Research Article

Secure Multi-Keyword Search and Access Control over Electronic Health Records in Wireless Body Area Networks

Yong Ding^{1,2}, Hui Xu³, Yujue Wang⁴, Fang Yuan⁵ and Hai Liang¹

¹Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin 541004, China

²Cyberspace Security Research Center, Peng Cheng Laboratory, Shenzhen 518055, China

³School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin 541004, China

⁴Hangzhou Innovation Institute, Beihang University, Hangzhou 310052, China

⁵Foreign Affairs and Communications Administration, Beijing 100045, China

Correspondence should be addressed to Hai Liang; lianghai@guet.edu.cn

Received 3 June 2021; Revised 21 July 2021; Accepted 19 August 2021; Published 29 September 2021

Academic Editor: Shui Yu

Copyright © 2021 Yong Ding et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless body area network (WBAN) consists of a number of sensors that are worn on patients to collect dynamic e-health records (EHRs) and mobile devices that aggregate EHRs. These EHRs are encrypted at mobile devices and then uploaded to the public cloud for storage and user access. To share encrypted EHRs with users effectively, help users retrieve EHRs accurately, and ensure EHRs confidentiality, a secure multi-keyword search and access control (SMKS-AC) scheme is proposed, which implements encrypted EHRs access control under the ciphertext-policy attribute-based encryption (CP-ABE). SMKs-AC provides multi-keyword search for accurate EHRs retrieval, supports the validation of decrypted EHRs, and traces and revokes the identity of users who leak private key. Security analysis shows that SMKs-AC is secure against chosen keyword and chosen plaintext attacks. Through theoretical analysis and experimental verification, the proposed SMKs-AC scheme requires less storage resources and computational costs on mobile devices than existing schemes.

1. Introduction

Wireless body area network (WBAN) consists of sensors and mobile devices. Sensors are used to monitor and collect patients's medical/health data. These data are aggregated on mobile devices, then transmitted to public cloud for storage, and shared with data users. However, due to the limited storage space, computing resources, and energy on mobile devices, it is important to keep computing lightweight on mobile devices. Besides, data security and privacy [1] are another important concern in WBAN, including the security of data transmission and storage [2] and access control at the user side [3].

In WBAN, electronic health records (EHRs) are outsourced to public cloud which cannot be directly controlled by the data owner. To ensure the privacy and security of data storage, data owners usually store EHRs in an encrypted

form in the cloud. However, the encrypted data should be shared with data users according to some access control policy. Attribute-based encryption (ABE) is an effective mechanism for fine-grained access control of data. In addition, when data users want to access the encrypted EHRs stored in the public cloud, they can retrieve the required data according to some keywords. Then, the data can be obtained through decryption. Although the single keyword search method can retrieve encrypted data, the search results may contain a large number of irrelevant data, which affects the retrieval accuracy. Therefore, an effective multi-keyword search on encrypted data and the validation of the correctness of decrypted data meet more practical needs.

The EHRs are not only highly private but also commercially valuable. These may promote authorized users to sell their private key for huge profits, so it is necessary to track the identity of the private key owners. Moreover, when

the private key of a data user is sold or compromised, the user's access authority shall be revoked. The decrypted data are able to be forged during transmission, and data users cannot recognize the forged data. Hence, it is imperative to verify the correctness of the data.

1.1. Our Contributions. Inspired by the LiST scheme [4], based on the schemes of LSABE [5] and Chen et al. [6], we propose a secure multi-keyword search and access control scheme (SMKS-AC) that supports multi-keyword search of encrypted EHRs and verification of decrypted EHRs in WBAN. It enables data users to search for EHRs more accurately and verify the correctness of decrypted EHRs after decryption so as to ensure the reliability and security of EHRs sharing between data owners and data users. In addition, CP-ABE is employed to achieve fine-grained access control over EHRs. If a user's private key is compromised, the system can also track and revoke the user's real identity. Our SMKs-AC scheme is suitable to the limited resources of mobile devices since only a few exponentiation operations are performed on mobile devices, while bilinear operations are transferred to the public cloud.

The rest of this paper is organized as follows. Section 2 reviews related works. Section 3 provides the preliminaries for the proposed scheme. Section 4 presents system model and security requirements. Section 5 presents a concrete SMKs-AC construction. Section 6 shows the security analysis, as well as the function and performance comparison with other schemes. Finally, Section 7 concludes this paper.

2. Related Works

To achieve fine-grained access control of outsourcing data, ABE provides a good method of data encryption and sharing. ABE is an extension of the identity-based encryption algorithm, which was first proposed by Amit and Waters [7]. It can be divided into two types, namely, key policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE). The first KP-ABE, proposed by Goyal et al. [8], associates ciphertext with a set of attributes, while the private key is associated with an access policy that controls which ciphertext users can decrypt. The CP-ABE is first proposed by Bethencourt et al. [9], which can realize complex access control on encrypted data. The main idea is to embed the user's attribute set into the private key, and the ciphertext contains the access policy that determines which user can access the ciphertext. The key can be used to decrypt the ciphertext for the user who can access the data. The advantage of the CP-ABE scheme is that encrypted data can be kept secret even if the storage server is untrusted [10], such as the public cloud.

In order to save storage resources and computing costs at mobile devices, Go et al. [11] introduced an outsourcing scheme that reduces the storage space of mobile devices by outsourcing encrypted medical data to cloud servers. It is also the first outsourcing scheme that can resist the malicious cloud server attacks. Li et al. [12] proposed an attribute-based

encryption scheme for verifiable outsourcing encryption and decryption. Their scheme not only reduces the storage cost of mobile devices but also requires only a small amount of computing overhead to complete the decryption. However, Guo et al. [13] found that there was a security problem in Li et al.'s scheme [12] and put forward an improved scheme. Fan et al. [14] proposed a secure and effective outsourcing computing algorithm to solve the problem of a large amount of computing in existing schemes for data encrypting on mobile devices. This algorithm aims to outsource most of the computing involved in encryption and decryption to the cloud, thereby reducing the cost of computing on mobile devices. In order to reduce the overhead of data transmission and the storage space occupied by encrypted data, Zhang et al. [15] proposed an outsourced data access control scheme with constant size, which can keep the encrypted data and the user's private key constant size.

When some users sell private keys for profit, the outsourced data will face the risk of disclosure. Yu et al. [16] proposed a scheme that can effectively protect the outsourced data. If a malicious user shares the decryption key for profit, the user's identity is required to be publicly verified and the request cannot be denied. Therefore, it is necessary to trace the user's identity in ABE. Zhang et al. [17] proposed a scheme that can change user attributes and track the identity of traitors. The results show that the scheme provides feasibility and reliability for practical application. Sethi et al. [18] constructed a multiauthority CP-ABE scheme, which not only provides the function of tracking the identity of malicious users who leak decryption keys but also provides the ability to outsource decryption to reduce the computational burden on users and also supports access policy updates.

Although ABE scheme can preserve the security and privacy of outsourced encrypted medical data, it still faces another problem in retrieving the required medical data in a large number of ciphertexts. The encryption algorithm directly makes the outsourced EHRs unreadable. In order to solve the problem of encrypted data search, Song et al. [19] proposed a scheme of using keywords to search encrypted data stored on untrusted servers without revealing any text information. After that many schemes have been proposed to search for encrypted data. For example, Vaanchig et al. [20] proposed a temporary and fuzzy keyword search public key encryption scheme, which can resist keyword guessing attacks and limit data retrieval time, thus enhancing the security of keyword search. Zhou et al. [21] proposed a public key encryption scheme, in which CP-ABE technology is used for fine-grained access control and keyword search of encrypted data. Their scheme is indistinguishable against adaptive selection keyword attacks.

Besides, the ABE scheme for keyword search of outsourced encrypted data has also been specifically studied in [22–25]. However, these schemes only support single keyword search, which limits the flexibility and accuracy of data retrieval. Sun et al. [26] proposed a multi-keyword search scheme based on CP-ABE, which supports auditing on search results. This scheme reduces a large number of irrelevant search results from cloud servers by narrowing the

search scope. Moreover, Long et al. [27] proposed a light-weight multi-keyword search algorithm based on attribute encryption, which not only supports multi-keyword search but also reduces the computing cost of mobile devices.

3. Preliminaries

3.1. Linear Secret-Sharing Scheme (LSSS)

Definition 1 (see [28]). A secret-sharing scheme Π over a set of parties $\mathcal{P}$ is called linear (over $\mathbb{Z}_p$) if

- (1) The shares for all parties form a vector over $\mathbb{Z}_p$.
- (2) There exists a matrix M with l rows and n columns called the share-generating matrix for Π . For all $i = 1, \dots, l$, the i -th row of M is labeled by a party $\rho(i)$, where ρ is a function from i to $\mathcal{P}$. Set the column vector $v = (s, r_2, \dots, r_n)$, where $s \in \mathbb{Z}_p$ is the secret to be shared and $r_2, \dots, r_n$ are randomly chosen. Mv is the vector of l shares of the secret s according to Π . The share $(Mv)_i$ belongs to party $\rho(i)$.

It is shown in [28] that every LSSS defined above has the linear reconstruction property. Let Π be an LSSS for access structure $\mathbb{A}$, $S \in \mathbb{A}$ be any authorized set, and $I \subset \{1, 2, \dots, l\}$ be defined as $I = \{i: \rho(i) \in S\}$. Then, there exist constants $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ such that if $\{\lambda_i\}_{i \in I}$ are valid shares of any secret s according to Π , then $\sum_{i \in I} \omega_i \lambda_i = s$. Moreover, these constants $\{\omega_i\}_{i \in I}$ can be found in time polynomial in the size of the share-generating matrix M .

3.2. Bilinear Groups. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of prime order p and g be a generator of $\mathbb{G}_1$. If $e: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ satisfies the following properties:

- (1) Bilinearity: for any $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$.
- (2) Nondegeneracy: there exist two elements $u, v \in \mathbb{G}_1$ such that $e(u, v) \neq 1$, where 1 is the identity element in $\mathbb{G}_2$.
- (3) Computability: $e(u, v)$ can be efficiently computed, for any $u, v \in \mathbb{G}_1$ then e is a bilinear map.

3.3. DBDH Assumption. The security of the proposed SMKS-AC construction is based on the following decisional bilinear Diffie-Hellman (DBDH) assumption.

Assumption 1 (decisional bilinear Diffie-Hellman (DBDH) assumption). Let $\mathbb{G}_1$ be a multiplicative cyclic group of prime order p with generator g . Let $a, b, c \in \mathbb{Z}_p^*$ be randomly selected. If an adversary $\mathcal{A}$ is given g, g^a, g^b, g^c , and $e(g, g)^{abc}$, it is difficult for $\mathcal{A}$ to distinguish $e(g, g)^{abc}$ from a random element R in $\mathbb{G}_2$. The adversary $\mathcal{A}$ has advantage ϵ in solving the DBDH assumption if $\text{Adv}_{DBDH} = |\Pr[\mathcal{A}(g, g^a, g^b, g^c, e(g, g)^{abc}) = 0] - \Pr[\mathcal{A}(g, g^a, g^b, g^c, R) = 0]| \geq \epsilon$.

4. System Model and Security Requirements

In this section, we introduce the system model and security requirements of SMKS-AC.

4.1. System Model. As shown in Figure 1, the system of SMKS-AC mainly consists of the following four types of entities: data owner, medical staff who are regarded as data users, public cloud (PC), and key generation center (KGC).

- (1) *WBAN (Data Owner).* WBAN contains many sensors, which are embedded or worn on patients to collect medical data. The collected data are aggregated and transmitted to personal digital assistant (PDA) through wireless channel. Keywords are extracted from EHRs to describe health information, and an EHR can be described by multiple keywords. Then, these keywords form a keyword group, and the corresponding EHR is encrypted under a specific access policy. Finally, these encrypted EHRs are outsourced to the PC.
- (2) *Medical Staff (Data User).* Medical staff, as data users, have their own specific set of attributes. Data users are authorized to search encrypted EHRs according to their set of attributes. A data user can generate keyword trapdoor and send it to the PC to realize data retrieval. As long as the search keywords are included in the keyword group describing the corresponding EHR and the user's set of attributes satisfies the access policy, the searched encrypted EHR will be returned. Then, the user uses his/her private key to decrypt and verify the decrypted EHR to ensure the correctness of EHR.
- (3) *PC.* PC has almost unlimited storage space and computing power, which can be used to store encrypted EHRs and respond to users' data retrieval requests. In SMKS-AC, PC first verifies whether the keywords in the retrieval request are in the keyword group of the retrieved data. Then, the PC helps transform the retrieved ciphertext into a ciphertext that users can decrypt through lightweight computing.
- (4) *KGC.* KGC generates public parameters for the whole system and distributes private key to each data user. The user's attributes set is embedded in the private key to implement access control. When the user's private key is maliciously disclosed, KGC can trace the identity of the private key holder and add it to the revocation list.

4.2. Security Requirements. In WBAN, in order to ensure the availability, privacy, and security of EHRs, a secure data access control scheme supporting multi-keyword search needs to meet the following security requirements.

4.2.1. Confidentiality of EHRs. The EHRs should be encrypted before being outsourced to the PC for remote

FIGURE 1: System model.

storage. Only when the user's set of attributes meets the access policy in the ciphertext, the data can be decrypted by the user.

4.2.2. Accuracy of Retrieved EHRs. When the users retrieve data, there will be a lot of redundant data in the single keyword search results. Thus, it is necessary to use multi-keyword search to improve the accuracy of retrieval results.

4.2.3. Verifiability of EHRs. Since the ciphertext is outsourced to PC, in order to prevent PC from tampering with the ciphertext, data users need to verify the accuracy of the retrieved data after decrypting with their own private key.

5. A SMKS-AC Construction

In this section, we present a SMKS-AC construction and depict the running procedure in Figure 2.

5.1. System Setup. Let $g \in \mathbb{G}_1$ be a generator of group $\mathbb{G}_1$ of prime order p . Let $e: \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a bilinear map, $H: \{0, 1\}^* \rightarrow \mathbb{Z}_p^*$ and $H_1: \{0, 1\}^* \rightarrow \kappa$ be two collision-resistant hash functions, and κ be the key space of the symmetric encryption algorithm (i.e., AES, DES, and so on). KGC selects random elements $\alpha, a, \lambda, \tau \in \mathbb{Z}_p^*$ and $k_1 \in \kappa$ and computes

$$\begin{aligned} f &= g^\tau, \\ Y &= e(g, g)^\alpha, \\ Y_0 &= e(g, f), \\ h &= g^\lambda. \end{aligned} \quad (1)$$

The public parameter and the master secret key of the system are $PP = (g, p, e, h, f, Y, Y_0, g^a, f^a, H, H_1)$ and $MSK = (\alpha, a, \lambda, \tau, k_1)$, respectively.

5.2. Key Generation. Let id be the user's identity and $S = \{\xi_1, \xi_2, \dots, \xi_k\} \subseteq \mathbb{Z}_p^*$ be the data user's attributes set, where k represents the number of attributes in the attribute set. KGC selects random elements $r, \varrho, u', u'' \in \mathbb{Z}_p^*$ and computes

$$\begin{aligned} \delta &= \text{Enc}_{k_1}(id), \\ D_1 &= g^{a/(\lambda+\delta)}, \\ D_2 &= \delta, \\ D_{3,i} &= g^{r/(\xi_i+\tau)}, \\ D_4 &= g^{(a-ar)/(\lambda+\delta)}, \\ D_5 &= \varrho, \\ \Psi_1 &= D_1^{u'}, \\ \Psi_2 &= (D_4^\varrho)^{u''}, \\ \Psi_{3,i} &= (D_{3,i}^\varrho)^{u''}. \end{aligned} \quad (2)$$

The public key $PK_{id,S}$ and secret key $SK_{id,S}$ of the data user are $PK_{id,S} = (\Psi_1, \Psi_2, \{\Psi_{3,i}\}_{i \in [k]})$ and $SK_{id,S} = (D_1, D_2, \{D_{3,i}\}_{i \in [k]}, D_4, D_5, u', u'')$, respectively.

5.3. Encryption. Data owner chooses keywords for an EHR m to form a keyword set $KW = \{kw_1, kw_2, \dots, kw_{l_1}\}$ and constructs the l_1 degree polynomial:

$$\varphi(x) = \eta_{l_1}x^{l_1} + \eta_{l_1-1}x^{l_1-1} + \dots + \eta_0 \text{ mod } p, \quad (3)$$

where $H(kw_1), H(kw_2), \dots, H(kw_{l_1})$ are the l_1 roots of $\varphi(x) = 1$. Let M be a $l \times n$ matrix and ρ be a function that associates M_i to attributes, where l and n represent the rows and columns of M , respectively, and M_i is the i -th row of M , $i = 1, 2, \dots, l$. Data owner randomly selects $Y \in \mathbb{G}_2^*$ and computes

$$\begin{aligned} \zeta &= H_1(Y), \\ C_m &= \text{Enc}_\zeta(m), \\ s &= H(m, Y). \end{aligned} \quad (4)$$

Then, the data owner chooses a random vector $\vec{v} = (s, y_2, y_3, \dots, y_n)^T \in \mathbb{Z}_p^n$, where $y_2, y_3, \dots, y_n \in \mathbb{Z}_p^*$ are random elements. For $i \in [l]$, the data owner computes

$$s_i = M_i \cdot \vec{v} \text{ mod } p, \quad (5)$$

selects random elements $s', s'' \in \mathbb{Z}_p^*$, and calculates

FIGURE 2: The flow of SMKS-AC construction.

$$\begin{aligned}
C_0 &= Y \cdot Y^s, & T_0 &= u \cdot (u')^{-1}, \\
C_1 &= g^s, & T_1 &= u \cdot (u'')^{-1}, \\
C_2 &= h^s, & T_2 &= D_2, \\
C_{3,i} &= \rho(i) \frac{s_i}{s}, & T_3 &= u \cdot u_0 \cdot l_2^{-1}, \\
C_{3,i}' &= \frac{s_i}{s'}, & T_4 &= Y_0^u, \\
C_4 &= f^{as''}, & T_{5,j} &= u_0^{-1} \sum_{i=1}^{l_2} H(kw_i)^j, \quad j \in \{0, 1, \dots, l_1\}. \\
C_5 &= Y_0^s Y^{ss'}, & & \\
C_6 &= g^{as'}, & & \\
C_j &= \frac{\eta_j}{s'}, \quad j \in \{0, 1, \dots, l_1\}. & &
\end{aligned} \tag{6}$$

Then, the ciphertext $CT = (C_0, C_1, C_2, \{C_{3,i}, C_{3,i}'\}_{i \in \{1, \dots, l\}}, C_4, C_5, C_6, \{C_j\}_{j \in \{0, 1, \dots, l_1\}}, C_m)$ and the access policy (M, ρ) are outsourced to PC.

5.4. Trapdoor Generation. If a data user wants to search for EHRs containing the keyword set $KW' = \{kw_1, kw_2, \dots, kw_{l_2}\}$, the user needs to construct keywords trapdoor $T_{KW'}$. The data user chooses random elements $u, u_0 \in \mathbb{Z}_p^*$ and computes

The keyword trapdoor $T_{KW'} = (T_0, T_1, T_2, T_3, T_4, \{T_{5,j}\}_{j \in \{0, 1, \dots, l_1\}})$ is sent to PC.

5.5. Transform. After receiving keywords trapdoor $T_{KW'}$ from a data user, the PC first verifies whether the following equation is true:

$$T_4 \cdot e(\Psi_1^{T_0}, C_1^{T_2} C_2) = C_5 \sum_{j=0}^{l_1} C_j T_{5,j}. \tag{8}$$

If so, the PC outputs 1 means $KW' \subset KW$. Otherwise, the PC outputs 0. And then, PC checks whether the attribute set S associated with $T_{KW'}$ meets the access policy (M, ρ) associated with ciphertext CT .

- (1) If the PC outputs 0 or the attribute set S associated with $T_{KW'}$ does not satisfy the access policy (M, ρ) , the Transform algorithm aborts.
- (2) If the PC outputs 1 and S associated with $T_{KW'}$ satisfies (M, ρ) , the Transform algorithm continues as follows. Let $I \subset [l]$ be $I = \{i: \rho(i) \in S\}$, and there exists a set of constants $\{\omega_i \in \mathbb{Z}_p\}_{i \in I}$ such that

$$\sum_{i \in I} \omega_i M_i = (1, 0, \dots, 0). \quad (9)$$

The PC computes

$$\begin{aligned} I' &= e\left(C_6, \prod_{i \in I} \Psi_{3,i}^{C_{3,i'} \omega_i}\right) \cdot e\left(C_4, \prod_{i \in I} \Psi_{3,i}^{C_{3,i'} \omega_i}\right), \\ \Gamma &= (I')^{T_1}, \\ \Gamma' &= e(\Psi_2, C_1^{T_2} C_2)^{T_1}. \end{aligned} \quad (10)$$

Therefore, $C' = \Gamma \cdot \Gamma' = e(g, g)^{\alpha \text{qsu}}$.

Finally, the transformed ciphertext $\text{CT}' = (C_0, C', C_m)$ is sent back to the data user.

5.6. Decryption and Verification. After receiving the transformed ciphertext $\text{CT}' = (\text{CT}_1, \text{CT}_2, C_m)$, the data user calculates

$$\begin{aligned} Y' &= \frac{\text{CT}_1}{(\text{CT}_2)^{1/(u D_5)}}, \\ \zeta' &= H_1(Y'), \end{aligned} \quad (11)$$

$$\begin{aligned} m' &= \text{Dec}_{\zeta'}(C_m), \\ s_1 &= H(m', Y'). \end{aligned} \quad (12)$$

If

$$\text{CT}_1 = Y' \cdot Y^{s_1}, \quad (13)$$

$$\text{CT}_2 = Y^{s_1 u D_5}, \quad (14)$$

both hold, then m' is outputted.

5.7. User Tracing. If the private key of some authorized user is leaked, KGC is able to perform the following two algorithms to track its real identity.

5.7.1. Keycheck. Suppose $S = \{\xi_1, \xi_2, \dots, \xi_k\}$. KGC checks whether the private key $\text{SK}_{\text{id}, S}$ is in the form of $(D_1, D_2, \{D_{3,i}\}_{i \in [k]}, D_4, D_5, u', u'')$, where $D_1, D_{3,i}, D_4 \in \mathbb{G}_1$ and $D_2, D_5, u', u'' \in \mathbb{Z}_p^*$. Then, the KGC verifies the following equation:

$$e(D_4, h \cdot g^{D_2})^k \cdot e\left(g^a, \prod_{i \in [k]} D_{3,i}^{\xi_i}\right) \cdot e\left(f^a, \prod_{i \in [k]} D_{3,i}\right) = Y^k. \quad (15)$$

If it is satisfied, the algorithm outputs 1 or 0 otherwise.

5.7.2. Trace. If the KeyCheck algorithm outputs 0, it means that the private key $\text{SK}_{\text{id}, S}$ does not need to be traced; in this case, the Trace algorithm outputs $\perp$. Otherwise, KGC can use the master key MSK to recover the identity of the private key holder by calculating $\text{id} = \text{Dec}_{k_1}(D_2)$.

5.8. User Revocation. After the Trace algorithm is completed, the access right of the holder who leaked the private key $\text{SK}_{\text{id}, S}$ needs to be revoked. Therefore, KGC stores the component $D_2 = \delta$ of the key containing the user's identity information on the PC. When a user sends a data retrieval request and submits keywords trapdoor $T_{\text{KW}'}$, the PC checks whether the component $D_2 = \delta$ in $T_{\text{KW}'}$ is in the revocation list. If so, the user's data retrieval request is rejected.

6. Scheme Analysis

6.1. Correctness and Security Analysis. This section analyzes and proves the correctness and security of SMKS-AC construction.

Theorem 1. *The proposed SMKS-AC construction is correct.*

Proof. In order to prove the correctness of the SMKS-AC construction, we only need to show that equations (8), (11), and (15) hold.

First, PC can determine whether the keyword set KW' searched by the data user satisfies $\text{KW}' \subset \text{KW}$ by verifying equation (8), where KW is keyword set in CT. Since

$$\begin{aligned} T_4 \cdot e(\Psi_1^{T_0}, C_1^{T_2} C_2) &= Y_0^u \cdot e\left(D_1^{u' \cdot u \cdot (t_1)^{-1}}, g^{s_\delta} g^{\lambda s}\right) \\ &= e(g, f)^u \cdot e\left(g^{au/(\lambda+\delta)}, g^{(\lambda+\delta)s}\right) \\ &= e(g, f)^u \cdot e(g, g)^{aus}, \\ C_5^{T_3} \sum_{j=0}^{l_1} C_j^{T_{5,j}} &= \left(Y_0^{s'} Y^{ss'}\right)^{u \cdot u_0 \cdot (l_2)^{-1}} \sum_{j=0}^{l_1} \left(\eta_j / s' \cdot (u_0)^{-1} \sum_{\ell=1}^{l_2} H(\text{kw}_i)^j\right) \\ &= (e(g, f) \cdot e(g, g)^{\alpha s})^{u \cdot (l_2)^{-1}} \sum_{j=0}^{l_1} \left(\eta_j \cdot \sum_{\ell=1}^{l_2} H(\text{kw}_i)^j\right) \\ &= (e(g, f) \cdot e(g, g)^{\alpha s})^u, \end{aligned} \quad (16)$$

we have $KW' \subset KW$.

Second, after receiving the PC's transformed ciphertext, the data user can recover the random element Y by calculating equation (11), so as to decrypt and verify the message. Since $C' = \Gamma \cdot \Gamma' = (I')^{T_1} \cdot \Gamma'$, we have

$$\frac{C_0}{(C')^{1/(uD_5)}} = \frac{Y \cdot e(g, g)^{\alpha s}}{(e(g, g)^{\alpha qsu})^{1/uq}} = Y, \quad (17)$$

where

$$\begin{aligned} I' &= e\left(C_6, \prod_{i \in I} \Psi_{3,i}^{C_{3,i} \cdot \omega_i}\right) \cdot e\left(C_4, \prod_{i \in I} \Psi_{3,i}^{C'_{3,i} \cdot \omega_i}\right) \\ &= e\left(g^a, g^{rqu'' \sum_{i \in I} \rho(i) s_i \omega_i / (\xi_i + \tau)}\right) \cdot e\left(g^{\tau a}, g^{rqu'' \sum_{i \in I} s_i \omega_i / (\xi_i + \tau)}\right) \\ &= e(g, g)^{arqu'' \sum_{i \in I} \rho(i) s_i \omega_i / (\xi_i + \tau) + \sum_{i \in I} \tau s_i \omega_i / (\xi_i + \tau)} \\ &= e(g, g)^{arqu'' s}, \\ \Gamma' &= e(\Psi_2, C_1^{T_2} C_2)^{T_1} \\ &= e\left(g^{(\alpha - ar)/(\lambda + \delta)qu''}, g^{s(\lambda + \delta)}\right)^{u(u'')^{-1}} \\ &= e(g, g)^{(\alpha - ar)qsu}, \\ C' &= \left(e(g, g)^{arqu'' s}\right)^{T_1} e(g, g)^{(\alpha - ar)qsu} \\ &= e(g, g)^{arqu'' s \cdot u(u'')^{-1}} e(g, g)^{(\alpha - ar)qsu} \\ &= e(g, g)^{\alpha qsu}. \end{aligned} \quad (18)$$

Third, if the data user sells the private key for profit, KGC can verify the authenticity of the sold key by verifying equation (15) to determine whether the identity of the user holding the private key is worth tracking due to

$$\begin{aligned} &e(D_4, h \cdot g^{D_2})^k \cdot e\left(g^a, \prod_{i \in [k]} D_{3,i}^{\xi_i}\right) \cdot e\left(f^a, \prod_{i \in [k]} D_{3,i}\right) \\ &= e\left(g^{(\alpha - ar)/(\lambda + \delta)}, g^\lambda g^\delta\right)^k \cdot e\left(g^a, g^{i \in [k] \sum_{i \in [k]} \tau \xi_i / (\xi_i + \tau)}\right) \cdot e\left(g^{\tau a}, g^{i \in [k] \sum_{i \in [k]} \tau / (\xi_i + \tau)}\right) \\ &= e(g, g)^{(\alpha - ar)k} \cdot e(g, g)^{ar \cdot \sum_{i \in [k]} \xi_i / (\xi_i + \tau)} \cdot e(g, g)^{ar \cdot \sum_{i \in [k]} \tau / (\xi_i + \tau)} \\ &= e(g, g)^{\alpha k} \end{aligned} \quad (19)$$

Therefore, the proposed SMKS-AC construction is correct. $\square$

Theorem 2. *The proposed SMKS-AC construction can ensure the EHRs confidentiality. Under the DBDH assumption, the SMKS-AC construction is indistinguishable against chosen keyword and chosen plaintext attack (IND-CKCPA).*

Proof. The form of ciphertext CT in the SMKS-AC construction is similar to that in [4]. Compared with the SMKS-AC construction, the data owners of [4] only extract one keyword in processing an EHR. In our construction, in order to improve the accuracy of the data retrieval by data users, the data owners of SMKS-AC construction are able to extract multiple keywords from an EHR. The specific proof of Theorem 2 is similar to Theorem 1 in [4]. Therefore, the SMKS-AC construction is IND-CKCPA secure. $\square$

Theorem 3. *The proposed SMKS-AC construction can resist collusion attacks against multiple users.*

Proof. In a multi-user system, collusion attack is an important attack type. Authorized users can collude with each other to generate a new key and gain extra privileges. Whereas in our solution, for each user, KGC selects a set of random numbers based on the user's attributes to generate the user's private key. Users who intend to collude with each other cannot combine their private keys to generate a new valid key. Since each user's private key is generated by different random numbers, they are not compatible with each other. Therefore, the proposed SMKS-AC construction is secure against collusion attacks. $\square$

6.2. Comparison. This section compares SMKS-AC construction with existing schemes in terms of function, storage, and computation overheads. The comparison is shown in Tables 1–3, respectively.

As shown in Table 1, in addition to the proposed SMKS-AC construction, both schemes [5, 29] provide multi-keyword search function. Data users are allowed to verify the message after decryption in [4] and SMKS-AC construction. Since the user's private key may be used maliciously, both [4] and the SMKS-AC construction provide user trace and user revocation functions, which are not considered in [5, 22, 29].

The storage and computing resources of mobile devices in WBAN are limited. In practical applications, storage and computation costs on mobile devices need to be considered. Let $|PP|$, $|SK|$, $|CT|$, and $|T_{KW'}|$ represent the sizes of the public parameter, private key of data user, ciphertext, and the keywords trapdoor, respectively. $|\mathbb{G}_1|$, $|\mathbb{G}_2|$, and $|\mathbb{Z}_p|$ denote the length of an element in groups $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{Z}_p$, respectively. Let $|S|$ be the size of attribute set S , l be the number of rows in M , $|U|$ be the size of the universe attribute set U , and l_1 be the size of the keyword set KW . Besides, E_1 and E_2 represent exponentiation operations on groups $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. P is the symbol of bilinear pairing operation.

Table 2 shows the storage cost comparison. It can be seen that only the size of public parameter in Wang et al.'s scheme [29] is related to U , which is the number of all the attributes in the whole system. As the number of system attributes increases, the size of public parameter also increases, which do not meet the actual needs. Although the length of public parameters of schemes [5, 22] is smaller than that of SMKS-AC construction, the sizes of user's private key, ciphertext, and the keywords trapdoor are, respectively, larger than

TABLE 1: Function comparison.

	Access control	Multi-keyword search	Verify decryption	User trace	User revocation
Yang et al.'s scheme [4]	✓	×	✓	✓	✓
Zhang et al.'s scheme [5]	✓	✓	×	×	×
Wang et al.'s scheme [22]	✓	×	×	×	×
Wang et al.'s scheme [29]	✓	✓	×	×	×
SMKS-AC scheme	✓	✓	✓	✓	✓

“✓” represents supported; “×” represents not supported.

TABLE 2: Storage cost comparison.

	PP	SK	CT	T _{KW'}
Yang et al.'s scheme [4]	$5 \mathbb{G}_1 + 2 \mathbb{G}_2 $	$(S + 1) \mathbb{G}_1 + 5 \mathbb{Z}_p $	$4 \mathbb{G}_1 + 2 \mathbb{G}_2 + 2l \mathbb{Z}_p $	$7 \mathbb{Z}_p $
Zhang et al.'s scheme [5]	$4 \mathbb{G}_1 + \mathbb{G}_2 $	$(S + 3) \mathbb{G}_1 + \mathbb{Z}_p $	$(l + 4) \mathbb{G}_1 + 2 \mathbb{G}_2 + (l_1 + 4) \mathbb{Z}_p $	$ \mathbb{G}_1 + \mathbb{G}_2 + (l_1 + 2) \mathbb{Z}_p $
Wang et al.'s scheme [22]	$5 \mathbb{G}_1 $	$(2 S) \mathbb{G}_1 + \mathbb{Z}_p $	$(l + 2) \mathbb{G}_1 + 3 \mathbb{G}_2 + \mathbb{Z}_p $	$2 \mathbb{G}_1 + \mathbb{Z}_p $
Wang et al.'s scheme [29]	$(4 + U) \mathbb{G}_1 + 3 \mathbb{G}_2 $	$3 \mathbb{G}_1 + \mathbb{Z}_p $	$(2l + l_1 + 4) \mathbb{G}_1 + \mathbb{G}_2 $	$(3 + l_1) \mathbb{G}_1 $
SMKS-AC scheme	$5 \mathbb{G}_1 + 2 \mathbb{G}_2 $	$(S + 2) \mathbb{G}_1 + 4 \mathbb{Z}_p $	$4 \mathbb{G}_1 + 2 \mathbb{G}_2 + (2l + l_1) \mathbb{Z}_p $	$ \mathbb{G}_2 + (l_1 + 4) \mathbb{Z}_p $

TABLE 3: Computation cost comparison.

	Key generation	Encryption	Trapdoor	Decryption
Yang et al.'s scheme [4]	$(S + 1)E_1$	$4E_1 + 3E_2$	0	E_2
Zhang et al.'s scheme [5]	$(S + 4)E_1$	$(2l + 4)E_1 + 3E_2 + 3P$	$E_1 + E_2 + P$	E_2
Wang et al.'s scheme [22]	$3E_1$	$5E_1 + 3E_2$	$2E_1$	$E_1 + P$
Wang et al.'s scheme [29]	$(S + 6)E_1$	$(4 + l + l_1)E_1 + E_2 + P$	$(3 + l_1)E_1$	$E_1 + E_2 + P$
SMKS-AC scheme	$(S + 2)E_1$	$4E_1 + 3E_2$	E_2	E_2

those of SMKS-AC construction, which will increase the storage burden and data transmission time of mobile devices. Although private key size of scheme [29] is smaller than that of SMKS-AC construction, the sizes of ciphertext and the keywords trapdoor in SMKS-AC construction are, respectively, smaller than those of [29]. Note that the length of the elements in group $\mathbb{Z}_p$ is much smaller than that in group $\mathbb{G}$. In addition, the sizes of private key, ciphertext and keywords trapdoor in SMKS-AC scheme are, respectively, larger than those in scheme [4]. The main reason is that the SMKS-AC construction realizes multi-keyword search, which can ensure the accuracy of data retrieval, while scheme [4] can only support single keyword search.

Table 3 shows the computation cost comparison, where only the cost of exponentiation and bilinear pairing operations are considered. In the Key Generation algorithm, the KGC can use $|S| + 2$ exponentiation operations on group $\mathbb{G}_1$ to get the user's private key in SMKS-AC construction. Except for [4], other schemes require more computation than ours. The Encryption algorithm is executed on mobile devices with limited resources. In order to save storage space on mobile devices, the EHRs should be encrypted immediately and then transferred to the PC, which require high encryption efficiency. In our scheme and the scheme [4], only four exponentiation operations on group $\mathbb{G}_1$ and three exponentiation operations on group $\mathbb{G}_2$ are required to generate a ciphertext. However, the other schemes require other redundant operations overhead. Since the Trapdoor

Generation and Decryption algorithms are executed on the user's mobile device, a small amount of computations is required to get the keywords trapdoor and decryption data to meet the system needs. In addition, compared with the scheme [4] and our construction, the other schemes need to carry out bilinear pairing operations and exponentiation operations.

6.3. Experimental Analysis. In this section, we implement the code based on the Pairing-Based Cryptography Library (PBC-0.5.14, <https://crypto.stanford.edu/pbc/>). The experimental simulation is run on a virtual machine with 4-core 8 GB memory, 64 bit Linux Ubuntu 18.04.5 operating system, and Intel (R) Core (TM) i5-8265U CPU @ 1.60 GHz 1.80 GHz. The element of cyclic group is 512 bits, and the length of p is 160 bits.

Figure 3 compares the running time of each phase in SMKS-AC construction with that in other schemes. Since other schemes had no trace and revocation phases, we only compared the time of Key Generation algorithm, Encryption algorithm, Trapdoor Generation algorithm, Transform algorithm, and Decryption algorithm. Due to the limited resources of mobile devices, it is particularly necessary to consider the complexity of algorithms executed by data owners and users.

Figure 3(a) shows the time required for the Key Generation algorithm, which is executed by the KGC. It can be

FIGURE 3: Computation overhead. (a) Key Generation time. (b) Encryption time. (c) Trapdoor Generation time. (d) Transform time. (e) Decryption time.

seen that the scheme in [22] takes the least time to generate the key. With the increase in the number of attributes, the time required remains unchanged. However, our SMKS-AC construction generates different keys for different attributes in the attribute set of data users; thus, the time user key generation will increase as the number of attributes in the attribute set increases.

Figure 3(b) indicates the time required for the data owner to execute the Encryption algorithm. The computational cost of SMKS-AC construction and the scheme in [22] roughly keeps constant, while the computational cost of [5] is increasing dynamically. The main reason is that the generation time of ciphertext in scheme [5] is related to the number of rows in matrix M . As the number of rows increases, the time to run the Encryption algorithm also increases.

It can be seen from Figure 3(c) that the time cost of the Trapdoor Generation algorithm has nothing to do with the number of attributes, and the generation time of trapdoor by SMKS-AC construction is shorter than that of the existing schemes. Note that the SMKS-AC construction and scheme [5] support multi-keyword search to improve the search accuracy, while scheme [22] only supports single keyword search.

As indicated in Figure 3(d), the SMKS-AC construction takes more time than the existing schemes. It is worth noting that the Transform algorithm is implemented by the PC, which has almost unlimited computing power and resources.

Figure 3(e) shows the decryption time of encrypted data by data users. The SMKS-AC construction takes almost the same decryption time as the scheme [5], and scheme [22] takes relatively much time. Also, the SMKS-AC construction satisfies the verification on the message after decryption.

7. Conclusion

In WBAN, in order to achieve secure sharing of outsourced EHRs with a large number of users, we proposed a SMKS-AC construction supporting secure multi-keyword search and access control. SMKS-AC provides fine-grained access control and verifiability of decrypted EHRs, multi-keyword search over encrypted EHRs, user's identity tracking, and revocation. Security analysis showed that the SMKS-AC construction can resist chosen keyword and chosen plaintext attacks and collusion attacks. Theoretical analysis and experiments demonstrate that our SMKS-AC construction is more effective and takes lower computational cost than existing related solutions.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This article was supported in part by the National Key R&D Program of China under project 2020YFB1006003, the

National Natural Science Foundation of China under projects 61772150, 61862012, and 61962012, the Guangdong Key R&D Program under project 2020B0101090002, the Guangxi Natural Science Foundation under grants 2018GXNSFDA281054, 2019GXNSFFA245015, and 2019GXNSFGA245004, the Peng Cheng Laboratory Project of Guangdong Province under grant PCL2018KP004, the Innovation Projects of GUET Graduate Education under grants 2021YCXSL16 and 2021YCXSL15, the Guangxi Young Teachers' Basic Ability Improvement Program under grant 2021KY0214, and the Open Program of Guangxi Key Laboratory of Cryptography and Information Security under grant GCIS201930.

References

- [1] A. Gazi Imtiyaz, S. Jimmye, and J. G. Kaiser, *Security and Privacy of E-Health Data*, pp. 199–214, Springer, Singapore, 2021.
- [2] C. Yang, X. Tao, F. Zhao, and Y. Wang, "Secure data transfer and deletion from counting bloom filter in cloud computing," *Chinese Journal of Electronics*, vol. 29, no. 2, pp. 79–86, 2020.
- [3] X. Zhou, J. Liu, Z. Zhang, and Q. Wu, "Secure outsourced medical data against unexpected leakage with flexible access control in a cloud storage system," *Security and Communication Networks*, vol. 2020, no. 7, pp. 1–20, 2020.
- [4] Y. Yang, X. Liu, R. H. Deng, and Y. Li, "Lightweight sharable and traceable secure mobile health system," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 1, pp. 78–91, 2020.
- [5] K. Zhang, J. Long, X. Wang, H.-N. Dai, K. Liang, and M. Imran, "Lightweight searchable encryption protocol for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4248–4259, 2021.
- [6] Y. Chen, Z. L. Jiang, S. M. Yiu, J. K. Au, and X. Wang, "Fully secure ciphertext-policy attribute based encryption with security mediator," *Information and Communications Security*, vol. 8958, pp. 274–289, 2015.
- [7] S. Amit and B. Waters, "Fuzzy identity-based encryption," *Lecture Notes in Computer Science*, vol. 3494, no. 1, p. 457, 2005.
- [8] V. Goyal, O. Pandey, S. Amit, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security*, pp. 89–98, CCS 2006, Alexandria, VA, USA, November 2006.
- [9] J. Bethencourt, S. Amit, and B. Waters, "Ciphertext-policy attribute-based encryption," *IEEE Symposium on Security and Privacy*, vol. 321, 2007.
- [10] Y. Wang, L. Wei, X. Tong, X. Zhao, and M. Li, "CP-ABE based access control for cloud storage," *Advances in Intelligent Systems and Computing*, vol. 455, pp. 463–472, 2017.
- [11] O. Go, R. Safavi-Naini, and L. F. Zhang, "Outsourcing scheme of ABE encryption secure against malicious adversary," *Computers & Security*, vol. 86, pp. 437–452, 2019.
- [12] Z. Li, W. Li, Z. Jin, H. Zhang, and Q. Wen, "An efficient ABE scheme with verifiable outsourced encryption and decryption," *IEEE Access*, vol. 7, pp. 29023–29037, 2019.
- [13] Z. Guo, B. Liu, K. Zhao, and C. Feng, "Comment on 'an efficient abe scheme with verifiable outsourced encryption and decryption,'" *IEEE Access*, vol. 8, pp. 202483–202486, 2020.
- [14] K. Fan, T. Liu, K. Zhang, H. Li, and Y. Yang, "A secure and efficient outsourced computation on data sharing scheme for

- privacy computing,” *Journal of Parallel and Distributed Computing*, vol. 135, pp. 169–176, 2020.
- [15] Z. Zhang, W. Zhang, and Z. Qin, “Fully constant-size CP-ABE with privacy-preserving outsourced decryption for lightweight devices in cloud-assisted IoT,” *Security and Communication Networks*, vol. 2021, no. 6676862, pp. 1–16, 2021.
 - [16] G. Yu, X. Ma, Z. Cao, G. Zeng, and W. Han, “Accountable CP-ABE with public verifiability: how to effectively protect the outsourced data in cloud,” *International Journal of Foundations of Computer Science*, vol. 28, no. 6, pp. 705–723, 2017.
 - [17] R. Zhang, L. Hui, S. Yiu, X. Yu, L. Zou, and Jiang, “A traceable outsourcing CP-ABE scheme with attribute revocation,” in *Proceedings of the 2017 16th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/11th IEEE International Conference on Big Data Science and Engineering/14th IEEE International Conference on Embedded Software and Systems*, pp. 363–370, IEEE, Sydney, Australia, August 2017.
 - [18] 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, Article ID 102435, 2020.
 - [19] D. X. Song, D. Wagner, and P. Adrian, “Practical techniques for searches on encrypted data,” in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 44–55, Berkeley, California, USA, May 2002.
 - [20] N. Vaanchig, Z. G. Qin, and Z. Qin, “Public key encryption with temporary and fuzzy keyword search,” *Mathematical Biosciences and Engineering*, vol. 16, no. 5, pp. 3914–3935, 2019.
 - [21] Y. Zhou, S. Zheng, and L. Wang, “Privacy-preserving and efficient public key encryption with keyword search based on CP-ABE in cloud,” *Cryptography*, vol. 4, no. 28, p. 28, 2020.
 - [22] H. Wang, J. Ning, X. Huang, G. Wei, G. Sen Poh, and X. Liu, “Secure fine-grained encrypted keyword search for e-healthcare cloud,” *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1307–1319, 2021.
 - [23] Y. Miao, R. Deng, X. Liu, K.-K. Raymond Choo, H. Wu, and H. Li, “Multi-authority attribute-based keyword search over encrypted cloud data,” *IEEE Transactions on Dependable and Secure Computing*, vol. 18, pp. 1667–1680, 2019.
 - [24] H. Yin, Z. Qin, J. Zhang, H. Deng, F. Li, and K. Li, “A fine-grained authorized keyword secure search scheme with efficient search permission update in cloud computing,” *Journal of Parallel and Distributed Computing*, vol. 135, pp. 56–69, 2020.
 - [25] J. Cui, H. Zhou, Y. Xu, and H. Zhong, “OOABKS: online/offline attribute-based encryption for keyword search in mobile cloud,” *Information Sciences*, vol. 489, pp. 63–77, 2019.
 - [26] J. Sun, L. Ren, S. Wang, and X. Yao, “Multi-keyword searchable and data verifiable attribute-based encryption scheme for cloud storage,” *IEEE Access*, vol. 7, pp. 66655–66667, 2019.
 - [27] J. Long, K. Zhang, X. Wang, and H.-N. Dai, “Lightweight distributed attribute based keyword search system for internet of things,” *Security, Privacy, and Anonymity in Computation, Communication, and Storage*, vol. 11637, pp. 253–264, 2019.
 - [28] B. Amos, *Secure Schemes for Secret Sharing and Key Distribution*, PhD thesis, Israel Institute of Technology Sivan, Haifa, 1996.
 - [29] S. Wang, S. Jia, and Y. Zhang, “Verifiable and multi-keyword searchable attribute-based encryption scheme for cloud storage,” *IEEE Access*, vol. 7, pp. 50136–50147, 2019.