International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 2579-2591

https://doi.org/10.70517/ijhsa463216

Enhancing Data Protection in Cloud Computing Environments with Encryption and Access Control Mechanisms

Chengcheng Gao[1], Mingwei Liu[2], Ning Li[3], Xingda Gao[1], Xibao Wang[1] and Shizhu Wu[4,*]
[1] The Network Security Lab, Yunding Technology Co., Ltd., Jinan, Shandong, 250000, China
[2] Information Technology Department, Shandong Jiuzhou Xintai Information Technology Co., Ltd., Jinan, Shandong, 250000, China
[3] Ministry of Information and Technology, Shandong Rural Credit Cooperatives Union, Jining, Shandong, 250000, China
[4] Information Technology Department, Shandong Sunshine Digital Technology Co., Ltd., Binzhou, Shandong, 256600, China
Corresponding authors: (e-mail: 13365318522@163.com).

Abstract Data privacy protection in cloud computing environment faces severe challenges, and traditional encryption techniques are difficult to meet the flexible access control requirements. In this study, a cloud computing data protection scheme that integrates attribute-based encryption and access control mechanism is proposed to solve the problem of data security and access control in cloud storage environment. Methodologically, the CP-ABE encryption technique is used in combination with the XACML access control framework to construct a protection mechanism that contains three key phases: system initialization, data storage and data access. The experimental analysis shows that the CP-ABE scheme shows a significant advantage when the number of attributes increases, and the average number of pseudo-tuples increases from 1.5 to 2.25 when the number of attributes increases from 3 to 4. The performance test shows that in the policy attribute revocation scenario, the CP-ABE scheme reduces the computational overhead at the data owner side, and the average number of pseudo-tuples significantly decreases when the number of tuples increases from 1k to 4k, and the average number of pseudo-tuples decreases from 7.6 to 0.94.The CP-ABE scheme with the introduction of joint attributes not only reduces the computational burden on the data owner, but also significantly reduces the overall computational overhead when accessing more attributes of the structural tree, and at the same time ensures the forward and backward security of the data, which realizes the efficient protection and flexible access control of the data in the cloud environment.

Index Terms cloud computing, attribute-based encryption, access control, CP-ABE, data protection, computational overhead

I. Introduction

In today's digital era, cloud computing has become an important means for enterprises and individuals to store, process and share data [1], [2]. However, with the wide application of cloud computing, data security and privacy protection, issues are becoming more and more prominent, and encryption technology and access control mechanisms have received widespread attention as important strategies for protecting data security in cloud computing environments [3]-[5].

The basic principle of encryption technology is by encrypting the data so that no one else can directly read or understand the data content except legally authorized users [6], [7]. In cloud computing, encryption technology can be categorized into two aspects: data transmission encryption and data storage encryption [8]. Through the combined use of data transmission encryption and data storage encryption, the security of cloud data can be effectively protected [9], [10]. With the continuous development of cloud computing, encryption technology will also be continuously innovated and improved to cope with the increasingly complex security challenges [11], [12]. And the access control mechanism is a more applied information security technology, which restricts user data access by effectively verifying the user's identity and privileges [13]-[15]. This technique can avoid unauthorized users from accessing sensitive data, which is conducive to making the security and integrity of data protected [16], [17]. The application of access control technology is particularly important in cloud computing environment, because the data stored in the cloud, users can use the Internet to access anytime and anywhere, which will lead to a significant increase in the risk of illegal access to data [18]-[21]. In this case, the application of access control mechanisms can ensure that only authorized users have access to important data [22], [23]. Access control mechanisms usually involve authentication and privilege management [24]. Identity verification is to confirm the identity of the user, mainly by user name and password verification, data certificate verification and other methods, the management of rights is mainly based on the identity of the user, to assign the appropriate access rights [25]-[27].

The rapid development of information technology has promoted the widespread application of cloud computing services, and enterprises and individuals are increasingly storing data in the cloud to enjoy the convenience and elasticity of scalability brought by cloud computing. However, storing sensitive data with third-party cloud service providers also raises serious data security and privacy protection issues. In cloud service environments, data storage and computation are usually controlled by different entities, and traditional access control and security mechanisms are difficult to meet the complex security requirements in distributed environments. Especially in multi-user collaboration scenarios, it becomes an important challenge to ensure that data can only be accessed by users with specific attributes or meeting specific conditions. In addition, the security status of the cloud platform itself is also a concern, as malicious attackers may illegally access sensitive data stored in the cloud through various means.

Traditional encryption techniques can safeguard data confidentiality but lack flexible access control capabilities. Symmetric encryption algorithms are efficient but complex in key management, while asymmetric encryption algorithms have high computational overhead. These methods usually treat encryption and access control as independent problems, making it difficult to realize fine-grained access policies. In this context, novel security mechanisms that balance data encryption and access control become necessary.

Attribute-based encryption (ABE), as a novel public key encryption technique, offers the possibility to realize fine-grained access control. It allows the data owner to encrypt the data based on user attributes or access policies, and only users who fulfill specific conditions can decrypt the data. However, the pure ABE scheme suffers from low computational efficiency and complex attribute revocation in practical applications. On the other hand, standardized access control frameworks such as XACML provide flexible policy expression and enforcement mechanisms, but lack cryptographic protection capabilities. This study proposes a comprehensive solution for the complex requirements of data protection and access control in cloud computing environment. First, the security threats and access control requirements in cloud environment are deeply analyzed to clarify the protection objectives; second, the advantages of attribute-based encryption and access control models are integrated to construct a unified security framework; third, the joint attribute concept is introduced to optimize the performance of CP-ABE scheme; fourth, a complete system architecture is designed, including three key phases, namely, system initialization, data storage, and data access; Finally, the security and performance advantages of the scheme are verified through experiments, especially in terms of computation overhead, forward security and backward security. Through this research, it is expected to provide theoretical foundation and practical guidance for data security and access control in cloud computing environment.

II. Encryption technology

II. A.Basic concepts

Definition 1 Bilinear pairs

Let G_1 and G_2 both be multiplicative cyclic groups of order p, where p is a prime, g is a generator of group G_1, and Z_p is a prime cyclic group of order p, and define a mapping relation $e: G_1 \times G_1 \to G_2$ on G_1 and G_2, which is called a bipartite pair when e satisfies the following property:

(1) Bilinear: for any $g_1, g_2 \in G_1$ and any $a, b \in Z_p$, it satisfies $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$.

(2) Non-degeneracy: there exists an element $g \in G_1$ that satisfies $e(g, g) \neq 1$.

(3) Computability: there exist efficient polynomial algorithms making it possible to compute $e(u, v)$ efficiently for any $u, v \in_{G_1}$.

Definition 2 Lagrange interpolation theorem

If given $n+1$ distinct points $(x_0, y_0), (x_1, y_1), \cdots, (x_n, y_n)$ in the plane, where $x_0, x_1, \cdots, x_n$ are each distinct, there exists a unique polynomial $f(x)$ of order not exceeding n, which can be uniquely determined as polynomial $f(x)$ by the following formula:

$$f(x) = \sum_{i=1}^{n} (f(x_i) \cdot l_i(x)) = \sum_{i=1}^{n} \left(f(x_i) \cdot \prod_{j=0, j \neq i}^{n} \frac{(x - x_j)}{(x_i - x_j)} \right) \tag{1}$$

where $l_i(x) = \prod_{j=0, j \neq i}^{n} \frac{(x - x_j)}{(x_i - x_j)}$, and $l_i(x)$ is the Lagrange coefficient.

Definition 3 DBDH assumption

The adjudicated bilinear Diffie-Hellman (DBDH) problem refers to the definition of the multiplicative cyclic groups of order p and bilinear mappings $e: G_1 \times G_1 \to G_2$, where p is a prime, for both G_1 and G_2. Given the generating element $g \in G_1$, choose $a,b,c \in Z_p, T \in G_2$, decide whether $T = e(g,g)^{abc}$ holds, and if the equation holds, say (g, g^a, g^b, g^c, T) is a DBDH tuple, Algorithm A outputs 1, otherwise it outputs 0. If the advantage of Algorithm A in solving the DBDH problem is satisfied:

$$Adv^{DBDH} = |\Pr\left[A(g, g^a, g^b, g^c, e(g,g)^{abc}) = 1\right] - \Pr\left[A(g, g^a, g^b, g^c, T) = 1\right]| \geq \varepsilon \tag{2}$$

Then the algorithm is said to solve the DBDH problem by ε advantage.

The DBDH assumption is said to hold for the group (G_1, G_2) if there does not exist an algorithm in polynomial time that solves the DBDH problem with at least ε advantage. That is, $(g, g^a, g^b, g^c, e(g,g)^{abc})$ and (g, g^a, g^b, g^c, T) are indistinguishable in polynomial time.

II. B.Mechanisms for secret sharing

The secret sharing one has the advantages of high security and easy-to-understand principle, which is more widely used and is the classic scheme of secret sharing, so the following is an example of Shamir threshold secret sharing scheme [28].

The basic idea of Shamir's secret sharing scheme is that for a threshold (t,n), where $(t < -n)$, given n participants to share the secret s, only t of them are needed to recover the shared secret s, and the shared secret cannot be recovered when there are fewer than t participants.Shamir's secret sharing scheme utilizes the classical Lagrange interpolation theorem with the following procedure:

(1) Assume that the shared secret to be shared is $s \in Z_q$, where q is a prime number, and the shared secret s is to be assigned to n participants $q_i (1 \leq i \leq n)$.

(2) Arbitrarily select $t-1$ random numbers $\{a_1, a_2, \cdots, a_{t-1}\} \in Z_q$ to construct a $t-1$ -order polynomial $f(x) = s + a_1 x + \cdots + a_{t-2} x^{t-2} + a_{t-1} x^{t-1}$.

(3) Arbitrarily select n non-zero elements $\{x_1, x_2, \cdots, x_{t-1}\} \in Z_q$, compute $y_i = f(x_i)$, and assign (x_i, y_i) to each participant q_i, where $1 \leq i \leq n$.

(4) According to the Lagrange interpolation theorem, (x_i, y_i) is the point on the $t-1$ -order polynomial $f(x) = s + a_1 x + \cdots + a_{t-2} x^{t-2} + a_{t-1} x^{t-1}$ through t points one can uniquely determine a polynomial $f(x)$ of order $t-1$, and then determine s according to $f(0) = s$. That is, the shared secret s can be recovered through the shared secret share of t participants, and the shared secret s cannot be recovered when the shared secret share of participants is less than t.

The Linear Secret Sharing Scheme (LSSS) is a generalization of the Shamir Threshold Secret Sharing Scheme. A linear secret sharing scheme defined on a set P of participants is linear needs to satisfy the following two conditions:

(1) The secret share of each participant constitutes a vector on Z_p.

(2) There exists a linear matrix M consisting of m rows and n columns. For $i = 1, ..., m$, $\rho(i)$ is a set of $\{1, 2, ..., m\}$ to ρ, and $\rho(i)$ denotes the attribute associated with the i th row M_i of the matrix M. Given a column vector $\vec{v} = (s, r_2, \cdots, r_n)^T$, where $s \in Z_p$ is the secret to be shared, $r_2, \cdots, r_n \in Z_p$ is randomly chosen, the vector $M \cdot \vec{v}$ denotes the secret share of the secret to be shared s into m shares, which are assigned to each of the m participants, there are secret shares $\lambda_i = (M \cdot \vec{v})_i$, that is, $M_i \cdot \vec{v}$ assigned to the participant $p(i)$:

$$M \cdot \vec{v} = \begin{pmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & \ddots & \vdots \\ m_{m1} & \cdots & m_{mn} \end{pmatrix} \cdot \begin{bmatrix} s \\ r_2 \\ \vdots \\ r_n \end{bmatrix} = \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_m \end{bmatrix} \tag{3}$$

It is proved that the LSSS matrix satisfies the above conditions with linear reconstruction characteristics. It is defined as follows: let the linear secret sharing scheme of access structure A satisfy the above conditions, let $S \in A$ be the authorized set, define the set $I \in \{1,\cdots,m\}$ is $I = \{i : \rho(i) \in S\}$, and exist the constant $\{\omega_i \in Z_p\}_{i\in I}$ satisfies $\sum_{i\in I} \omega_i M_i = (1,0,\cdots,0)$, if $\{\lambda_i\}$ is the valid secret share of s, then there is $\Sigma_i \in I \omega_i \lambda_i = s$.

Moreover, these constants $\{\omega_i \in Z_p\}_{i\in I}$ can be found in polynomial time in the matrix M, and no such constants exist for any unauthorized set. That is, if $\{\lambda_i\}$ is a valid secret share of s, by means of a set of constants computed in polynomial time $\{\omega_i \in Z_p\}_{i\in I}$, and $\sum_{i\in I} \omega_i \lambda_i = s$, the shared secret s can be recovered.

II. C.Attribute-based encryption scheme

II. C. 1) Attribute Encryption Based on Key Policies

In the key policy-based attribute encryption scheme KP-ABE, the key corresponds to an access control policy and the ciphertext corresponds to a set of attributes, and the key can be used to decrypt the ciphertext if and only if the set of attributes associated with the ciphertext satisfies the access control policy associated with the key. The access control policy is formulated by the message receiver, and the access control structure is embedded in the key, and the message receiver can successfully decrypt the message when and only when the access control structure of the message receiver satisfies the attribute set of the message sender [29]. KP-ABE is more suitable for query applications, such as directional broadcasting and pay-per-view systems, etc., where the message receiver formulates an access control policy, qualifies the pay-per-view The corresponding video ciphertext can be successfully decrypted by the message receiver only when the attributes associated with the ciphertext satisfy the access control policy.

II. C. 2) Attribute Encryption Based on Ciphertext Policy

In the ciphertext policy-based attribute encryption CP-ABE scheme, the ciphertext corresponds to an access control policy and the key corresponds to a set of attributes, and the key can be used to decrypt the ciphertext if and only if the set of attributes associated with the key satisfies the access control policy associated with the ciphertext. Different from KP-ABE, CP-ABE has the message sender formulate the access policy, embed the access structure into the ciphertext, and decrypt the message when and only when the set of attributes satisfies the access policy, which is more widely used in reality, and is suitable for access control applications, such as e-healthcare systems and social networking sites. The user can successfully decrypt the message [30].

The specific construction process of the CP-ABE scheme includes the following steps:

(1) System initialization Setup

This step for randomized algorithms, authorized by a third party trusted center, set G_1 and G_2 is order to p the multiplication of cyclic group, including p is a prime number, g is a group of G_1 generation, Z_p is a prime number p order cyclic group, and defines the G_1 and G_2 on the mapping relation $e : G_1 X G_1 \to G_2$ for bilinear mapping, Define the mapping $H : \{0,1\} \to G_2$, choose a random number $\alpha, \beta \in Z_p$, generate principal private key and public key PK MSK is as follows:

$$PK = \left\{G_1, g, h = g^{\beta}, f = g^{1/\beta}, e(g,g)^{\alpha}\right\} \tag{4}$$

$$MSK = \left\{g^{\alpha}, \beta\right\} \tag{5}$$

(2) Key Generation KeyGen

This step is a randomization algorithm, executed by a third-party trusted authorization center, which inputs PK, MSK, and the set of attributes A_u provided by the visitor U_x, to generate the user's decrypted private key SK. where $r \in Z_p$ is a random number, and for each attribute j in the set of attributes Au provided by the user, randomly select $r_j \in Z_p$:

$$SK = \left\{D = g^{(\alpha+r)/\beta}, \forall j \in A_u : D_j = g^r \cdot H(j)^{r_j}, D_j^{'} = g^{r_j}\right\} \tag{6}$$

(3) EncryptEncrypt

This step is a randomization algorithm that inputs the data to be encrypted M, the public key PK, and the access structure Tu to generate the ciphertext CT.The specific process is as follows:

The access structure T_u is the access structure tree, starting from the root node r and going up and down, for each node in the access control tree, a polynomial q_x with a random number of times of d_x is chosen, where $d_x = k_x - 1$, and k_x is the threshold value of the node. Limit value.

For the root node r, randomly choose $s \in Z_p$ such that $q_r(0) = s$; for the general node $x, q_x(0) = q_{parent(x)}(index(x))$.

Let Y be the set of all leaf nodes in the access structure tree T_u, denoting the set of all attributes in this access structure tree, and $attribute(y)$ denotes the attribute associated with node y. The ciphertext CT is computed by the following formula:

$$CT = \{T, \tilde{C} = M \cdot e(g,g)^{\alpha s}, C = h^s, \forall y \in Y : C_y = g^{q_y(0)}, C'_y = H(attribute(y))^{q_y(0)}\} \tag{7}$$

(4) Decrypt

This step uses a recursive algorithm to decrypt the corresponding plaintext data M by inputting the ciphertext CT, the decryption sum key SK associated with the user attribute set Au, and the node x in the access structure tree.The specific process is as follows:

Define a recursive function DecryptNode (CT, SK, x) where x is a node in the access structure tree.

If x is a leaf node in the access structure tree, let $i = attribute(x)$, if $i \in A_u$, then:

$$\begin{aligned} e(CT, SK, x) &= \frac{e(C_x, D_i)}{e(C_x, D'_i)} \\ &= \frac{e(g^{q_x(0)}, g^r \cdot H(i)^{r_i})}{e(H(attribute(x))^{q_x(0)}, g^{r_i})} \\ &= \frac{e(g^{q_x(0)}, g^r) e(g^{q_x(0)}, H(i)^{r_i})}{e(H(i)^{q_x(0)}, g^{r_i})} \\ &= \frac{e(g,g)^{q_x(0) \cdot r} e(g, H(i))^{q_x(0) \cdot r_i}}{e(g, H(i))^{q_x(0) \cdot r_i}} \\ &= e(g,g)^{q_x(0) \cdot r} \end{aligned} \tag{8}$$

If $i \notin A_u$, then DecryptNode $(CT, SK, x) = \perp$.

If x is a non-leaf node in the access structure tree, proceed as follows: for all leaf nodes z of x, recursively compute $F_Z = \text{DecryptNode}(CT, SK, z)$ such that S_x denotes the set consisting of k_x child nodes of node x, and for all nodes in the set S_x satisfy $F_z \neq L$. If no such set S_x exists, the output of the algorithm is empty. If it exists, let $i = index(z), S'_x = \{index(z) : z \in S_x\}$, and compute it as follows using the Lagrange interpolation theorem:

$$\begin{aligned} F_x &= \prod_{z \in S_x} F_z^{\Delta_{iS'_x}(0)} \\ &= \prod_{z \in S_x} (e(g,g)^{r \cdot q_s(0)})^{\Delta_{i,S'_x}(0)} \\ &= \prod_{z \in S_x} (e(g,g)^{r \cdot q_{parent(s)}(index(s))})^{\Delta_{i,S'_x}(0)} \\ &= \prod_{z \in S_x} e(g,g)^{r \cdot q_x(i) \cdot \Delta_{i,S'_x}(0)} \\ &= e(g,g)^{r \cdot q_x(0)} \end{aligned} \tag{9}$$

Call the function DecryptNode (CT, SK, x) on the root node r of the access structure tree T_u. If the set of user attributes A_u satisfies the access structure tree T_u, i.e., $T(A_u)=1$, DecryptNode $(CT, SK, r)=e(g,g)^{r \cdot s}$, and the formula for computing the decrypted ciphertext M is as follows:

$$\begin{aligned}(C, D) &= e(h^s, g^{(\alpha+r)/\beta}) \\ &= e(g^{\beta \cdots}, g^{(\alpha+r)/\beta}) \\ &= e(g,g)^{(\alpha+r)s}\end{aligned} \tag{10}$$

$$\begin{aligned}M &= \tilde{C} \cdot \frac{DecryptNode(CT, SK, r)}{e(C, D)} \\ &= \tilde{C} \cdot \frac{e(g,g)^{r \cdots}}{e(g,g)^{(\alpha+r)s}} \\ &= \frac{\tilde{C}}{e(g,g)^{\alpha s}}\end{aligned} \tag{11}$$

III. Access control scheme based on attribute-based encryption

III. A. Problem analysis

Because in cloud storage access control, it will involve the operation of many data resources, for some private content and so on can utilize the cryptographic mechanism. Based on the research about a large number of cryptographic mechanisms, combined with the current cloud computing access control model, this paper chooses to use a combination of ABE and attribute-based encryption scheme, on the basis of ABE, combined with the advantages of attribute-based encryption scheme, to obtain an access control model as shown in Fig. 1, to analyze the problems existing therein, and to propose a corresponding model [31].

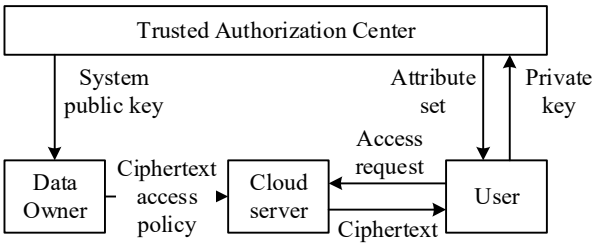

Figure 1: Access control model

III. B. Program design

III. B. 1) System initialization

In this scheme, firstly, the cloud storage provider saves the entity attributes (subject, resource and environment attributes) and the access permission attribute sets in the postgresql database.The XACML mechanism, based on the set of entity attributes and access permission attribute sets, utilizes the Attribute Selection Module to select certain attributes to generate the XACML access control policy, which is stored in the cloud. The trusted authorization center calls the attribute selection module and selects the mixed attribute set A consisting of the user's basic and access authority attribute sets, and generates the master key MK and the system public key PK by using Setup of CP-ABE.

The task of attribute authority in the scheme is for the construction, maintenance, and organization of entity attributes and other operations. Entity attributes are mainly composed of attributes including subject (name, ID, unit, section, position, title, age, etc.), resource (name, ID, creation time, owner ID), and environment (current network status, storage utilization, time). It divides the user's attribute set can be divided into two parts: the basic and the attribute set of access rights.

Setup generates the master key MK and the system public key PK, PK is public to all members in the system, while MK is kept by the Trusted Authorization Center (TA). Remember that the attribute set of subject is $S=\{attr_1, attr_2, \cdots attr_n\}$, the attribute set of resource is $R=\{attr_1, attr_2, \cdots attr_n\}$, the attribute set of environment is

$E=\{attr_1,attr_2,\cdots attr_n\}$, the attribute set of basic attributes is $B=\{attr_1,attr_2,\cdots attr_n\}$, the attribute set of access privilege is $P=\{attr_1,attr_2,\cdots attr_n\}$, and the attribute set of hybrid is $A=S\cup R\cup E=B\cup P$.

The system establishment algorithm is mainly executed by the third-party authorization center during the system initialization, which mainly generates the system public key and master key.The core idea of the Setup algorithm is to generate the public key and master key of the system by using the security parameters of the system and the mixed attribute set of the user, and by using the knowledge of cryptography such as bilinear pairs.

The input of the algorithm is the system security parameter λ and the mixed attribute set A. The output of the algorithm is the system public key PK and the master private key MK. First, a suitable encoding function ρ is chosen to map each attribute $attr_i$ to a unique value s_i, i.e., $\rho(attr_i)=s_i\in G$, where G and G_τ are the groups of two primes of order p. g is a generating element of G defining a bilinear map on G. Secondly the parameters α,β are chosen randomly from the set of integers Z_p, and finally the values of PK and MK are computed with $PK=\{G,g,h=g^\beta,e(g,g)^\alpha\},MK=(\beta,g^\alpha)$.

III. B. 2) Storage of data

According to the definition of the security model, it can be seen that the cloud storage provider is not completely feasible, so the data sender to ensure the confidentiality of the data when storing the data to the cloud storage center is required to perform encryption operations on the data first. Considering the low encryption efficiency of CP-ABE itself, it is necessary to judge the security level of the data and other resources, select the algorithm of the corresponding level, perform symmetric or asymmetric encryption operations, and generate the key and ciphertext before performing CP-ABE. This scheme encrypts the key in the Encrypt() encryption processing of CP-ABE.

Assuming the entity is F, using symmetric key algorithm E, the corresponding key is K_F, the corresponding ciphertext is C_F, and the ciphertext after CP-ABE encryption is C_T, and the detailed process is as follows:

1)First define the unique identity ID_F of F, randomly select the key K_F, select different encryption algorithms according to the content of the entity and so on, and encrypt F to get C_F.

2) Define the access structure tree T based on the access privilege attribute customized by the owner of F and call the CP-ABE encryption algorithm to execute $Encrypt(PK,K_F,T)$ with T as the parameter to encrypt K_F to produce the ciphertext CT_T. $Encrypt(PK,K_F,T)$.

The inputs to the algorithm are the system public key PK, the symmetric key K_F, the access structure tree T, and the output of the algorithm is the encrypted ciphertext CT_T. First, let each node of the access structure tree T be x and pick the polynomial q_x. Let the threshold value of node x be k_x, and compute the order d_x of the polynomial q_x with $d_x=k_x-1$. Second, for the root node r a number s is randomly chosen from the set of integers z_p, and other values are chosen according to polynomial interpolation to determine the polynomial $q_r(0)=s$. Again, for accessing other nodes x in the structured tree T, such that $q_x(0)=q_{parent(x)}(ind(x))$. Then a randomly chosen d_x point is chosen and the complete polynomial is determined in turn. Finally, the encrypted ciphertext is output.

3) The data sender uploads the ciphertext C_F, the encrypted key CT_τ, and the set of access privilege attributes A_c to the cloud storage center, and then the cloud service provider runs the algorithms related to the data storage, etc., and saves C_F and C_τ. List of basic information about the data.

III. B. 3) Access to data

When a cloud user is interested in a resource F such as cloud data, the cloud service provider will cloud obtain the list of access privilege attribute sets of F and judge the user's access control according to it. The specific implementation of the access control on the F stored in the cloud storage center is accomplished by the combined efforts of two parts, one of which is the cloud service provider and the other is the sender of the F. The detailed flow of access is as follows:

1) Firstly, the user password authentication request is made through the cloud service provider, if it fails, an error message is returned, otherwise the user is given a

The user is issued a token etc. for the next step.

2) Within the validity period of the token, for the user's access request according to the XACML framework mechanism, PAP selects the F policy from the policy library, PIP selects the relevant subject, resource, environmental attributes and other information from the database, PDP determines whether the user's attribute set

satisfies the two aspects of this PAP and PIP based on the content of the above information. If it is allowed, the next operation is carried out, otherwise an error message is returned.

3) When the user requests to browse F_1 for the first time, the authorization center generates the corresponding private key based on the mixed attribute set A composed of the user's basic attribute set and access permission attribute set. When the user accesses F_1 again, the cloud storage service provider checks whether its access permission attribute set has changed according to the access permission list, and if it has changed, it reacquires the corresponding private key, otherwise, it directly proceeds to step 4.

The Generate Private Key algorithm is mainly executed by the third-party authorization center during the system initialization, which mainly generates the user's private key SK_u .The core idea of the KeyGen algorithm is to get the user's private key by using the user's acquired master key and the mixed attribute set as well as cryptographic bilinear pairs and other calculations.

The input to the algorithm is the master key MK and the set of mixed attributes A. The output of the algorithm is the user's private key SK_u . First, the parameters α, β, γ are randomly chosen from the set of integers Z_p , where G and G_T are two groups of order prime p , and g is a generating element of G defining a bilinear mapping on G . Next, for each attribute j in turn, compute $D_j = g^{\gamma} H(j)^{\gamma}$ and $D_j^{'} = g^{\gamma}$, and finally, compute the value of SK_u , $SK_u = (D, \forall j \in A : D_j D_j^{'})$.

4) Execute the decryption algorithm Decrypt (CT, SK, PK) , first get the symmetric key K_F , and then use K_F to get the content that the cloud visitor needs to browse.

The decryption algorithm is mainly executed by the client when it uses the decryption operation on F and mainly decrypts the corresponding F . The core idea of the $Decrypt(CT_T, SK_u, PK)$ algorithm is to decrypt the corresponding file using the user's private key, the system's public key, and the user's encrypted file, and to decrypt the corresponding file using the knowledge of cryptography's bilinear reciprocity.

The input to the algorithm is the private key SK_u , the encrypted symmetric key CT_T and the system public key PK , and the output of the algorithm is the symmetric key K_F . First, check whether the ciphertext is corrupted or not, if it is intact continue, otherwise return an error. Second, according to whether the x node's is a leaf node or not, discussed in cases, first x is a leaf node, calculate $DecryptNode(CT, SK, x) = e(C_x, D_i) / e(C_x^{'}, D_i^{'}) = e(g, g)^{q_x(0) r}$; otherwise, recursively compute the x -child node values with the formula $F_z = DecryptNode(CT_T, SK_u, z)$, and let S_x be a collection of size k_x , and if all of the child nodes can be passed through T , then $F_z \neq \perp$. Instead, return an error. Compute $R = F_x = \prod_{z \in S_x} F_z^{\Delta_{iS_x}^{(0)}} = e(g, g)^{\eta q_x(0)}$, and finally output the decrypted symmetric key $K_F = C \cdot R / e(C, D)$.

IV. Security and performance analysis

IV. A. Security analysis

IV. A. 1) Data security

The data security is mainly divided into the security of ciphertext data and the security of key ciphertext. It is analyzed as follows:

For the key ciphertext, the SCP-ABE scheme used to encrypt the key is encrypted in the ABE-AC scheme, and the security of this operation has been proven. When a policy is changed, the security of ciphertext data mainly includes the security of adding policy attributes and the security of revoking policy attributes. When the policy attribute is added, the ABE-AC does not leak additional information to the CSP. In the process of policy attribute revocation, it is divided into two situations: ordinary attribute revocation and joint attribute revocation. When revoking a normal attribute, the data owner needs to send $k_f^{'} \cdot k_f^{-1}$ and $s^{'} - s$ to the CSP. When revoking the union attribute, the data master DO needs to send $k_f^{'} \cdot k_f^{-1}, s^{'} - s$ and $s_j^{'} - s_j$ to the CSP, so this article needs to analyze the CSP when it has $k_f^{'} \cdot k_f^{-1}$, $s^{'} - s$ and the impact on the security of your data after $s_j^{'} - s_j$. Since $k_f^{'}, s^{'}$ and $s_j^{'}$ are all randomly generated parameters by the data owner, and have no relationship with the key ciphertext C_K , and the parameters are in the following form $k_f^{'} \cdot k_f^{-1}$, $s^{'} - s$ and $s_j^{'} - s_j$, so CSPs have $k_f^{'} \cdot k_f^{-1}$, $s^{'} - s$, and $s_j^{'} - s_j$ are just a few random parameters after changes, thus ensuring the security of the solution.

For the data ciphertext C_f, this paper first uses symmetric encryption, and then saves it in the form of data ciphertext C_f in the CSP. When there is a policy change, the information that the CSP can obtain is the data C_f in ciphertext state and the random number $k_f^{'} \cdot k_f^{-1}$ after the form change, which is secure for the data ciphertext.

IV. A. 2) Forward and backward security

The re-encryption operation of the ABE-AC scheme ensures the backward and forward security of the data. When a user is not granted permission before, he cannot decrypt the ciphertext because he cannot get the encryption key, which is backward security. When a user's privilege is revoked, the CSP immediately performs the re-encryption operation and encrypts the re-encryption key by utilizing the access structure, and the revoked user cannot get the re-encryption key, and the decryption key that he used to have will lose the ability to decrypt the ciphertext, which is forward security.

IV. B. Performance analysis

This article selects the U.S. Census Bureau's adult database, which is linked to http://archive.ics.uci.edu/ml/datasets/Adult. The tuples in the database include 14 categorical attributes, including "age, job category, final weight, education level, education number, marital status, occupation, relationship, race, gender, capital gains, capital loss, hours worked per week, nationality." Select Occupation as the sensitive attribute and other attributes as the quasi-identifier attributes. The domain size of Occupation is 14, which means that there are 14 different selectable values in the Occupation column. The results are shown in Figure 2, from which it can be concluded that the proposed algorithm has a similar number of pseudo-tuples as the traditional m-invariance algorithm, and the use of pseudo-nuples is used to maintain the consistency of the distribution of the original QI groups, so the use of this algorithm and the traditional m-invariance algorithm has little impact on the number of pseudo-tuples.

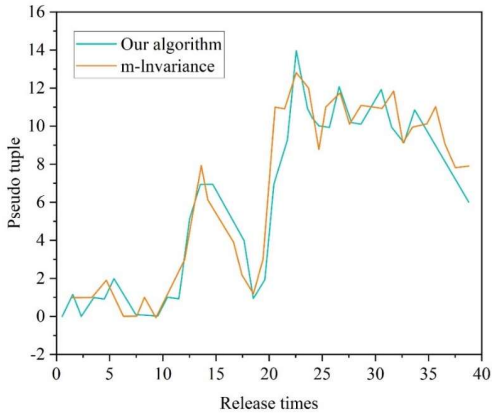

Figure 2: The number of pseudo-meta-groups and the number of releases

The relationship between the variable m and the average number of pseudo-tuples is shown in Figure 3. When m = 3, the average number of pseudo-tuples is 1.5, and when m = 4, the average number of pseudo-tuples in this algorithm is about 2.25, and the average number of pseudo-tuples in the traditional m-invariant algorithm is about 2.2. It can be seen that the number of pseudo-tuples increases when m increases, since a larger m implies more optional sensitivities, resulting in a higher likelihood of imbalanced buckets.

The variation of the number of pseudo-tuples between the present algorithm and the traditional m-invariant algorithm when the number of tuples r varies is shown in Fig. 4. The average number of pseudo-tuples of the present algorithm is about 7.6 when r = 1k, and the average number of pseudo-tuples of the traditional m-invariant algorithm is about 7.5, and the average number of pseudo-tuples of the present algorithm is about 0.94 when r = 4k and the average number of pseudo-tuples of the traditional m-invariant algorithm is about 1.1, and the average number of pseudo-tuples of the conventional m-invariant algorithm is about 0 when r = 8k. It can be seen that, when r increases, the pseudo-tuples' number decreases because a larger r means that there are more tuples to fill the bucket and more opportunities to balance the bucket.

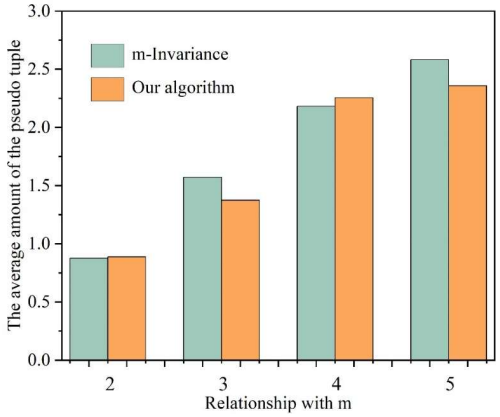

Figure 3: The relationship between the pseudo-element and m

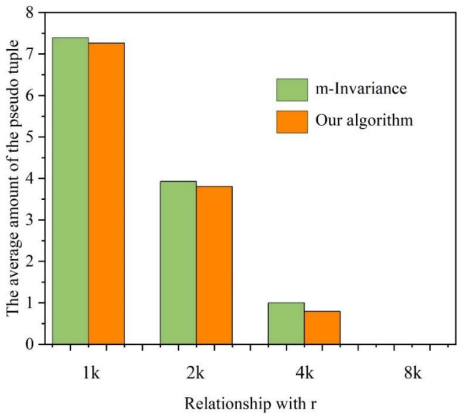

Figure 4: The relationship between the pseudo-element and m

IV. C. Experimental analysis

In this section, we experimentally verify the total elapsed time when the plaintext is finally obtained in the above two scenarios, with the environment and configuration of VS2012 Windows7 64-bit, Intel(R) Core(TM) i7-14590 CPU @ 3.60GHz and 12GB RAM. The time consumptions of one modulo power operation, multiplication operation, and bilinear pair operation in the scheme are calculated using the bilinear development kit PBC library file with version number pbc-0.5.1vc, respectively. This section compares the computational overhead at the DO side and the total computational overhead of the two schemes, KP-ABE, an attribute-based cryptographic access control scheme based on key policy, and CP-ABE, an attribute-based cryptographic access control scheme based on ciphertext policy, at the time of policy attribute addition and policy attribute revocation, respectively.

The results related to the computation overhead at the DO side when policy attributes are added are shown in Figure 5. The experimental results show that when adding policy attributes, with the increase of the number of attributes, the advantage of its joint attributes is slowly highlighted, and the growth rate of the computation overhead of the CP-ABE scheme is lower than that of the KP-ABE scheme.

The total computational overhead of the two schemes when policy attributes are added is shown in Fig. 6. The experimental results show that when the number of attributes is relatively small, the computational overhead is higher than the KP-ABE scheme because the CP-ABE scheme decrypts one more time than the KP-ABE scheme when decrypting at the user's end, but as the number of attributes increases and the joint attributes increase, the computational overhead of CP-ABE will be less than that of the KP-ABE scheme.

The computational overhead of the two schemes at the DO side during policy attribute revocation is shown in Fig. 7. The experimental results show that when the policy attribute is revoked, for the re-encryption operation of the data ciphertext, the KP-ABE scheme is completed at the DO side, for this reason, the KP-ABE scheme needs the DO side to perform an SCP-ABE decryption, as well as encryption operation of the data ciphertext, while the CP-ABE scheme gives the re-encryption of the data ciphertext to the CSP for processing, which does not need to be processed at the DO side, which greatly reduces the computational overhead at the DO side. This greatly reduces the computation overhead at the DO side.

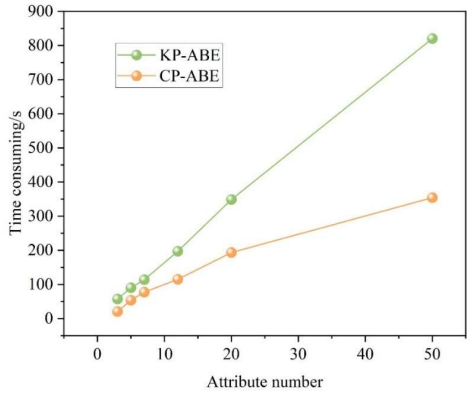

Figure 5: Do end calculation overhead

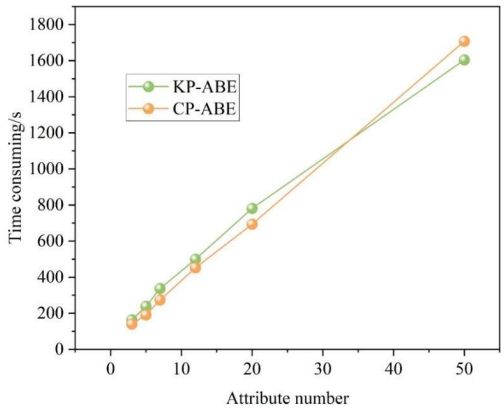

Figure 6: The total calculation overhead of the two schemes

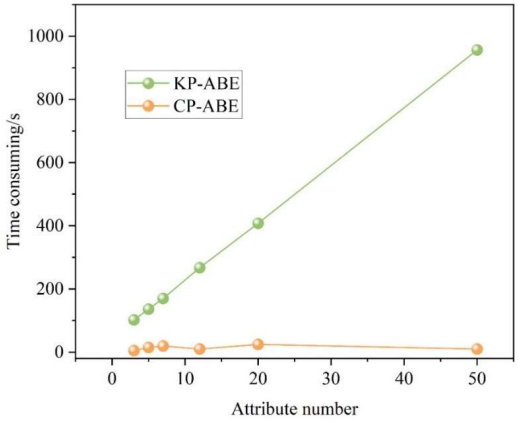

Figure 7: The two schemes are calculated at the do

The total computational overhead of the two schemes when the policy attributes are withdrawn is shown in Fig. 8. The experimental results show that the total computational overhead of the CP-ABE scheme is higher than that of the KP-ABE scheme when the number of attributes is relatively small, but as the number of attributes increases and the advantage of joint attributes is emphasized, the computational overhead of the KP-ABE scheme is higher than that of the CP-ABE scheme.

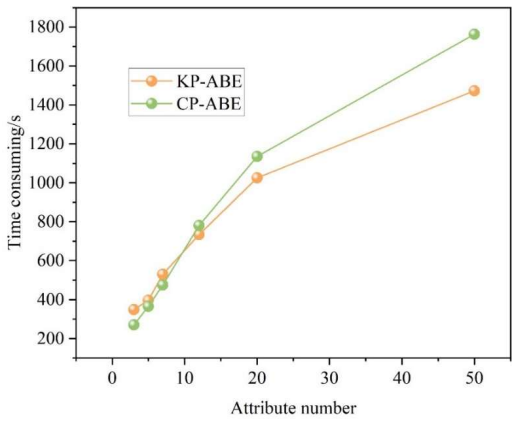

Figure 8: The total cost of the two schemes

In summary, for the same access structure tree, the computational overhead at the DO side of the CP-ABE scheme is less than that of the KP-ABE scheme, both when policy attributes are added and when policy attributes are withdrawn. As for the total computation overhead, as the number of attributes in the access structure tree increases, the number of AND thresholds included in the tree may also increase, and the number of joint attributes may also increase as a result. Therefore the growth rate of the number of re-encryptions for the CP-ABE scheme slowly decreases as the number of attributes in the tree increases. The number of re-encryptions for KP-ABE grows linearly with the number of attributes in the access structure tree being equal to the number of attributes in the tree.

Therefore, compared to the CP-ABE scheme, the CP-ABE scheme with the introduction of joint attributes not only reduces the computational overhead at the DO side, but also reduces the total computational overhead to a large extent when the number of attributes accessing the structure tree is large.

V. Conclusion

Through the theoretical analysis and experimental validation of attribute-based encryption (ABE) based data protection schemes for cloud computing, the following conclusions are drawn:

The CP-ABE scheme shows excellent performance in dealing with complex access control scenarios. The experimental data shows that when the number of attributes increases from 3 to 4, the average number of pseudo-tuples increases from 1.5 to 2.25, which proves that the scheme has good adaptability to the growth of the number of attributes. Especially in the policy attribute revocation session, the CP-ABE scheme delegates the re-encryption operation to the cloud service provider, which significantly reduces the computational pressure at the data owner's end, and avoids the resource consumption of the data decryption and re-encryption that the data owner needs to perform in the KP-ABE scheme.

In terms of overall computational overhead, when the number of tuples increases from 1k to 4k and then to 8k, the average number of pseudo-tuples is 7.6, 0.94, and close to 0, respectively, indicating that the scheme's efficiency is improved in handling large-scale datasets. The introduction of the optimized design of joint attributes enables the CP-ABE scheme to have a lower growth rate of computational overhead than the KP-ABE scheme when the number of attributes increases, especially when more attributes of the structure tree are accessed.

The security analysis confirms that the scheme ensures both forward and backward security of data, effectively preventing unauthorized users from accessing sensitive information. Forward security is achieved by immediately performing a re-encryption operation that invalidates the decryption key of the revoked user, while backward security ensures that unauthorized users cannot decrypt the historical data.

In summary, the combined encryption and access control scheme constructed in this study provides an efficient, secure and flexible solution for data protection in cloud computing environment.

References

[1] Thakur, K., Tao, L., Wang, T., & Ali, M. L. (2017). Cloud computing and its security issues. Application and Theory of Computer Technology, 2(1), 1-10.

[2] Ahmed, I. (2019). A brief review: security issues in cloud computing and their solutions. Telkomnika (Telecommunication Computing Electronics and Control), 17(6), 2812-2817.

[3] Inukollu, V. N., Arsi, S., & Ravuri, S. R. (2014). Security issues associated with big data in cloud computing. International Journal of Network Security & Its Applications, 6(3), 45.

[4] Gou, Z., Yamaguchi, S., & Gupta, B. B. (2017). Analysis of various security issues and challenges in cloud computing environment: a survey. Identity Theft: Breakthroughs in Research and Practice, 221-247.

[5] Rao, B. T. (2016). A study on data storage security issues in cloud computing. Procedia Computer Science, 92, 128-135.
[6] Liu, G. (2022). The application of data encryption technology in computer network communication security. Mobile information systems, 2022(1), 3632298.
[7] Van Daalen, O. L. (2023). The right to encryption: Privacy as preventing unlawful access. Computer Law & Security Review, 49, 105804.
[8] Geng, Y. (2019). Homomorphic encryption technology for cloud computing. Procedia Computer Science, 154, 73-83.
[9] Shabir, M. Y., Iqbal, A., Mahmood, Z., & Ghafoor, A. (2016). Analysis of classical encryption techniques in cloud computing. Tsinghua Science and Technology, 21(1), 102-113.
[10] Kumar, L., & Badal, N. (2019, April). A review on hybrid encryption in cloud computing. In 2019 4th international conference on internet of things: smart innovation and usages (IoT-SIU) (pp. 1-6). IEEE.
[11] Shukla, D. K., Dwivedi, V. K., & Trivedi, M. C. (2021). Encryption algorithm in cloud computing. Materials Today: Proceedings, 37, 1869-1875.
[12] Daniel, A., Shaba, S. M., Momoh, M. O., Chinedu, P. U., & Nwankwo, W. (2021). A computer security system for cloud computing based on encryption technique. Computer Engineering and Applications, 10(1), 41-53.
[13] El Sibai, R., Gemayel, N., Bou Abdo, J., & Demerjian, J. (2020). A survey on access control mechanisms for cloud computing. Transactions on Emerging Telecommunications Technologies, 31(2), e3720.
[14] Sultana, T., Almogren, A., Akbar, M., Zuair, M., Ullah, I., & Javaid, N. (2020). Data sharing system integrating access control mechanism using blockchain-based smart contracts for IoT devices. Applied Sciences, 10(2), 488.
[15] Ravinder Reddy, B., & Anil Kumar, A. (2019, December). Survey on access control mechanisms in cloud environments. In International Conference on Advances in Computational Intelligence and Informatics (pp. 141-149). Singapore: Springer Singapore.
[16] Ouaddah, A., Mousannif, H., Abou Elkalam, A., & Ait Ouahman, A. (2017). Access control in the Internet of Things: Big challenges and new opportunities. Computer networks, 112, 237-262.
[17] Karatas, G., & Akbulut, A. (2018). Survey on access control mechanisms in cloud computing. Journal of Cyber Security and Mobility, 1-36.
[18] Younis, Y. A., Kifayat, K., & Merabti, M. (2014). An access control model for cloud computing. Journal of Information Security and Applications, 19(1), 45-60.
[19] Sifou, F., Kartit, A., & Hammouch, A. (2017, September). Different access control mechanisms for data security in cloud computing. In Proceedings of the 2017 International Conference on Cloud and Big Data Computing (pp. 40-44).
[20] Mulimani, M., & Rachh, R. (2017). Analysis of access control methods in cloud computing. International Journal of Education and Management Engineering, 7(3), 15.
[21] Indu, I., Anand, P. R., & Bhaskar, V. (2018). Identity and access management in cloud environment: Mechanisms and challenges. Engineering science and technology, an international journal, 21(4), 574-588.
[22] Mahmood, G. S., Huang, D. J., & Jaleel, B. A. (2019). A secure cloud computing system by using encryption and access control model. Journal of Information Processing Systems, 15(3), 538-549.
[23] Almutairi, S., Alghanmi, N., & Monowar, M. M. (2021). Survey of centralized and decentralized access control models in cloud computing. International Journal of Advanced Computer Science and Applications, 12(2).
[24] Ma, H., Zhang, R., Sun, S., Song, Z., & Tan, G. (2019). Server-aided fine-grained access control mechanism with robust revocation in cloud computing. IEEE Transactions on Services Computing, 15(1), 164-173.
[25] Kanimozhi, S., Kannan, A., Suganya Devi, K., & Selvamani, K. (2019). Secure cloud-based e-learning system with access control and group key mechanism. Concurrency and computation: Practice and experience, 31(12), e4841.
[26] Abdul, A. M., Mohammad, A. A. K., Venkat Reddy, P., Nuthakki, P., Kancharla, R., Joshi, R., & Kannaiya Raja, N. (2022). Enhancing Security of Mobile Cloud Computing by Trust-and Role-Based Access Control. Scientific Programming, 2022(1), 9995023.
[27] Wang, J., Liu, J., & Zhang, H. (2017). Access Control Based Resource Allocation in Cloud Computing Environment. Int. J. Netw. Secur., 19(2), 236-243.
[28] Lei Zhang,Mingzeng Cao,Jing Li,Chenglin Zhang & Lili He. (2024). A novel collaborative privacy protection scheme based on verifiable secret sharing and trust mechanism. Computing,107(1),23-23.
[29] P. Nayudu & Krovi Sekhar. (2023). Secured Access Policy in Ciphertext-Policy Attribute-Based Encryption for Cloud Environment. Computer Systems Science and Engineering,46(1),1079-1092.
[30] Xiaodan Yan,Shanshan Tu,Hisham Alasmary & Fengming Huang. (2023). Multiauthority Ciphertext Policy-Attribute-Based Encryption (MA-CP-ABE) with Revocation and Computation Outsourcing for Resource-Constraint Devices. Applied Sciences,13(20).
[31] Hanlei Cheng,Sio Long Lo & Jing Lu. (2024). A blockchain-enabled decentralized access control scheme using multi-authority attribute-based encryption for edge-assisted Internet of Things. Internet of Things,26,101220-.