneyama, and K. Ohta, "Attribute-based encryption with partially hidden encryptor-specified access structures," in *Applied cryptography and network security, 6th international conference, ACNS 2008*, vol. 5037, pp. 111–129, New York, NY, USA, 2008.

[32] K. Emura, A. Miyaji, A. Nomura, K. Omote, and M. Soshi, "A ciphertext policy attribute-based encryption scheme with constant ciphertext length," in *Information security practice and experience, 5th international conference, ISPEC 2009*, vol. 5451, pp. 13–23, Xi'an, China, 2009.

[33] J. Lai, R. H. Deng, and Y. Li, "Fully secure cipertext-policy hiding CP-ABE," in *Information security practice and experience -7th international conference, ISPEC 2011*, vol. 6672, pp. 24–39, Guangzhou, China, 2011.

[34] V. Goyal, A. Jain, O. Pandey, and A. Sahai, "Bounded ciphertext policy attribute based encryption," in *Automata,*

Languages and Programming, 35th International Colloquium, ICALP 2008, vol. 5126, pp. 579–591, Reykjavik, Iceland, 2008.

[35] A. B. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption," in *Advances in cryptology - EUROCRYPT 2010, 29th annual international conference on the theory and applications of cryptographic techniques*, vol. 6110, pp. 62–91, Monaco / French Riviera, 2010.

[36] B. Waters, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Public Key Cryptography - PKC 2011 - 14th International Conference on Practice and Theory in Public Key Cryptography*, vol. 6571, pp. 53–70, Taormina, Italy, 2011.

[37] Y. Zhang, D. Zheng, and R. H. Deng, "Security and privacy in smart health: Efficient policy-hiding attribute-based access control," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 2130–2145, 2018.

[38] J. Hao, C. Huang, J. Ni, H. Rong, M. Xian, and X. S. Shen, "Fine-grained data access control with attribute-hiding policy for cloud-based IoT," *Computer Networks*, vol. 153, pp. 1–10, 2019.

[39] J. Li, H. Wang, Y. Zhang, and J. Sheng, "Ciphertext-policy attribute-based encryption with hidden access policy and testing," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 7, pp. 3339–3352, 2016.

[40] J. Bonneau, A. Miller, J. Clark, A. Narayanan, J. A. Kroll, and E. W. Felten, "Sok: Research perspectives and challenges for bitcoin and cryptocurrencies," in *2015 IEEE Symposium on Security and Privacy, SP 2015*, pp. 104–121, San Jose, CA, USA, 2015.

[41] L. Lamport, R. E. Shostak, and M. C. Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4, no. 3, pp. 382–401, 1982.

[42] N. Szabo, "Formalizing and securing relationships on public networks," *First Monday*, vol. 2, no. 9, 1997.

[43] F. Guo, Y. Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "CP-ABE with constant-size keys for lightweight devices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 763–771, 2014.

[44] R. L. Rivest, L. Adleman, and M. L. Dertouzos, "On data banks and privacy homomorphisms," *Foundations of secure computation*, vol. 4, no. 11, pp. 169–180, 1978.

[45] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009*, pp. 169–178, Bethesda, MD, USA, 2009.

[46] D. Boneh, E. Goh, and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts," in *Theory of cryptography, second theory of cryptography conference, TCC 2005*, vol. 3378, pp. 325–341, Cambridge, MA, USA, 2005.

[47] L. Ibraimi, Q. Tang, P. H. Hartel, and W. Jonker, "Efficient and provable secure ciphertext-policy attribute-based encryption schemes," in *Information security practice and experience, 5th international conference, ISPEC 2009*, vol. 5451, pp. 1–12, Xi'an, China, 2009.