Dynamic Permission Access Control Model Based on Privacy Protection

Qikun Zhang
Zhengzhou University of Light Industry
Liang Zhu
Zhengzhou University of Light Industry
Yimeng Wu
zhengzhou Technical College
Jianyong Li
Zhengzhou University of Light Industry
Yinghui Meng
Zhengzhou University of Light Industry
Sikang Hu (sihang_hu@163.com)
Beijing Institute of Technology https://orcid.org/0000-0002-5103-066X

Research Article

Keywords: privacy protection, access control, attribute-based encryption, information security, hidden attribute authentication

Posted Date: December 21st, 2021

DOI: https://doi.org/10.21203/rs.3.rs-1044200/v1

Dynamic permission access control model based on privacy protection

Qikun Zhang[1], Liang Zhu[1], Yimeng Wu[2], Jianyong Li[1], Yinghui Meng[1*] and Sikang Hu[3*]

[1]School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, China.

[2]Department of Architectural Engineering, Zhengzhou Technical College, Zhengzhou, 450000, China.

[3]School of Computer Science and Technology, Beijing Institute of Technology, Beijing, 100081, China.

*Corresponding author(s). E-mail(s): yinghuimeng@126.com; sikang_hu@bit.edu.cn;

Abstract

Access control technology is one of the key technologies to ensure safe resource sharing. Identity authentication and authority distribution are two key technologies for access control technology to restrict unauthorized users from accessing resources and resources can only be accessed by authorized legal users. However, user privacy protection and frequent permission changes are two thorny issues that need to be solved urgently by access control technology. To deal with these problems, this paper proposes a dynamic access control technology based on privacy protection. Compared with existing access control technologies, the main advantages of this paper are as follows: 1) encrypt and hide the attributes of entities, and use attribute-based identity authentication technology for identity authentication, which not only achieves the purpose of traditional identity authentication, but also ensures the attributes and privacy of entities are not leaked; 2) Binding resource access permissions with entity attributes, dynamically assigning and adjusting resource access control permissions through changes in entity attributes, making resource access control more fine-grained and more flexible. Security proof and performance analysis show that the proposed protocol safe under the hardness assumption of the discrete logarithm problem (DLP)

and the decision bilinear Diffie-Hellman (DBDH) problem. Compared with the cited references, it has the advantages of low computational complexity, short computational time, and low communication overhead.

Keywords: privacy protection, access control, attribute-based encryption, information security, hidden attribute authentication

1 Introduction

The innovation and development of artificial intelligence, big data and 5G technology have promoted the birth of new applications such as smart community, smart transportation and smart city. One of its core technologies is the secure resource sharing, information exchange and transmission among multiple entities. Access control is a core technology that guarantees secure resource sharing and information exchange among entities. Under the access control technology, only the legitimate terminal that meets the access policy can access the resources of the network platform. Not only the security of system resources is improved, but also the flexibility of access to system resources is enhanced. In recent years, there has been more and more research on access control technology, involving many fields such as medical, industrial, corporate and personal. On this basis, many access control schemes have been proposed, such as early autonomous access control and mandatory access control, later identity based access control and recent research hotspot attribute based access control, etc. They solve many practical problems.

However, with the progress of network technology, access control technology is also facing new challenges. The huge and complex data exchange not only increases the threat to data security, but also increases the risk of terminal privacy leakage. It is difficult to realize the hierarchical division and classified storage of resources. In the process of resource sharing, the terminal needs to undertake huge computing and communication tasks. In addition, it is still difficult to solve the problem of terminal member revocation or terminal attribute weight revocation, etc. In order to solve these problems, this paper proposes a dynamic permission access control model based on privacy protection (PP-DPAC), which uploads the encrypted data to the resource storage service platform. If the terminal member wants to obtain resources, its identity information is verified first, then its attribute weight information is verified. Only when the identity and attribute weights are satisfied, the terminal can obtain resources. Under the dual authentication mechanism, the security of resources is improved. Terminal members can only access resources with corresponding confidentiality level or resources with lower confidentiality level. Fine grained access control is implemented. A large number of computing tasks are transferred to the server, so that the terminal can efficiently complete data sharing and truly realize light load. In addition, terminal members can obtain higher level of confidentiality resources by upgrading their permissions. If they

are punished, their access permissions will be reduced. This greatly increases the security of privacy and the dynamic flexibility of access.

1.1 Contributions

In this paper, a PP-DPAC model is proposed. The advantages and main contributions of the paper are as follows:

(1) Hidden attribute authentication. Terminal members need to be authenticated before participating in resource sharing. By improving the traditional attribute based authentication scheme, this paper proposes a terminal identity authentication scheme with hidden attributes. In the scheme, not only the identity information of the terminal is hidden, but also the attribute information of the terminal is hidden by algorithm. In this way, the leakage of personal privacy is avoided.

(2) Dynamic and fine-grained access control. Terminal members with different numbers of attributes have different access permissions. Terminal members access resources at the corresponding level and below according to the number of their own attributes. When terminal members have lower permissions, they can access higher-level resources by upgrading their permissions. In other words, authenticated terminal members can access resources at multiple levels through upgrade or downgrade.

(3) High security. Shared resources need to be encrypted before storage, and then uploaded to the resource storage service platform. When terminal members obtain resources, they not only need to verify identity, but also need to have enough attribute weights to calculate the decryption key and decrypt the ciphertext. This double guarantee mechanism can resist collusion attacks and has high security.

2 Related work

With the development and application of various new technologies, the application scenario of access control is also expanding. However, the existing access control technology is difficult to meet the requirements of complex application scenarios, such as cloud computing, edge computing and industrial Internet of things (IoT). In recent years, many scholars have carried out in-depth research on access control technology in combination with specific application scenarios. An access control protocol combined with blockchain is proposed in [1, 2]. Blockchain has the characteristics of decentralization, which can better solve the third-party trust problem. At the same time, the scheme uses blockchain technology to record the attribute information of the terminal members, which not only facilitates the management of the terminal, but also enhances the security of the system. The algorithm part of the scheme adopts a modular design. This method not only facilitates the later management and maintenance, but also improves the flexibility of the scheme. In [3], an access control scheme supporting data privacy protection and policy hiding is proposed. The scheme designs a flexible access structure. LSSS matrix combined with policy

hiding can better ensure the security of data access. At the same time, this scheme not only supports user revocation, but also is efficient and lightweight. This greatly reduces the computing and communication load of the terminal. Finally, the security and performance of the scheme are analyzed, and the feasibility of the scheme is analyzed through simulation experiments. In [4, 5], an efficient access control scheme based on privacy protection is proposed. The scheme uses the characteristics of hash function and the structure of binary tree to design a specific scheme to limit the decision-making process of the model. The scheme uses a binary search tree based on hash function to protect the attributes of terminal members, so it reduces the risk of user privacy leakage. At the same time, this method is also applied to the server to effectively process requests from various terminals. In [6, 7], a secure access control scheme based on attribute signcryption is proposed. Under the assumption of discrete logarithm theory, the scheme uses attribute hiding and zero knowledge proof technology, which can not only protect the privacy of terminal members, but also reduce the leakage of attributes in the process of data sharing. In addition, the scheme uses the server to partially decrypt the ciphertext resources. This method effectively reduces the computing task of terminal members.

A secure access control model based on ciphertext policy and blockchain is proposed in [8]. The scheme uses blockchain technology to solve the problems of single point of failure and third-party trust. The scheme improves and optimizes the traditional hiding strategy, which can not only meet the security sharing of data, but also protect the privacy of access strategy. Finally, the security of the scheme is proved from many aspects. The comparison shows that the scheme has great advantages in many aspects. In [9], a new cryptographic access control framework is proposed. The programme supports the establishment of multiple authority centres. The scheme adopts a new security encryption strategy, which is bound with ciphertext to resist known attacks. Users no longer use private key signature verification, but adopt a more convenient attribute token. Although this method is convenient and lightweight, there is still a risk of token leakage. In [10, 11], a flexible multi authority data storage scheme is proposed. This scheme does not need an authority center to distribute the key. At the same time, a series of attributes are used to realize flexible data storage, which completely eliminates the security risks caused by key distribution. In addition, the scheme transfers the computing task of the terminal to the assistant node, which reduces the load of the terminal node. A ciphertext strategy for media cloud is proposed in [12, 13]. Compared with other attribute based encryption schemes, this scheme has better performance. Legitimate users only need two hash operations when switching access to resources, which greatly reduces the calculation time. At the same time, the scheme also supports user revocation, which is easy to expand and maintain in later work. In [14], an access control scheme supporting attribute hiding is proposed. By hiding the attributes, the scheme can better protect the privacy information of terminal members. In addition, this scheme also proposes an attribute location mechanism, which can help authorized terminals locate

attribute information and decrypt resources. The final simulation comparison shows that the efficiency of this scheme is better.

In [15, 16], a user and attribute revocable access control scheme is proposed.The scheme supports the revocation of the user or the user's attributes, and the terminal members cannot access the original resources after revocation. At the same time, in the scheme, users can add or reduce attributes according to their own needs, and the amount of calculation is small in this process. Through the security analysis of the scheme, the scheme can effectively resist known attacks. In [17], a fine-grained access control scheme based on attribute and blockchain is proposed. In this scheme, a cooperation mechanism is applied, which can authorize users in emergency, and this mechanism can be verified. The scheme adopts the outsourcing method to build a trusted node to perform the main computing and communication tasks, and writes the information into the blockchain through transaction. A resource sharing scheme combining blockchain and ABE is proposed in [18]. In this scheme, data providers can set access policies for shared data ciphertext to achieve fine-grained access control. At the same time, the data demander communicates directly with the data provider, and the data provider can provide the key for the legitimate data demander. In addition, this scheme also supports the keyword search function. The server locates resources and returns information according to the keywords provided by the data demander. In [19], a secure and efficient access control scheme is proposed. It can prevent privacy disclosure threats without introducing third-party trusted entities. In this scheme, a new hash search tree is introduced to protect sensitive attributes. In addition, the server does not need to know the user's private information to correctly process the access request through the hash binary search tree.

In [20, 21], a traceable access control scheme is proposed. In this scheme, the data provider can not only set the access policy of shared resources, but also update them in time, which makes the scheme more flexible and efficient. At the same time, combined with blockchain technology, the scheme can track the terminals that maliciously access resources and maliciously leakage private keys. Finally, the effectiveness of the scheme is proved. An access control scheme for Internet of things is proposed in [22]. In the scheme, the terminal device encrypts its own data and then uploads it to the server, which solves the problem of over authorization by the third party or the entrusted node. At the same time, the scheme can limit the unauthorized operation of the application, so it can protect the privacy information of the terminal device. Finally, it is proved that the scheme is safe. An access control model in smart health application scenario is proposed in [23]. In this system, when the smart health record is encrypted, the attribute value of the access policy is hidden and only the attribute name is displayed, which well protects personal privacy. It uses a small number of bilinear pairs to complete the decryption of smart health records, which greatly improves the efficiency of decryption. In [24], a cross domain access control scheme for cloud sharing is proposed. Combined with blockchain technology, the scheme not only solves the problem of single

point of failure, but also can trace the access records of the terminal. At the same time, by extending the traditional scheme, this scheme designs a cross domain cooperation mechanism through smart contract, which can realize the cooperative operation of multiple trust institutions and generate decryption keys for users. The analysis shows that the scheme has good performance. In [25], a proactive dynamic secure data scheme is proposed. It uses attribute-based access control to protect the private information of financial users. At the same time, in this solution, it uses semantic access technology to generate attribute-based access methods to provide flexibility for access control. In order to protect the integrity and security of data, this scheme takes the client as the core method to effectively avoid the impact of unexpected operation on the server. In addition, due to data access restrictions based on configuring user attributes, this model can continue to provide a high level of security.

An access control scheme based on blockchain smart contract is proposed in [26]. In this scheme, multiple smart contracts are designed, such as the master contract, to manage the access control of data between users. The authentication management contract is used to authenticate the identity of terminal members and store registration records. Intelligent detection contract is used to detect illegal behaviors in the system and punish users. In addition, the scheme can reduce the consumption of computing and communication energy to a certain extent. In [27], an access control scheme for fog computing scenario is proposed. It aims to reduce the cost of calculation and ensure the confidentiality of data. In this scheme, the fog device bears the main computational cost of the encryption and decryption stages. Therefore, the calculation cost of the sender and the receiver will be reduced, which greatly improves the efficiency of data exchange. At the same time, the user's private key is generated through multi-authority, which enhances data security. In [28], a secure and efficient access control scheme using smart contract is proposed. Different from the previous attribute based access control, this scheme carries out safe and efficient data sharing by setting up multiple smart contracts. The most important is the smart contract for access control, which completes the established security policy by setting the access rights of resources and verifying the identity information of terminal members. In addition, there is a judgment contract and a registration contract. Judgment contract is mainly to punish the violations in the process of system operation. The registration contract is used to manage the information of the terminal. In [29], a fine-grained access control scheme with decentralized capability is proposed. In this scheme, resources are stored on their own devices instead of other third-party large storage devices. At the same time, the access control of resources is based on the user's identity. The management of user rights is realized through smart contract. Finally, the feasibility of the scheme is proved. In [30], a traceable access control protocol supporting emergency authorization is proposed. In the scheme, smart contracts are used to define some rules to deal with emergencies, and the duration of emergency visits is also regulated. In addition, patients can also restrict the distribution of permissions for personal health records.

By reading the above references, researchers have made a lot of contributions to data sharing and access control. At the same time, there are some shortcomings, such as the privacy leakage of terminal members, the lack of clear classification of shared resources, and the problem of user authority revoking. To solve these problems, we proposed a dynamic permission access control model based on privacy protection (PP-DPAC), and optimized it in terms of personal privacy information protection, lightweight and security. Through comparative analysis, the effect of this scheme is better.

3 Basic theory

3.1 Bilinear mapping

This paper is based on the basic theory of bilinear mapping; some basic knowledge related to bilinear mapping will be described in this section.

Let G_1 be an additive group on elliptic curve and G_2 is a multiplicative group. Both of them have the same prime order q, where $q \geq 2^{\ell} + 1$, and ℓ is a security parameter. G_1 is generated by g_1, that means $G_1 = \langle g_1 \rangle$, and the discrete logarithm problems of G_1 and G_2 are difficult. We call e an admissible pairing, if $e : G_1 \times G_1 \rightarrow G_2$ satisfies the follow properties:

(1) bilinearty: For all $\mu, \nu \in G_1$, and $a, b \in \mathbb{Z}_q^*$, there is $e(a\mu, b\nu) = e(\mu, \nu)^{ab}$;

(2) Non-degeneracy: There exits $\mu, \nu \in G_1$, such that $e(\mu, \nu) \neq 1$;

(3) Computability: For all $\mu, \nu \in G_1$, there exits a efficient way to calculate $e(\mu, \nu)$.

Inference1. For all $\mu_1, \mu_2, \nu \in G_1$, there is $e(\mu_1 + \mu_2, \nu) = e(\mu_1, \nu)e(\mu_2, \nu)$.

Definition 1. Discrete Logarithm problem (DLP). Given an equation$Y = aQ$, where $Y, Q \in G_1$ and $a < Q$. If a and Q are given, it is easy to calculate Y. But if Y and Q are given, it will be difficult to calculate a.

Definition 2. Inverse Computational Diffie-Hellman (ICDH) Problem: The ICDH problem is given g_1, ag_1 and abg_1, for some $a, b \in \mathbb{Z}_q^*$ to compute $(ab/a)g_1$.

3.2 Lagrange Interpolation Theorem

Generally, if it is known that the function value corresponding to the function $y = f(x)$ is $y_0, y_1, ..., y_n$ at different $n + 1$ points $x_0, x_1, ..., x_n$, in other words, the function passes through these $n + 1$ points $(x_0, y_0), (x_1, y_1), ..., (x_n, y_n)$, we can consider constructing a polynomial $y = P_n(x)$ that passes through these $n + 1$ points and the degree does not exceed n, so that it satisfies: $P_n(x_k) = y_k, k = 0, 1, ..., n$.

To estimate any point ξ, where $\xi \neq x_i, i = 0, 1, 2, ..., n$, the value of $P_n(\xi)$ can be used as the approximate value of the exact value $f(\xi)$. This method is called interpolation method. The formula $P_n(x_k) = y_k, k = 0, 1, ..., n$ is called the interpolation condition, the minimum interval of x_i is $[a, b]$, where $a = \min\{x_0, x_1, ..., x_n\}$, $b = \max\{x_0, x_1, ..., x_n\}$.

General form application method. There are n points $(x_0, y_0), (x_1, y_1), ..., (x_{n-1}, y_{n-1})$ on the plane. Now make a function $f(x)$ to make the image pass through these n points. The specific implementation is as follows:

Let set D_n be a set of subscripts about point (x, y), where $D_n = \{0, 1, ..., n-1\}$ and construct n polynomials $p_j(x), j \in D_n$. For any $k \in D_n$, there are $p_k(x)$ and $B_k = \{i \mid i \neq k, i \in D_n\}$, so that $p_k(x) = \prod_{i \in B_k} \frac{x - x_i}{x_k - x_i}$, Where $p_k(x)$ is a polynomial of degree $n-1$ and satisfies $p_k(x_m) = 0$ and $p_k(x_k) = 1$ for any $m \in B_k$, so that $L_n(x) = \sum_{j=0}^{n-1} y_j p_j(x)$. The interpolation polynomial $L_n(x)$ of the form above is called Lagrange interpolation polynomial.

4 The proposed resource sharing model

The main idea of the model are as follows: The resources stored in the shared database have different security levels. Terminal members have different levels of access rights. Only the terminal members meeting the access rights can decrypt the corresponding resources. In the model, terminal members are divided into two categories, one is data sharers, and the other is data acquirers. A terminal member can be either a data sharer or a data acquirer. Data sharers first set access rights according to the confidentiality level of resources. Then it calculates the encryption key according to the access rights and encrypts the resources to obtain the ciphertext. Finally, the ciphertext resources and the corresponding description information are uploaded to the resource sharing platform. The data requester first checks whether its attribute weight meets the access rights. Then it requests access to and downloads ciphertext resources. Finally, it calculates the decryption key according to its attribute weight and decrypts the downloaded ciphertext resources.

4.1 System Model

Data requesters can upgrade or revoke their attributes according to the access rights of the target resources. This method can not only flexibly and dynamically restrict resource access, but also make the resource access control between entities more secure and efficient. The dynamic permission access control model based on privacy protection is shown in Figure 1.

Certificate authority (CA): Generate public / private key pairs for the resource storage service platform and terminal devices. At the same time, it authenticates the identity and attributes of terminal members, and distributes the corresponding attribute weights for each terminal attribute.

Resource storage service platform(RSSP): Publish each terminal's attribute sequence, identity and its public key, and provide a public display platform for terminals to publish information such as plaintext keywords for shared ciphertext resources, ciphertext information description, and ciphertext encryption attribute weights.

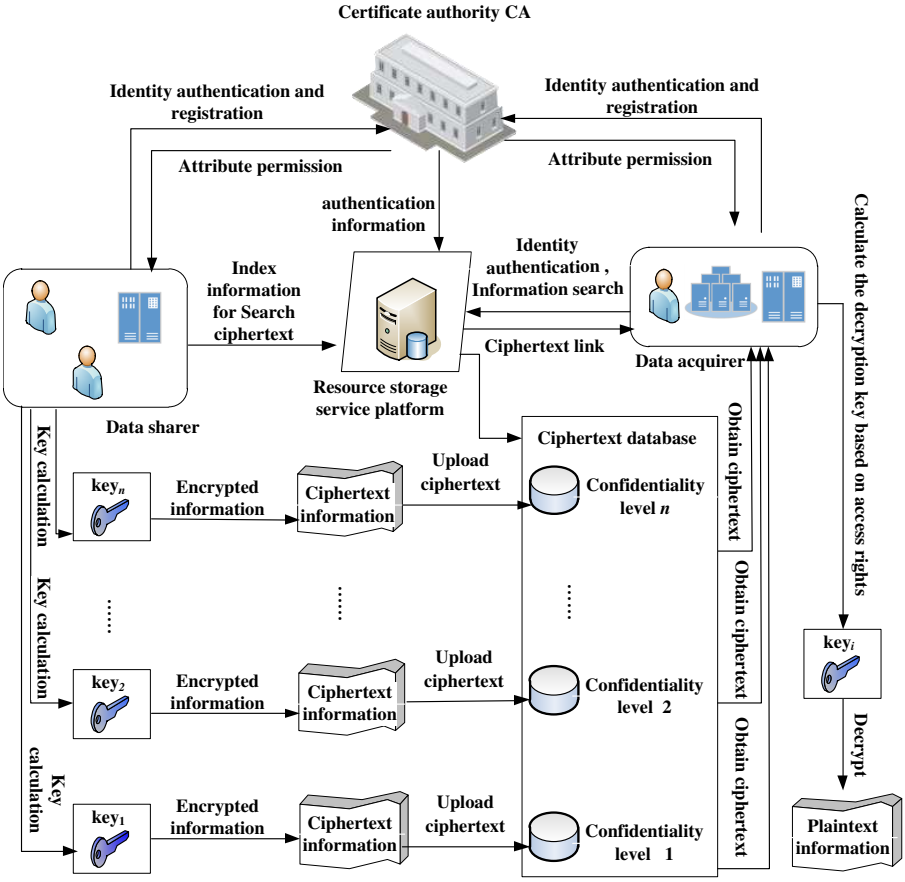

Fig. 1 The system model

Ciphertext database (CD): Store ciphertext resource information shared by each terminal.

4.2 Initialization

In this work, assuming that the information sharing network contains a certification authority CA and n terminal members. CA is mainly used to verify terminal identity and generate system parameters and system master keys. The set of n terminal members is denoted as $U = \{u_1, u_2, ..., u_n\}$, and the corresponding identity set is $ID = \{id_{u_1}, id_{u_2}, ..., id_{u_n}\}$. The sequence of constraint attributes for all access to network resources is $Attr_{seq} = A_1 \mid A_2 \mid ... \mid A_i \mid A_j \mid ... \mid A_R$, where $i < j, A_i < A_j (i, j, R \in N^*)$. The corresponding constraint attribute set is $Attr_{set} = \{A_1, A_2, ..., A_R\}$. And the attribute sequence of the terminal member u_i is $attr_{seq_i} = a_{u_{i,1}} \mid a_{u_{i,2}} \mid ... \mid a_{u_{i,r}} (1 \leq i \leq n)$, and the corresponding ordered attribute set is $attr_{set_i} = \{a_{u_{i,1}}, a_{u_{i,2}}, ..., a_{u_{i,j}}, ..., a_{u_{i,r}}\}$,

where $a_{u_{i,j}} < a_{u_{j,j+1}}$, $attr_{set_i} \subseteq Attr_{set}$. r denotes the number of attributes of terminal member u_i.

Assuming G_1 and G_2 are an additive group and a multiplicative group on the elliptic curve of prime order q, respectively. The discrete logarithm over G_1 and G_2 are difficult, $g_1 \in G_1$ is a generator of G_1. Parameter $e : G_1 \times G_1 \to G_2$ is a computable bilinear mapping. $H_1 : \{0,1\}^* \to \mathbb{Z}_q^*, H_2 : G_1 \to \mathbb{Z}_q^*$ are two hash functions.

KeyGen. $KeyGen(1^\lambda) \to (PK_E, SK_E)$: The CA can run $KeyGen(1^\lambda)$ to generate public / private key pairs (SK_E, PK_E) for the entity, where $SK_E \in \mathbb{Z}_q^*$ and $PK_E = SK_E g_1$.

The CA runs the $KeyGen(1^\lambda)$ to obtain a public/private key pair (SK_{CA}, PK_{CA}), where $SK_{CA} \in \mathbb{Z}_q^*$ and $PK_{CA} = SK_{CA} g_1$. At the same time, CA runs $KeyGen(1^\lambda)$ to generate public key and private key for RSSP and each terminal member in the system respectively and distribute them through secure channel. Suppose the public/private key pair of RSSP is (SK_{RSSP}, PK_{RSSP}), where $SK_{RSSP} \in \mathbb{Z}_q^*$, $PK_{RSSP} = g_1 SK_{RSSP}$. The public/private key pair of the terminal member $u_i \in U(1 \le i \le n)$ is (sk_{u_i}, pk_{u_i}), where $sk_{u_i} \in \mathbb{Z}_q^*$, $pk_{u_i} = g_1 sk_{u_i}$. The system parameters are $params = (PK_{CA}, PK_{RSSP}, q, G_1, G_2, g_1, e, H_1, H_2)$.

4.3 Hidden attribute authentication

1) CA broadcasts the attribute sequence set for accessing network resources and corresponding sequence numbers $\{(A_1, S_1), (A_2, S_2), ..., (A_R, S_R)\}$, where $A_i(1 \le i \le R)$ represents attribute, S_i represents the serial number corresponding to the attribute A_i.

2) Each terminal user $u_i(1 \le i \le n)$ with an ordered attribute set $attr_{set_i} = \{a_{u_{i,1}}, a_{u_{i,2}}, ..., a_{u_{i,r}}\}$, where $a_{u_{i,j}} < a_{u_{i,j+1}}(1 \le j < r)$, chooses a random positive integer $s_{u_i} \in \mathbb{Z}_q^*$ and calculates $\vartheta_{i,0} = s_{u_i} PK_{CA}$, $\vartheta_{i,1} = s_{u_i} a_{u_i,1} g_1, \vartheta_{i,2} = s_{u_i} a_{u_i,2} g_1, ..., \vartheta_{i,r} = s_{u_i} a_{u_i,r} g_1$, $o_i = s_{u_i} H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r}) PK_{CA}$. Then, u_i sends $\{id_{u_i}, pk_{u_i}, o_i, \vartheta_{i,0}, (\vartheta_{i,1}, S_1), (\vartheta_{i,2}, S_2), \ldots, (\vartheta_{i,r}, S_r)\}$ to CA.

3) After receiving the messages $\{id_{u_i}, pk_{u_i}, o_i, \vartheta_{i,0}, (\vartheta_{i,1}, S_1), (\vartheta_{i,2}, S_2), \ldots, (\vartheta_{i,r}, S_r)\}$, CA calculates $\eta_i = SK_{CA}^{-1} o_i = s_{u_i} H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r}) g_1$ and $\psi_k = SK_{CA}^{-1} \vartheta_{i,0} A_k = s_{u_i} g_1 a_{u_i,k} (k = 1, 2, ..., r)$. Then, CA verifies the identity of u_i by the equation $H_1(id_{u_i})\eta_i =? H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r}) pk_{u_i}$ and $H_2(\psi_1 \| \psi_2 \|, ..., \| \psi_r) =? H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})$. If it holds, CA selects a random numbers $\iota_{CA,k} \in \mathbb{Z}_q^* (1 \le k \le r)$ for each attribute $a_{u_{i,k}}$, CA computers $\chi_{i,k} = \iota_{CA,k} \vartheta_{i,k}$ and its signature $\delta_i = SK_{CA}(\iota_{CA,1} a_{u_i,1} + \iota_{CA,2} a_{u_i,2} + ... + \iota_{CA,r} a_{u_i,r}) g_1$. (Note that for any attributes $a_{u_i,j}$ and $a_{u_l,k}$ of different terminals u_i and $u_l (i \ne l)$, if $j = k$, then $\iota_{CA,j} = \iota_{CA,k}$). Then, CA sends messages $\{PK_{CA}, \delta_i, (\chi_{i,1}, \chi_{i,2}, ... \chi_{i,r})\}$ to the register terminal u_i.

4) After receiving the messages $\{PK_{CA}, \delta_i, (\chi_{i,1}, \chi_{i,2}, ... \chi_{i,r})\}$ from CA, $u_i(1 \le i \le n)$ calculates $T_{i,1} = {s_{u_i}}^{-1} \chi_{i,1} = \iota_{CA,1} a_{u_i,1} g_1, T_{i,2} = {s_{u_i}}^{-1} \chi_{i,2} = \iota_{CA,2} a_{u_i,2} g_1, ..., T_{i,r} = {s_{u_i}}^{-1} \chi_{i,r} = \iota_{CA,r} a_{u_i,r} g_1$ and $\mu_i = (T_{i,1} + T_{i,2} + ... + T_{i,r})$. Then, u_i verifies the identity of CA and attribute weight $T_{i,k}$ of $a_{u_{i,k}} (1 \le k \le$

r) by equation $e(\delta_i, g_1) = e(\mu_i, PK_{CA})$. If it holds, u_i obtains the attribute weight $T_{i,k}$ corresponding to each of its attribute $a_{u_i,k}(1 \leq k \leq r)$, Each terminal member has successfully registered.

5) Due to the service characteristics of RSSP and its direct connection with CA, it can obtain all the attributes $\{A_1, A_2, ..., A_R\}$ of the system. Then, according to the above steps, RSSP can calculate all attribute weights $\{T_{i,1} = \iota_{CA,1}A_1g_1, T_{i,2} = \iota_{CA,2}A_2g_1, ..., T_{i,R} = \iota_{CA,R}A_Rg_1\}$.

6) Finally, according to the messages $\{id_{u_i}, pk_{u_i}, o_i, \vartheta_{i,0}, (\vartheta_{i,1}, S_1), (\vartheta_{i,2}, S_2), \ldots, (\vartheta_{i,r}, S_r)\}$ sent by u_i, CA sends the information $\{pk_{u_i}, S_{i,1}, S_{i,2}, ..., S_{i,r}\}$ of each u_i to RSSP, where the attribute serial number $(S_{i,1}, S_{i,2}, ..., S_{i,r})$ of u_i corresponds to the network attribute serial number $S_1, S_2, ..., S_r$.

4.4 Calculation of access permissions for shared resources

Data sharer $u_i(1 \leq i \leq n)$ sets corresponding access permissions according to the security level of shared resources. Only terminal members with the attribute set $attr_{set_{i,m}} = \{a_{u_{i,1}}, a_{u_{i,2}}, ..., a_{u_{i,j}}, ..., a_{u_{i,t}}\}(j, t \in N^*, t \leq r)$ can access the resource $m_{u_i,i} \in M^*$ (M^* is the plaintext space). The calculation of resource access permissions is as follows:

1) u_i randomly selects the encryption parameter $\beta_{u_i,m} \in \mathbb{Z}_q^*$ and uses the encryption parameter to calculate the encryption key $k_{u_i,m} = H_2(\beta_{u_i,m}g_1)$. Then u_i encrypts resource $m_{u_i,i}$ to obtain ciphertext $c_{i,m} = k_{u_i,m} \oplus m_{u_i,i}$. and sets the attribute weight according to the confidentiality of resource $m_{u_i,i}$. Only the terminal meeting the attribute weight can access and decrypt resource $m_{u_i,i}$. Assume that access to resource $m_{u_i,i}$ requires t different attribute weights. The attribute serial number corresponding to the t attribute weights is $(S_{i,1}, S_{i,2}, ..., S_{i,t})$. u_i calculates $\hbar = \beta_{u_i,m}PK_{RSSP}$, $H_1(c_{i,m})$ and signature $\sigma_{i,m} = sk_{u_i}^{-1}H_1(keyword_{i,m}S_{i,1}S_{i,2}...S_{i,t-1}\hbar c_{i,m})\, g_1$. Then u_i sends the message $\{id_{u_i}, pk_{u_i}, (S_{i,1}, S_{i,2}, ..., S_{i,t-1}), \hbar, \sigma_{i,m}, c_{i,m}, keyword_{i,m}, H_1(c_{i,m})\}$ to the RSSP.

2) After RSSP receiving the message $\{id_{u_i}, pk_{u_i}, (S_{i,1}, S_{i,2}, ..., S_{i,t-1}), \hbar, \sigma_{i,m}, c_{i,m}, keyword_{i,m}, H_1(c_{i,m})\}$, it first calculates hash values $H_1(c_{i,m})$ and $e(\sigma_{i,m}, pk_{u_i}) = e(H_1(keyword_{i,m}S_{i,1}S_{i,2}...S_{i,t-1}\hbar c_{i,m})\, g_1, g_1)$ to verify the integrity of ciphertext resources and the legitimacy of terminal member u_i's identity respectively. If it is correct, RSSP calculates the encryption key $k_{u_i,m} = H_2\left(SK_{RSSP}^{-1}\hbar\right)$ and randomly selects $t-1$ random numbers $b_{i,1}, b_{i,2}, ..., b_{i,t-1} \in \mathbb{Z}_q^*$ to construct the polynomial $f(x) = b_{i,t-1}x^{t-1} + b_{i,t-2}x^{t-2} + ... + b_{i,1}x + k_{u_i,m}$. Then it inputs the hash value $\{H_2(T_{i,1}), H_2(T_{i,2}), ..., H_2(T_{i,t})\}$ of the attribute weight corresponding to the attribute serial number to the polynomial $f(x)$. $f(x)$ outputs t function values $\{f_{i,1}, f_{i,2}, ..., f_{i,t}\}$. Finally, the RSSP publishes the information $\{id_{u_i}, pk_{u_i}, ((f_{i,1}, S_{i,1}), (f_{i,2}, S_{i,2}), ..., (f_{i,t}, S_{i,t})), keyword_{i,m}\}$ on the public information sharing platform, and the ciphertext $c_{i,m}$ is stored in the ciphertext database CD.

5 The proposed access control model

5.1 Dynamic resource access control

1) The data acquirer $u_j(1 \leq j \leq n)$ searches the ciphertext resource $c_{i,m}$ and the attribute serial number $(S_{j,1}, S_{j,2}, ..., S_{j,t})$ corresponding to the ciphertext resource $c_{i,m}$ on the public information sharing platform according to the keyword $keyword_{i,m}$. If the data acquirer u_j has the attribute $a_{u_i,1}, a_{u_i,2}, ..., a_{u_i,t}$ corresponding to the attributes serial number$(S_{j,1}, S_{j,2}, ..., S_{j,t})$, u_j has the access right to access the resource. u_j calculates $\sigma_{j,m} = sk_{u_j}^{-1} H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1$ and sends the message $\{id_{u_j}, pk_{u_j}, (S_{j,1}, S_{j,2}, ..., S_{j,t}), \sigma_{j,m}, keyword_{i,m}\}$ to RSSP to apply for access to the resource.

2) After RSSP receives the message $\{id_{u_j}, pk_{u_j}, (S_{j,1}, S_{j,2}, ..., S_{j,t}), \sigma_{j,m}, keyword_{i,m}\}$ sent by u_j, it will match the attribute serial number set $(S_{j,1}, S_{j,2}, ..., S_{j,t})$ in the sent message with the platform $(S_{j,1}, S_{j,2}, ..., S_{j,r})(t \leq r)$ published on the information sharing platform (that is, whether u_j has the attribute access rights it claims). If it matches, RSSP calculates $\phi_{j,m} = H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1$ and verifies the identity of terminal member u_j by calculating whether equation $e(\sigma_{j,m}, pk_{u_j}) = e(\phi_{j,m}, g_1)$ holds. If the equation holds, RSSP will provide the ciphertext $c_{i,m}$'s link corresponding to the keyword $keyword_{i,m}$ to u_j.

3) After downloading the ciphertext $c_{i,m}$, u_j uses the corresponding attribute serial number $(y_{u_{i,j}}, S_{i,j})(j = 1, 2, ...t)$ obtained from the information sharing platform and the corresponding attribute weights $T_{i,j}(j = 1, 2, ..., t)$ of u_j , the polynomial $g(x) = \sum_{j=1}^{t} y_{u_{i,j}} \prod_{\varpi=1, \varpi \neq j}^{t} \frac{x - H_2(T_{i,\varpi})}{H_2(T_{i,j}) - H_2(T_{i,\varpi})}$ is recovered according to the Lagrange theorem. That is, $g(x) = f(x)$, u_j calculates the decryption key $g(0) = k_{u_i,m}$ of ciphertext $c_{i,m}$ to obtain plaintext information $m_{u_i,i} = c_{i,m} \oplus k_{u_i,m}$.

5.2 Permission upgrade

If u_j upgrade membership attributes, it can obtain the corresponding level of resource access permissions. Suppose the previous attribute set of u_j is $attr_{set_j} = \{a_{u_j,1}, a_{u_j,2}, ..., a_{u_j,r}\}(j, r \in N^*, r < R)$, it can only reproduce the polynomial constructed by the corresponding attribute weight set $\{T_{j,i} \mid i = 1, 2, \ldots, r\}$ and its subset to calculate the decryption key of the corresponding ciphertext and decrypt the ciphertext resource. If u_j obtains a new member attribute $a_{u_j,r+1}$, it can apply to CA to obtain the attribute weight $T_{j,r+1}$ corresponding to $a_{u_j,r+1}$. Then u_j can upgrade the polynomial constructed from the corresponding attribute weight set $\{T_{j,i} \mid i = 1, 2, \ldots, r+1\}$ and its subset to calculate the decryption key of the corresponding ciphertext, and decrypt the ciphertext resource. The permission application process is as follows:

1) u_j calculates $\vartheta_{j,r+1}{=}s_{u_j}a_{u_j,r+1}g_1$, $o_j = s_{u_j}H_2(\vartheta_{j,r+1})PK_{CA}$, $\vartheta_{j,0} = s_{u_i}PK_{CA}$. Then, u_j sends $\{id_{u_j}, pk_{u_j}, o_j, \vartheta_{j,0}, (\vartheta_{j,r+1}, S_{r+1})\}$ to CA.

2) After receiving the messages $\{id_{u_j}, pk_{u_j}, o_j, \vartheta_{j,0}, (\vartheta_{j,r+1}, S_{r+1})\}$, CA calculates $\eta_j = SK_{CA}^{-1}o_j = s_{u_j}H_2(\vartheta_{j,r+1})g_1$, $\psi_k = SK_{CA}^{-1}\vartheta_{i,0}A_k = s_{u_j}g_1a_{u_j,k}(k = 1, 2, ..., r+1)$ and verifies the identity of u_j by calculating whether the equation $H_1(id_{u_j})\eta_j = H_2(\vartheta_{j,r+1})pk_{u_j}$, $H_2(\psi_1 \|\psi_2\|, ..., \|\psi_{r+1}) = ?H_2(\vartheta_{i,1}\|\vartheta_{i,2}...\|\vartheta_{i,r+1})$ holds. If the verification is successful, CA randomly selects a positive integer $\iota_{CA,r+1} \in \mathbb{Z}_q^*$ for attribute $a_{j,r+1}$, and calculates $\chi_{j,r+1} = \iota_{CA,r+1}\vartheta_{j,r+1}$ and $\delta_j = SK_{CA}\iota_{CA,r+1}a_{u_j,r+1}g_1$. Then CA sends the information $\{PK_{CA}, \delta_j, \chi_{j,r+1}\}$ to the terminal member u_j.

3) After u_j receives the message $\{PK_{CA}, \delta_j, \chi_{j,r+1}\}$, u_j calculates $T_{j,r+1} = {s_{u_j}}^{-1}\chi_{j,r+1} = \iota_{CA,r+1}a_{u_j,r+1}g_1$ and verifies whether the identity of CA and the attribute $a_{u_j,r+1}$ corresponding to the attribute weight $T_{j,r+1}$ by calculating whether the equation $e(\delta_j, g_1) = e(T_{j,r+1}, PK_{CA})$ holds. If the verification is successful, u_j obtains the attribute weight $T_{j,r+1}$ corresponding to attribute $a_{u_j,r+1}$.

At this time, The set of attribute weights of u_j is $\{T_{j,i} \mid i = 1, 2, \ldots, r+1\}(r < R)$. u_j can not only construct the polynomial by the set $\{T_{j,1}, T_{j,2}, ..., T_{j,r}\}$, but also reproduce the polynomial constructed by the set $\{T_{j,1}, T_{j,2}, ..., T_{j,r}, T_{j,r+1}\}$, then calculate the decryption key of the corresponding ciphertext, and upgrade the access authority of the resource.

5.3 Permission revocation

When terminal members are punished, such as reduced trust or illegal operations, certain specific resource access rights may be cancelled. Suppose the current attribute set of terminal member u_j is $attr_{set_j} = \{a_{u_j,1}, a_{u_j,2}, ..., a_{u_j,r}\}(j, r \in N^*, r < R)$. u_j is punished and an attribute $a_{u_j,r}$ is cancelled, then the attribute set of u_j becomes $\widetilde{attr}_{set_j} = \{a_{u_j,1}, a_{u_j,2}, ..., a_{u_j,r-1}\}$. u_j can only use the corresponding attribute weights $\{T_{j,i} \mid i = 1, 2, \ldots, r-1\}$ to access lower-level shared resources. The process of revoking attribute $a_{u_{j,r}}$ of u_j is as follows:

1) CA broadcasts a notice of revocation of serial number $S_{j,r}$ of attribute $a_{u_j,r}$ of u_j;

2) After receiving the notification, RSSP updates the information of u_j in the information sharing platform, that is, cancels the $S_{j,r}$ item in the u_j column.

3) CA selects a random numbers $\widetilde{\iota}_{CA,r} \in \mathbb{Z}_q^*(\widetilde{\iota}_{CA,r} \neq \iota_{CA,r})$ for the attribute $a_{u_i,r}$ of each $u_i(1 \leq i \leq n, i \neq j)$, CA computers $\widetilde{\chi}_{i,r} = \widetilde{\iota}_{i,r}\vartheta_{i,r}(1 \leq i \leq n, i \neq j)$ and its signature $\widetilde{\delta}_{CA} = SK_{CA}\widetilde{\iota_{CA},r}a_{u_i,r}g_1$. Then, CA broadcasts messages $\{PK_{CA}, \widetilde{\delta}_{CA}, (\widetilde{\chi}_{1,r}, \widetilde{\chi}_{2,r}, ..., \widetilde{\chi}_{j-1,r}, \widetilde{\chi}_{j+1,r}, ..., \widetilde{\chi}_{n,r})\}$ to all the register terminal u_i.

4) After receiving the messages $\{PK_{CA}, \widetilde{\delta}_{CA}, (\widetilde{\chi}_{1,r}, \widetilde{\chi}_{2,r}, ..., \widetilde{\chi}_{j-1,r}, \widetilde{\chi}_{j+1,r}, ..., \widetilde{\chi}_{n,r})\}$ from CA, $u_i(1 \leq i \leq n, i \neq j)$ calculates $\widetilde{T}_{i,r} = {s_{u_i}}^{-1}\widetilde{\chi}_{i,r} = \widetilde{\iota}_{i,r}a_{i,r}g_1$. Then, u_i verifies the identity of CA and attribute weight $\widetilde{T}_{i,r}$ of $a_{u_i,r}(1 \leq i \leq$

n) by equation $e(\delta_{CA}, g_1) = e(\widetilde{T}_{i,r}, PK_{CA})$. If it holds,$u_i$ obtains the attribute weight $\widetilde{T}_{i,r}$ corresponding to its attribute $a_{u_i,k}(1 \leq k \leq r)$. u_i updates the previous attribute weight $T_{i,r}$ with $\widetilde{T}_{i,r}$. At this time, u_j cannot calculate the new attribute weight $\widetilde{T}_{j,r}$, and u_j can only access low-level shared resources.

Through the above authority update process, the access authority of terminal members to access certain shared resources can be dynamically upgraded or downgraded.

6 Correctness and Security Analysis

In this section, we discussed the PP-DPAC protocol. The first is the proof of the correctness of the PP-DPAC protocol, then the security of the PP-DPAC protocol is analyzed.

6.1 Correctness

The following theorem proves the correctness of PP-DPAC protocol.

Theorem 1: If any terminal member $u_i(1 \leq i \leq n)$ has a legal attribute sequence set $attr_{set_i} = \{a_{u_{i,1}}, a_{u_{i,2}}, ..., a_{u_{i,r}}\}$, it can satisfy the correctness of the equation $H_1(id_{u_i})\eta_i =?H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})pk_{u_i}$ and $e(\delta_i, g_1) = e(\mu_i, PK_{CA})$, then complete the registration.

Proof. Since $\delta_i = SK_{CA}(\iota_{CA,1}a_{u_i,1} + \iota_{CA,2}a_{u_i,2} + ... + \iota_{CA,r}a_{u_i,r})g_1$, $o_i = s_{u_i}H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})PK_{CA}$, $\vartheta_{i,r}$=$s_{u_i}a_{u_i,r}g_1$, $\chi_{i,k} = \iota_{CA,k}\vartheta_{i,k}$, $T_{i,r} = {s_{u_i}}^{-1}\chi_{i,r} = \iota_{CA,r}a_{u_i,r}g_1$, $\mu_i = (T_{i,1} + T_{i,2} + ... + T_{i,r})$ and according to the properties of the bilinear pairings, it is proved as follows:

$$
\begin{aligned}
&H_1(id_{u_i})\eta_i \\
&= H_1(id_{u_i})SK_{CA}^{-1}o_i \\
&= H_1(id_{u_i})SK_{CA}^{-1}s_{u_i}H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})PK_{CA} \\
&= H_1(id_{u_i})s_{u_i}H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})g_1 \\
&= sk_{u_i}H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})g_1 \\
&= H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})pk_{u_i} \\
&e(\delta_i, g_1) \\
&= e(SK_{CA}(\iota_{CA,1}a_{i,1} + \iota_{CA,2}a_{i,2} + ... + \iota_{CA,r}a_{i,r})g_1, g_1) \\
&= e((\iota_{CA,1}a_{i,1} + \iota_{CA,2}a_{i,2} + ... + \iota_{CA,r}a_{i,r})g_1, g_1)^{SK_{CA}} \\
&= e((\iota_{CA,1}a_{i,1} + \iota_{CA,2}a_{i,2} + ... + \iota_{CA,r}a_{i,r})g_1, SK_{CA}g_1) \\
&= e((\iota_{CA,1}a_{i,1} + \iota_{CA,2}a_{i,2} + ... + \iota_{CA,r}a_{i,r})g_1, PK_{CA}) \\
&= e((T_{i,1} + T_{i,2} + ... + T_{i,r}), PK_{CA}) \\
&= e(\mu_i, PK_{CA})
\end{aligned}
$$

According to the above two equations, if terminal members have a legal set of attributes, they can Verify successfully and complete registration.

Theorem 2: If any terminal member u_j has a legal attribute weight set $\{T_{j,1}, T_{j,2}, ..., T_{j,r}\}$, it can satisfy the correctness of the equation $e(\sigma_{j,m}, pk_{u_j}) = e(\phi_{j,m}, g_1)$ and the key $k_{u_i,m}$. It can obtain the plaintext resource.

Proof. Since $\sigma_{j,m} = sk_{u_j}^{-1} H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1$, $\phi_{j,m} = H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1$ and according to the properties of the bilinear pairings, it is proved as follows:

$e(\sigma_{j,m}, pk_{u_j})$
$=e(sk_{u_j}^{-1} H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1, pk_{u_j})$
$=e(sk_{u_j}^{-1} H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1, pk_{u_j})^{sk_{u_j}^{-1}}$
$=e(H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1, sk_{u_j}^{-1} pk_{u_j})$
$= e(H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1, g_1)$
$= e(\phi_{j,m}, g_1)$

After the above equation is verified, the terminal member obtains $(y_{u_{i,}k}, T_{j,k})(k = 1, 2, ..., r)$ from the RSSP, and u_j calculates the decryption key. Then u_j restore the $(r-1)$ degree polynomial $f(x) = b_{i,t-1}x^{r-1} + b_{i,t-2}x^{r-2} + ... + b_{i,1}x + k_{u_i,m}$ by using the polynomial $g(x) = \sum_{i=1}^{t} y_{u_{j,i}} \prod_{\varpi=1,\varpi\neq i}^{t} \frac{x - H_2(T_{j,\varpi})}{H_2(T_{j,i}) - H_2(T_{j,\varpi})} (t = r)$ according to the Lagrange theorem. Since $\{T_{i,k} = T_{j,k}\}(k = 1, 2, ..., r)$ and $y_{u_i,k}$ is the same, u_j can get $g(0)$=$k_{u_i,m}$. So **Theorem 2** is proved.

6.2 Security Analysis

Theorem 3: For any terminal member $u_i(1 \leq i \leq n)$, if it has attribute $a_{u_i,r}$, it can obtain the attribute weight $T_{i,r}$ corresponding to the attribute $a_{u_i,r}$.

Proof. For terminal member u_i with attribute $a_{u_i,r}$, it can calculate $\vartheta_{i,0} = s_{u_i} PK_{CA}$ $\vartheta_{i,1} = s_{u_i} a_{u_i,1} g_1, \vartheta_{i,2}$=$s_{u_i} a_{u_i,2} g_1, ..., \vartheta_{i,r}$=$s_{u_i} a_{u_i,r} g_1$, $o_i = s_{u_i} H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r}) PK_{CA}$. After receiving the message $\{id_{u_i}, pk_{u_i}, o_i, \vartheta_{i,0}, (\vartheta_{i,1}, S_1), (\vartheta_{i,2}, S_2), \ldots, (\vartheta_{i,r}, S_r)\}$ from the terminal member u_i ,CA calculates $\eta_i = SK_{CA}^{-1} o_i = s_{u_i} H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r}) g_1$, $\psi_k = SK_{CA}^{-1} \vartheta_{i,0} A_k = s_{u_i} g_1 a_{u_i,k} (k = 1, 2, ..., r)$ and verifies the identity of u_i by the equation $H_1(id_{u_i})\eta_i$ =?$H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r}) pk_{u_i}$, $H_2(\psi_1 \| \psi_2 \|, ..., \| \psi_r)$ = ?$H_2(\vartheta_{i,1} \| \vartheta_{i,2} ... \| \vartheta_{i,r})$. Then CA selects a random numbers $\iota_{CA,k} \in \mathbb{Z}_q^*(1 \leq k \leq r)$ for each attribute $a_{u_i,k}$ and calculates $\chi_{i,k} = \iota_{CA,k} \vartheta_{i,k}$. After receiving the messages $\{PK_{CA}, \delta_i, (\chi_{i,1}, \chi_{i,2}, ... \chi_{i,r})\}$ from CA, u_i can obtain attribute weight $T_{i,r}$ corresponding to the attribute $a_{u_i,r}$ by calculating $T_{i,r} = {s_{u_i}}^{-1} \chi_{i,r} = \iota_{CA,r} a_{u_i,r} g_1$.

Theorem 4: If the terminal member $u_j(1 \leq j \leq n, i \neq j)$ does not have attribute $a_{u_j,r}$, it cannot obtain the attribute weight $T_{j,r}$ corresponding to the attribute $a_{u_j,r}$.

Proof. if the terminal member u_j wants to obtain the attribute weight $T_{j,r}$, it needs to calculate the attribute weight $T_{i,r}$ (Due to $T_{j,r} = T_{i,r}$). It can intercept the dialogue between CA and u_i in network communication to obtain information such as $\chi_{i,r}$, $\vartheta_{i,r}$. Then u_j tries to calculate $T_{i,r}$. But because s_{u_i} is a private parameter of u_i , u_j cannot be calculated. Assume that u_j can calculate $T_{i,r}$ from $\chi_{i,r}$ and $\vartheta_{i,r}$. Here is$\chi_{i,k} = \iota_{CA,k} \vartheta_{i,k}$, $T_{i,r} = {s_{u_i}}^{-1} \chi_{i,r} = \iota_{CA,r} a_{u_i,r} g_1$, $\vartheta_{i,r}$=$s_{u_i} a_{u_i,r} g_1$. Let $ag_1 = \vartheta_{i,k}$, $abg_1 = \chi_{i,k}$, $b = \iota_{CA,r}$, the terminal member u_j constructs algorithm $\mathbb{A}$ to calculate $\iota_{CA,k}$

and $T_{i,r} = \iota_{CA,r} a_{u_i,r} g_1 = a_{u_i,r} b g_1$. Solving $\mathbb{A}$ is equivalent to solving the ICDH problem. The ICDH problem is a difficult assumption, so u_j cannot calculate attribute weight $T_{i,r}$ through $\chi_{i,r}$ and $\vartheta_{i,r}$. The above proof shows that only legal terminals have legal attributes to obtain corresponding attribute weights, and illegal terminals or illegal attributes cannot obtain correct attribute weights.

Theorem 5: Any terminal member $u_j (1 \leq j \leq n, i \neq j)$ with an attribute set $attr_{set_j} = \{a_{u_{j,1}}, a_{u_{j,2}}, ..., a_{u_{j,r}}\} (j, r \in N^*, r < R)$ can decrypt the resources of higher confidentiality level by upgrading to obtain attributes $a_{u_{j,r+1}}$.

Proof. According to **Theorem 3**, if terminal member $u_j (1 \leq j \leq n, i \neq j)$ has an attribute set $attr_{set_j} = \{a_{u_{j,1}}, a_{u_{j,2}}, ..., a_{u_{j,r}}\} (j, r \in N^*, r < R)$, it can obtain an attribute weight set $\{T_{j,1}, T_{j,2}, ..., T_{j,r}\}$. Then combined with the attribute sequence set obtained from the information sharing platform, r points $(y_{u_{j,i}}, T_{j,i}) (i = 1, 2, ..., r)$ can be formed. Then u_j restore the $(r-1)$ degree polynomial $f(x) = b_{i,t-1} x^{t-1} + b_{i,t-2} x^{t-2} + ... + b_{i,1} x + k_{u_i,m}$ by using the polynomial $g(x) = \sum_{i=1}^{t} y_{u_{j,i}} \prod_{\varpi=1, \varpi \neq i}^{t} \frac{x - H_2(T_{j,\varpi})}{H_2(T_{j,i}) - H_2(T_{j,\varpi})} (t = r)$ according to the Lagrange theorem. That is, $g(x) = f(x)$, u_j calculates the decryption key $g(0) = k_{u_i,m}$. In the same way, terminal members with attributes can restore r degree polynomial $f(x) = b_{i,t} x^r + b_{i,t-1} x^{r-1} + b_{i,t-2} x^{r-2} + ... + b_{i,1} x + k_{u_i,m}$ after upgrading. Then u_j calculates the decryption key $g(0) = k_{u_i,m}$ to decrypt higher-level resources according to $g(x) = \sum_{i=1}^{t} y_{u_{j,i}} \prod_{\varpi=1, \varpi \neq i}^{t} \frac{x - H_2(T_{j,\varpi})}{H_2(T_{j,i}) - H_2(T_{j,\varpi})} (t = r + 1)$.

Theorem 6: Any terminal member, if it is downgraded, u_j cannot decrypt the resources of the corresponding sensitivity level before the downgrade.

Proof. if terminal member $u_j (1 \leq j \leq n, i \neq j)$ has an attribute set $attr_{set_j} = \{a_{u_{j,1}}, a_{u_{j,2}}, ..., a_{u_{j,r}}\} (j, r \in N^*, r < R)$, it can obtain an attribute weight set $\{T_{j,1}, T_{j,2}, ..., T_{j,r}\}$. Then combined with the attribute sequence set obtained from the information sharing platform, r points $(y_{u_{j,i}}, T_{j,i}) (i = 1, 2, ..., r)$ can be formed. Then according to the Lagrange interpolation theorem $g(x) = \sum_{i=1}^{t} y_{u_{j,i}} \prod_{\varpi=1, \varpi \neq i}^{t} \frac{x - H_2(T_{j,\varpi})}{H_2(T_{j,i}) - H_2(T_{j,\varpi})} (t = r)$, u_j can calculate the decryption key $x = g(0)$. When the level of u_j decreases, his attribute authority parameter changes to $\{T_{j,1}, T_{j,2}, ..., \widetilde{T}_{j,r}\}$. The last point formed by the combination will become $(y_{u_{j,r}}, \widetilde{T}_{j,r})$ different from the previous $(y_{u_{j,r}}, T_{j,r})$. According to the Lagrange interpolation theorem $g(x) = \sum_{i=1}^{t} y_{u_{j,i}} \prod_{\varpi=1, \varpi \neq i}^{t} \frac{x - H_2(T_{j,\varpi})}{H_2(T_{j,i}) - H_2(T_{j,\varpi})} (t = r)$, the calculated decryption key is $\widetilde{x} = g(0)$. Because of $\widetilde{x} \neq x$, the degraded terminal members cannot decrypt the resources of the original sensitivity level.

Theorem 7: For different members to collude with different attribute combinations, this model can resist by registering and calculating permissions. In other words, this model can resist collusion attacks.

Proof. For this model, the attribute serial number set $(S_{j,1}, S_{j,3}, S_{j,4}, ..., S_{j,t})$ of u_j will be sent directly to RSSP by CA during the registration process. Suppose u_j wants to access the resource whose attribute serial number set is $(S_{j,1}, S_{j,2}, S_{j,3}, ..., S_{j,t})$. It needs to form the attribute sequence set $(S_{j,1}, S_{j,2}, S_{j,3}, ..., S_{j,t})$ together with the conspirators. Then u_j calculates $\sigma_{j,m} = sk_{u_j}^{-1} H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1$ and sends the message $\{id_{u_j}, pk_{u_j}, (S_{j,1}, S_{j,2}, ..., S_{j,t}), \sigma_{j,m}, keyword_{i,m}\}$ to RSSP to apply for access to the resource. Then RSSP will match the attribute serial number set $(S_{j,1}, S_{j,2}, ..., S_{j,t})$ in the sent message with the platform $(S_{j,1}, S_{j,2}, ..., S_{j,r})(t \leq r)$ published on the information sharing platform. If it matches, RSSP calculates $\widetilde{\phi}_{j,m} = H_1(S_{j,1} \| S_{j,2} \| ... \| S_{j,t} \| keyword_{i,m}) g_1$ and verifies the identity of u_j by calculating whether equation $e(\sigma_{j,m}, pk_{u_j}) = e(\widetilde{\phi}_{j,m}, g_1)$ holds. If the equation holds, RSSP will provide the ciphertext $c_{i,m}$'s link corresponding to the keyword $keyword_{i,m}$ to u_j. However, the actual value calculated by the RSSP is $\phi_{j,m} = H_1(S_{j,1} \| S_{j,3} \| ... \| S_{j,t} \| keyword_{i,m}) g_1$. Since $\phi_{j,m} \neq \widetilde{\phi}_{j,m}$, $e(\sigma_{j,m}, pk_{u_j}) \neq e(\phi_{j,m}, g_1)$. u_j cannot get the ciphertext link. In other words, this model can resist collusion attacks.

7 Performance Analysis

In addition to security analysis, performance analysis is an important aspect of evaluating the efficiency of the model. In this section, we comprehensively evaluate our model in terms of time complexity and computational consumption. In addition, according to the information provided in literature [31], we compare the models of literature [31–34] with our model, and the results show that our model is more efficient.

In terms of time costs, we use the Java programming language and the Java-based encryption library JPBC library (the library version is JPBC -2.0.0). The configuration of the computer is that the processor is Intel(R) Core(TM) 2 i5-7500 3.4Ghz, running Windows 10 operating system. In terms of operation, compared with bilinear operation and exponentiation operation, addition operation and multiplication operation consume very short time, which is negligible in performance analysis. The data is shown in Table 1:

Suppose r represents the size of the terminal attribute set. n represents the number of access tree nodes. $| l |$ represents the number of rows of the matrix. $| c |$ represents the number of columns of the matrix. t represents the size of the attribute set required by the access policy. $| k |$ represents the number of revoked users. Based on this, we establish a calculation complexity analysis table, as shown in Table 2.

In order to facilitate quantitative analysis of the computational time consumption of the five models, suppose the number of access tree nodes is $n = 10$. The number of rows and columns of the matrix is $| l | = | c | = 5$. The number of revoked users is $| k | = 0$. Based on this, we compare and analyze the three different stages of the five models, as shown in Figures 2, 3, and 4.

Table 1 Execution Time of Each Algorithm

The algorithm	Execution Time
Modular inverse operation (T_{inv})	$T_{inv} \approx 0.0042$ms
Multiplication operation (T_{mul})	$T_{mul} \approx 0.0011$ms
Exponentiation operation (T_{exp})	$T_{exp} \approx 6.6432$ms
Elliptic curve point addition (T_{pa-ecc})	$T_{pa-ecc} \approx 0.003$ms
Elliptic curve point multiplication (T_{sm-ecc})	$T_{sm-ecc} \approx 0.0036$ms
Hash operation (T_h)	$T_h \approx 0.0001$ms
Bilinear pairing (T_{bp})	$T_{bp} \approx 4.3183$ms
Bilinear pairing point addition (T_{pa-bp})	$T_{pa-bp} \approx 0.014$ms
Bilinear pairing scalar multiplication (T_{sm-bp})	$T_{sm-bp} \approx 0.2038$ms

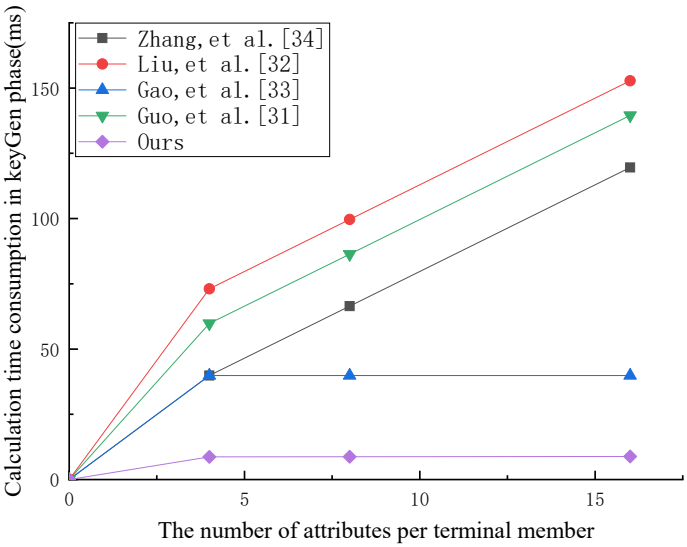

Fig. 2 The time cost in the encryption phase

It can be seen from Figure 2 that in the keyGen stage, our model consumes the least computational time. Followed by the model of Gao et al. [33], the model of Zhang et al. [34], and the model of Guo et al. [31]. The scheme of Liu et al. [32] consumes the most computational time. Among the five models, the model of Liu et al. [32], the model of Guo et al. [31], and the model of Zhang et al. [34] have a faster increase in computing time consumption as the attributes of terminal members increase, and they are not suitable for large-scale attribute application scenarios.

As can be seen from Figure 3, in the encryption phase, the calculation time consumption of the five schemes is independent of the number of attributes of

Table 2 Computational complexity of our model and the models in the other four protocols

Models	keyGen	Encrypt	Decrypt
Zhang, et al. [34]	$(2+r)T_{\exp}+T_{mul}$	$(2+3 \mid l \mid)T_{\exp}+$ $(1+2 \mid l \mid)T_{mul}$	$tT_{\exp}+tT_{mul}+$ $(1+2t)T_{bp}$
Liu,et al. [32]	$(7+r)T_{\exp}+2T_{mul}$	$(3+3 \mid l \mid +4 \mid k \mid)T_{\exp}+$ $(1+2 \mid l \mid + \mid k \mid)T_{mul}$	$(2+2t+2 \mid k \mid)T_{\exp}+$ $(2 \mid k \mid +2t)T_{mul}+$ $(1+2t+2 \mid k \mid)T_{bp}$
Gao,et al. [33]	$(1+ \mid l \mid)T_{\exp}$	$(2+ \mid l \times c \mid)T_{\exp}+$ $(2+ \mid l \times c \mid)T_{mul}$	$rT_{mul}+(1+r)T_{bp}$
Guo,et al. [31]	$(5+r)T_{\exp}+2T_{mul}$	$(4+ \mid l \times c \mid + \mid l \mid)T_{\exp}+$ $(1+ \mid l \times c \mid)T_{mul}$	$(1+2t)T_{\exp}+$ $2tT_{mul}+3T_{bp}$
Ours	$2(r+1)T_{pa-ecc}+$ $rT_{inv}+2T_{bp}+$ $(r+1)T_{mul}+T_h$	$2T_{pa-ecc}+T_{inv}$ $+T_{mul}+3T_h$	$(t+1)T_{mul}+T_{inv}$ $+T_{pa-ecc}+4T_h$

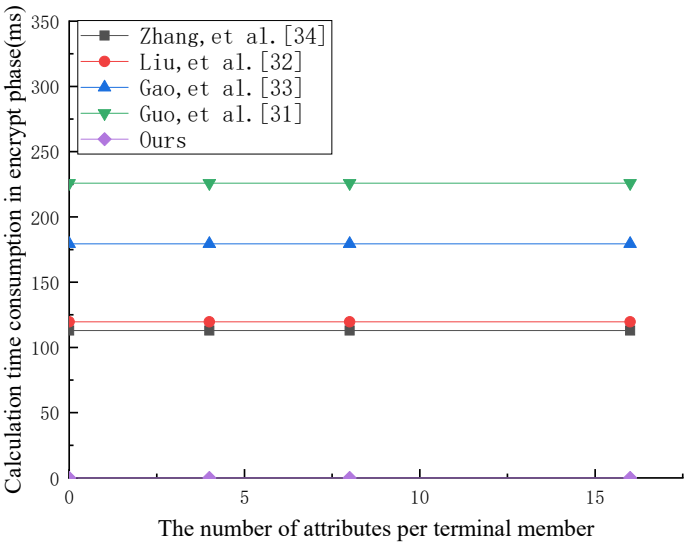

Fig. 3 The time cost in the decryption phase

terminal members. The model of Guo et al. [31] consumes the most computational time. Followed by the model of Gao et al. [33], the model of Liu et al. [32], and the model of Zhang et al. [34]. Among them, Liu et al. [32]'s model and Zhang et al. [34]'s model consume almost the same calculation time. Our model consumes the least computational time.

It can be seen from Figure 4 that in the decryption stage, the model of Liu et al. [32] consumes the most computational time. Followed by the model of Zhang et al. [34], the model of Guo et al. [31], and the model of Gao et al. [33]. Our model consumes the least computational time.

8 Conclusion

In order to solve the problem of privacy leakage in access control, improve the security of the data sharing system. Based on the analysis of current access control research results and existing problems, this paper proposes a dynamic permission access control model based on privacy protection. First, a hidden attribute authentication method is proposed, which can not only hide the identity information of terminal members, but also hide attribute information. It greatly protects the personal privacy of users. It also proposes a dynamic, flexible and fine-grained access rule, which can allow users to access resources with different sensitivity levels through upgrade or downgrade. In addition, it proposes a two-factor authentication mechanism. Under this mechanism, terminal members not only need to authenticate and register, but also need to

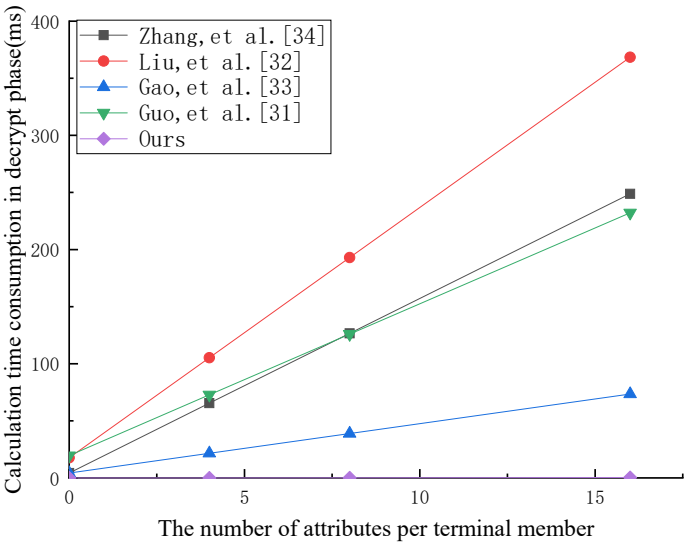

Fig. 4 The communication cost of each model

use attributes to calculate decryption keys to decrypt resources. It can resist collusion attacks. Finally, the correctness and safety of the model is proved, and its performance is analyzed. The results show that the model has higher security and better performance.

Acknowledgments. This work is supported by National Natural Science Foundation of China under Grant (No.61772477, 61971380, U1804263 and 62072037), and the key technologies R & D Program of Henan Province (No.212102210089, 212102210171, 212102210075), and the Key scientific research project plans of higher education institutions in Henan Province (Grant No.21zx014).

References

[1] Ding S, Cao J, Li C, et al. A novel attribute-based access control scheme using blockchain for IoT[J]. IEEE Access, 2019, 7: 38431-38441.

[2] Zhang Q, Zhu L, Li Y, et al. A group key agreement protocol for intelligent internet of things system. Int J Intell Syst. 2021; 1- 24. https://doi.org/10.1002/int.22644

[3] Zhong H, Zhu W, Xu Y, et al. Multi-authority attribute-based encryption access control scheme with policy hidden for cloud storage[J]. Soft Computing, 2018, 22(1): 243-251.

[4] Xu Y, Zeng Q, Wang G, et al. An efficient privacy-enhanced attribute-based access control mechanism[J]. Concurrency and Computation: Practice and Experience, 2020, 32(5): e5556.

[5] Ma Y, Shen M, Zhao Y, et al. Opponent portrait for multiagent reinforcement learning in competitive environment. Int J Intell Syst. 2021; 1- 14. https://doi.org/10.1002/int.22594

[6] Xu Q, Tan C, Fan Z, et al. Secure multi-authority data access control scheme in cloud storage system based on attribute-based signcryption[J]. IEEE Access, 2018, 6: 34051-34074.

[7] Zhang Q, Zhu L, Wang R, et al. Group key agreement protocol among terminals of the intelligent information system for mobile edge computing. Int J Intell Syst. 2021; 1- 20. https://doi.org/10.1002/int.22544

[8] Gao S, Piao G, Zhu J, et al. TrustAccess: A Trustworthy Secure Ciphertext-Policy and Attribute Hiding Access Control Scheme based on Blockchain[J]. IEEE Transactions on Vehicular Technology, 2020.

[9] Zhu Y, Yu R, Ma D, et al. Cryptographic Attribute-Based Access Control (ABAC) for Secure Decision Making of Dynamic Policy With Multiauthority Attribute Tokens[J]. IEEE Transactions on Reliability, 2019, 68(4): 1330-1346.

[10] Sandor V K A, Lin Y, Li X, et al. Efficient decentralized multi-authority attribute based encryption for mobile cloud data storage[J]. Journal of Network and Computer Applications, 2019, 129: 25-36.

[11] Li Y, Yao S, Zhang R, Yang C. Analyzing host security using D-S evidence theory and multisource information fusion. Int J Intell Syst. 2021; 36: 1053–1068. https://doi.org/10.1002/int.22330

[12] Li H, Deng L, Yang C, et al. An enhanced media ciphertext-policy attribute-based encryption algorithm on media cloud[J]. International Journal of Distributed Sensor Networks, 2020, 16(2): 1550147720908196.

[13] Zhang Q, Li Y, Wang R, Liu L, Tan Y-a, Hu J. Data security sharing model based on privacy protection for blockchain-enabled industrial Internet of Things. Int J Intell Syst. 2021; 36: 94-111. https://doi.org/10.1002/int.22293

[14] Hao J, Huang C, Ni J, et al. Fine-grained data access control with attribute-hiding policy for cloud-based IoT[J]. Computer Networks, 2019, 153: 1-10.

[15] Imine Y, Lounis A, Bouabdallah A. Revocable attribute-based access control in mutli-autority systems[J]. Journal of Network and Computer Applications, 2018, 122: 61-76.

[16] Zhang N, Xue J, Ma Y, Zhang R, Liang T, Tan Y-A. Hybrid sequence-based Android malware detection using natural language processing. Int J Intell Syst. 2021; 36: 5770- 5784. https://doi.org/10.1002/int.22529

[17] Zhang Y, Li B, Liu B, et al. An Attribute-Based Collaborative Access Control Scheme Using Blockchain for IoT Devices[J]. Electronics, 2020, 9(2): 285.

[18] Wang S, Zhang Y, Zhang Y. A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems[J]. Ieee Access, 2018, 6: 38437-38450.

[19] Xu Y, Zeng Q, Wang G, et al. An efficient privacy-enhanced attribute-based access control mechanism[J]. Concurrency and Computation: Practice and Experience, 2020, 32(5): e5556.

[20] Guo L, Yang X, Yau W C. TABE-DAC: Efficient Traceable Attribute-Based Encryption Scheme With Dynamic Access Control Based on Blockchain[J]. IEEE Access, 2021, 9: 8479-8490.

[21] Li Y, Wang X, Shi Z, Zhang R, Xue J, Wang Z. Boosting training for PDF malware classifier via active learning. Int J Intell Syst. 2021; 1- 19. https://doi.org/10.1002/int.22451

[22] Yan H, Wang Y, Jia C, et al. IoT-FBAC: Function-based access control scheme using identity-based encryption in IoT[J]. Future Generation Computer Systems, 2019, 95: 344-353.

[23] Zhang Y, Zheng D, Deng R H. Security and privacy in smart health: Efficient policy-hiding attribute-based access control[J]. IEEE Internet of Things Journal, 2018, 5(3): 2130-2145.

[24] Qin X, Huang Y, Yang Z, et al. A Blockchain-based access control scheme with multiple attribute authorities for secure cloud data sharing[J]. Journal of Systems Architecture, 2021, 112: 101854.

[25] Qiu M, Gai K, Thuraisingham B, et al. Proactive user-centric secure data scheme using attribute-based semantic access controls for mobile clouds in financial industry[J]. Future Generation Computer Systems, 2018, 80: 421-429.

[26] Sultana T, Almogren A, Akbar M, et al. Data sharing system integrating access control mechanism using blockchain-based smart contracts for IoT

devices[J]. Applied Sciences, 2020, 10(2): 488.

[27] Wang Q, Lv G, Sun X. Distributed Access Control with Outsourced Computation in Fog Computing[C]//2019 Chinese Control And Decision Conference (CCDC). IEEE, 2019: 2446-2450.

[28] Zhang Y, Kasahara S, Shen Y, et al. Smart contract-based access control for the internet of things[J]. IEEE Internet of Things Journal, 2018, 6(2): 1594-1605.

[29] Xu R, Chen Y, Blasch E, et al. Blendcac: A blockchain-enabled decentralized capability-based access control for iots[C] //2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData). IEEE, 2018: 1027-1034.

[30] Rajput A R, Li Q, Ahvanooey M T, et al. EACMS: emergency access control management system for personal health record based on blockchain[J]. IEEE Access, 2019, 7: 84304-84317.

[31] Guo L, Yang X, Yau W C. TABE-DAC: Efficient Traceable Attribute-Based Encryption Scheme With Dynamic Access Control Based on Blockchain[J]. IEEE Access, 2021, 9: 8479-8490.

[32] Liu Z, Xu J, Liu Y, et al. Updatable ciphertext-policy attribute-based encryption scheme with traceability and revocability[J]. IEEE Access, 2019, 7: 66832-66844.

[33] Gao S, Piao G, Zhu J, et al. TrustAccess: A trustworthy secure ciphertext-policy and attribute hiding access control scheme based on blockchain[J]. IEEE Transactions on Vehicular Technology, 2020, 69(6): 5784-5798.

[34] Zhang Y, He D, Choo K K R. BaDS: Blockchain-based architecture for data sharing with ABS and CP-ABE in IoT[J]. Wireless Communications and Mobile Computing, 2018, 2018.