Data Access Control Scheme Based on Blockchain and Outsourced Verifiable Attribute-Based Encryption in Edge Computing

Chao Ma[1], Xiaojun Jin[2], Song Luo[1*], Yifei Wei[2], and Xiaojun Wang[3]
[1]China Academy of Information and Communications Technology, Beijing 100045, China
[e-mail: machao@caict.ac.cn, luosong@caict.ac.cn]
[2] Beijing University of Posts and Telecommunications, Beijing 100876, China
[e-mail: jinxiaojun@bupt.edu.cn, weiyifei@bupt.edu.cn]
3 Dublin City University, Dublin, Dublin 9, Ireland
[e-mail : xiaojun.wang@dcu.ie]
*Corresponding author: Song Luo

*Received April 13, 2022; revised July 3, 2023; accepted July 16, 2023;
published July 31, 2023*

Abstract

The arrival of the Internet of Things and 5G technology enables users to rely on edge computing platforms to process massive data. Data sharing based on edge computing refines the efficiency of data collection and analysis, saves the communication cost of data transmission back and forth, but also causes the privacy leakage of a lot of user data. Based on attribute-based encryption and blockchain technology, we design a fine-grained access control scheme for data in edge computing, which has the characteristics of verifiability, support for outsourcing decryption and user attribute revocation. User attributes are authorized by multi-attribute authorization, and the calculation of outsourcing decryption in attribute encryption is completed by edge server, which reduces the computing cost of end users. Meanwhile, We implemented the user's attribute revocation process through the dual encryption process of attribute authority and blockchain. Compared with other schemes, our scheme can manage users' attributes more flexibly. Blockchain technology also ensures the verifiability in the process of outsourcing decryption, which reduces the space occupied by ciphertext compared with other schemes. Meanwhile, the user attribute revocation scheme realizes the dynamic management of user attribute and protects the privacy of user attribute.

Keywords: Blockchain; CP-ABE; Edge Computing; Date Sharing; Attribute Revocation

http://doi.org/10.3837/tiis.2023.07.011 **ISSN : 1976-7277**

1. Introduction

Due to the increasing popularity of the Internet of Things (IoT), the access of massive devices forms a heterogeneous environment, which will generate a great deal data to be transmitted to the data center through the network at the same time, and the network delay is inevitable. Edge computing provides a new way to solve the high delay problem in the process of mass data transmission of Internet of Things devices. In edge computing, the edge nodes are set between the cloud server and the client, and the storage and computing tasks are completed by the edge nodes, which solve the real-time problem of the outsourcing system to a large extent. However, it also brings huge challenges of security and privacy protection[2]. Therefore, access control for IoT systems with edge nodes has become a new research hotspot. Most of the existing schemes use data encryption to ensure user security and privacy, but the traditional encryption interface algorithms have security problems in the edge computing environment with a large amount of tasks. As the security and reliability requirements of large-scale data acquisition become higher, how to acquire and manage massive data securely and efficiently in the edge computing environment has become a difficult problem.

Attribution-based encryption (ABE) schemes[6] include KP-ABE[7]and CP-ABE[8], among which CP-ABE applies to cloud computing scenarios in most cases. However, ABE algorithm has a defect in efficiency, that is the computational cost of decryption stage is very high, and the more attributes involved in the access strategy, the more time decryption will take. Since many devices in the Internet of Things do not have enough computing and storage capacity, this problem makes it difficult for ABE algorithm to be widely applied in mobile devices with limited resources. In order to solve the problem that ABE algorithm takes a long time to decrypt and is difficult to apply on devices with limited resources, [14] firstly proposed an outsourced attribute encryption scheme. In the scheme [14], the client provides the transformation key (also known as the outsourcing decryption key) to the cloud, and the cloud uses the transformation key to pre-decrypt the ABE ciphertext and obtain the pre-decrypted ciphertext.Although the cloud server is pre-decrypted, it still cannot get any information about the original data from the ciphertext obtained from pre-decryption. Users can decrypt the ciphertext pre-decrypted by the cloud and recover the original data from it, and the calculation process is very easy, thus greatly reducing the computing cost of users. The solution is essentially a two-stage decryption between the cloud and the user. Most of the computing cost of decryption is borne by the cloud, while only a small part of the computing cost is borne by the user. [15] adds verifiability on the basis of outsourcing decryption to ensure that users can effectively check whether the transformation is completed correctly. Blockchain is cutting-edge technology in the field of data sharing, as a kind of decentralized distributed ledger, using data logically independent space, based on a new kind of distributed architecture and computing paradigm has been used in the areas of data sharing, the combination of blockchain and margin calculation can offer edge security problem in the calculation of the solution. In terms of attribute cancellation, many studies directly revoke the user [16] instead of revoking the attributes of the user. This paper uses the blockchain to achieve the revocation of user attributes.

Few existing schemes have studied the use of attribute encryption for data access control in blockchain-based edge computing. This paper has made the following contributions on the basis of existing research:

1) We propose a multi-agency attribute authorization data access control architecture based on blockchain and attribute-based encryption in edge computing, which has the

characteristics of outsourcing decryption, verifiability, attribute revocation, protection of user attribute privacy.

2) We put forward the outsourcing decryption of multi-authorized agencies in the marginal calculation. It can be verified by the outsourcing decryption of the blockchain, which uses the blockchain to provide verification. It has an advantage in the storage space of the ciphertext compared to other methods.

3) We propose the revocation of user attribute keys based on blockchain, which can realize the attributes of the user through dynamically managing the attribute key of the user.

2. Related work

There are already some blockchain-based attribute encryption schemes. [1] proposed a access control scheme based on multi-authorization center ABE algorithm and blockchain. Authorization nodes in blockchain can be used as multiple attribute authorization centers. This distributed method effectively solves the problem that the authority of a single attribute authorization center is too large in the traditional ABE scheme and is prone to single point of failure. In this scheme, all data is encrypted using attribute-based encryption algorithms, and the encrypted data is stored in the blockchain. However, this method does not outsource the decryption process, so the application scenario will be limited.

How to implement effective access control scheme in the combination of blockchain and edge computing is also a hot research direction. In edge computing, there are a large number of network and computing resources distributed in the edge, and blockchain itself has the characteristics of decentralization and distribution. Integrating blockchain and edge computing into one system can provide users with safe and effective network, storage and computing services[4]. In[17], a multi-authorization outsourcing attribute encryption scheme (TLMO-ABE) containing time and location information is proposed. The innovation of this scheme is to add two factors of place and time to the attributes, so that the data user must access the corresponding data within the scope of time and place. The scheme also uses multiple attribute authority to manage user attributes, and the attribute key is generated and distributed by multiple attribute authority, which solves the security problem and performance bottleneck caused by a single authority. In order to achieve effective access control and management of the Internet of Things, [3] based on the existing traditional integrated blockchain and edge computing architecture, an access control model of the Internet of Things system including edge computing is proposed. Specifically, the scheme designs a three-tier IoT data access control architecture based on attribute encryption, while access control decisions are made by nodes on the blockchain by jointly executing smart contracts. The edge nodes in this scheme are responsible for encrypting, decrypting and transmitting data. The "SC-ABAC" access control model is proposed in [3] to realize the effective access control and management of the Internet of Things. However, this scheme does not deal with the environment in which the data itself exists and personal privacy information, so there may be privacy leakage problems. In [5], some nodes trusted by data consumer together calculate the user's attribute keys and store them. Then, in order to prevent key abuse problem, the key transfer transaction data structure is defined to realize the accountability of CP-ABE algorithm. Finally, attribute encryption and blockchain technology can be effectively combined, encryption and decryption process is carried out on the chain, sensitive text stored process is carried out on the chain, to achieve the chain on the chain of collaborative computing. However, this scheme has no outsourcing and verification process, so its application scope is limited. [12] proposed an efficient decryption access control scheme based on large domain attributes using cloud computing outsourcing decryption. The decryption process is completed by the cloud, which greatly reduces the ciphertext size, decryption time, and computing resource consumption. At the same time, the cloud has no access to any knowledge about the raw data.

There are some researches on attribute encryption based access control in blockchain. [9] proposed a WBAN privacy protection policy enhanced by mobile edge computing (MEC), and used attribute encryption for identity authentication to ensure the reliability of data sources. The scheme also designs a hybrid signature algorithm based on the decentralized MEC paradigm of blockchain, which realizes the efficient transmission of private information. The scheme also designs an optimized model of Merkle tree, which makes the source of the patient's medical data traceable and highly reliable by authenticating the nodes. Overall, the scheme has high reliability and is suitable for scenarios with high demand for data security.[10] given full consideration to the technical characteristics of blockchain, the malicious attack behavior in blockchain and the privacy issues involved in the access strategy of CP-ABE scheme were studied. The ciphertext, access structure and key structure are redesigned to protect the privacy information in the access policy. The solution also uses computing outsourcing to improve overall system efficiency. In [11], a new blockchain-based secure and trusted access control scheme TrustAccess that supports ciphertext policy and attribute privacy is proposed, which ensures the privacy of policy and attribute while realizing trusted access. Due to the low efficiency and poor scalability of the traditional CP-ABE scheme in the decryption process, [11] also proposed the OHP-CP-ABE optimization scheme, which meets the large-scale access requirements and has the scalability.

In the aspect of attribute revocation, some researches[17] directly revoke the user rather than revoke the user's attribute. In our scheme, the attribute revocation is realized by using blockchain.

3. Preliminaries

3.1. Bilinear maps

G_0 and G_1 are two multiplicative cyclic groups of prime order p. g is the generator of G_0 and e is a bilinear mapping, $e:G_0\times G_0\rightarrow G_1$. The bilinear mapping e has the following properties:

1) Bilinearity: $e(g^a, g^b)=e(g, g)^{ab}$, for all $g \in G_0$ and $a, b \in Z_p$

2) Nondegeneracy: $e(g, g)\neq 1$

3)Computable: For $\forall g_1, g_2 \in G_0$, there is an algorithm that makes $e(g_1, g_2)$ computable.

3.2. Attribute and Access Policy(J. Lai et al.2011[13])

Attribute and access policy are important part of attribute encryption. We assume that the attribute set of the system is $M=(V_1, ..., V_i, ..., V_n)$, and the attribute set of a user is $S=\{Att_1, Att_2, ..., Att_n\}$.Access policy can be expressed as $A = \{W_1, ..., W_i, ..., W_n\}$, in which $W_i \subseteq V_i$.

If the attribute set S of the user satisfies the access policy A, then k_i and $j_i \in W_i$ are present.

In the study of [13], both the access policy in ciphertext and the attribute set of users can be converted into vectors. The predicate OR(I1, I2) can be encoded as:

$$p(x_1, x_2) = (x_1 - I_1)\cdot(x_2 - I_2) \tag{1}$$

AND(I_1, I_2) can be encoded as:

$$p(x_1, x_2) = (x_1 - I_1) + r(x_2 - I_2), r \in Z_N \tag{2}$$

Through the OR and AND polynomial coding, tree structure of the access policy can be transformed to vector $\vec{x}$, similarly the user's attribute set of S can be converted into $\vec{v}$, when S meet T, we can get $\vec{x} \cdot \vec{v} = 0$.

4. System Model

This section introduces the overall architecture of the scheme proposed in this paper, including the specific implementation details of the scheme, and the implementation of smart contracts involving storage, outsourcing decryption and attribute revocation in the blockchain.

4.1. Architecture

The full nodes in the blockchain are the edge server, which have computing power and storage capacity. The data source is from the resource-constrained devices, which have limited computing and storage capacity. Each edge server collects data from multiple end devices and provides services. The edge computing environment may involve multiple domains, multiple attributes, terminal devices, edge servers, and attribute authority jointly form a blockchain system, on which immutable data sharing transaction records are stored. The system has the following roles:

(1) Edge nodes (ENs) :EN have storage and computing capabilities. It processes DU's access requests in real time, pre-decrypts the ciphertext requested by DU, and returns the pre-decrypted ciphertext to DU.

(2) Data owner (DO), DO is the maker of the access policy in ciphertext, encrypting to generate ciphertext for raw data and uploading it to blockchain. A DO can be an edge node or a user.

(3) Data user (DU), DU obtains the pre-decryption result from the EN end and is responsible for local decryption to obtain plaintext data.

(4) Attribute authorization(AA), AA is the attribute authorization authority and distributes the attribute private key to DU.

DO and DU can be either edge servers or terminals. Our scheme supports multiple AA Settings. DU can request different attribute keys from AAs, and these AAs don't need to communicate with each other. We used a decentralized method to carry out the key generation process, so that the key parts were issued by n independent AA for different vector elements v_i. We assume that AA_i generates attribute keys for attribute i, and then compose these keys into a corresponding attribute vector v = (v_1, v_i, ..., v_n), Attribute sets are represented as U={w_1, w_2, w_3, .., w_i... w_n}, w_i represents the *ith* class attribute, each class attribute corresponds to multiple attribute values. Suppose that w_i has j attribute values, denoted as $S_{w, i}$={$v_{i, 1}$, $v_{i, 2}$... $v_{i, j}$}, | $S_{w, i}$| = j. Suppose the data visitor has n1 attributes and the set of attributes L={L_1, L_2, .. L_{n1}}, where L_i=$v_{i, j}$. The data owner has n^2 attributes, set W={W_1, W_2, ..., W_{n2}}, W_i = $v_{i, j}$. Set p edge nodes as blockchain nodes, and act as attribute authorization centers to authorize terminal devices.

Fig. 1 shows the system architecture and the data sharing process of our scheme:

1) DO starts the ciphertext storage contract and saves the ciphertext CT in the blockchain.

2) DU requests ciphertex.

3) AA issues the corresponding attribute key SK_i to attribute i of DU. Note that the SK_i here is obtained after AA is encrypted with DU's RSA public key.
4) Attribute key Sk_i is encrypted by the blockchain system to generate DK_i and then sent to DU.
5) DU encrypts SK_i again and sends the outsourced decryption key TK_i to the blockchain system.
6) EN uses TK_i to execute the outsourced decryption contract to realize partial decryption of CT and generate CT'.
7) CT' is sent to DU.
8) DU finishes the final decryption locally to get the plaintext data m. Each DU has a unique GID, and the RSA algorithm is used to generate both public and private keys for DU.

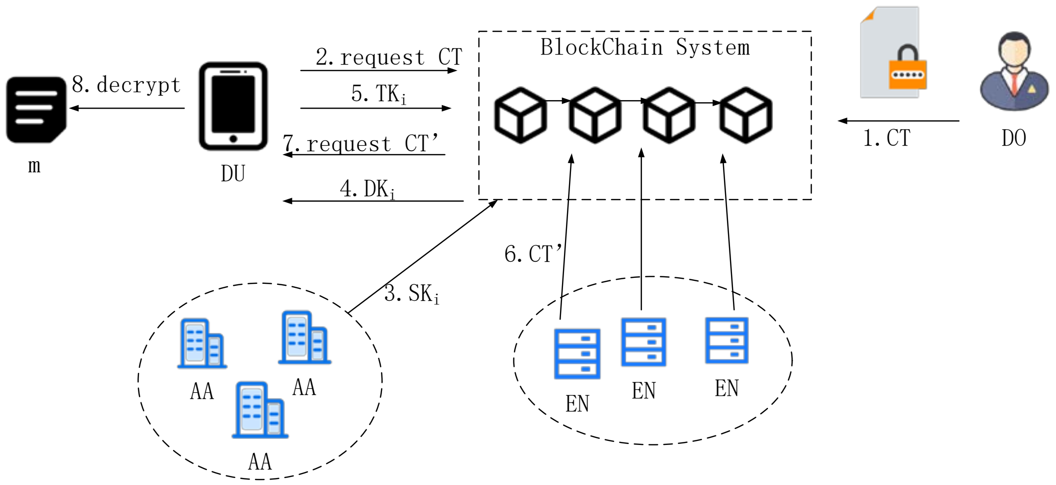


Fig. 1. System architecture

the notations used in this paper are listed in **Table 1**.

Table 1. The notations used in this paper

Notation	Description
EN	edge node
DO	data owner
DU	data user
AA	attribute authorization
U	attribute sets
CT	ciphertext
CT'	partial decryption of CT
GP	global public parameter
PK_i	The public key of AA
PK_j	The public key of DU
GID	Globally Identifier
K_i	attribute key
$SK_{i, GID, v}$	encrypted key of K_i
$SK_{i, GID, v}'$	encrypted key of $SK_{i, GID, v}$
z	a random value selected by DU
TK_i	outsourced decryption key
m	shared information

4.2. Construction

The system contains the following algorithms:

(1) Global Setup(1^λ)

We first input the safety parameter λ and run the group generator ϱ (1^λ) to obtain (p, q, r, g_1, g_2, G_1, G_2, G_T, e), Let g_1 and g_2 be the two generators of G_1 and G_2. Select a random matrix A, $A \in Z^{(k+1)(k)}$, and a random matrix U, $U \in Z^{(k+1)(k+1),}$ We use compound order bilinear groups G_1, G_2, where the group order is the product of three prime numbers: N=pqr, hash function H:{0, 1}* →G_2, which can map the global identity GID to G_2. e:$G_1 \times G_2 \rightarrow G_T$ is a bilinear mapping. The global public parameter GP is

$$GP = \{g_1, g_2, g_1^A, g_1^{U^T A}, e(g_1, g_2)\} \tag{3}$$

(2)Authority Setup(GP, AA$_i$)

There are multiple AAs in the system, and each AA has multiple attributes. The algorithm input samples a random matrix W for the algorithm, resulting in a vector α_i, a random number $\sigma_i \in Z$, and the attribute institution stores the key. The public key of AA is defined as follows:

$$PK_i = \{g_1^{W_i^T A}, e(g_1, g_2)^{\alpha_i^T A}, y_i = g_2^{\sigma_i}\} \tag{4}$$

In addition to the GID, DU has its own RSA public key. The public key of DU is

$$PK_j = \{GID_j, c_j\} \tag{5}$$

c_j is the RSA public key, r_j is the RSA private key, $c_j * r_j$ mod N=1, Where c_j is exposed to everyone in the system, r_j is saved locally by the user. In KeyGen(), AA generates the attribute key for DU and uses c_j to encrypt the attribute key.

(3)Encrypt ({PK$_i$}, x, m) → CT

DO takes the public key {PK$_i$} of AAs, access policy vector x, and shared information m as input, and finally outputs ciphertext CT. x = (x_1, . . ., x_n) ∈ Z, is the access policy vector, and the vector $s \in Z$ is randomly selected to calculate:

$$C_0 = g_1^{As} \tag{6}$$

$$C_i = g_1^{(x_i U^T + W_i^T)As} \tag{7}$$

$$C' = m \cdot \prod_{i=1}^{n} e(g_1, g_2)^{\alpha^T As} = m \cdot e(g_1, g_2)^{\alpha^T As} \tag{8}$$

$$CT = \{C_0, \{C_i\}, C'\} \tag{9}$$

Where $\alpha = \sum_{i=1}^{n} \alpha_i$.

(4)KeyGen(GID, PK$_i$, SK$_{i, v)}$ → DK$_{i, GID, v.}$

AA, associated with attribute i, selects the public key PK$_i$ of AA related to the attribute i of DU, the GID of DU, and the attribute vector corresponding to v of DU for calculation

$$\mu_i = \sum_{j=1}^{i-1} H(y_j^{\sigma_i}, GID, v) - \sum_{j=i+1}^{n} H(y_j^{\sigma_i}, GID, v) \tag{10}$$

It is easy to check that

$$\sum_{i=1}^{n} \mu_i = 0 \tag{11}$$

We take k+1 hash function, to generates g_2^h, where h∈Z. We define

$$H(GID,v)=(H_1(GID,v),...,H_{k+1}(GID,v))^T=g_2^h \tag{12}$$

$$K_i=g_2^{\alpha_i-v_iW_ih+\mu_i} \tag{13}$$

DU has its own RSA public key c_i, AA uses DU's RSA public key to encrypt K_i, and the final algorithm outputs key $SK_{i,\,GID,\,v}$ .The algorithm outputs keys $SK_{i,\,GID,\,v}$

$$SK_{i,GID,v}=\{K_i^{c_i},g_2^h\} \tag{14}$$

$SK_{i,\,GID,\,v}$ are first sent to the blockchain system, and the blockchain node encrypts $SK_{i,\,GID,\,v}$ again for subsequent attribute revocation. All nodes maintain a table, in which the attribute key corresponding to each GID is recorded. Attribute i of GID corresponds to a key $d_{GID,\,i}$, $d_{GID,\,i}$ is secretly stored by the blockchain node. The blockchain system sends the encrypted keys $SK_{i,\,GID,\,v}$'to the user,

$$SK_{i,GID,v}'=\{K_i^{c_i}\cdot g_2^{d_i},g_2^{h\cdot d_i}\} \tag{15}$$

After the user gets $SK_{i,\,GID,\,v}$ ', decrypts it with his own RSA private key r_i, then DU gets the $DK_{i,\,GID,\,v}$.

$$DK_{i,GID,v}=\{K_i\cdot g_2^{d_{GID,i}},g_2^{h\cdot d_{GID,i}}\} \tag{16}$$

$DK_{i,\,GID,\,v}$ is the attribute key that the user gets associated with attribute i. The essence of revocation attribute is to change the user's attribute key. If the blockchain system does not encrypt this time, the attribute key obtained by DU will always be kept by the user and cannot be changed, so the attribute revocation cannot be completed. What is stored in the blockchain is the $SK_{i,\,GID,\,v}$ 'encrypted by DU's RSA public key, not the true attribute key.

(5) OutKeyGen (*z*, {$DK_{i,\,GID,\,v}$}) → (TK_i) .

The OutKeyGen algorithm is executed by the user and ultimately generates the transformation key TK_i. DU selects a random value *z*∈ Z, then calculate TK_i

$$TK_i=\{K_i^z\cdot g_2^{zd_{GID,i}},g_2^{zhd_{GID,i}}\ \ \forall i\} \tag{17}$$

The TK_i is recorded on the blockchain. Decryption requires obtaining both $d_{GID,\,i}$ and *z*.

(6) **Transform (GP, TK, CT) →CT'.**

The Transform algorithm is executed by blockchain nodes through smart contracts. The CT is encrypted under the access matrix (A, ρ), If DU has a secret key $\{K_{\rho(x),\,GID}\}$ for a subset of row A_x of A such that (1, 0, ..., 0) within the span of these lines, the decrypt process as follows

$$\begin{aligned}&e(C_0,\prod_{i=1}^n K_i)\cdot e(\prod_{i=1}^n C_i^{v_i},H(GID,v))\\&=e(g_1^{As},g_2^{z\cdot\sum_{i=1}^n\alpha_i-v_iW_ih+\mu_i+d_{GID,i}})\cdot e(g_1^{\sum_{i=1}^n v_i(x_iU^T+W_i^T)As},g_2^{zh\cdot d_{GID,i}})\\&=e(g_1,g_2)^{\alpha^TAsz\sum_{i=1}^n d_i-Asz\cdot\sum_{i=1}^n v_ih^TW_i^T+d_{GID,i}}\cdot e(g_1,g_2)^{<x,v>h^TU^TAsz\sum_{i=1}^n d_i+Asz\cdot\sum_{i=1}^n v_ih^TW_i^T+d_{GID,i}}\\&=e(g_1,g_2)^{\alpha^TAsz\sum_{i=1}^n d_{GID,i}}\cdot e(g_1,g_2)^{<x,v>h^TU^TAsz\sum_{i=1}^n d_{GID,i}}\\&=CT'\end{aligned} \tag{18}$$

If the attribute satisfies the policy, <x, v>=0, then

$$CT'=e(g_1,g_2)^{\alpha^TAsz\sum_{i=1}^n d_{GID,i}} \tag{19}$$

The ENs on the Blockchain are further calculated $\sum_{i=1}^n d_i$ according to the locally saved d_i, then decryption of CT ' $e(g_1,g_2)^{\alpha^TAsz}$ is obtained. The decryption result is sent to DU offline. In this step, the outsourcing decryption process realized by smart contract consumes the most computing resources in the whole attribute encryption.

(7)OutDecrypt (CT, CT', *z*) → {*m*, ⊥} .

The OutDecrypt algorithm is run by DU. DU calculate：

$$\frac{C'}{CT'}=\frac{m\cdot e(g_1,g_2)^{\alpha^T Asz}}{e(g_1,g_2)^{\alpha^T Asz}}=m \tag{20}$$

In this step DU is decrypted to obtain the original data m, which requires very little computation.

4.3. Smart Contracts

A blockchain node executing a smart contract has the same input and executes the same code, so it also has the same output. Our scheme mainly includes three smart contracts, storage contract, outsourcing decryption contract and attribute revocation contract. The characteristics of smart contract ensure that the process of storage, outsourcing decryption and attribute revocation is open, transparent and verifiable. The functions of these three contracts are described below.

1) Storage contract

DO links the encrypted data through the storage contract, enter the ciphertext CT and Signature, and link the content as $Tx_{storage}$ = {CT, CheckCode, signature}. $Tx_{storage}$ consists of three parts, wherein CT is the cipher text, CheckCode is the hash of the plaintext m, The DO uses its private key to digitally sign the ciphertext to generate the signature. CheckCode=H(CT), Therefore, any DU can use CheckCode to verify the integrity of the shared ciphertext. Assume that DO has an RSA private key of r and a public key of c, Signature is used to prove that the CT was indeed sent by DO, Signature= H(CT)· r. The blockchain node only needs to calculate L=Signature·c mod N when verifying the signature. If L=CheckCode, the signature verification is successful.

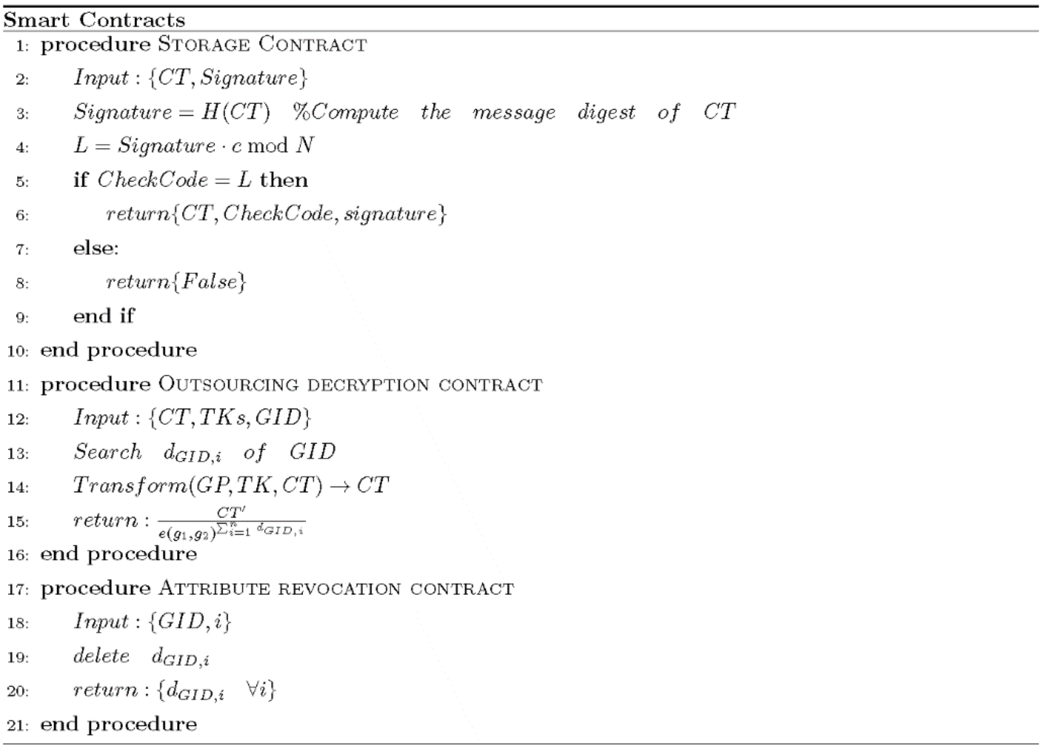
2) Outsourcing decryption contract:

Triggered by DU, DU inputs CT, GID and its own conversion key TKs, and finally obtains proxy decryption result. Blockchain nodes conduct proxy decryption operation through smart contracts, making the process of proxy decryption open and transparent, and ensuring the correctness of decryption results.

3) Attribute revocation contract

The attribute update contract input is GID, and the attribute to be revoked is i, and S_{AA} is the attribute set corresponding to attribute authority AA. Finally, all blockchain nodes delete the key $d_{GID, i}$ corresponding to attribute I of the locally saved GID. In this way, the blockchain cannot perform outsourcing decryption, and users cannot decrypt data with the local key. Thus, the effect of revoking user attributes is achieved.

Smart contracts perform operations such as ciphertext storage, outsourced decryption and attribute retraction, making these processes open and transparent and giving full play to the security and reliability of blockchain.



```
Smart Contracts
procedure STORAGE CONTRACT
    $Input : \{CT, Signature\}$
    $Signature = H(CT)$  $\%Compute\ the\ message\ digest\ of\ CT$
    $L = Signature \cdot c \bmod N$
    if $CheckCode = L$ then
        $return\{CT, CheckCode, signature\}$
    else:
        $return\{False\}$
    end if
end procedure
procedure OUTSOURCING DECRYPTION CONTRACT
    $Input : \{CT, TKs, GID\}$
    $Search\ \ d_{GID,i}\ \ of\ \ GID$
    $Transform(GP, TK, CT) \to CT$
    $return : \frac{CT'}{e(g_1,g_2)^{\sum_{i=1}^{n} d_{GID,i}}}$
end procedure
procedure ATTRIBUTE REVOCATION CONTRACT
    $Input : \{GID, i\}$
    $delete\ \ d_{GID,i}$
    $return : \{d_{GID,i}\ \ \forall i\}$
end procedure
```

4.4. Security Analysis

a. Prevent collusive attacks

In order to prevent users with different attribute keys from gathering attribute keys to launch collusion attacks, in our scheme, AA will embed the user's GID into the user's attribute key when generating attribute keys for the user, so that each attribute key of the user is bound to its globally unique GID. As a result, users cannot combine their attribute keys to decrypt ciphertext during decryption. The decryption program must recover the blind factor s in $e(g_1, g_2)^s$ by pairing the key of the attribute, identity pair (i, GID) with the ciphertext element to obtain s[14].

b. Attribute key abuse problem

1)AA

When the edge server generates the outsourced decryption key, only part of the decrypted data can be obtained, but not the plaintext data. Multi-institution attribute authorization ensures that the complete attribute key of the user is generated by multiple AAs together, instead of a single AA controlling the attribute key of all users. This also solves the problem of single point of failure and key abuse of attribute authorization to a certain extent.

2) Blockchain node

In this paper, AA generates the attribute key and encrypts it with the user's RSA public key to get $SK_{i, GID, v}$. The blockchain node gets it and then encrypts the attribute key with $d_{GID, i}$ to get $SK_{i, GID, v}$ '. The blockchain node gets the key encrypted by AAs with the DU's RSA public key, so if it is not decrypted by the user's private key, the blockchain node cannot decrypt the ciphertext. DU gets the key $SK_{i, GID, v}$ 'after the blockchain encryption, and gets the proxy decryption key TK_i after the encryption again, which is sent to the blockchain system. User to unlock, you must pass a blockchain system agent, so blockchain system can revoke users corresponding $d_{GID, i}$, make Outsourcing decryption process can not be calculated $\sum_{i=1}^{n} d_i$ to get the result of the attributes revocation, and revocation of attribute key is done through smart contracts, This also means that every attribute revocation is approved by all nodes of the blockchain, so there is no problem in this system that someone will revoke the user's attribute at will.

c. Privacy protection

Since GID is included in the generation process of attribute key SK_i, when the blockchain node encrypts SK_i, it does not know what attribute corresponds to the user. Therefore, the attribute privacy of the data user is protected. AAs generate attribute keys for DU, but the attribute key set $\{SK_i\}$ of a DU does not necessarily come from the same AA. Therefore, an AA usually cannot grasp all attribute information of the user. Compared with the scheme using a single AA, our scheme is more private.

d. Indistinguishability

Theorem 1: The CP-ABE scheme in Waters[18] satisfies the selective CPA- security, so our scheme is also selective CPA-security.

We define a safe game between an adversary *A* and a challenger *C* in this paper, named Exp^{ind}. *A* attempts to recognize two randomly generated encrypted messages without sufficient attribute keys. The Exp^{ind} security game is defined as follows:

Initialisation –*A* chooses a challenge access policy $\Psi^*=(A^*, \rho^*)$ and sends it to *C*.

Setup –A and C both run the Global Setup algorithm to generate the global public parameters and get their public and private keys.

Queries phase –A queries the attribute keys, which doesn't satisfy the access policy.

Challenge –*A* sends two plaintexts M_0 and M_1 to *C*. *C* chooses a random bit b ∈ {0, 1}, and encrypts M_b under $\Psi^*=(A^*, \rho^*)$, and sends the result E_b to *A*.

Queries phase 2 –*A* can request some queries as in Queries Phase 1.

Guess – *A* tries to guess which message M_b' where b' ∈ {0, 1} corresponds to the challenge ciphertext E_b. The advantage of the adversary to win the game is:

$$Adv_A[Exp^{Conf}(1^{\xi})] = |\Pr[b = b'] - \frac{1}{2}| \tag{21}$$

Definition 1. Our scheme is CPA-secure if the $Adv_A[Exp^{Conf}(1^{\xi})]$ is negligible for all probabilistic polynomial time adversaries.

Proof: We define adversary *A* running the Exp^{ind} security game with *B* who is running the Lewko et al[19]'s CPA-security game with a challenger *C*. The interaction process of A, B, and C is described below:

Initialisation –A gives B a challenge access policy $\Psi^*=(A^*, \rho^*)$,

Setup - *B* runs the Global Setup algorithm to generate the global public parameters GP. Finally, it outputs the GP defined as

$$GP = \{g_1, g_2, g_1^{A}, g_1^{U^T A}, e(g_1, g_2)\} \tag{22}$$

B asks C to execute the Authority Setup algorithm to generate the public key $PK_i = \{g_1^{W_i^T A}, e(g_1, g_2)^{\alpha_i^T A}, y_i = g_2^{\sigma_i}\}$.

Queries phase 1 –A issues a key query by submitting a set of attributes S_j and his GID. Then, *B* calls *C* to generate and return $SK_{i,GID,v}' = \{K_i^{c_i} \cdot g_2^{d_i}, g_2^{h \cdot d_i}\}$ to A.

Challenge –*A* sends $\{M_0, M_1\} \in G_T$ to *B*. *B* chooses a bit b ∈ {0, 1} and sends M_b to *C*. *C* selects a random bit b ∈ {0, 1} and encrypts a message $M_b \in \{M_0, M_1\}$ as the challenge ciphertext E_{Db}. The challenger computes the challenge ciphertext as follows:

$$C_0 = g_1^{As}, \; C_i = g_1^{(x_iU^T+W_i^T)As}, \; C^{'} = m\cdot\prod_{i=1}^{n} e(g_1,g_2)^{\alpha^T As} = M_b\cdot e(g_1,g_2)^{\alpha^T As} \tag{23}$$

Then, *C* sends the generated ciphertext E_{Db} to *A*.

Queries phase 2 – *A* continues to request some queries and *B* answers as in Queries Phase 1.

Guess –*A* chooses a bit b'. Then, B sends b' to C as its guess about b. A guesses the generated challenge ciphertext E_{Db} by trying one of the two plaintext messages M_0 and M_1. Hence, *A* tries to distinguish between $C_{01}=M_1e(g_1, g_2)^s$ and $C_{02}=M_2e(g_1, g_2)^s$.

Exp^{ind}'s game has a smaller probability of breaking than Lewko-Game because *B* must win the game for *A* to get the correct E_{Db} value and try to guess *B*'s value as such a probability $\Pr[Exp_A^{Lewko}(1^{\zeta})] \geq \Pr[Exp_A^{conf-real}(1^{\zeta})]$, and the advantage of *A* is negligible. Therefor, our scheme satisfies the indistinguishability.

5. Comparisons and Performance Analysis

From **Table 2**, we can see how our scheme compares with other schemes. The scheme in [15] has the function of verifiable outsourcing decryption, but it lacks the privacy protection process of attributes or policies, and the function of attribute revocation, which is not convenient to manage user attributes. The multiplicative homomorphic ElGamal cryptosystem is used in [11] to ensure the attribute privacy in the authorization verification process. This scheme has high security and privacy, but the decryption process is conducted by the user. As the decryption process requires complex calculations, it is not appropriate for the scenario with a great number of terminal devices, and the scheme has no attribute revocation procedure. The scheme in [10] combines attribute encryption and blockchain technology for data sharing, has the process of outsourcing attribute decryption, and has privacy protection and verifiability, and has excellent performance. However, this scheme lacks an attribute revocation procedure. When applied to the edge computing scenario, our scheme can provide multi-institution attribute authorization, outsourcing decryption, verifiability, attribute privacy protection, attribute revocation and other properties, which has great advantages compared with other literatures. The characteristics of multi-mechanism make the system of this paper has high scalability. The feature of outsourcing decryption makes the scheme in this paper applicable to scenarios involving resource-constrained devices. Verifiability is a natural feature of blockchain, and our scheme also has the feature of privacy protection, which protects the user's attribute privacy.

Table 2. Comparison with other scheme

scheme	Multi-agency attribute authorization	Outsourcing decryption	verifiable	Privacy protection	Attribute revocable
[15]		√	√		
[11]				√	
[10]		√	√	√	
Ours	√	√	√	√	√

The experimental environment of this paper uses ubuntu 20.04 system, python3.7, and the CPU is Intel(R) Core(TM) i7-9750H CPU@2.60GH. In terms of performance, it can be seen from **Fig. 2** that the most time-consuming process is the decryption. When the number of attributes is 10, the decryption time is about 100ms. When the number of attributes reaches 50, the decryption time is about 2500ms, with exponential growth. However, mass terminal devices are usually limited in computing power, storage capacity and other resources, and it is difficult to do decryption. As long as security and privacy are guaranteed, outsourcing the decryption process requiring complex calculations to edge servers with relatively rich computing resources is a good solution. Therefore, outsourcing decryption is very suitable for edge computing scenarios, and the attribute revocation mechanism based on blockchain in this paper can flexibly manage the attributes of end users.

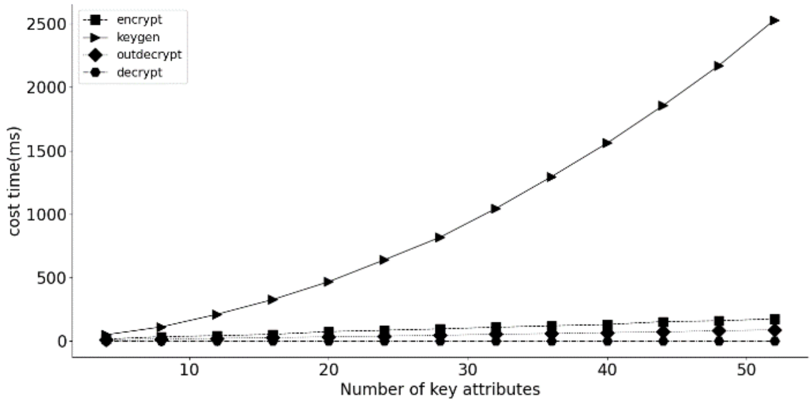


Fig. 2. Latency of our scheme

Due to the natural verifiability of blockchain, compared with the scheme [15], which adds verification marks to the ciphertext, it can be seen from **Fig. 3** that when the number of attributes is the same, the ciphertext in our scheme occupies a smaller space and the gap between the occupied space will become larger as the number of attributes increases.

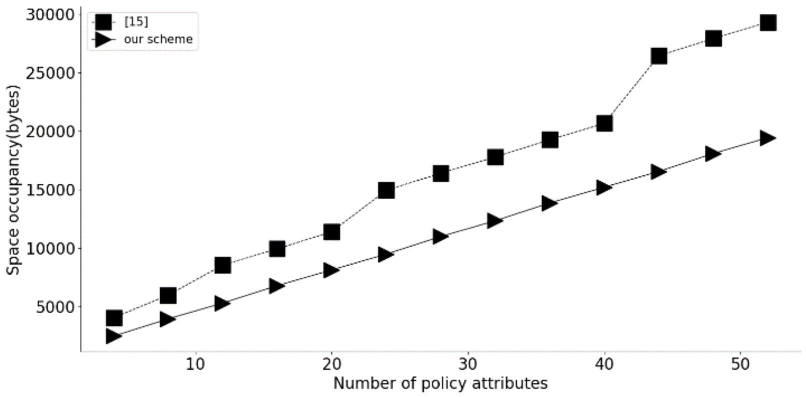


Fig. 3. Compare the space occupied of CT

6. Conclusion

This paper studies the data access control scheme based on CP-ABE in edge computing. Our scheme has the characteristics of outsourcing decryption, verifiability, protection of attribute privacy, etc., which is more suitable for the edge computing scenario with many devices with limited resources, and proposes a user attribute revocation scheme based on blockchain. After attribute revocation, users will not be able to decrypt related ciphertext. Therefore, users' access permissions can be controlled at a fine-grained level. In future research, we will consider the efficiency problem. While blockchain brings security and privacy to various fields, there are also some efficiency problems, which are mainly caused by the consensus algorithm adopted with the blockchain system. Different consensus algorithms in the blockchain may introduce different system delays, and the end-user experience will also depend on the performance of the blockchain system. Therefore, efficient consensus algorithm in edge computing based on blockchain will be a meaningful topic for our future research.

References

[1] Guan Y, Guo S, Li P, et al., “Secure and Verifiable Data Access Control Scheme With Policy Update and Computation Outsourcing for Edge Computing,” in *Proc. of 2020 IEEE 26th International Conference on Parallel and Distributed Systems (ICPADS)*, IEEE, 398-405, 2020. Article (CrossRef Link)

[2] Zhou Jun, Shen Huajie, Lin Zhongyun, Cao Zhenfu, Dong Xiaolei, “Research Advances on Privacy Preserving in Edge Computing,” *Journal of Computer Research and Development*, 57(10), 2027-2051, 2020. Article (CrossRef Link)

[3] ZHANG Jie, XU Shanshan, YUAN Lingyun, “Internet of things access control model based on blockchain and edge computing,” *Journal of Computer Applications*, 42(07), 2104-2111, 2022. Article (CrossRef Link)

[4] Yang R, Yu F R, Si P, et al., “Integrated Blockchain and Edge Computing Systems: A Survey, Some Research Issues and Challenges,” *IEEE Communications Surveys & Tutorials*, 21(2), 1508-1532, 2019. Article (CrossRef Link)

[5] Zhang Xiaodong, Chen Taowei, Yu Yimin, Duan Zhengtai, Gao Jian, “Model of block-chain data sharing based on ABE,” *Application Research of Computers*, 2021(08), 2278-2283, 2021. Article (CrossRef Link).

[6] Sahai A, Waters B, “Fuzzy identity-based encryption,” in *Proc. of Annual international conference on the theory and applications of cryptographic techniques*, 457-473, 2005. Article (CrossRef Link)

[7] Goyal V, Pandey O, Sahai A, et al., “Attribute-based encryption for fine-grained access control of encrypted data,” in *Proc. of the 13th ACM conference on Computer and communications security*, 89-98, 2006. Article (CrossRef Link)

[8] Bethencourt J, Sahai A, Waters B, “Ciphertext-policy attribute-based encryption,” in *Proc. of 2007 IEEE symposium on security and privacy (SP'07)*, IEEE, 321-334, 2007. Article (CrossRef Link)

[9] Zhen Y, Liu H, “Distributed privacy protection strategy for MEC enhanced wireless body area networks,” *Digital Communications and Networks*, 6(2), 229-237, 2020. Article (CrossRef Link)

[10] Zhang Z, Ren X, “Data security sharing method based on CP-ABE and blockchain,” *Journal of Intelligent & Fuzzy Systems*, 40(2), 2193-2203, 2021. Article (CrossRef Link)

[11] Gao S, Piao G, Zhu J, et al., “Trustaccess: A trustworthy secure ciphertext-policy and attribute hiding access control scheme based on blockchain,” *IEEE Transactions on Vehicular Technology*, 69(6), 5784-579, 2020. Article (CrossRef Link)

[12] Fu X, Nie X, Wu T, et al., "Large universe attribute based access control with efficient decryption in cloud storage system," *Journal of Systems and Software*, 135, 157-164, 2018. Article (CrossRef Link)

[13] Green M, Hohenberger S, Waters B, "Outsourcing the Decryption of ABE Ciphertexts," in *Proc. of 20th USENIX Security Symposium (USENIX Security 11)*, 2011.

[14] Lai J, Deng R H, Li Y, "Fully secure cipertext-policy hiding CP-ABE," in *Proc. of Information Security Practice and Experience: 7th International Conference, ISPEC 2011*, Guangzhou, China, 24-39, 2011. Article (CrossRef Link)

[15] J. Lai, R. H. Deng, C. Guan and J. Weng, "Attribute-Based Encryption With Verifiable Outsourced Decryption," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 8, pp. 1343-1354, 2013. Article (CrossRef Link).

[16] Ge C, Susilo W, Baek J, et al., "Revocable attribute-based encryption with data integrity in clouds," *IEEE Transactions on Dependable and Secure Computing*, 19(5), 2864-2872, 2022. Article (CrossRef Link)

[17] PENG Hongyan, LING Jiao, QIN Shaohua, DENG Jianfeng, "Attribute-Based Encryption Scheme for Edge Computing," *Computer Engineering*, 2021(1), 37-43, 2021. Article (CrossRef Link)

[18] Waters B, "Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization," in *Proc. of Public Key Cryptography–PKC 2011: 14th International Conference on Practice and Theory in Public Key Cryptography*, Taormina, Italy, 53-70, 2011. Article (CrossRef Link)

[19] A. Lewko and B. Waters, "Decentralizing attribute-based encryption," in *Proc. of Advances in Cryptology–EUROCRYPT 2011*, 568–588, 2011. Article (CrossRef Link)

Chao Ma received his PhD in Communications Engineering from Aston University, Birmingham, UK in 2014. He joined the China Academy of Information and Communications Technology, as the Director of the Business Development Department of the Institute of Industrial Internet and Internet of Things in 2020. He is also the rapporteur of the Q3 of the ITU-T SG20, the registered expert of ISO/IEC JTC1 SC41 IoT Technical Committee, the expert of W3C Web of Things, Zigbee Alliance expert, Zhijiang Laboratory expert. Research areas include IoT and smart city architecture and standards, blockchain technology and applications, industrial Internet.

Xiaojun Jin received the B.S. degree from Xiangtan University in 2019.He is currently studying for a doctor's degree in Electronic Science and Technology from Beijing University of Posts and Telecommunications (BUPT), His current research interests are in blockchain, privacy computing and federated learning.

Song Luo received his master's degree in Communication and Information System from the Academy of Telecommunication Science and Technology in 2006. He is the deputy director of the Institute of Industrial Internet and Internet of Things of the China Academy of Information and Communications Technology, the head of the domestic counterpart group of the ITU-T Internet of Things and Smart City Study Group (SG20), and the associate rapporteur of the ITU-T SG20 Q3. His current research interests include Industrial Internet policies and technologies, basic common technical standards and industrial development of the Internet of Things.

Yifei Wei received the B.Sc. and Ph.D. degrees in electronic engineering from Beijing University of Posts and Telecommunications (BUPT, China), in 2004 and 2009, respectively. He was a visiting Ph.D. student in Carleton University (Canada) from 2008 to 2009. He was a postdoctoral research fellow in the Dublin City University (Ireland) in 2013. He was the vice dean of school of science in BUPT from 2014 to 2016. He was a visiting scholar in the University of Houston (USA) from 2016 to 2017. He is currently a professor and the vice-dean of school of electronic engineering at BUPT. His current research interests are in intelligent optimization of network resources, deep learning and blockchain technology.

Xiaojun Wang received the B.Eng. degree in computer and communications and the M.Eng. degree in computer applications from Beijing University of Posts and Telecommunications (BUPT), China, in 1984 and 1987 respectively. He was a Lecturer in BUPT from 1987 to 1989. He received the Ph.D. degree in electronic engineering from Staffordshire University (then Staffordshire Polytechnic), England, U.K., in 1993. He joined the School of Electronic Engineering, Dublin City University, Ireland, as an Assistant Lecturer in 1992, where he is currently a Senior Lecturer. His current research interests include energy-efficient networking, network security, and hardware acceleration of cryptographic algorithms.