Hindawi
Security and Communication Networks
Volume 2021, Article ID 9606240, 13 pages
https://doi.org/10.1155/2021/9606240

Research Article

Speech Encryption Scheme Based on Ciphertext Policy Hierarchical Attribute in Cloud Storage

Qiuyu Zhang, Zhenyu Zhao, and Minrui Fu

School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China

Correspondence should be addressed to Zhenyu Zhao; zzhenyu@lut.edu.cn

Received 22 July 2021; Accepted 18 October 2021; Published 31 October 2021

Academic Editor: Luigi Coppolino

In order to ensure the confidentiality and secure sharing of speech data, and to solve the problems of slow deployment of attribute encryption systems and fine-grained access control in cloud storage, a speech encryption scheme based on ciphertext policy hierarchical attributes was proposed. First, perform hierarchical processing of the attributes of the speech data to reflect the hierarchical structure and integrate the hierarchical access structure into a single-access structure. Second, use the attribute fast encryption framework to construct the attribute encryption scheme of the speech data, and use the integrated access to the speech data; thus, the structure is encrypted and uploaded to the cloud for storage and sharing. Finally, use the hardness of decisional bilinear Diffie–Hellman (DBDH) assumption to prove that the proposed scheme is secure in the random oracle model. The theoretical security analysis and experimental results show that the proposed scheme can achieve efficient and fine-grained access control and is secure and extensible.

1. Introduction

With the rapid development of cloud computing and multimedia technology, cloud storage has become one of the most promising application platforms to solve the explosive growth of data sharing [1]. It can not only save costs but also facilitate the storage and sharing of multimedia data. However, the Elastic Compute Service (ECS) is a basic computing component composed of CPU, memory, operating system, and cloud hard drive, like local PCs and physical servers, and it is not a completely trusted third-party server. When the user outsources the data to the ECS, the user will lose the control of the data, especially for the sensitive speech and other multimedia data [2].

In recent years, in order to ensure user privacy and data security, data are usually encrypted and stored in the form of ciphertext to the cloud. Public-key cryptography provides a powerful mechanism to protect the confidentiality of data storage and information transmission. When the data owner wants to share certain information with the data user, must know exactly the data user wants information. In many real-world applications, the data owner wants to share certain information based on some credential policies for the data user. Attribute-based encryption (ABE) schemes provide a powerful method to achieve cloud data security and fine-grained access control. However, the existing ABE scheme cannot be fairly evaluated and compared in terms of security and performance. Therefore, in order to ensure data confidentiality and fine-grained access control, scholars proposed a ciphertext-policy attribute-based encryption (CP-ABE) [2] scheme suitable for cloud storage, in which the ciphertext is associated with the access structure defined by the data owner and the attribute private key is associated with the properties set of the relevant data user. This scheme has become the preferred encryption technology to solve the challenging problem of secure data sharing in cloud storage.

Speech is an important information carrier in audio. As the most direct and convenient multimedia application to convey information, speech contains important and sensitive confidential contents under certain circumstances, such as meetings, court evidence, military instructions, communication recordings, education system, and health care. These sensitive information contents involving national and corporate secrets and personal privacy require special attention when storing and sharing. The shared data generally have the characteristic of multilevel hierarchy, especially in the fields of health care, railway transportation, electric power, and military. However, the hierarchical structure of shared data and multiauthority access control are not fully utilized in the CP-ABE scheme. The hierarchical structure can realize fine-grained data access and multilevel hierarchy data file sharing and resist collusion attacks of multiauthority access control in the cloud storage system. In most existing schemes, attributes are considered at the same hierarchy, while in real-world applications, attributes are always at different hierarchies.

To overcome such drawbacks, the proposed scheme adopts speech data with different duration as the object of study, and the hypothetical military command scenario sets relevant attribute parameters. According to the multi-attribute characteristics of the instruction scenario, such as confidentiality level, participating units, and operational properties, it performs multilevel hierarchical processing and formulates a hierarchical access policy to process the speech data. A speech encryption scheme based on ciphertext policy hierarchical attribute is proposed. The contributions of this work are as follows:

(1) An attribute encryption scheme suitable for speech data is constructed. Using the faster and more secure type-III pairings, only a few pairings are needed for encryption and decryption, which effectively improves the rate of speech data decryption and does not limit the size of access policy and attribution. It is suitable for speech data encryption scenarios under complex attributes.

(2) The hierarchical model of access structure is used to solve the sharing problem of speech data with multilevel hierarchy attributes. Speech data are encrypted with an integrated access structure, which can provide fine-grained access control and improves encryption efficiency.

(3) Using the DBDH assumption to prove the security of the proposed scheme, which has higher encryption and decryption efficiency and lower complexity.

The rest of this paper is organized as follows: In Section 2, we have reviewed existing literature related to CP-ABE. Section 3 describes the preliminaries including the bilinear map, the access structures, the hierarchical access structures, and the DBDH assumption. Section 4 describes the system model, algorithm definition, and security model of the proposed scheme in detail. Section 5 gives the detailed performance analysis. Section 6 gives the experimental results and the performance analysis compared with other related methods, and Section 7 summarizes some conclusions of this paper.

2. Related Works

At present, the existing CP-ABE scheme is shown in Table 1.

Lian et al. [17] proposed a large universe CP-ABE with efficient attribute-level user revocation, which divides the master key into the delegation key and the secret key and sends to the cloud provider and user separately, thus realizing attribute revocation, reducing the computational load of the central authority, and effectively saving the storage space. Li et al. [18] proposed a lightweight data sharing scheme for mobile cloud computing, which changes the structure of access control tree to make it suitable for mobile cloud environment; it introduces attribute description fields to implement lazy-revocation and reduces the user revocation cost. Bayat et al. [19] proposed an efficient no-pairing and revocable ABE data sharing scheme based on elliptic curve cryptography, which solves the complex problem of bilinear pairing operation. Namasudra [20] proposed an efficient and secure access control model for resource and knowledge sharing in the cloud computing environment based on distributed hash table, in order to improve the performance and security of sharing. Vaanchig et al. [21] proposed a key-escrow-free multiauthority ciphertext-policy attribute-based encryption scheme with dual-revocation; it realizes the data access control scheme of the collaborative cloud storage system. Yu et al. [4] proposed an attribute-revoking mechanism without updating the key and a hybrid cloud storage model, which solves the problem of public cloud trust management. Arthur Sandor et al. [14] proposed a decentralized multiauthority attribute-based scheme for mobile cloud data storage, which does not require a trusted central to publish system parameters and generate the user secret key, thus improving data confidentiality and reducing the risk of privacy leakage.

The hierarchical access structure helps build an access structure for the fine-grained and multiple permissions of cloud storage. The access structure of all subfiles is integrated into a single-access structure, and the hierarchical files are encrypted with the integrated access structure, and the ciphertext related to attributes can be shared. Li et al. [22] proposed an efficient extended file hierarchy CP-ABE scheme, which solves the flexible access control of users in cloud storage and saves storage space and computation cost. Wang et al. [13] proposed a hierarchical encryption scheme based on an identity-based encryption system and ciphertext policy attributes to solve the problem of fine-grained access control and proposed an extensible revocation scheme to effectively revoke user access rights. Wang et al. [12] proposed an efficient cloud computing file hierarchical attribute encryption scheme, saving ciphertext storage space and encryption time cost. Yang et al. [23] used mandatory access control method, attribute-based encryption, and combined with the characters of classified and graded data, and proposed a secure label-based access control model in object storage to achieve fine-grained access control to a large number of resources with classification and grade in cloud storage.

In recent years, the CP-ABE scheme has not been utilized efficiently in real life. Sowjanya and Dasgupta [24] used the CP-ABE scheme to provide a security framework for the wireless body area networks, and it also has a user/attribute revocation mechanism. Liang et al. [25] proposed a privacy protection distributed attribute encryption scheme based on the Lewko and Waters scheme [26] by introducing global

Table 1: Different CP-ABE category descriptions.

CP-ABE category	Description
Revocable CP-ABE constructions [3, 4]	It can achieve fine-grained revocable access control. According to the fine-grained difference, the revocation mechanism is divided into user revocation and attribute revocation; according to nonrevoked users, it is divided into indirect revocation and direct revocation.
Traceable CP-ABE constructions [5]	It can realize fine-grained and traceable access control mechanism, responsible for user traceability and attribute authority responsibility, which can be divided into white-box traceability and black-box traceability.
Policy-hiding CP-ABE constructions [6]	Hide related attributes and have a fine-grained access control mechanism for attribute privacy protection. It can be divided into fully hidden and partially hidden. Currently, there is no specific and completely hidden CP-ABE scheme.
Policy-updating CP-ABE constructions [7]	Involving proxy re-encryption, it is impossible to change the secret access policy. In an emergency, the policy update policy can be adopted. Used to implement fine-grained access control for policy updates.
Multiauthority CP-ABE constructions [8]	It can realize fine-grained distributed access authority and can be divided into centralized CP-ABE and distributed CP-ABE according to whether there is a central organization.
Hierarchical CP-ABE constructions [9–13]	It can achieve hierarchical fine-grained access control. The delegation of access privilege is organized in a hierarchical manner.
Online/offline CP-ABE constructions [14]	In order to reduce the amount of calculation of the data owner and attribute authority, offline encryption or offline key generation can be used.
Outsourced CP-ABE constructions [15, 16]	In order to support resource constrained users (data users, data owners, and central authority), laborious calculations are outsourced to third-party servers in the encryption and decryption process.

identities (GIDs) to security share personal health record and communicate health status with hospitals or doctors. Meng et al. [27] aiming at the problem of illegal access to private and sensitive information in smart cities proposed a new keyword search CP-ABE scheme to encrypt or decrypt Internet of things (IoT) data in cloud storage. Ali et al. [9] proposed an efficient multiauthority access control scheme for the employee attribute scenarios of large companies, which realized privacy protection, multiauthority access control, and fine-grained access control to stored data. De Oliveira et al. [28] aimed at the problem that the healthcare professional could not obtain the patient's complete electronic medical records (EMRs) in the medical emergency situation and proposed a protocol based on CP-ABE scheme, through which all the treatment teams participating in the emergency rescue can security decrypt the relevant data of the patient's EMRs. Pournaghi et al. [29] provided a scheme for recording and storing medical data based on blockchain technology and attribute encryption, which realized fine-grained access control of medical patient data and secure storage on blockchain. In order to solve the problem that searchable encryption technology does not consider the fine-grained search authority of data users, Niu et al. [30] used attribute-based encryption technology to achieve fine-grained access control of data and used the tamperproof feature of the blockchain to ensure the keyword ciphertext security.

In summary, most of the existing CP-ABE schemes encrypt text or random numbers, and as the number of attributes increases and the complexity of the access policy increases, the encryption and decryption time will also increase, and the deployment cost will become very expensive. At present, the CP-ABE scheme solution has been mainly used in the medical system and the IoT, and there are few application scenarios for encrypting speech data. Therefore, we present a ciphertext-policy attribute-based speech encryption scheme under different attribute hierarchies. The proposed scheme makes corresponding attributes and policies according to the assumed multilevel characteristics of military command speech scenes and adopts MNT224 curves to realize the pairings operation of asymmetric bilinear maps [31], improves the efficiency of decryption data, and adjusts the hierarchical access policy scheme for speech scenarios. The scheme does not need to change the public parameters or encryption algorithm and considers the access policy optimized for personalized users.

3. Preliminaries

3.1. Bilinear Maps. Let G_0 and G_T are two multiplicative cyclic groups with a prime order, where g is a generator in G_0, and e: $G_0 \times G_0 \longrightarrow G_T$ is the bilinear map that satisfies the following properties:

(1) Bilinear: for any u, $v \in G_0$ and a, $b \in Z_p$, $e(u^1, v^b) = e(u, v)^{ab}$, where $Z_p = \{0, 1, 2, \ldots, p\}$

(2) Nondegeneracy: $e(u, v) \neq 1$

(3) Computability: for any $\forall u$, $v \in G_o$, there is a polynomial time algorithm for calculating $e(u, v)$

Let GroupGen is an asymmetric paired group generator, input parameter 1^λ, and generate three groups of multiplicative cyclic groups G_0, G_1, and G_T with a prime order p. If there is no valid homomorphic calculation between two multiplicative groups, the pairing is asymmetric or Type-III. Type-I pairing has serious security problems [32], Type-III pairing can be easily converted to Type-I (by taking $G_0 = G_1$), and using the Type-III pairing cryptogram protocol (such as ABE), the main reason is to improve encryption and decryption performance and security [33]. Type-III pairing structure has been deployed in several practical applications, such as the zk-SNARK algorithm is used to protect the privacy of blockchain transactions [34].

3.2. Access Structure. The access structure is a logical structure that describes the access control policy; it specifies a set of attributes required to access a certain ciphertext speech and defines the authorized sets and the unauthorized sets. The access structure in ABE defines an internal relationship between user access rights and access control policies, which are described as follows.

Let $P = \{P_1, P_2, \ldots, P_n\}$ represent the set of participants, and let $A = \{A | A \subseteq \{P_1, P_2, \ldots, P_n\}\}$. The set $A \subseteq 2^P$ is monotonic, if and only if for any subset $B, C \subseteq P$, if $B \in A$, and $B \subseteq C$, then $C \in A$. If A is a nonempty subset (monotonic) in $P = \{P_1, P_2, \ldots, P_n\}$, $A \subseteq 2^{\{P1, P2, \ldots, Pn\}} \backslash \{\emptyset\}$. The subset in the set A is called the authorized set, and the set that is not in the set A is called the unauthorized set.

In the proposed CP-ABE scheme, attributes are participants. The attribute set that can satisfy the associated ciphertext speech access structure is the authorization set defined above, the users attribute set that can decrypt the ciphertext speech legally and correctly. Monotonicity means that after an authorized user obtains more attributes, he cannot lose his own privileged attributes. Unless otherwise specified in this paper, the access structure is monotonous.

3.3. Linear Secret Sharing Scheme (LSSS). Let p be a prime order, U be the attribute universe, and $P = \{P_1, P_2, \ldots, P_n\}$ represents the set of participants. The access structure of a secret sharing scheme II on Z_p is linear on U, if and only if II is composed of the following two conditions:

(1) Each attribute has a secret random number $s \in Z_p$ to be shared and generates a vector on Z_p

(2) For each access structure S on U, there is a shared generating matrix $\mathbf{M} \in Z_{\mathrm{p}}^{l \times n}$, let $\mathbf{M}$ be a matrix of size $l \times n$; ρ:$\{\forall i \in [1, l]$:$\mathbf{M}_i\} \longrightarrow \{\forall i \in [1, l]$:$P_{\mathrm{i}}\}$ is a mapping that maps each participant to a certain row vector in matrix $\mathbf{M}$, that is $\rho(i) = P_i$, where $\mathbf{M}_i$ is the i-th row in $\mathbf{M}$ and satisfies the following condition:

In the process of generating the shared matrix, firstly, generate a random column vector $v = (s, y_2, y_3, \ldots, y_n)^T$, where $y_2, y_3, \ldots, y_{\mathrm{n}} \in Z_p$; then, calculate $\mathbf{M} \cdot v$ to generate a l-dimensional row vector, each element of the vector $\lambda_{\rho(i)} = \mathbf{M}_{(i)} \bullet v$ will be kept by a participant $\rho(i)$. $(\mathbf{M}, \rho)$ is the policy of access structure S.

The matrix $\mathbf{M} \in Z_p$ is $n_1 \times n_2$ in the proposed scheme, and the mapping π:$\{1, 2, \ldots, n_1\} \in U$, Lewko and Waters [35] proposed a simple and effective way to convert any monotonic Boolean formula F into $(\mathbf{M}, \pi)$, so that each row in $\mathbf{M}$ corresponds to the input in F, and the number of columns in the matrix is the same as the number of AND gates in F, and each element in $\mathbf{M}$ is 0, 1, or −1.

3.4. Hierarchy Access Tree. Let Γ be a hierarchical access tree structure [10] divided into k access levels. The node of the access tree is represented as (p, q). p represents the number of rows of the node (from top to bottom), and q represents the number of columns of the node (from left to right). As shown in Figure 1, each node can be expressed as $A = (1, 1)$, $B = (2, 1)$, $C = (2, 2)$, $D = (3, 1)$, $E = (3, 3)$, $F = (4, 1)$, $G = (4, 2)$, and for the convenience of describing the access tree Γ, the following definitions are made:

(1) (p, q) represents a node of the access tree Γ. If (p, q) is a leaf node, it is represented as an attribute. If (p, q) is a nonleaf node, it is represented a threshold gate: "AND", "OR", etc. In the figure, node C is represented as an attribute, and node E is represented as an AND gate.

(2) $(p_{\mathrm{i}}, q_{\mathrm{i}})$ $(i \in [1, k])$ represent the level nodes of the access tree Γ; Γ is divided into k access levels, and the levels of the nodes are arranged in descending order. (p_1, q_1) is the highest level, and (p_k, q_k) is the lowest level. As shown in the figure, (p_2, q_2) represents the second level.

(3) $\mathrm{num}_{(p, q)}$ represents the number of child nodes of the node Γ in access tree, as shown in the figure, $\mathrm{num}_B = 2$.

(4) $k_{(p, q)}$ represents the threshold of access tree Γ node, and $1 \leq k_{(p, q)} \leq \mathrm{num}_{(p, q)}$. As shown in the figure, $k_{\mathrm{E}} = 2$ means "AND" gate.

(5) parent(p, q) represents the parent node of the access tree Γ. As shown in the figure, parent$(3, 1) =$ parent$(B) = A$.

(6) Transport node represents if a child node contains at least one threshold gate, the child node is a transport node. As shown in the figure, A, B, and E are transport nodes.

(7) TN−CT(p, q) represents the threshold set of the child nodes of the transport node (p, q) in the access tree Γ. As shown in the figure, TN−CT$(A) = \{B\}$, TN−CT$(B) = \{E\}$.

3.5. Decision Bilinear Diffie–Hellman (DBDH) Assumption. The DBDH assumption is defined in the form of a game. A challenger B selects a set of groups G_0 with a prime order p according to the security parameters of the system. Let e: $G_0 \times G_0 \longrightarrow G_{\mathrm{T}}$ be an effective bilinear mapping, select generator $g \in G_o$ and random parameters $a, b, c, \in Z_{\mathrm{p}}$. The DBDH of the bilinear group G_0 and G_{T} is assumed as follows: given the input g, g_a, g^b, $g^c \in G_o$, an adversary A needs to distinguish the tuple $e(g, .g)^{abc}$ from the random element $R \in G_T$, which can distinguish between $e(g, .g)^{abc}$ and R. The advantage of defining algorithm B to solve the DBDH problem [22] is defined as

$$\mathrm{Adv}_A^{\mathrm{DBDH}} = \left| \Pr\left[B\left(g, g^a, g^b, g^c, e(g, g)^{abc}\right) = 0 - \Pr\left[B\left(g, g^a, g^b, g^c, R\right) = 0 \right] \right] \right|. \tag{1}$$

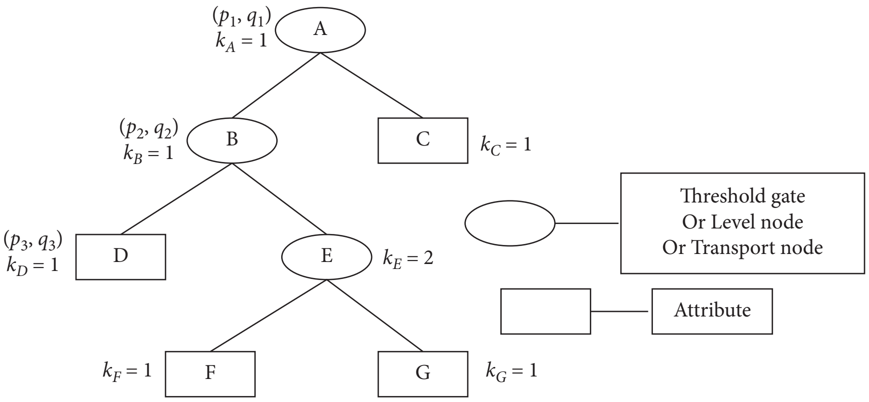

Figure 1: General hierarchical access structure.

If there is no probabilistic polynomial time (PPT) algorithm B to solve the DBDH problem under the bilinear group G_0 and G_T with a non-negligible advantage, then the DBDH assumption is established.

4. The Proposed Scheme

4.1. System Model. Figure 2 shows the system model of the ciphertext policy hierarchical attribute encryption scheme. In this scenario, the data owner is the military command center. Data users are different according to the actual situation. For example, command posts at all levels of a certain army, combat personnel at all levels, intelligence departments of various units, etc., have different access rights, such as the specific content of the access instruction and the basic elements of the instruction, and there are different attributes, such as a certain division-level cadre, a certain regiment-level cadre, a certain intelligence personnel, and a certain fire control system operator.

The system model of the proposed scheme consists of four different entities: central authority (CA), data owner (DO), data user (DU), and cloud service provider (CSP) as shown in Figure 2. Figure 3 shows an example of a hierarchical access structure.

(1) Central authority (CA): It is a completely honest and trusted entity that performs user registration of cloud storage and generates a private key for each data user through interaction with the user. This entity mainly executes two algorithms: *Setup* and *Keygen*.

(2) Cloud service provider (CSP): It is a semitrusted entity related to modern military systems, which can honestly perform assigned tasks and return right results. CSP does not participate in the implementation of access control or the encryption and decryption process, and only authorized users can obtain data. However, it hopes to discover sensitive contents as much as possible. In the proposed scheme, ciphertext speech storage and transmission services are provided.

(3) Data owner (DO): There is a large amount of speech data that needs to be stored and shared in the cloud. In the encryption system proposed by this scheme, DO has m speech data and k access levels, $msg = \{m_1, m_2, \ldots, m_k\}$, where m_1 is the highest level in the access structure and m_k is the lowest level.

(4) Data user (DU): As a cloud user, DU obtains a private key whose attributes conform to the access structure. Data users want to access large amounts of data in the cloud. First, download the corresponding ciphertext speech, and then perform the decryption operation of the proposed scheme. If the user can decrypt m_1, then the user can also decrypt m_2, m_3, $\ldots$, m_k.

The proposed scheme assumes application scenarios for speech encryption in cloud storage, such as conferences, court recordings, or military commands. When the military command center uploads instructions to the cloud service provider, it performs related operations as the data owner. The military command center divides the command *msg* into two elements, m_1 and m_2, where m_1 may include basic command elements such as command level, command purpose, and command information. In addition to these basic command elements, m_2 also contains the specific content, execution steps, and troops participating in the war and equipment used. In the framework constructed in Figure 3, the central authority confirms the data user's request for access and generates some parameters. Shared speech data include hierarchical access policies, and a speech data or file is divided into sublevels that are located at different access levels. If the speech data or files of the same hierarchical structure can be encrypted with the integrated access structure, the storage consumption after encryption and the time consumption of encryption can be saved.

According to the actual application scenario, the data owner adopts a ciphertext policy attribute encryption scheme with a hierarchical access structure and uses different access policies to encrypt the speech data m_1 and m_2.

As shown in Figure 3, a certain divisional cadre needs to access basic instruction elements such as instruction level, instruction purpose, and instruction information in order to quickly respond and execute the instructions issued. The detailed information that a regiment-level cadre needs to access includes specific instruction elements such as specific

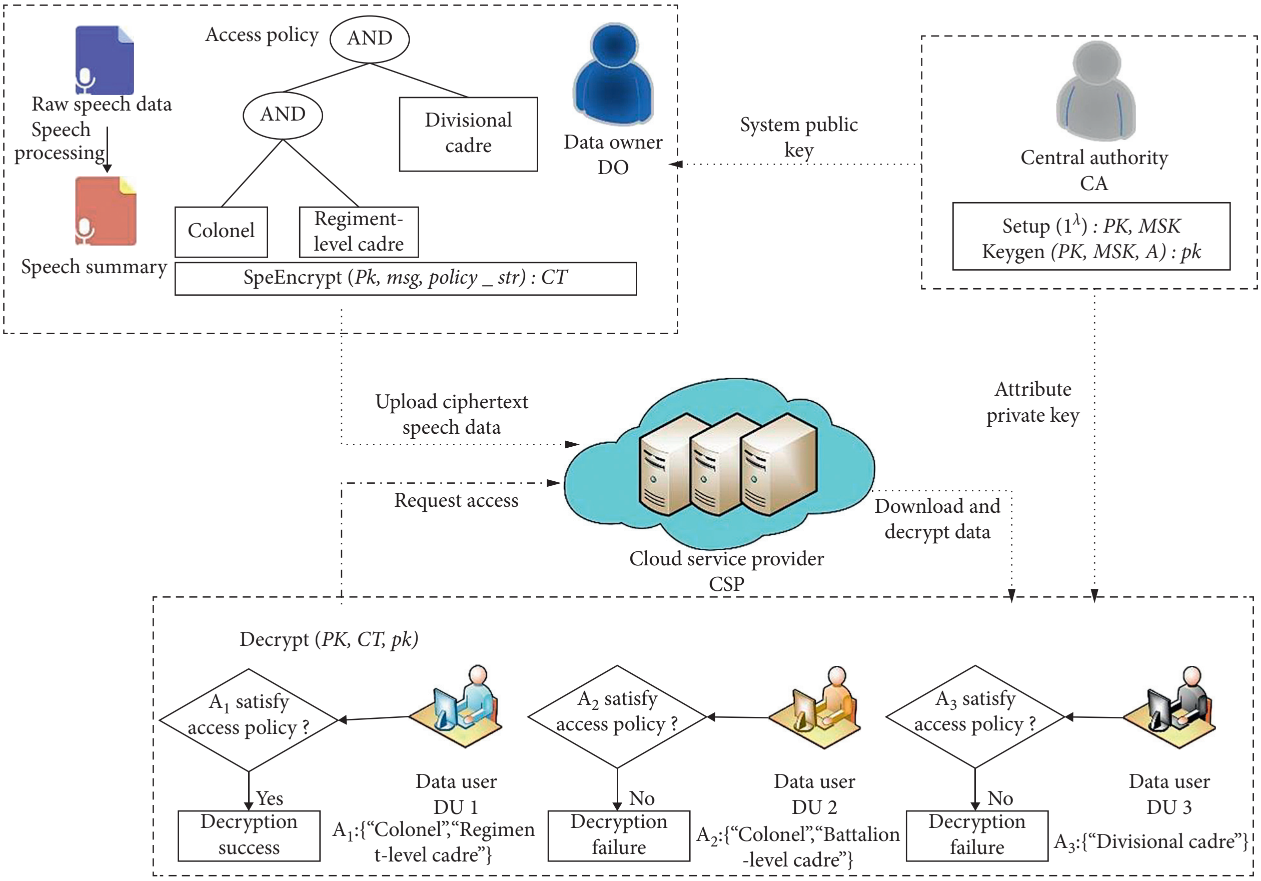

FIGURE 2: System model of ciphertext policy hierarchical attribute encryption scheme.

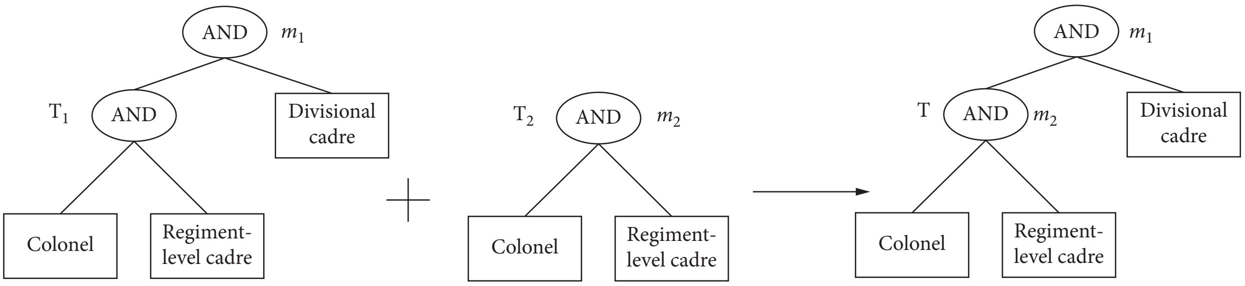

FIGURE 3: Example of hierarchical access structure.

content, execution steps, and combat personnel equipment. Suppose that the command center sets the access structure of m_1 to T_1{("Colonel" AND "Regiment-level cadre") AND "Divisional cadre"} and the access structure of m_2 is set to T_2{"Colonel" AND " Divisional cadre "}. Encrypted speech data m_1 and m_2 need to be encrypted twice with access structures T_1 and T_2, respectively, to generate ciphertext data CT_1 and CT_2. In these two access structures, T_2 is a subset of T_1 and T_1 is an extension of the T_2 access structure. Therefore, there is a hierarchical relationship, and the ciphertext CT can be generated through the integrated access structure T to encrypt the speech data m_1 and m_2, thus solving multilevel speech data or file sharing issues. Encryption complexity and encryption overhead will be significantly reduced, and data users can decrypt all authorized speech data or files by generating keys through the transport node in Figure 1.

4.2. Concrete Construction. The proposed scheme is based on the ciphertext policy hierarchical attribute encryption scheme. In order to reduce the computational complexity and encryption and decryption time, a Type-III pairing group is adopted. Let G_1 and G_2 be an asymmetric pairing group with a prime order p, and e: $G_1 \times G_2 \longrightarrow G_T$ is an asymmetric bilinear map. λ is a security parameter that determines the size of the set. Use the hash function G_2, which is a random oracle model and map any binary string to the elements G_1.

(1) *Setup*(1^{λ})⟶(MSK, PK): The setting algorithm gives a security parameter λ as input and does not accept any input other than the implicit security parameter and outputs a public key PK and a master key MSK

The algorithm executes GroupGen(1^{λ}) to input the security parameter λ and generates a pair of asymmetric paired groups G_1 and G_2 with a prime order p, where g is the generator of G_1, and h is the generator of G_2. Randomly choose $\alpha, \beta \in Z_p$, $\gamma \in Z_p{}^{*}$, and output the public key PK as shown in equation (2):

$$PK = \left(h^{\alpha}, e(g,h)^{\alpha\gamma+\gamma}\right). \tag{2}$$

The system master secret key MSK is shown in equation (3):

$$\mathrm{MSK} = \left(g, h, \alpha, \beta, g^{\gamma}\right). \tag{3}$$

(2) *KeyGen*(PK, MSK, A)⟶(pk): The key generation algorithm takes the public key PK, the master secret key MSK, and a set of attributes describing the key as input and outputs a private key pk.

The algorithm randomly chooses δ, σ_A, $\sigma' \in Z_p$, $A \in A$ is a set of attributes of attribute A, where k_0, k, k_p are shown in equations (4)–(6):

$$k_0 = h^{\beta\delta}, \tag{4}$$

$$k = G_2(\alpha t)^{\beta\delta/\alpha t} \cdot g^{\sigma_A/\alpha t}, \tag{5}$$

$$k_p = g^{\gamma} \cdot G_2(01t)^{\beta\delta/\alpha t} \cdot g^{\sigma_{\prime}/\alpha t}, \tag{6}$$

where $t = 1{,}2$. g, h, α, β, g^{v} and other elements come from the master secret key MSK. Output user private key pk = (k_0, k, k_p).

(3) *SpeEncrypt*(PK, msg, S)⟶(CT): the speech encryption algorithm takes the public key PK, the speech message digest *msg* after the original speech processing, and the integrated access structure S as input. The algorithm encrypts *msg* and generates ciphertext CT.

The algorithm opens and reads the format and data of the wav file, returns the information of the wav file format at one time, and obtains a tuple including the number of channels, the number of quantization bits, the sampling frequency, and the number of sampling points. Read waveform data, sound data, and transfer the length of the data that needs to be read. The speech waveform data is converted into the number of channels and the number of quantization bits, and the read binary data is converted into a computable data msg. Randomly choose $\eta \in Z_p$, M is a matrix with n_1 rows and n_2 columns; $i = 0, 1, \ldots, n_1$; $l = 1, 2, 3$. Output ciphertext CT = (ct_0, …, ct_{n1}, cp), where ct_{il}, c_p are shown in equations (7) and (8).

$$ct_{il} = G_2(\pi(i)l)^{\eta} \cdot \prod_{j=1}^{n_2}\left[G_2(jl)^{\eta}\right]^{(M)_{i,j}}, \tag{7}$$

$$cp = \mathrm{msg} \cdot e(g,h)^{(\alpha\gamma+\gamma)\eta}. \tag{8}$$

(4) *Decrypt*(PK, CT, pk)⟶(msg or ⊥): the decryption algorithm takes the public key PK, the ciphertext CT containing the integrated access policy S, and the private key pk is the private key of any set of attributes in the attribute set $A = \{A_1, A_2, \ldots, A_{\mathrm{i}}\}$, the decryption algorithm is

$$\mathrm{Decrypt} = \frac{cp \cdot e\left(ct_{il}, h^{\beta\delta}\right)}{e\left(k_p \cdot k(A_i), h^{\alpha\eta}\right)}. \tag{9}$$

If the attribute A_{i} satisfies the corresponding policy in the access structure S, the corresponding ciphertext data can be decrypted, then the algorithm will decrypt the ciphertext data and return the message msg, otherwise the output decryption fails ⊥.

4.3. System Security Model. The proposed scheme assumes that the potential attacker of the cloud storage system is that each DU is considered a dishonest malicious user and may try to obtain data access permissions beyond the access permissions. It is assumed that the adversary A in this system means that the unauthorized user does not have enough attributes to satisfy the encrypted data access policy and will not decrypt the encrypted data. The security model of the scheme is based on the security model of the classic CP-ABE scheme [22, 26]. Assuming that the access structure has only one level node, the CPA security game between adversary A and challenger B is defined as follows:

Initialization: A selects an access structure S* that he wants to challenge and gives it to the challenger B.

Setup: B executes the algorithm of the proposed scheme, outputs the PK, and gives it to A.

Phase1: A does multiple private key queries on the attribute set $A = \{A_1, A_2, \ldots, A_{\mathrm{i}}\}$, none of the attribute set A_{i} satisfies S*, and runs the ***KeyGen*** algorithm to execute these queries.

Challenge: A selects two pieces of data m_1 and m_2 of equal size and needs to accept query operations. Randomly selects m_k, where $k \in \{1{,}2\}$, and encrypts it under the access structure S*, and returns the generated ciphertext CT* to A.

Phase2: A repeatedly makes the queries as the same as the phase 1.

Guess: A outputs a guess k^{*} of k. A wins this game if $k = k^{*}$. The advantage for A in the above game $\varepsilon = Adv^{CPA}{}_{A}(1^{\lambda})$ is

$$A\ dv_{A}^{CPA}\left(1^{\lambda}\right) = \left|\Pr\left[k = k^{*}\right] - \frac{1}{2}\right|. \tag{10}$$

Definition 1. The proposed scheme is secure if no PPT adversary is able to win the above mentioned security game with a non-negligible advantage ε.

5. Scheme Performance Analysis

5.1. Theoretical Analysis. The encryption scheme of the proposed scheme is provided for the entity data owner and the data user, and assuming the data owner is the scheme set above, m hierarchical speech data with k access levels are shared in the cloud storage.

Data owner (DO) computing cost: the proposed scheme provides a hierarchical model of access structure and achieves multilevel speech data sharing, and the speech data is encrypted using an integrated access structure. Therefore, the data owner only needs to run the encryption algorithm once to encrypt different levels of speech data to generate ciphertext data. The public key of the system only needs to be calculated once, and the generation of the private key only needs to be calculated once, thus improving the encryption efficiency of the data owner.

Data user (DU) computing cost: in the decryption process, since the transmission is added to the access structure with k level nodes, the data user can decrypt the authorized data according to its own attributes. In addition, the traditional Boolean formula is replaced with LSSS, which only requires a small amount of pairing calculations to pair the data during encryption and decryption, thus improving the time efficiency of data users.

5.2. Security Analysis

Theorem 1. *If the adversary has a non-negligible advantage in a defined secure game under the random oracle model, then at least one probabilistic polynomial time simulator* C *can solve the DBDH problem with a non-negligible advantage. That is, assuming that there is a polynomial time adversary A at the non-negligible advantage $\varepsilon = Adv^{A}{}_{CPA}(1^{\lambda})$, which breaks the CPA security of this scheme, the advantage ε/2 can be constructed to solve the DBDH problem, where ε is the advantage to solve the DBDH assumption problem.*

Proof: Given the defined asymmetric bilinear mapping e security parameter P^{λ} = (G1, G2, GT, g, h, p), the challenger B chooses a', b', c', $z \in Z_{p}{}^{*}$, and a random bit value $v \in \{0, 1\}$. If $v = 0$, B creates Z = e(g, h)$a'b'c'$; otherwise, Z = e(g, h)z, assuming that the simulator C gives the B tuple (g, h, ga', gb', gc', $h^{b\prime}$, $h^{c'}$, Z), then Cwill play the role of B in the subsequent security games.

Initialization: The simulator C runs the adversary A; A describes the access structure S* that wants to challenge and gives it to C.

Setup: C computes the public key PK = e(g, h) $a'c'$ + c' and sends it to A. Furthermore, choose a challenging access structure S* and send it to A.

Phase1: The adversary A queries the private key of the attribute set $A = \{A_1, A_2, \ldots, A_i\}$, and there is no A_i that satisfies the access structure S*. For any attribute $j \in A_{\mathrm{i}}$, A randomly selects $a_j{}' \in Z_p$ and computes the private key as shown in equation (11):

$$\begin{gathered} D = h^{\beta\delta},\ D' = G_2\left(a_j't\right)^{\beta\delta/\alpha t} \cdot g^{\sigma_a/a_t'}, \\ D'' = g^{\gamma} \cdot G_2\left(01t\right)^{\beta\delta/\alpha t} \cdot g^{\sigma/\alpha t}. \end{gathered} \tag{11}$$

Send the PK=(D, D', D'', $\forall j \in$ A) to the adversary A.

Challenge: A selects two pieces of data m_1 and m_2 of equal size and send them to C; C randomly selects a piece of data m_k, where $k \in \{1, 2\}$, and encrypts it under the access structure S*. C computes the ciphertext data CT* and sends it to A.

Phase2: the adversary A repeatedly makes the queries as the same as the query phase1 operation.

Guess: A outputs a guess of k^* of k, If $k = k^*$, the simulator C outputs 0, which means $Z = e(g,h)^{a'b'c'}$. Else, it outputs 1, which means $Z = e(g,h)^{z}$. If Z=e(g, h)^z, the simulator C generates an effective ciphertext CT* under the advantage $1/2 + \varepsilon$ in the above-mentioned way, where ε is the advantage of the adversary A guessing a right bit:

$$\begin{aligned} &\Pr\left[C\left(g, h, g^{a'}, g^{b\prime}, g^{c'}, h^{b\prime}, h^{c'}, Z = e(g,h)^{a'b\prime c'}\right) = 0\right] \\ &\quad = \frac{1}{2} + \varepsilon. \end{aligned} \tag{12}$$

If Z = e(g, h)^z, the data m_k is completely hidden from the adversary A, so the inequality $k \neq k^*$ holds with an advantage 1/2:

$$\Pr\left[C\left(g, h, g^{a'}, g^{b\prime}, g^{c'}, h^{b\prime}, h^{c'}, Z = e(g,h)^{Z}\right) = 0\right] = \frac{1}{2}. \tag{13}$$

It can be concluded that the compute of the advantage in the above CPA secure game is defined as

$$\mathrm{Adv}_{C}^{\mathrm{CPA}} = \frac{1}{2} \times \left(\frac{1}{2} + \varepsilon\right) + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} = \frac{\varepsilon}{2}. \tag{14}$$

□

5.3. Fine-Grained Access Control and Flexible Data Sharing. The proposed scheme is based on the CP-ABE scheme, which can realize DO's precise control of the speech data. DO builds an integrated access structure to encrypt the speech data according to the access policy of each speech data that wants to be encrypted and shared. The access policy describes the individual attributes of DU. For example, "Position: Battalion Cadre" AND ("Position: Company Cadre "AND" Rank: Captain") allows users with a battalion cadre or a position or military rank of major or lower to successfully access data, achieving fine-grained access control and flexible sharing.

5.4. User Revocable. The problem of denying access requests from users who have been revoked in the proposed scheme can be realized by embedding time stamp mechanism in the private key of the DU, which can ensure that the DU updates its attribute parameters to access the encrypted speech data again. This ensures that DU does certain lazy revocation of access control. More complex and complete revoking mechanisms, such as using proxy re-encryption mechanism to recalculate ciphertext, or using attribute authority (AA) to constantly update public parameters and issue key credentials to users who have not been revoked, are beyond the scope of the proposed scheme. Reference [17] further discusses and studies the mechanism of attribute revocation and user revocation.

6. Performance Comparison and Experimental Simulation

6.1. Performance Comparison of Different Schemes. In order to reflect the advantages of the proposed scheme, the evaluation indexes between the proposed scheme and ABE schemes in References [16, 26, 30] are compared from the aspects of function and storage cost, such as access structure, speech encryption, speech application scenarios, ciphertext, and key size. The comparison results are shown in Table 2. Table 3 defines the symbols used in the evaluation of indicators in Table 2.

It can be seen from Table 2 that Reference [26] is a classic CP-ABE scheme, which realizes the basic functions and uses access tree to construct access policy, the implementation uses a 160-bit elliptic curve group based on the super-singular (SS) curve $y = x^3 + x$ over a 512-bit finite field, and the PBC library can compute pairings in approximately 5.5 ms. Reference [16] converts the CP-ABE scheme into an asymmetric bilinear mapping, using the symmetric elliptic curve (SS512) of the Charm library. Use this scheme to encrypt speech data and compare it with the experimental results of this paper. Reference [30] uses a cloud-assisted attribute-based searchable encryption scheme on blockchain, which uses C programming language and uses 512-bit elliptic curve domain to construct Type-I bilinear pairings. The proposed scheme realizes the encryption of multiple speech data through hierarchical access tree structure and linear secret sharing scheme, and it does not limit the number of user attributes and is suitable for complex speech scenarios. In order to improve efficiency, the proposed scheme uses prime order groups. In order to improve security, uses asymmetric encryption mechanism.

6.2. Experiment Analysis. In the experiment, the Charm-Crypto [36] is an ABE framework under Python, which integrates libraries such as OpenSSL, PBC, GMP, and other related architectures in the field of network security and implements the proposed scheme based on the cpabe toolkit [26]. On a laptop with Windows 10 operating system, the hardware environment is Intel Core i5-4210H 2.9 GHz, and the running memory is 16 GB using a VMware Workstation 15 Pro virtual machine to build the Ubuntu 20.04 Linux operating system with 4 GB of memory. Linux Python 3.8 and Charm-Crypto 0.50 software version are used, and the speech data from THCHS-30 [37], a Chinese speech database released by the Center for Speech and Language Technology (CSLT) of Tsinghua University, are used to conduct experiments.

The proposed scheme converts the CP-ABE into asymmetric bilinear map. The Charm library uses only asymmetric metric groups and uses the Type-III MNT224 curve supported by the Charm library with order security 96 bit. All the following runtimes are the result of running 10 times and averaging under the premise that MNT224 security 96 bit. The evaluation performance indicators required by the scheme are shown in Table 4, and the user-defined parameters are shown in Table 5.

The threshold gates in the attribute policy are all connected by "AND" gate. Encryption of data requires access policy attribute X, decryption requires policy attribute Y, $X = Y = 2, 4, \ldots, 20$. Convert the access policy to Boolean formulas, and then use the method of Water et al. [38] to convert Boolean formulas to LSSS. The advantage is that the generated matrix has only 0, 1, and −1 options, and the reconstructed coefficient is only 0 or 1. Figure 4 shows the results of the comparison between different parameters of the proposed scheme and the running time of each algorithm.

Figures 4(a) and 4(b) shows the comparisons of the number of attributes of data users, the number of attributes in the access structure, and the running time of each algorithm. Keep the same access policy, it can be seen from Figure 4(a) that the number of user attributes increases, the private key generation time will slowly increase, and the encryption and decryption time will not increase as a result. Keep the number of attributes in the private key the same; it can be seen from Figure 4(b) that the number of attributes in the access policy increases, the encryption time will slowly increase. Since the increase of the attribute will increase the pairing operation of the bilinear mapping, it will affect the running time. Figures 4(b) and 4(d) shows the comparisons of the number of access structure hierarchies and the size of encrypted speech data and time. It can be concluded that an increase in the number of access hierarchies will lead to an increase in attributes in the access policy; therefore, the encryption time will also increase. The access hierarchy policy is embedded in the data, so the size of the speech data will not affect the increase in encryption time, but it will affect the decryption time. As the data increases, the decryption time will also increase because it needs to be performed multiple pairing operations in the ciphertext, and the plaintext data can be decrypted when certain policy is

Table 2: Comparison of evaluation indicators for ABE.

Evaluation index number	Reference [16]	Reference [26]	Reference [30]	Proposed
1	Type-I	Type-I	Type-I	Type-III
2	LSSS	Access tree	Access tree	LSSS
3	✓	✗	✗	✓
4	✓	✗	✓	✓
5	✗	✗	✗	✓
6	✗	✗	✗	✓
7	$(4\lvert A_u\rvert + 1)L_{G_1}$	$(2\lvert A_u\rvert + 1)L_{G_0}$	$(2\lvert A_u\rvert + 2)L_{G_0}$	$3(\lvert A_u\rvert + 1)L_{G_1} + 3L_{G_2}$
8	$5n_1 L_{G_1} + 3L_{G_2}$	$[2\sum_{i=1}^{k} \lvert A_{c_i}\rvert + k]L_{G_0} + kL_{G_T}$	$(2\lvert X\rvert + 1)L_{G_0} + L_{G_T}$	$3n_1 L_{G_1} + 3L_{G_2}$

*1, elliptic curve type; 2, access structure; 3, attribute coverage: large attribute universe; 4, multiuser; 5, speech encryption; 6, speech application scenarios; 7, user private key size; 8, ciphertext size.

Table 3: Symbol definition.

Symbol	Definition description
A_u	A user's set of attributes
A_{c_i}	Attributes associated with ciphertext
L_*	The bit length of the * element
$\lvert * \rvert$	Number of * elements
k	Number of files
n_1	Number of rows of matrix **M**
$\lvert X\rvert$	The set of leaf nodes of the access tree in Reference [30]

Table 4: Data user-defined system performance metrics list.

Metrics	Definitions
Setup	Enter a safety parameter randomly, the more the number of cycles, the higher the efficiency and the shorter the running time.
Keygen	In addition to inputting the public key, the master key, and a set of attributes, a random number is selected on a finite field and the attribute private key is generated. The more the number of cycles, the higher the efficiency and the shorter the running time.
Encryption time	It takes time to encrypt plaintext data into ciphertext data. If the system consumes less time, the efficiency of the system is better.
Decryption time	It takes time to ciphertext data into plaintext data using generated keys. The honest and trusted third-party service provider generates the key. If the system consumes less time, the efficiency of the system is better.
Number of attributes	The number of attributes that are given to user/person to encrypt/decrypt the data or/and to access data stored at cloud. Attributes are given to data users by authorized person or entity.
Size of data	The size of the speech data to be encrypted and the owner of the data to be encrypted.

Table 5: Data user-defined parameters list.

Entity name	Attributes/parameters
Military personnel details	Name, position, education, military rank, affiliation, department, privileges
Command details	Command level, participating troops, participants, specific contents, execution steps

satisfied. From Figure 4, it can be concluded that the decryption operation time of the proposed scheme will not increase linearly with the increase of the number of attributes, so it is suitable for the large attribute universe scheme and is suitable for complex multiattribute speech encryption scenarios.

Figure 5 shows the experimental comparison results between the proposed scheme and Sethi's scheme (2020) [16] in terms of efficiency.

As shown in Figure 5(a), the key generation time of the proposed scheme increases with the number of attributes, and the key generation time increases linearly with the increase of the number of attributes, but the time efficiency of the proposed scheme is significantly better than that of Reference [16]. Figure 5(b) shows the encryption time of the proposed scheme does not increase with the increase of attributes. Because the proposed scheme transforms the access strategy from Boolean formulas to linear secret

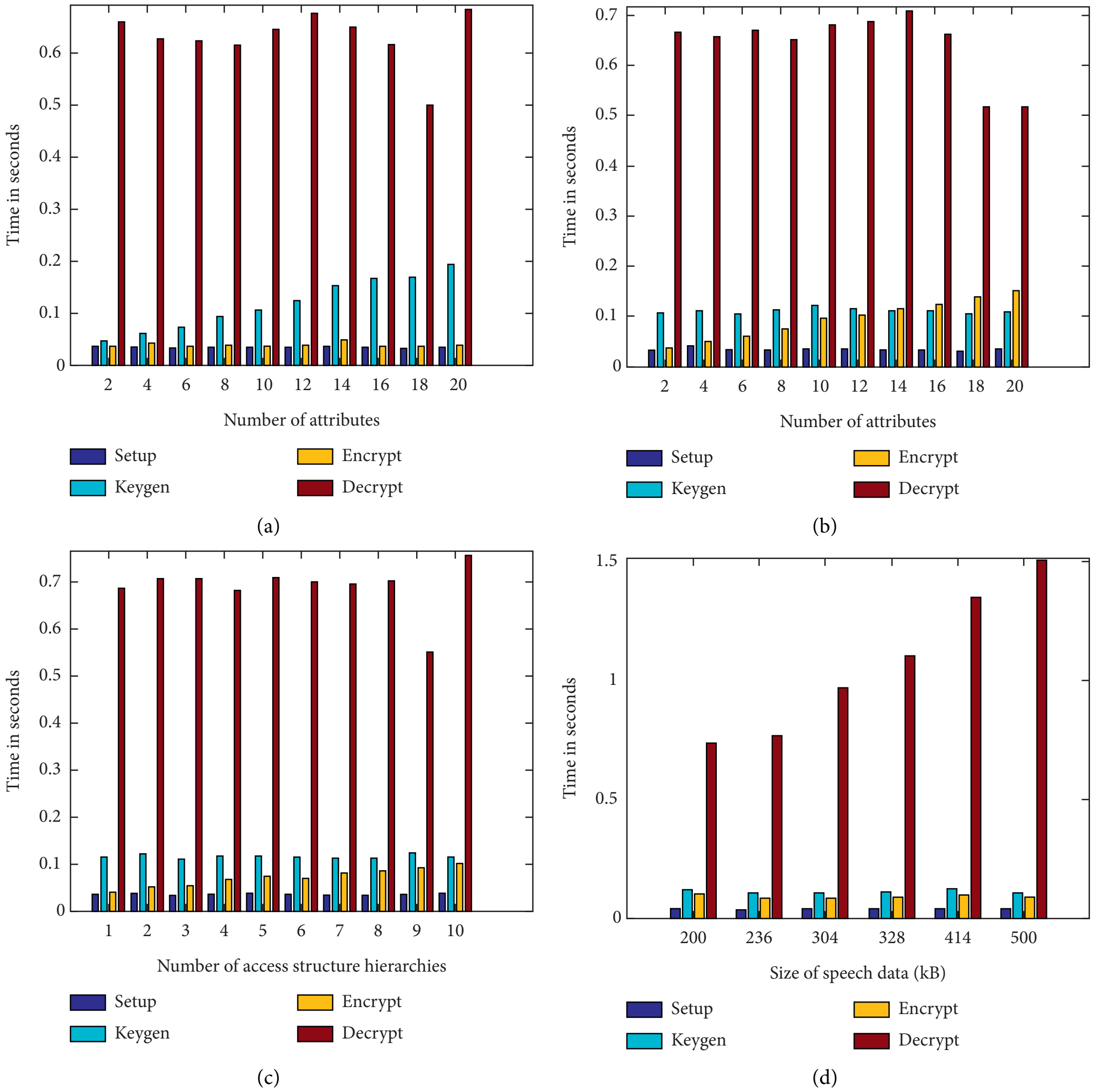

Figure 4: Comparison of different parameters and running time of each algorithm. (a) Comparison of the number of attributes and time. (b) Comparison of the number of attributes and time in the access policy. (c) Comparison of the number of access structure hierarchies and time. (d) Comparison of the size of encrypted speech and time.

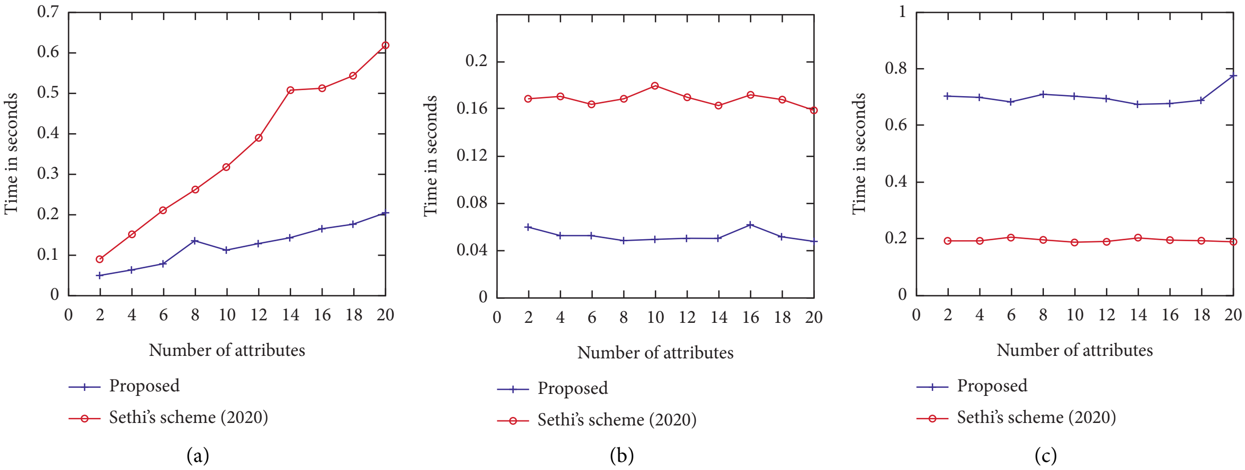

Figure 5: Comparison of the efficiency of the scheme in this article and Sethi's scheme (2020). (a) Comparison of key generation efficiency. (b) Comparison of encryption efficiency. (c) Comparison of decryption efficiency.

sharing scheme, the time efficiency of encryption is improved and the structure advantage hierarchy access tree is fully utilized and provides the speech user more convenient and fast service efficiency and data collection efficiency. It can be seen from Figure 5(b) that the encryption efficiency is better than that of Reference [16]. It can be seen from Figure 5(c) that the decryption efficiency of the proposed scheme is not as efficient as that of Reference [16]. The scheme of Reference [16] uses Boolean formulas for exponential operation during decryption, and the proposed scheme performs multiplication in cyclic groups.

7. Conclusions

In order to achieve the secure storage and sharing of speech data and fine-grained access control in the cloud environment, in this paper, a speech encryption scheme based on ciphertext policy hierarchical attribute for multiattribute speech scenes and relevant attributes and access policies suitable for military speech command application scenarios are constructed. The proposed scheme is suitable for large attribute universe scenes, and it uses the characteristics of multiple attributes of the speech scene to perform hierarchical processing to reflect the hierarchical structure, constructs the access policy into an integrated structure, and uses the attribute fast encryption framework to construct the attribute encryption scheme of the speech data; adopts asymmetric bilinear map, performs pairing operations, encrypts speech data with an integrated access structure and saves storage space and calculations, and achieves fine-grained access control. Theoretical analysis and experiments show that the proposed scheme can effectively improve the efficiency of key and ciphertext generation by using hierarchical access trees, solve the problems of slow deployment of attribute encryption systems and fine-grained access control, and can be further applied to the actual speech application scenarios, such as railway transportation and electric power.

Data Availability

Previously reported speech data were used to support this study and are available at https://arxiv.org/abs/1512.01882 and are cited at relevant places within the text as reference [37].

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (grant nos. 61862041 and 61363078).

References

[1] P. Sharma, R. Jindal, and M. D. Borah, “Blockchain technology for cloud storage: a systematic literature review,” *ACM Computing Surveys*, vol. 53, no. 4, pp. 1–32, 2020.

[2] Y. Zhang, R. H. Deng, S. Xu, J. Sun, Q. Li, and D. Zheng, “Attribute-based encryption for cloud computing access control: a survey,” *ACM Computing Surveys*, vol. 53, no. 4, pp. 1–41, 2020.

[3] R. R. Al-Dahhan, Q. Wen, P. Yu, G. M. Lee, and K. Kifayat, “Survey on revocation in ciphertext-policy attribute-based encryption,” *Sensors*, vol. 19, no. 7, pp. 1659–1717, 2019.

[4] P. Yu, Q. Wen, W. Ni et al., “Decentralized, revocable and verifiable attribute-based encryption in hybrid cloud system,” *Wireless Personal Communications*, vol. 106, no. 2, pp. 719–738, 2019.

[5] J. Ling, J. Chen, J. Chen, and W. Gan, “Multiauthority attribute-based encryption with traceable and dynamic policy updating,” *Security and Communication Networks*, vol. 2021, pp. 1–13, 2021.

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

[7] J. Li, S. Wang, Y. Li et al., “An efficient attribute-based encryption scheme with policy update and file update in cloud computing,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 12, pp. 6500–6509, 2019.

[8] P. S. Challagidad and M. N. Birje, “Efficient multi-authority access control using attribute-based encryption in cloud storage,” *Procedia Computer Science*, vol. 167, pp. 840–849, 2020.

[9] M. Ali, J. Mohajeri, M.-R. Sadeghi, and X. Liu, “A fully distributed hierarchical attribute-based encryption scheme,” *Theoretical Computer Science*, vol. 815, pp. 25–46, 2020.

[10] X. Liu, J. Ma, J. Xiong, and G. Liu, “Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data,” *International Journal on Network Security*, vol. 16, no. 6, pp. 437–443, 2014.

[11] B. Chandrasekaran, Y. Nogami, and R. Balakrishnan, “An efficient file hierarchy attribute based encryption using optimized tate pairing construction in cloud environment,” *Journal of Applied Security Research*, vol. 15, no. 2, pp. 270–278, 2020.

[12] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, “An efficient file hierarchy attribute-based encryption scheme in cloud computing,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1265–1277, 2016.

[13] G. Wang, Q. Liu, J. Wu, and M. Guo, “Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers,” *Computers & Security*, vol. 30, no. 5, pp. 320–331, 2011.

[14] V. K. Arthur Sandor, Y. Lin, X. Li, F. Lin, and S. Zhang, “Efficient decentralized multi-authority attribute based encryption for mobile cloud data storage,” *Journal of Network and Computer Applications*, vol. 129, pp. 25–36, 2019.

[15] R. Z. Du, P. W. Yan, and Y. Liu, “Fine-grained attribute update and outsourcing computing access control scheme in fog computing,” *Journal of Communications*, vol. 42, no. 3, pp. 160–170, 2021.

[16] K. Sethi, A. Pradhan, and P. Bera, “Practical traceable multi-authority CP-ABE with outsourcing decryption and access policy updation,” *Journal of Information Security and Applications*, vol. 51, pp. 102435–102451, 2020.

[17] H. Lian, Q. Wang, and G. Wang, “Large universe ciphertext-policy attribute-based encryption with attribute level user

revocation in cloud storage," *The International Arab Journal of Information Technology*, vol. 17, no. 1, pp. 107–117, 2020.

[18] R. Li, C. Shen, H. He, X. Gu, Z. Xu, and C.-Z. Xu, "A lightweight secure data sharing scheme for mobile cloud computing," *IEEE Transactions on Cloud Computing*, vol. 6, no. 2, pp. 344–357, 2017.

[19] M. Bayat, M. Doostari, and S. Rezaei, "A lightweight and efficient data sharing scheme for cloud computing," *International Journal of Electronics and Information Engineering*, vol. 9, no. 2, pp. 115–131, 2018.

[20] S. Namasudra, "An improved attribute-based encryption technique towards the data security in cloud computing," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 3, pp. e4364–e4379, 2019.

[21] N. Vaanchig, H. Xiong, W. Chen, and Z. Qin, "Achieving collaborative cloud data storage by key-escrow-free multi-authority CP-ABE scheme with dual-revocation," *International Journal on Network Security*, vol. 20, no. 1, pp. 95–109, 2018.

[22] J. Li, N. Chen, and Y. Zhang, "Extended file hierarchy access control scheme with attribute based encryption in cloud computing," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 2, 2019.

[23] T. F. Yang, P. S. Shen, X. Tian, and R.-Q. Feng, "Access control mechanism for classified and graded object storage in cloud computing," *Journal of Software*, vol. 28, no. 9, pp. 2334–2353, 2017.

[24] K. Sowjanya and M. Dasgupta, "A ciphertext-policy Attribute based encryption scheme for wireless body area networks based on ECC," *Journal of Information Security and Applications*, vol. 54, pp. 102559–102569, 2020.

[25] P. Liang, L. Zhang, L. Kang, and J. Ren, "Privacy-preserving decentralized ABE for secure sharing of personal health records in cloud storage," *Journal of Information Security and Applications*, vol. 47, pp. 258–266, 2019.

[26] J. Bethencourt, S. Amit, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the IEEE Symposium On Security And Privacy*, pp. 321–334, Oakland, CL, USA, September 2007.

[27] F. Meng, L. Cheng, and M. Wang, "Ciphertext-policy attribute-based encryption with hidden sensitive policy from keyword search techniques in smart city," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1–22, 2021.

[28] M. T. de Oliveira, A. Bakas, E. Frimpong et al., "A break-glass protocol based on ciphertext-policy attribute-based encryption to access medical records in the cloud," *Annals of Telecommunications*, vol. 75, no. 3-4, pp. 103–119, 2020.

[29] S. M. Pournaghi, M. Bayat, and Y. Farjami, "MedSBA: a novel and secure scheme to share medical data based on blockchain technology and attribute-based encryption," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 11, pp. 4613–4641, 2020.

[30] S. F. Niu, Y. Y. Xie, P. P. Yang, and X. Du, "Cloud-assisted attribute-based searchable encryption scheme on blockchain," *Journal of Computer Research and Development*, vol. 58, no. 4, pp. 811–821, 2021, in Chinese.

[31] S. Agrawal and M. Chase, "F*AME: fast attribute-based message encryption*," in *Proceedings of the 2017 ACM SIGSAC Conference On Computer And Communications Security*, pp. 665–682, Dallas, TX, USA, November, 2017.

[32] S. Galbraith, "New discrete logarithm records, and the death of type 1 pairings," 2014, https://ellipticnews.wordpress.com/2014/02/01/new-discrete-logarithm-records-and-the-death-of-type-1-pairings/.

[33] S. D. Galbraith, K. G. Paterson, and N. P. Smart, "Pairings for cryptographers," *Discrete Applied Mathematics*, vol. 156, no. 16, pp. 3113–3121, 2008.

[34] R. Blum and T. Bocek, "Superlight–A permissionless, light-client only blockchain with self-contained proofs and BLS signatures," in *Proceedings of the 2019 IFIP/IEEE Symposium On Integrated Network And Service Management (IM)*, pp. 36–41, Washington DC, USA, April, 2019.

[35] A. Lewko and B. Waters, "Unbounded HIBE and attribute-based encryption," in *Proceedings of the Annual International Conference On the Theory And Applications Of Cryptographic Techniques*, pp. 547–567, Springer, Heidelberg, Berlin, May 2011.

[36] J. A. Akinyele, C. Garman, I. Miers et al., "Charm: a framework for rapidly prototyping cryptosystems," *Journal of Cryptographic Engineering*, vol. 3, no. 2, pp. 111–128, 2013.

[37] D. Wang and X. Zhang, "THCHS-30: A free chinese speech corpus," *Center for Speech and Language Technology (CSLT) at Tsinghua University*, Beijing, China, 2015, https://arxiv.org/abs/1512.01882.

[38] B. Waters, "Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization," in *Proceedings of the International Workshop On Public Key Cryptography*, pp. 53–70, Springer, Heidelberg, Berlin, March 2011.