This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

Efficient Policy-Hiding and Large Universe Attribute-Based Encryption with Public Traceability for Internet of Medical Things

Peng Zeng, *Member, IEEE*, Zhiting Zhang, Rongxing Lu, *Senior Member, IEEE*, and Kim-Kwang Raymond Choo, *Senior Member, IEEE*

***Abstract*—Modern day medical systems are closely integrated and interconnected with other systems, such as those comprising Internet of Medical Things (IoMT) devices, for example to facilitate remote healthcare services during pandemics (e.g., COVID-19). Attribute-based encryption (ABE) is a promising cryptographic primitive to support fine-grained access control in the ciphertext environment; in other words, ABE can potentially be used to ensure data confidentiality and user privacy in the IoMT ecosystem. In this paper, we propose an efficient partially-policy-hidden and large universe ABE scheme with public traceability to construct a practical IoMT system (hereafter referred to as PTIoMT). The system is designed to achieve the following features: 1) the access policy is partially hidden: only nonsensitive attribute labels/names are displayed, while sensitive attribute values are hidden in the encrypted electronic health records (EHRs); 2) the number of the attributes is independent of the public parameters and thus can be arbitrarily large; 3) any user who discloses the decryption key can be efficiently tracked; and 4) fewer bilinear pairing operations are required during the decryption process. The security analysis and performance evaluation demonstrate the security and efficiency of PTIoMT.**

***Index Terms*—Mobile health, access control, ciphertext-policy attribute-based encryption, full public traceability, privacy-preserving, Internet of Medical Things**

I. Introduction

Rapid advances in Internet of Things (IoT) devices and systems [1–3] and electronic health systems, as well as other supporting technologies and infrastructures (e.g., 5G), have contributed to an increasingly digitalized and interconnected society. Medical IoT, or Internet of Medical Things (IoMT) [4–7], is also increasingly commonplace. There are many benefits associated with the deployment of IoMT-based systems, such as enabling medical practitioners to diagnose patients remotely and in real-time, and real-time sharing and accessing of patients' electronic health records (EHRs) [8]. The importance of IoMT-based systems is reinforced in the COVID-19 related lock-down and stay-at-home regime. Data collected/sent from IoMT devices may contain personal and sensitive information, such as the patient's medical and family history, diagnosis, allergy, and medication. Given the sensitivity of such information, there is a clear need to ensure the confidentiality of EHRs even in authoritarian countries. Hence, there have been significant efforts in designing solutions, such as those based on attribute-based encryption (ABE) [9] to facilitate fine-grained access control [10] and data sharing [11] features in IoMT ecosystem.

ABE can be categorized into key-policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE) [12], in terms of the different forms of ciphertext and key expressions. Specifically, in CP-ABE, the access formula is embedded in the ciphertext and the decryption key of each user is related to an attribute set. A user is allowed to access a ciphertext only if his/her attribute set matches the ciphertext's access formula. Although conventional CP-ABE can be used to achieve fine-grained access control, it is generally not suitable for deployment in IoMT-based system due to challenges such as the partial hiding policy problem and the large universe problem [13].

Fig. 1 depicts a typical IoMT-based system based on conventional CP-ABE, where a data owner uploads his/her encrypted EHRs with the access structure ({Hospital: People's hospital} AND {Occupation: Psychologists}) OR ({ID Number: 116-728-682})[1]. Since the access structure is embedded in the encrypted EHR, anyone who obtain the ciphertext could also infer that the user with ID number 116-728-682 suffers from mental illness. This is a clear violation of patient privacy, and we refer to this as the partial hiding policy problem.

The size of the attribute space directly affects the expression ability of ABE, which is closely related to the access control capability of ciphertext. In conventional CP-ABE schemes, all attributes are determined when the public parameters are established and no further attributes can be added subsequently. In many application scenarios, however, the attributes available

The work is supported in part by the National Natural Science Foundation of China (NSFC) under Grant Nos. 62072184, 61601129, the National Key R&D Program of China under Grant No. 2017YFB0802302, the NSFC-Zhejiang Joint Fund for the Integration of Industrialization and Informatization under Grant No. U1509219, the Key Lab of Information Network Security of Ministry of Public Security (The Third Research Institute of Ministry of Public Security) under Grant No. C18603, the Shanghai Natural Science Foundation under Grant No. 17ZR1408400. K.-K. R. Choo is supported only by the Cloud Technology Endowed Professorship. (Corresponding authors: Zhiting Zhang and Kim-Kwang Raymond Choo)

P. Zeng and Z. Zhang are with the Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, Shanghai, China (e-mail: pzeng@sei.ecnu.edu.cn, ztzhangecnu@outlook.com).

R. Lu is with the Faculty of Computer Science, University of New Brunswick, Fredericton, NB, Canada (e-mail: rlu1@unb.ca).

K.-K. R. Choo is with the Department of Information Systems and Cyber Security, and the Department of Electrical and Computer Engineering, and Department of Computer Science, University of Texas at San Antonio, San Antonio, TX 78249, USA (e-mail: raymond.choo@fulbrightmail.org).

[1]We consider the case where the data user needs to access his/her EHRs from some public cloud. In addition, to minimize storage overhead, the data owner may delete his/her EHRs data after outsourcing them to the public cloud, and there is a risk that the local data may be lost due to the system or hard disk failure.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

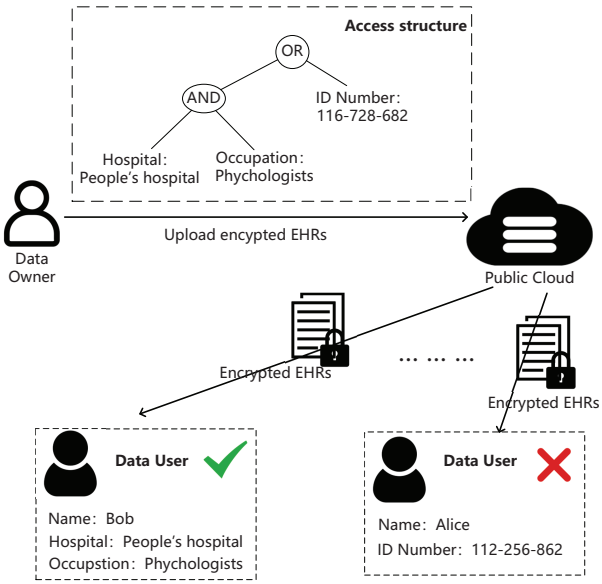

Fig. 1: A typical IoMT architecture with conventional CP-ABE

in the system are expected to be "infinite" (super-multinomial size) and new attributes can be added at any time. This is the so-called large universe problem. In addition, the scale of IoMT-based systems can also be dynamically adjusted and the functional requirements vary regularly. Hence, it is necessary to design a solution that supports large universe to meet the attribute requirement in IoMT.

In addition to the above partial hiding policy and large universe challenges, we also need to consider the key abuse issue. As an example, we assume that there are two users, Alice and Bob, whose respective attribute sets are {Alice, Financial Secretary, Software Engineering} and {Bob, Financial Secretary, Software Engineering}. Thus, Alice and Bob have the same decryption key (denoted by sk) corresponding to {Financial Secretary, Software Engineering}, and both are able to decrypt the ciphertext with the access policy ({Financial Secretary} AND {Software Engineering}). If an unauthorized copy of sk is found in the darkweb or a competitor, how can we determine whether Alice or Bob leaks the information. This is referred to as the key abuse problem. This problem arises because the key has no specific identity information, and is only related to an attribute set owned by several users. Hence, this reinforces the importance of also supporting traceability in CP-ABE, in order to facilitate leakage tracing (and forensic investigation).

Our contributions: In this paper, we propose an IoMT access control system with partial policy hiding and key traceability (hereafter denoted as PTIoMT), designed to ensure data security, user privacy and mitigate key abuse in the IoMT ecosystem. The key building block of PTIoMT is a new policy-hiding and large universe CP-ABE scheme (hereafter denoted as PH-LU-CPABE) with full public traceability, which enables PTIoMT to provide the following essential features:

1) Partial policy hiding: In PTIoMT, each attribute consists of an attribute name and an attribute value. The attribute values carrying sensitive information in the access policy are hidden in the ciphertext, for example see Fig. 2.

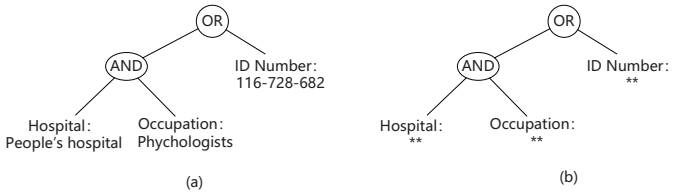

Fig. 2: (a) An access structure; (b) Partially policy-hiding

2) Large universe: PTIoMT has no restriction on the number of attributes. In other words, the attribute universe can be exponential but the size of public parameters remains constant.
3) Efficient test: To enhance decryption efficiency, an efficient decryption test algorithm is utilized in our new PH-LU-CPABE scheme prior to the final decryption of ciphertext.
4) Public traceability: If a decryption key is leaked / abused, anyone can track the owner of the key. We embed the user's identity information into the decryption key as an essential component to the decryption process. Any modification of the identity information shall result in the failure of decryption.
5) Expressiveness and Security: PTIoMT is a flexible and secure IoMT-based system since the underlying PH-LU-CPABE scheme supports flexible access policies and achieves full security under the standard model.

The rest of this paper is organized as follows. In the next section, we will briefly review the extant literature, prior to presenting the preliminaries in Section III. Then, the system architecture and security model are presented in Section IV. In Section V, we present our PTIoMT system, whose security and performance analysis are presented in Sections VI and VII, respectively. We conclude this paper in Section VIII.

II. Related Work

In 2005, Sahai and Waters introduced the concept of fuzzy identity encryption [9], and a year later Goyal et al. [12] proposed a concrete ABE scheme based on monotone access policy in which ABE was further divided into KP-ABE and CP-ABE. In 2007, Bethencourt et al. [14] presented the first CP-ABE scheme based on the generic group model. Since then, a number of CP-ABE schemes have been proposed in the literature [15–18]. In these schemes, ciphertext is generally associated with the access policy specified by the encryptor and the decryption key is associated with the user's attribute set. Access is granted only when the attribute set in the decryption key satisfies the access formula associated with the ciphertext. However, these schemes do not take into account the disclosure of the attribute privacy and thus they are unlikely to be candidates for IoMT deployment.

To protect user's attribute privacy, several CP-ABE schemes with the policy hidden property have been designed [19–25]. Based on Waters's scheme [15] and bloom filter technique [26], for example, Yang et al. [25] presented an access control mechanism to protect privacy. However, no formal security

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

TABLE I: Summary of notations

Notation	Description
$[\ell]$	The set $\{1, 2, \ldots, \ell\}$
$\{x_i\}_{i\in I}$	Another representation of the set $\{x_i : i \in I\}$
$\lvert X\rvert$	The cardinality of the set X
$x \xleftarrow{R} X$	The operation of selecting an element x from set X uniformly at random
$\mathcal{S} = (\mathcal{I}_S, S)$	An attribute set consisting of the attribute name set $\mathcal{I}_S$ and the attribute value set S
$\mathsf{sk}_{\mathsf{id},\mathcal{S}}$	A decryption key associated with the identity id and the attribute set $\mathcal{S}$
$\mathbb{A} = (A, \rho, \mathcal{T})$	An access policy with share generation matrix A, mapping ρ from the rows of A to attribute name space, and authorized value set $\mathcal{T}$
$\mathsf{CT}_{\mathbb{A}}$	A ciphertext under access policy $\mathbb{A}$
$\mathcal{X}_{A,\rho}$	The collection of minimum authorized sets on (A, ρ)

proof was presented. More recently in 2018, Zhang et al. [13] presented a partially attribute hidden CP-ABE scheme and proved its full security based on the dual system encryption technology. Zhang et al.'s scheme, however, does not support traceability and thus suffers from the key abuse problem.

There are a large number of CP-ABE schemes in the literature [27–35], which are designed to mitigate key abuse. For example, Li et al. [27] introduced the concept of accountable CP-ABE to avoid unauthorized key sharing and Katz et al. [28] introduced the concept of traceability for predicate encryption system. However, Li et al.'s scheme only supports the ciphertext policy of AND gate and Katz et al.'s scheme incurs additional overhead that is linear to the number of system users (in order to achieve traceability). In order to reduce computation cost, Liu et al. [29, 30] presented a white-box traceable scheme and a black-box traceable CP-ABE scheme. However, neither scheme is practical for IoMT deployment because the white-box traceable scheme requires an additional table to record user identities and the black-box traceable CP-ABE scheme has relatively large public parameter and ciphertext sizes. Ning et al. [31, 32] presented two CP-ABE schemes that support both traceability and large universe. In addition, Ning et al. [33] proposed a white-box traceable CP-ABE scheme based on non-interactive commitments. However, these three schemes do not support decryption testing or access policy hiding. Hahn et al. [34] and Wu et al. [35] presented two CP-ABE schemes with policy-hiding and traceability, but both schemes do not support decryption testing and large universe. To the best of our knowledge, there is no known CP-ABE scheme in the literature that addresses both key abuse and policy hiding while supporting large universe and decryption test simultaneously. As explained earlier, it is crucial for a practical CP-ABE system with a large number of IoT (and IoMT) devices to have all these features.

III. Preliminaries

Table I summarizes the notations used in this paper.

A. Basic Concepts

Definition 1 (Access Policy [36]): Let $\mathcal{U}$ be an attribute universe. A collection $\mathbb{A} \subseteq 2^{\mathcal{U}}$ is called monotone if $\forall\ \mathcal{X} \in \mathbb{A}$ and $\mathcal{Y} \in 2^{\mathcal{U}}$: $\mathcal{X} \subseteq \mathcal{Y}$ implies $\mathcal{Y} \in \mathbb{A}$. An (monotone) access structure on $\mathcal{U}$ is a (monotone) collection $\mathbb{A}$ with $\mathbb{A} \subseteq 2^{\mathcal{U}} \backslash \{\emptyset\}$.

Definition 2 (Linear Secret Sharing Scheme (LSSS) [21]): Let $\mathcal{S} = (\mathcal{I}_S, S)$ be an attribute set in which $\mathcal{I}_S \subseteq \mathbb{Z}_N$ is the attribute name set and $S = \{s_i : i \in \mathcal{I}_S\}$ is the corresponding attribute value set. Assume that $\mathbb{A} = (A, \rho, \mathcal{T})$ is an access policy, where A is an ℓ by n share generation matrix, ρ is a mapping from the set $[\ell]$ to the attribute name space $\mathbb{Z}_N$, and $\mathcal{T} = (t_{\rho(1)}, t_{\rho(2)}, \ldots, t_{\rho(\ell)})$ is an ℓ-dimensional attribute value vector that is sensitive, and therefore will be hidden in ciphertext. We say that $\mathcal{S} = (\mathcal{I}_S, S)$ satisfies $\mathbb{A} = (A, \rho, \mathcal{T})$ if there exists a subset $\mathcal{X} \subseteq [\ell]$ such that

1) $\{\rho(x) : x \in \mathcal{X}\} \subseteq \mathcal{I}_S$.
2) $s_{\rho(x)} = t_{\rho(x)}$ for any $x \in \mathcal{X}$.
3) $\mathcal{X}$ is an authorized set on (A, ρ). It means that there are $\lvert\mathcal{X}\rvert$ constants w_x, $x \in \mathcal{X}$, such that $\sum_{x\in\mathcal{X}} w_x A_x = (1, 0, \ldots, 0)$, where A_x is the x-th row of A.

We also call $\mathcal{X}$ a minimum authorized set on (A, ρ) if $\mathcal{X}$ is an authorized set on (A, ρ) and no any proper subset $\mathcal{X}'$ of $\mathcal{X}$ satisfies this condition. We denote by $\mathcal{X}_{A,\rho}$ the set of all the minimum authorized sets on (A, ρ).

Definition 3 (Composite Order Bilinear Groups [37]): Let κ be a security parameter and BGGen a bilinear group generator that takes 1^κ as input and outputs a four-tuple $(N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e})$, where p_1, p_2, p_3, p_4 are different prime numbers, $\mathbb{G}, \mathbb{G}_T$ are two cyclic groups of the same composite order $N = p_1p_2p_3p_4$, and $\hat{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a bilinear mapping with the following properties:

1) Bilinear: $\hat{e}(x^a, y^b) = \hat{e}(x, y)^{ab}$ for any $x, y \in \mathbb{G}$ and $a, b \in \mathbb{Z}_N$.
2) Non-degenerate: $\exists\ g \in \mathbb{G}$ such that $\hat{e}(g, g)$ is a generator of $\mathbb{G}_T$.
3) Computable: the operations in $\mathbb{G}, \mathbb{G}_T$ and the bilinear mapping $\hat{e}$ are efficiently computable.

Let $\mathbb{G}_{p_i}$ be the subgroup of $\mathbb{G}$ of order p_i, $1 \le i \le 4$, then we have the so-called "orthogonal" properties: $\forall\ X_i \in \mathbb{G}_{p_i}$, $X_j \in \mathbb{G}_{p_j}$ with $1 \le i \ne j \le 4$, it has $\hat{e}(X_i, X_j) = 1_{\mathbb{G}_T}$.

B. Complexity Assumptions

The security of our proposed PH-LU-CPABE scheme is based on the four complexity assumptions as described below (see also [38, 39]). In the following, we assume that κ is a security parameter and $\mathsf{BGGen}(1^\kappa) \to (N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e})$.

Assumption 1: Let $g \xleftarrow{R} \mathbb{G}_{p_1}$, $x_3 \xleftarrow{R} \mathbb{G}_{p_3}$, $x_4 \xleftarrow{R} \mathbb{G}_{p_4}$, $T_1 \xleftarrow{R} \mathbb{G}_{p_1p_2}$, $T_2 \xleftarrow{R} \mathbb{G}_{p_1}$. Set $D = (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g, x_3, x_4)$. We say that BGGen satisfies the Assumption 1 if the advantage

$$\mathrm{Adv}^1_{\mathsf{BGGen},\mathcal{A}}(\kappa) = |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

is negligible for any probabilistic polynomial time (PPT) attacker $\mathcal{A}$.

Assumption 2: Let $g, x_1 \xleftarrow{R} \mathbb{G}_{p_1}$, $x_2, y_2 \xleftarrow{R} \mathbb{G}_{p_2}$, $x_3, y_3 \xleftarrow{R} \mathbb{G}_{p_3}, x_4 \xleftarrow{R} \mathbb{G}_{p_4}$, $T_1 \xleftarrow{R} \mathbb{G}_{p_1p_2p_3}$, $T_2 \xleftarrow{R} \mathbb{G}_{p_1p_3}$. Set $D = (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g, x_1x_2, y_2y_3, x_3, x_4)$. We say that BGGen satisfies the Assumption 2 if the advantage

$$\mathrm{Adv}^2_{\mathsf{BGGen},\mathcal{A}}(\kappa) = |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

is negligible for any PPT attacker $\mathcal{A}$.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

Assumption 3: Let $\alpha, s \in \mathbb{Z}_N$, $g \xleftarrow{R} \mathbb{G}_{p_1}$, $g_2, x_2, y_2 \xleftarrow{R} \mathbb{G}_{p_2}$, $x_3 \xleftarrow{R} \mathbb{G}_{p_3}$, $x_4 \xleftarrow{R} \mathbb{G}_{p_4}$, $T_1 = \hat{e}(g,g)^{\alpha s}$, $T_2 \xleftarrow{R} \mathbb{G}_T$. Set $D = (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g, g_2, g^{\alpha} x_2, g^s y_2, x_3, x_4)$. We say that BGGen satisfies the Assumption 3 if the advantage

$$\mathrm{Adv}^3_{\mathsf{BGGen},\mathcal{A}}(\kappa) = |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

is negligible for any PPT attacker $\mathcal{A}$.

Assumption 4: Let $r, s \in \mathbb{Z}_N$, $g, h \xleftarrow{R} \mathbb{G}_{p_1}$, $g_2, x_2, a_2, b_2, d_2 \xleftarrow{R} \mathbb{G}_{p_2}$, $x_3 \xleftarrow{R} \mathbb{G}_{p_3}$, $x_4, z, a_4, d_4 \xleftarrow{R} \mathbb{G}_{p_4}$, $T_1 = h^r a_2 a_4$, $T_2 \xleftarrow{R} \mathbb{G}_{p_1 p_2 p_4}$. Set $D = (N, \mathbb{G}, \mathbb{G}_T, \hat{e}, g, g_2, g^s b_2, h^s x_2, x_3, x_4, hz, g^r d_2 d_4)$. We say that BGGen satisfies the Assumption 4 if the advantage

$$\mathrm{Adv}^4_{\mathsf{BGGen},\mathcal{A}}(\kappa) = |\Pr[\mathcal{A}(D, T_1) = 1] - \Pr[\mathcal{A}(D, T_2) = 1]|$$

is negligible for any PPT attacker $\mathcal{A}$.

IV. System Architecture and Security Model

A. System Architecture

Fig. 3 shows the system architecture of our proposed PTIoMT, which consists of four entities: data owner (DO), data user (DU), public cloud[2], and attribute authority (AA).

- In our PTIoMT system, DO is a patient whose EHRs are generated by some wireless body sensor network (WBSN) or some IoMT devices. DO encrypts his/her EHRs using an efficient symmetric encryption algorithm (e.g., AES) with a symmetric key $\hat{k}$. Then the key $\hat{k}$ is encrypted by our proposed CP-ABE scheme PH-LU-CPABE with an access policy $\mathbb{A}$. Finally, the ciphertexts of both the EHRs and the key $\hat{k}$ are outsourced to some public cloud server.
- DU may be a medical researcher/practitioner who needs to access DO's EHRs in the public cloud. DU has a decryption key associated with his/her attribute set $\mathcal{S}$. If $\mathcal{S}$ satisfies the access policy $\mathbb{A}$ embedded in the ciphertext of $\hat{k}$, then DU can successfully recover $\hat{k}$ and uses it to decrypt the encrypted EHRs.
- Public cloud is a semi-trusted third party (i.e. honest but curious). While it executes all instructions as per system specification, it may also attempt to extract sensitive information from the stored EHRs. We assume that the public cloud has adequate storage capacity to store encrypted EHRs.
- AA is a fully trusted third party, which is responsible to generate the system public parameters and the private keys for the system users. In practice, AA can be a government agency or an trusted third-party entity.

Our proposed PTIoMT system has the following five phases, namely: Initialization, User Registration, Privacy-Aware EHR Outsourcing, Privacy-Aware EHR Access and Traitor Tracing (see Section V):

- Initialization: AA computes the public parameters par and a master key msk for the system. Each user is allowed to own par.
- User Registration: A user can join PTIoMT by submitting his/her identity id and attribute set $\mathcal{S}$ to AA. Then, AA computes a private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ according to the pair $(\mathsf{id}, \mathcal{S})$ and sends $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ to the user via a secure channel. The user can be either a data owner or a data user.
- Privacy-Aware EHR Outsouring: When the EHRs are generated from a WBSN or some IoMT devices, they will be encrypted prior to outsourcing. Specially, DO first selects an efficient symmetric encryption algorithm (e.g. AES) to encrypt the EHRs. Then the involved symmetric key $\hat{k}$ is encrypted by our proposed PH-LU-CPABE scheme with a specified access policy $\mathbb{A}$. Finally, the ciphertexts of the EHRs and $\hat{k}$, denoted respectively by $\langle \mathrm{EHRs} \rangle_{\hat{k}}$ and $\mathsf{CT}_{\mathbb{A}}$, are outsourced by DO to the public cloud. The sensitive attribute value set of $\mathbb{A}$ is hidden in $\mathsf{CT}_{\mathbb{A}}$.
- Privacy-Aware EHR Access: DU downloads both ciphertexts $\langle \mathrm{EHRs} \rangle_{\hat{k}}$ and $\mathsf{CT}_{\mathbb{A}}$ from the public cloud. Then, DU performs a decryption test to check whether his/her private key matches the access policy $\mathbb{A}$ embedded in $\mathsf{CT}_{\mathbb{A}}$. If yes, DU can decrypt $\mathsf{CT}_{\mathbb{A}}$ to obtain the symmetric key $\hat{k}$ and subsequently uses it to decrypt $\langle \mathrm{EHRs} \rangle_{\hat{k}}$ to get the corresponding EHRs.
- Traitor Tracing: When a private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ is found to be leaked or used inappropriately (e.g., accessing some patient's information without authorization), anyone can determine whether $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ is well-formed. If yes, then the offender can be determined via the identity id inverted in $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$; otherwise, it does not need to be traced.

B. Security Model

As mentioned in Section I, PTIoMT is mainly based on our newly proposed PH-LU-CPABE scheme, which comprises the following six algorithms:

- $\mathsf{Setup}(1^{\kappa}) \rightarrow (\mathsf{par}, \mathsf{msk})$: Given as input a security parameter κ, the algorithm outputs the system public parameters par and a master key msk.
- $\mathsf{KeyGen}(\mathsf{par}, \mathsf{msk}, \mathsf{id}, \mathcal{S}) \rightarrow \mathsf{sk}_{\mathsf{id},\mathcal{S}}$: Given as input the system public parameters par, the master key msk, an identity id and an attribute set $\mathcal{S}$ of some user, the algorithm outputs a private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$.
- $\mathsf{Encrypt}(\mathsf{par}, M, \mathbb{A}) \rightarrow \mathsf{CT}_{\mathbb{A}}$: Given as input the system public parameters par, a message M, and an access policy $\mathbb{A} = (A, \rho, \mathcal{T})$, the algorithm returns the ciphertext $\mathsf{CT}_{\mathbb{A}}$ of the message M associated with $\mathbb{A}$. To achieve privacy preserving in PH-LU-CPABE, the attribute value vector $\mathcal{T}$ is not embedded in $\mathsf{CT}_{\mathbb{A}}$.
- $\mathsf{Decrypt}(\mathsf{par}, \mathsf{CT}_{\mathbb{A}}, \mathsf{sk}_{\mathsf{id},\mathcal{S}}) \rightarrow M$ or $\perp$: Given as input the system public parameters par, the ciphertext $\mathsf{CT}_{\mathbb{A}}$ associated with an access formula $\mathbb{A}$, and a private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$, the algorithm outputs a message M if $\mathcal{S}$ satisfies $\mathbb{A}$. Otherwise, it outputs an error symbol $\perp$.
  - **Decryption test**: If $\mathcal{S}$ does not satisfy the access policy $\mathbb{A}$, it returns $\perp$ and terminates the decryption algorithm Decrypt. Otherwise, Decrypt will continue executing.
  - **Full decryption**: It outputs a message M.

[2]This can also be a community cloud (e.g. Texas medical cloud cluster) or a private cloud.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

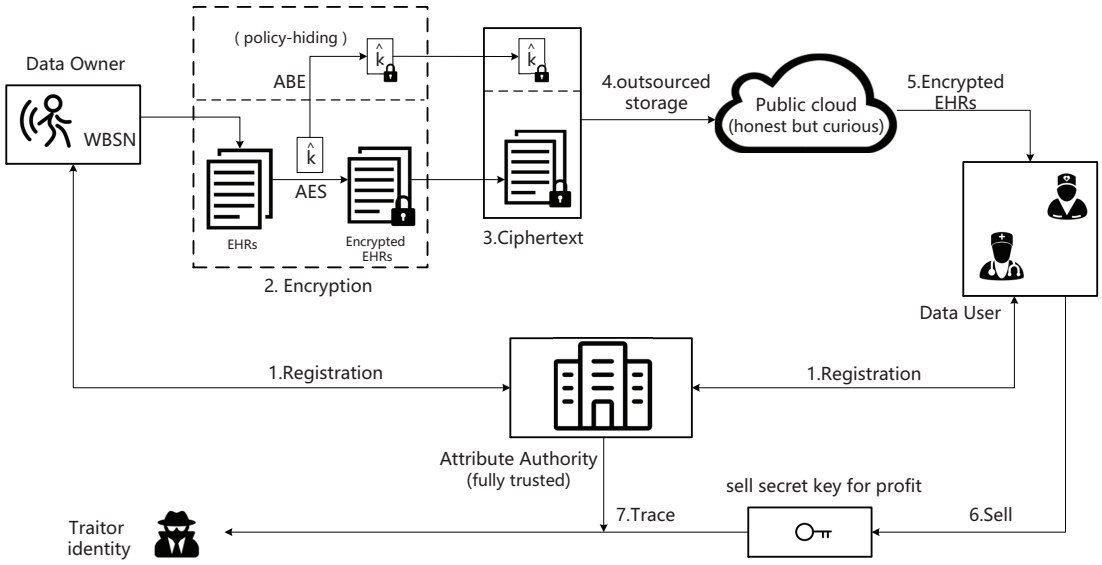

Fig. 3: PTIoMT system architecture

- $\mathsf{KeySanityCheck}(\mathsf{par}, \mathsf{sk}_{\mathsf{id},\mathcal{S}}) \rightarrow 1$ or 0: The algorithm takes as input the system public parameters par and a private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$. If $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ satisfies the key sanity check, it returns 1 (implying $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ is well-formed). Otherwise, it returns 0. KeySanityCheck is a deterministic algorithm [40, 41], and it is used to ensure the regularity of $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ before executing the traitor trace algorithm.
- $\mathsf{Trace}(\mathsf{par}, \mathsf{sk}_{\mathsf{id},\mathcal{S}}) \rightarrow \mathsf{id}$ or $\perp$: Given as input the system parameters par and a private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$, it first calls the algorithm KeySanityCheck to check whether $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ is well-formed. If yes, it extracts the owner identity id from $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$. Otherwise, it outputs $\perp$ which means that $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ does not need to be traced.

Let $\mathcal{E} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt}, \mathsf{KeySanityCheck}, \mathsf{Trace})$ be a PH-LU-CPABE scheme and $\mathcal{A}$ (resp. $\mathcal{B}$) a PPT attacker (resp. challenger) for $\mathcal{E}$. The security model for $\mathcal{E}$ is similar to the one in [13], except that each key query is associated with a specific identity. Specially, we consider the following security game, denoted by $\mathsf{Game}^{\mathsf{full}}_{\mathcal{E},\mathcal{A}}$.

1) Initialization: Given a security parameter κ, the challenger $\mathcal{B}$ executes $\mathsf{Setup}(1^\kappa)$ to generate the system public parameters par and the master key msk. Then $\mathcal{B}$ sends par to the attacker $\mathcal{A}$ and saves msk secretly.
2) Query-1: $\mathcal{A}$ adaptively queries the private keys for Q_1 identity and attribute set pairs $(\mathsf{id}_1, \mathcal{S}_1), (\mathsf{id}_2, \mathcal{S}_2), \ldots, (\mathsf{id}_{Q_1}, \mathcal{S}_{Q_1})$. For each pair $(\mathsf{id}_i, \mathcal{S}_i)$, $1 \leq i \leq Q_1$, $\mathcal{B}$ runs the algorithm $\mathsf{KeyGen}(\mathsf{par}, \mathsf{msk}, \mathsf{id}_i, \mathcal{S}_i)$ to get $\mathsf{sk}_{\mathsf{id}_i,\mathcal{S}_i}$, and sends $\mathsf{sk}_{\mathsf{id}_i,\mathcal{S}_i}$ to $\mathcal{A}$.
3) Challenge: $\mathcal{A}$ submits two messages M_i of the same size and two access policies $\mathbb{A}_i = (A, \rho, \mathcal{T}_i)$ to $\mathcal{B}$, $i = 0, 1$. Then $\mathcal{B}$ chooses a bit $\beta \xleftarrow{R} \{0, 1\}$ and returns $\mathsf{Encrypt}(\mathsf{par}, M_\beta, \mathbb{A}_\beta) \rightarrow \mathsf{CT}_{\mathbb{A}_\beta}$ to $\mathcal{A}$. Note that in this phase, the Q_1 attribute sets $\mathcal{S}_i$, $1 \leq i \leq Q_1$, in Query-1 are not allowed to satisfy $\mathbb{A}_0$ or $\mathbb{A}_1$.
4) Query-2: Similar to Query-1, $\mathcal{A}$ continues to query the private keys for the identity and attribute set pairs $(\mathsf{id}_{Q_1+1}, \mathcal{S}_{Q_1+1}), (\mathsf{id}_{Q_1+2}, \mathcal{S}_{Q_1+2}), \ldots, (\mathsf{id}_Q, \mathcal{S}_Q)$. $\mathcal{B}$ generates the corresponding private keys $\mathsf{sk}_{\mathsf{id}_i,\mathcal{S}_i}$ to $\mathcal{A}$ by running the algorithm $\mathsf{KeyGen}(\mathsf{par}, \mathsf{msk}, \mathsf{id}_i, \mathcal{S}_i)$, $Q_1 + 1 \leq i \leq Q$. Also, these attribute sets are not allowed to satisfy $\mathbb{A}_0$ or $\mathbb{A}_1$.
5) Guess: $\mathcal{A}$ outputs a guess bit $\beta' \in \{0, 1\}$. If $\beta' = \beta$, $\mathcal{A}$ wins the game $\mathsf{Game}^{\mathsf{full}}_{\mathcal{E},\mathcal{A}}$ and we denote the event by $\mathsf{Succ}^{\mathsf{full}}_{\mathcal{E},\mathcal{A}}$.

We say that $\mathcal{E}$ is fully secure if for any PPT attacker $\mathcal{A}$, the advantage

$$\mathsf{Adv}^{\mathsf{full}}_{\mathcal{E},\mathcal{A}}(\kappa) = \left| \Pr[\mathsf{Succ}^{\mathsf{full}}_{\mathcal{E},\mathcal{A}}] - 1/2 \right|$$

is negligible with respect to the security parameter κ.

V. Proposed PTIoMT system

In this section, we will present the concrete construction of our PTIoMT system. The system comprises the following five phases: Initialization, User Registration, Privacy-Aware EHR Outsourcing, Privacy-Aware EHR Access, and Traitor Tracing. We also remark that the six algorithms (i.e., Setup, KeyGen, Encrypt, Decrypt, KeySanityCheck, and Trace) in our PTIoMT system form a new partially-policy-hidden and large universe CP-ABE scheme, that is our PH-LU-CPABE scheme.

A. Initialization

AA selects a security parameter κ and executes the bilinear group generator $\mathsf{BGGen}(1^\kappa)$ to generate $(N = p_1p_2p_3p_4, \mathbb{G}, \mathbb{G}_T, \hat{e})$. Then, AA sets the attribute universe $\mathcal{U} = \mathbb{Z}_N$ and executes the following Setup algorithm.

• $\mathsf{Setup}(1^\kappa)$: AA selects $\alpha, a \xleftarrow{R} \mathbb{Z}_N$, $g, h \xleftarrow{R} \mathbb{G}_{p_1}$, $Z, X_4 \xleftarrow{R} \mathbb{G}_{p_4}$, and calculates $Y = \hat{e}(g, g)^\alpha$, $H = hZ$. The system public parameters and the master key are $\mathsf{par} = (N, g, g^a, Y, H, X_4)$ and $\mathsf{msk} = (\alpha, h)$, respectively.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

B. User Registration

Suppose that $U_{\mathsf{id},\mathcal{S}}$ is a system user (DO or DU) who owns the identity id and the attribute set $\mathcal{S} = (\mathcal{I}_S, S)$ with $\mathcal{I}_S \subseteq \mathbb{Z}_N$ and $S = \{s_i\}_{i\in\mathcal{I}_S}$. AA authorizes $U_{\mathsf{id},\mathcal{S}}$ the access right by the following algorithm.

• $\mathsf{KeyGen}(\mathsf{par}, \mathsf{msk}, \mathsf{id}, \mathcal{S})$: AA selects $t \xleftarrow{R} \mathbb{Z}_N$ and $R, R', R_i \xleftarrow{R} \mathbb{G}_{p_3}$, $\forall\ i \in \mathcal{I}_S$. Then, AA generates the private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ for $U_{\mathsf{id},\mathcal{S}}$ as the following:

$$\mathsf{sk}_{\mathsf{id},\mathcal{S}} = (\mathcal{S}, K, K', K'', \{K_i\}_{i\in\mathcal{I}_S}), \tag{1}$$

where $K' = g^t R'$, $K'' = \mathsf{id}$, $K = g^\alpha g^{at\mathcal{H}(K',K'')} R$, $K_i = (g^{s_i} h)^t R_i$, $i \in \mathcal{I}_S$, and $\mathcal{H} : \{0,1\}^* \to \mathbb{Z}_N$ is a secure collision-resistant hash function.

C. Privacy-Aware EHR Outsourcing

As discussed in Section IV-A, we use hybrid encryption for the outsourced EHRs. First, DO selects an efficient symmetric encryption algorithm (e.g., AES) and a symmetric key (denoted by $\hat{k}$) to encrypt his/her EHRs. Then, DO encrypts the symmetric key $\hat{k}$ using the following Encrypt algorithm with a specified access policy. Finally, DO outsources both ciphertexts to the public cloud (see Fig. 3).

• $\mathsf{Encrypt}(\mathsf{par}, M, \mathbb{A})$: Given as input the public parameters $\mathsf{par} = (N, g, g^a, Y, H, X_4)$, a message $M \in \mathbb{G}_T$, and an access policy $\mathbb{A} = (A, \rho, \mathcal{T})$ with an ℓ by n matrix A, a mapping ρ from $[\ell]$ to the attribute name space $\mathbb{Z}_N$, and an attribute value vector $\mathcal{T} = (t_{\rho(1)}, t_{\rho(2)}, \ldots, t_{\rho(\ell)})$ of length ℓ, the algorithm selects two n-dimensional random vectors $v = (s, v_2, \ldots, v_n)$ and $v' = (s', v'_2, \ldots, v'_n)$ over $\mathbb{Z}_N$ and $2\ell+2$ random subgroup elements $Z_0, Z_1 \xleftarrow{R} \mathbb{G}_{p_4}$, $Z_{0,x}, Z_{1,x} \xleftarrow{R} \mathbb{G}_{p_4}$ for $x \in [\ell]$. The ciphertext is calculated as

$$\mathsf{CT}_{\mathbb{A}} = \left((A,\rho), \tilde{C}_0, \hat{C}_0, \tilde{C}_1, \hat{C}_1, \{C_{0,x}, C_{1,x}\}_{x\in[\ell]}\right), \tag{2}$$

where we have $\tilde{C}_0 = Y^{s'}$, $\hat{C}_0 = g^{s'} Z_0$, $\tilde{C}_1 = M \cdot Y^{s-s'}$, $\hat{C}_1 = g^s Z_1$, $C_{0,x} = g^{aA_x\cdot v'} (g^{t_{\rho(x)}} H)^{-s'} Z_{0,x}$, $C_{1,x} = g^{aA_x\cdot v} (g^{t_{\rho(x)}} H)^{-s} Z_{1,x}$.

It should be noted that the sensitive attribute value vector $\mathcal{T}$ is not embedded in $\mathsf{CT}_{\mathbb{A}}$ and thus our PH-LU-CPABE scheme is partially policy-hidden.

D. Privacy-Aware EHR Access

DU downloads the ciphertexts of both the EHRs and the symmetric key $\hat{k}$ from the public cloud. Then, DU uses the following Decrypt algorithm to decrypt the asymmetric ciphertext to obtain $\hat{k}$. After this, DU can use $\hat{k}$ to decrypt the symmetric ciphertext to obtain the EHRs.

• $\mathsf{Decrypt}(\mathsf{par}, \mathsf{CT}_{\mathbb{A}}, \mathsf{sk}_{\mathsf{id},\mathcal{S}})$: Taken as input the public parameters $\mathsf{par} = (N, g, g^a, Y, H, X_4)$, the ciphertext $\mathsf{CT}_{\mathbb{A}} = ((A,\rho), \tilde{C}_0, \hat{C}_0, \tilde{C}_1, \hat{C}_1, \{C_{0,x}, C_{1,x}\}_{x\in[\ell]})$ and a private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}} = (\mathcal{S}, K, K', K'', \{K_i\}_{i\in\mathcal{I}_S})$ with $\mathcal{S} = (\mathcal{I}_S, S)$, the algorithm first computes the set $\mathcal{X}_{A,\rho}$ of all the minimum authorized sets on (A,ρ). Then, it performs the following operations:

1) Test Phase: It checks whether there is a minimum authorized set $\mathcal{X} \in \mathcal{X}_{A,\rho}$ such that $\{\rho(x)\}_{x\in\mathcal{X}} \subseteq \mathcal{I}_S$ and

$$\tilde{C}_0^{-1} = \hat{e}\left(\prod_{x\in\mathcal{X}} C_{0,x}^{w_x}, K'^d\right) \cdot \hat{e}\left(\hat{C}_0, K^{-1} \prod_{x\in\mathcal{X}} K_{\rho(x)}^{w_x\cdot d}\right),$$

where $d = \mathcal{H}(K', K'')$ and w_x, $x \in \mathcal{X}$, are $|\mathcal{X}|$ constants satisfying $\sum_{x\in\mathcal{X}} w_x A_x = (1, 0, \ldots, 0)$. If yes, then it goes into the following Decryption Phase with $\mathcal{X}$ and $\{w_x\}_{x\in\mathcal{X}}$. Otherwise, it outputs the symbol $\perp$ and terminates the decryption process.

2) Decryption Phase: It computes

$$F = \hat{e}\left(\prod_{x\in\mathcal{X}} C_{1,x}^{w_x}, K'^d\right) \cdot \hat{e}\left(\hat{C}_1, K^{-1} \prod_{x\in\mathcal{X}} K_{\rho(x)}^{w_x\cdot d}\right) = Y^{-s}.$$

Then we can get M by computing $M = \tilde{C}_0 \cdot F \cdot \tilde{C}_1$.

We mention that, besides the pair (A,ρ), the ciphertext $\mathsf{CT}_{\mathbb{A}}$ includes two parts, namely: $(\tilde{C}_0, \hat{C}_0, \{C_{0,x}\}_{x\in[\ell]})$ and $(\tilde{C}_1, \hat{C}_1, \{C_{1,x}\}_{x\in[\ell]})$. These two parts are involved in the Test Phase and Decryption Phase, respectively. The former is in fact a redundancy, while the latter is the encryption of message M. If the attribute set $\mathcal{S}$ of DU satisfies the access policy (A,ρ), then the redundant part enables DU to find a concrete minimum authorized set $\mathcal{X}$ and the corresponding coefficients w_x, $x \in \mathcal{X}$. Finally, DU can use the information and his/her private key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ to decrypt the second part and recover the message M.

E. Traitor Tracing

Based on the following KeySanityCheck and Trace algorithms, our proposed PTIoMT system enables any third party to track down the traitor when an unauthorized copy of some private key has been found on the market.

• $\mathsf{KeySanityCheck}(\mathsf{par}, \mathsf{sk}_{\mathsf{id},\mathcal{S}})$: Given as input the public parameters $\mathsf{par} = (N, g, g^a, Y, H, X_4)$ and a key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$, the algorithm first checks whether $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ has the form $((\mathcal{I}_S, S), K, K', K'', \{K_i\}_{i\in\mathcal{I}_S})$, where $\mathcal{I}_S \subseteq \mathbb{Z}_N$, $|S| = |\mathcal{I}_S|$, $K, K' \in \mathbb{G}$ and $s \in \mathbb{Z}_N$ for any $s \in S$, $K_i \in \mathbb{G}$ for any $i \in \mathcal{I}_S$. Then it verifies whether the equation

$$\hat{e}(g, K) = Y \cdot \hat{e}\left(g^a, K'^{\mathcal{H}(K',K'')}\right)$$

holds. If $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ passes both the two checks, it outputs 1 and 0 otherwise.

• $\mathsf{Trace}(\mathsf{par}, \mathsf{sk}_{\mathsf{id},\mathcal{S}})$: Taken as input the public parameters par and a key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$, it first calls the algorithm KeySanityCheck to check the sanity of $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$. If $\mathsf{KeySanityCheck}(\mathsf{par}, \mathsf{sk}_{\mathsf{id},\mathcal{S}}) \to 0$, it outputs $\perp$ which means that $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ is not well-formed. Otherwise, it outputs the identity $K'' = \mathsf{id}$ contained in the key $\mathsf{sk}_{\mathsf{id},\mathcal{S}}$ to be traced.

VI. Security Analysis

Recall that our proposed PTIoMT system is a hybrid encryption paradigm and its security relies mainly on the security of the underlying PH-LU-CPABE scheme. Thus, we focus on the security proof of the latter in this section.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

Theorem 1: Suppose the bilinear group generator BGGen satisfies Assumptions 1, 2, 3 and 4 (see Section III-B). Then, our PH-LU-CPABE scheme is secure in the standard model.

In our security proof, we introduce the notions of semi-functional ciphertext and semi-functional key, based on the approaches in [21, 38].

Semi-functional ciphertext: Let g_2 be a generator of the subgroup $\mathbb{G}_{p_2}$ and we can construct a semi-functional ciphertext as shown below.

• Use the encryption algorithm Encrypt to generate a regular ciphertext (refer to Eq. (2))

$$\mathsf{CT}_{\mathbb{A}} = \left((A,\rho), \tilde{C}_0, \hat{C}_0, \tilde{C}_1, \hat{C}_1, \{C_{0,x}, C_{1,x}\}_{x\in[\ell]}\right).$$

• Pick two vectors $u, u' \xleftarrow{R} \mathbb{Z}_N^n$ and $|\mathcal{I}_S| + 2\ell + 2$ integers $b, b' \xleftarrow{R} \mathbb{Z}_N$, $\gamma_x, \gamma'_x \xleftarrow{R} \mathbb{Z}_N$ for each $x \in [\ell]$, $z_i \xleftarrow{R} \mathbb{Z}_N$ for each $i \in \mathcal{I}_S$. Then the semi-functional ciphertext is of the following form:

$$\mathsf{CT}'_{\mathbb{A}} = \left((A,\rho), \tilde{C}'_0, \hat{C}'_0, \tilde{C}'_1, \hat{C}'_1, \{C'_{0,x}, C'_{1,x}\}_{x\in[\ell]}\right),$$

where we have

$$\begin{cases} \tilde{C}'_0 = \tilde{C}_0, & \hat{C}'_0 = \hat{C}_0 g_2^{b'}, & C'_{0,x} = C_{0,x} g_2^{A_x\cdot u' + \gamma'_x z_{\rho(x)}}, \\ \tilde{C}'_1 = \tilde{C}_1, & \hat{C}'_1 = \hat{C}_1 g_2^{b}, & C'_{1,x} = C_{1,x} g_2^{A_x\cdot u + \gamma_x z_{\rho(x)}}. \end{cases}$$

Semi-functional key: There exist three types of semi-functional keys $\mathsf{sk}^{\mathsf{j}}_{\mathsf{id},\mathcal{S}}$, $j = 1, 2, 3$. To construct them, we first use the algorithm KeyGen to generate a regular key

$$\mathsf{sk}_{\mathsf{id},\mathcal{S}} = (\mathcal{S}, K, K', K'', \{K_i\}_{i\in\mathcal{I}_S})$$

for an attribute set $\mathcal{S} = (\mathcal{I}_S, S)$ (refer to Eq. (1)). Then, we select random numbers $d, d', d_i \xleftarrow{R} \mathbb{Z}_N$ for $i \in \mathcal{I}_S$. The semi-functional key of Type 1 can be defined as:

$$\mathsf{sk}^1_{\mathsf{id},\mathcal{S}} = \left(\mathcal{S}, Kg_2^d, K'g_2^{d'}, K'', \{K_i g_2^{d' z_i}\}_{i\in\mathcal{I}_S}\right).$$

The semi-functional key of Type 2 is obtained by deleting the terms $g_2^{d'}$ and $g_2^{d' z_i}$ in $\mathsf{sk}^1_{\mathsf{id},\mathcal{S}}$. That is, we set

$$\mathsf{sk}^2_{\mathsf{id},\mathcal{S}} = \left(\mathcal{S}, Kg_2^d, K', K'', \{K_i\}_{i\in\mathcal{I}_S}\right).$$

Similarly, the semi-functional key of Type 3 is of the form:

$$\mathsf{sk}^3_{\mathsf{id},\mathcal{S}} = \left(\mathcal{S}, Kg_2^d, K'g_2^{d'}, K'', \{K_i g_2^{d_i}\}_{i\in\mathcal{I}_S}\right).$$

Based on the different forms of the challenge ciphertexts and the Q query keys, we introduce a series of games in Table II (see [21, 38]).

For simplicity, we denote by $\mathsf{Game}_a \overset{n}{\sim} \mathsf{Game}_b$ the indistinguishability of the two games Game_a and Game_b under Assumption n.

Lemma 1: For the games defined in Table II and the complexity Assumptions 1 to 4, we have the following six relations:

1) **R1**: $\mathsf{Game}_{\mathsf{Real}} \overset{1}{\sim} \mathsf{Game}_0$;
2) **R2**: $\mathsf{Game}_{k-1,3} \overset{2}{\sim} \mathsf{Game}_{k,1}$;
3) **R3**: $\mathsf{Game}_{k,1} \overset{2}{\sim} \mathsf{Game}_{k,2}$;
4) **R4**: $\mathsf{Game}_{k,2} \overset{2}{\sim} \mathsf{Game}_{k,3}$;
5) **R5**: $\mathsf{Game}_{Q,3} \overset{3}{\sim} \mathsf{Game}_{\mathsf{Final}_0}$;

TABLE II: The games to be used in our security proof [21, 38] (N: normal; SF: semi-functional; Type i: the semi-function key of type i, $1 \le i \le 3$)

Game	Ciphertext		Keys			
	N	SF	N	Type 1	Type 2	Type 3
$\mathsf{Game}_{\mathsf{Real}}$	$\surd$		$[Q]$			
Game_0		$\surd$	$[Q]$			
$\mathsf{Game}_{k,1}$		$\surd$	Left	$\{k\}$		$[k-1]$
$\mathsf{Game}_{k,2}$		$\surd$	Left		$\{k\}$	$[k-1]$
$\mathsf{Game}_{k,3}$		$\surd$	Left			$[k]$
$\mathsf{Game}_{\mathsf{Final}_0}$		$\surd^{\dagger}$				$[Q]$
$\mathsf{Game}_{\mathsf{Final}_1}$		$\surd^{\ddagger}$				$[Q]$

†A semi-functional ciphertext of a random message;
‡The elements $C'_{0,x}$ and $C'_{1,x}$ of the semi-functional ciphertext are chosen from $\mathbb{G}_{p_1p_2p_4}$

6) **R6**: $\mathsf{Game}_{\mathsf{Final}_0} \overset{4}{\sim} \mathsf{Game}_{\mathsf{Final}_1}$.

The key and ciphertext structures in our PH-LU-CPABE scheme are similar to those of [13], and there are just slight differences between the process of transforming a normal challenge ciphertext into a semi-functional ciphertext and the process of transforming query keys into semi-functional keys. As a result, we omit the proof of this lemma and interested readers are referred to Lemmas 1–6 in [13].

Now we present a proof sketch of Theorem 1. First, we can transmit the real security game $\mathsf{Game}_{\mathsf{Real}}$ to Game_0 by relation **R1**. Since Game_0 is the same game as $\mathsf{Game}_{0,3}$, and both $\mathsf{Game}_{0,3}$ and $\mathsf{Game}_{1,1}$ are indistinguishable because of relation **R2**, we can deduce that $\mathsf{Game}_{\mathsf{Real}}$ and $\mathsf{Game}_{1,1}$ are indistinguishable. Using relations **R3** and **R4**, we have $\mathsf{Game}_{\mathsf{Real}}$ is indistinguishable to $\mathsf{Game}_{1,3}$. By using repeatedly relations **R2**, **R3**, and **R4** sequentially, we can deduce that $\mathsf{Game}_{\mathsf{Real}}$ is indistinguishable to $\mathsf{Game}_{Q,3}$. Finally, based on relations **R5** and **R6**, it is obvious that $\mathsf{Game}_{\mathsf{Real}}$ and $\mathsf{Game}_{\mathsf{Final}_1}$ are indistinguishable. We also show the derivation path of the indistinguishability between games $\mathsf{Game}_{\mathsf{Real}}$ and $\mathsf{Game}_{\mathsf{Final}_1}$ in Fig. 4.

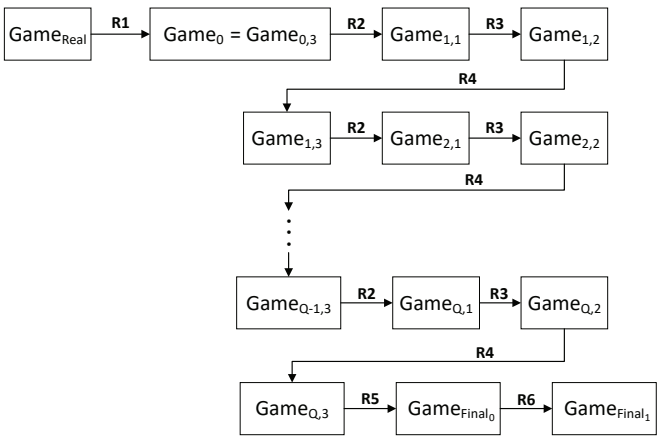

Fig. 4: The derivation path of the indistinguishability between $\mathsf{Game}_{\mathsf{Real}}$ and $\mathsf{Game}_{\mathsf{Final}_1}$

In games $\mathsf{Game}_{\mathsf{Final}_0}$ and $\mathsf{Game}_{\mathsf{Final}_1}$, all the keys are of the form $\mathsf{sk}^3_{\mathsf{id},\mathcal{S}}$ and none of them are effective for decrypting a semi-functional ciphertext. So in both games, the advantage of any attacker is negligible. This, together with the indistin-

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

guishability between $\mathsf{Game}_{\mathsf{Real}}$ and $\mathsf{Game}_{\mathsf{Final}_1}$, imply that the advantage of the attacker in $\mathsf{Game}_{\mathsf{Real}}$ is also negligible.

VII. Performance Evaluation

In this section, our performance evaluation metrics for our proposed PH-LU-CPABE scheme and the existing CP-ABE schemes of [13, 21, 23, 35] are functions, storage cost and computation overhead.

As shown in Table III, all five schemes achieve privacy-preserving, but only the schemes of [13, 21] and our scheme support decryption testing and are secure under the standard model. We also observe that only the schemes of [13, 23] and our scheme support large universe, and that only the scheme of [35] and our scheme achieve traceability. In other words, our scheme is the only one that achieves all six functions (i.e., privacy-preserving, large universe, decryption test, full security, standard model, and traceability).

TABLE III: Function comparison between our scheme and four other competing CP-ABE schemes supporting LSSS
(PP: privacy-preserving; LU: large universe; DT: decryption test; FS: full security; SM: standard model; GO: group order; T: traceability)

Scheme	PP	LU	DT	FS	SM	GO	T
[21]	√	×	√	√	√	composite	×
[23]	√	√	×	×	×	prime	×
[13]	√	√	√	√	√	composite	×
[35]	√	×	×	×	√	∅	√
Our	√	√	√	√	√	composite	√

In Table IV, we evaluate the storage costs of our PH-LU-CPABE scheme and the schemes of [13, 21]. These three schemes are based on the bilinear group of composite order. We observe that the size of the public parameters par in our scheme and the scheme of [13] is a constant number, while the value in the scheme of [21] is linear with the size of the attribute space. For the key size, all three schemes have $k+2$ subgroup elements, where k is the cardinality of the attribute name set $\mathcal{I}_S$. The key advantage of our scheme is that it reduces ℓ and 2ℓ subgroup elements in ciphertext to the schemes of [13] and [21], respectively, where ℓ is the row dimension of the share generation matrix A. Fig. 5 visually shows the advantage of our scheme over the scheme of [13], in terms of ciphertext size.

TABLE IV: Storage cost comparison between our scheme and two other schemes

Scheme	par		$\mathsf{sk}_{\mathsf{id},\mathcal{S}}$	$\mathsf{CT}_{\mathbb{A}}$	
	$\mathbb{G}_{p_i}$	$\mathbb{G}_T$	$\mathbb{G}_{p_ip_j}$	$\mathbb{G}_{p_ip_j}$	$\mathbb{G}_T$
[21]	$N+4$	1	$k+2$	$4\ell+2$	2
[13]	4	1	$k+2$	$3\ell+2$	2
Our	4	1	$k+2$	$2\ell+2$	2

Finally, we consider the computation overhead from three aspects, namely: encryption, decryption test, and decryption. As shown in Table V, the encryption cost of our PH-LU-CPABE scheme has ℓ and 2ℓ fewer exponential operations on $\mathbb{G}$ than those of [13] and [21], respectively. On the other hand, the number of the pair operations in our scheme and the scheme [13] is the constant number 2 during the decryption test process, while the number is $s+2$ in the scheme of [21],

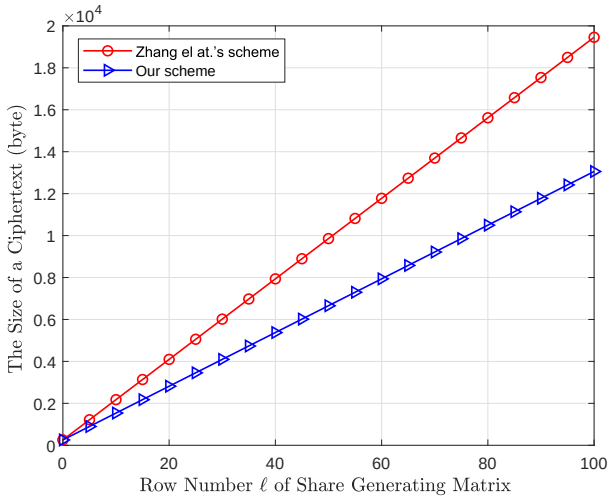

Fig. 5: Ciphertext size comparison between our scheme and the scheme of [13]

where s is the cardinality of the minimum authorized set $\mathcal{X}$. One of the main advantages of our scheme in computation overhead is that our scheme needs only to perform 2 pairing operations during the decryption process, while the number is $s+2$ in the schemes of [13, 21]. In other words, our scheme is more efficient than the schemes of [13, 21], since the pairing operation is much more time-consuming than the exponentiation operations in $\mathbb{G}$ or $\mathbb{G}_T$. To evaluate visually the performance of our scheme in terms of computation overhead, we simulate the encryption and decryption costs in Table V based on Java Pairing-Based Cryptography Library (JPBC) and using a personal computer (Intel(R) Core(TM) i5-8265U CPU @ 1.60GHz 1.80GHz and 8.00GB RAM) (see also Fig. 6 and Fig. 7).

TABLE V: Computation overhead comparison between our scheme and those of [13, 21] with decryption test
(Enc: encryption; DecT: decryption test; Dec: decryption; E/T: exponentiation operation in $\mathbb{G}/\mathbb{G}_T$; P: pairing operation)

Scheme	Enc		DecT			Dec		
	E	T	P	E	T	P	E	T
[21]	$7\ell+2$	2	$s+2$	s	s	$s+2$	s	s
[13]	$6\ell+2$	2	2	$2s$	0	$s+2$	s	s
Our	$5\ell+2$	2	2	$2s+1$	0	2	$2s+1$	0

VIII. Conclusion

In this paper, we proposed a partially-policy-hidden and traceable access control system (PTIoMT), which is designed to secure EHRs, ensure user privacy, and mitigate key abuse in IoTM-based systems. In PTIoMT, our proposed ciphertext-policy attribute-based encryption scheme (PH-LU-CPABE) is the core building block that enables us to support partially hidden access policy, large universe, decryption test, and traceability. In PH-LU-CPABE, only attribute names in an access policy are revealed, while the sensitive attribute values are hidden in the ciphertext. The attribute universe can be unbounded and the size of public parameters is constant.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

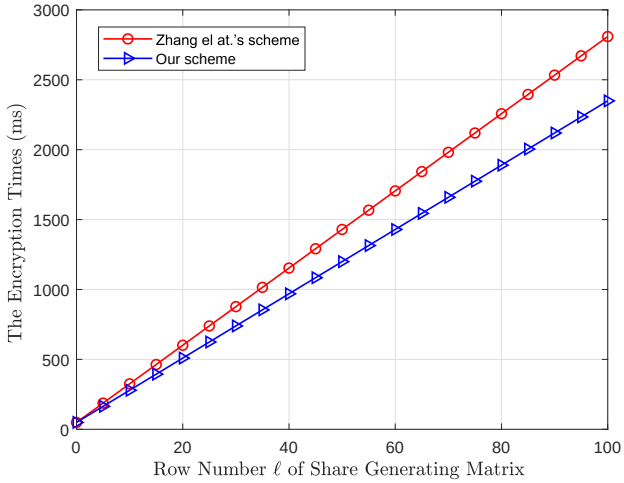

Fig. 6: Encryption cost comparison between our scheme and the scheme of [13]

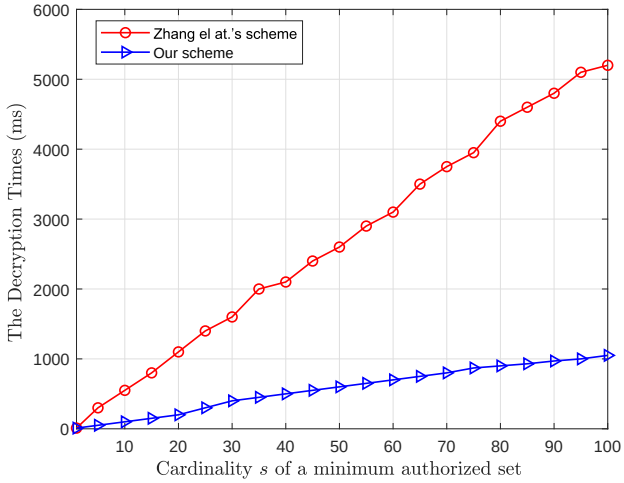

Fig. 7: Decryption cost comparison between our scheme and the scheme of [13]

To facilitate a more efficient decryption operation, PH-LU-CPABE adds a decryption test prior to the final decryption. Finally, PH-LU-CPABE achieves traceability by embedding the user's identity information into the decryption key. We demonstrated that our PH-LU-CPABE scheme is secure in the standard model and achieves better efficiency in terms of ciphertext size and computation overhead. To the best of our knowledge, PH-LU-CPABE is the first CP-ABE scheme that addresses both key abuse and the policy hiding while also supporting large universe and decryption test. A comparative performance evaluation of PTIoMT and the competing CP-ABE schemes presented in [13, 21, 23, 35] suggests the utility of PTIoMT.

In our system architecture, the attribute authority (AA) is assumed to be a fully trusted party. In reality, however, it may not be a reasonable assumption. Hence, one future research direction is to add the auditing function into our PTIoMT system, which is potentially an effective way to reduce the reliance on AA. In addition, there exists the concept of access revocation in ABE, which allows one to revoke the decryption right of the tracked data users. The revocation can occur either at the user level or at the attribute level. The latter (attribute revocation) is generally considered more fine-grained than the former. A natural follow-on question is how to achieve an efficient ABE scheme with large universe, policy-hiding, public traceability, and attribute revocation simultaneously. This is also another potential extension of our research.

REFERENCES

[1] Fei Liu, Chee-Wee Tan, Eric T.K. Lim, Ben Choi (2017) Traversing knowledge networks: an algorithmic historiography of extant literature on the Internet of Things (IoT), Journal of Management Analytics, 4:1, 3-34.

[2] Li, S., et al. 5G Internet of Things: A Survey. Journal of Industrial Information Integration, 10, 1-9, 2018.

[3] Caiming Zhang and Yong Chen. A Review of Research Relevant to the Emerging Industry Trends: Industry 4.0, IoT, Blockchain, and Business Analytics. Journal of Industrial Integration and Management, 5(1), 2020.

[4] Yang, S., et al. Semantic Inference on Clinical Documents: Combining Machine Learning Algorithms with an Inference Engine for Effective Clinical Diagnosis and Treatment. IEEE Access, 5, 3529-3546, 2017.

[5] Qi, J., et al Advanced Internet of Things for Personalized Healthcare Systems: A Survey. Pervasive and Mobile Computing, 41, 132-149, 2017.

[6] Yang, B., et al. Lifelogging Data Validation Model for Internet of Things enabled Healthcare System. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 48(1), 50-64, 2018.

[7] Yang, P., et al. The Internet of Things (IoT): Informatics methods for IoT-enabled health care. Journal of Biomedical Informatics, 87, 154-156, 2018.

[8] Germanakos P, Mourlas C, Samaras G. A mobile agent approach for ubiquitous and personalized eHealth information systems[C]//Proceedings of the Workshop on'Personalization for e-Health'of the 10th International Conference on User Modeling (UM'05). Edinburgh. 2005: 67-70.

[9] Sahai A, Waters B. Fuzzy identity-based encryption[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2005: 457-473.

[10] Zhang Y, Zheng D, Guo R, et al. Fine-grained access control systems suitable for resource-constrained users in cloud computing[J]. Computing and Informatics, 2018, 37(2): 327-348.

[11] Zhang Y, Wu A, Zheng D. Efficient and privacy-preserving attribute-based data sharing in mobile cloud computing[J]. Journal of Ambient Intelligence and Humanized Computing, 2018, 9(4): 1039-1048.

[12] Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data[C]//Proceedings of the 13th ACM conference on

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

Computer and communications security. Acm, 2006: 89-98.

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

[14] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption[C]//2007 IEEE symposium on security and privacy (SP'07). IEEE, 2007: 321-334.

[15] Waters B. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization[C]//International Workshop on Public Key Cryptography. Springer, Berlin, Heidelberg, 2011: 53-70.

[16] Goyal V, Jain A, Pandey O, et al. Bounded ciphertext policy attribute based encryption[C]//International Colloquium on Automata, Languages, and Programming. Springer, Berlin, Heidelberg, 2008: 579-591.

[17] Ibraimi L, Petkovic M, Nikova S, et al. Mediated ciphertext-policy attribute-based encryption and its application[C]//International Workshop on Information Security Applications. Springer, Berlin, Heidelberg, 2009: 309-323.

[18] Jung T, Li X Y, Wan Z, et al. Privacy preserving cloud data access with multi-authorities[C]//2013 Proceedings IEEE INFOCOM. IEEE, 2013: 2625-2633.

[19] Nishide T, Yoneyama K, Ohta K. Attribute-based encryption with partially hidden encryptor-specified access structures[C]//International conference on applied cryptography and network security. Springer, Berlin, Heidelberg, 2008: 111-129.

[20] Lai J, Deng R H, Li Y. Fully secure cipertext-policy hiding CP-ABE[C]//International conference on information security practice and experience. Springer, Berlin, Heidelberg, 2011: 24-39.

[21] Lai J, Deng R H, Li Y. Expressive CP-ABE with partially hidden access structures[C]//Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security, May 2-4, Seoul, Korea. 18-19.

[22] Zhang Y, Chen X, Li J, et al. Anonymous attribute-based encryption supporting efficient decryption test[C]//Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security. ACM, 2013: 511-516.

[23] Cui H, Deng R H, Wu G, et al. An efficient and expressive ciphertext-policy attribute-based encryption scheme with partially hidden access structures[C]//International Conference on Provable Security. Springer, Cham, 2016: 19-38.

[24] Jin C, Feng X, Shen Q. Fully secure hidden ciphertext policy attribute-based encryption with short ciphertext size[C]//Proceedings of the 6th International Conference on Communication and Network Security. ACM, 2016: 91-98.

[25] Yang K, Han Q, Li H, et al. An efficient and fine-grained big data access control scheme with privacy-preserving policy[J]. IEEE Internet of Things Journal, 2017, 4(2): 563-571.

[26] Dong C, Chen L, Wen Z. When private set intersection meets big data: an efficient and scalable protocol[C]//Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security. ACM, 2013: 789-800.

[27] Li J, Ren K, Kim K. A2BE: Accountable Attribute-Based Encryption for Abuse Free Access Control[J]. IACR Cryptology ePrint Archive, 2009, 2009: 118.

[28] Katz J, Schröder D. Tracing insider attacks in the context of predicate encryption schemes[J]. ACITA, 2011.

[29] Liu Z, Cao Z, Wong D S. Blackbox traceable CP-ABE: how to catch people leaking their keys by selling decryption devices on ebay[C]//Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security. ACM, 2013: 475-486.

[30] Liu Z, Cao Z, Wong D S. White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures[J]. IEEE Transactions on Information Forensics and Security, 2013, 8(1): 76-88.

[31] Ning J, Cao Z, Dong X, et al. Large universe ciphertext-policy attribute-based encryption with white-box traceability[C]//European Symposium on Research in Computer Security. Springer, Cham, 2014: 55-72.

[32] Ning J, Dong X, Cao Z, et al. White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(6): 1274-1288.

[33] Ning J, Cao Z, Dong X, et al. White-box traceable CP-ABE for cloud storage service: how to catch people leaking their access credentials effectively[J]. IEEE Transactions on Dependable and Secure Computing, 2018, 15(5): 883-897.

[34] Hahn C, Kwon H, Hur J. Efficient attribute-based secure data sharing with hidden policies and traceability in mobile health networks[J]. Mobile Information Systems, 2016, 2016.

[35] Wu A, Zhang Y, Zheng X, et al. Efficient and privacy-preserving traceable attribute-based encryption in blockchain[J]. Annals of Telecommunications, 2019: 1-11.

[36] Amos Beimel. Secure schemes for secret sharing and key distribution[M]. Technion-Israel Institute of technology, Faculty of computer science, 1996.

[37] Boneh D, Goh E J, Nissim K. Evaluating 2-DNF formulas on ciphertexts[C]//Theory of Cryptography Conference. Springer, Berlin, Heidelberg, 2005: 325-341.

[38] Lewko A, Okamoto T, Sahai A, et al. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption[C]//Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer, Berlin, Heidelberg, 2010: 62-91.

[39] De Caro A, Iovino V, Persiano G. Fully secure anonymous HIBE and secret-key anonymous IBE with short ciphertexts[C]//International Conference on Pairing-Based Cryptography. Springer, Berlin, Heidelberg, 2010: 347-366.

[40] Goyal V. Reducing trust in the PKG in identity based cryptosystems[C]//Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2007: 430-447.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051362, IEEE Internet of Things Journal

[41] Goyal V, Lu S, Sahai A, et al. Black-box accountable authority identity-based encryption[C]//Proceedings of the 15th ACM conference on Computer and communications security. ACM, 2008: 427-436.

[42] Au M H, Huang Q, Liu J K, et al. Traceable and retrievable identity-based encryption[C]//International Conference on Applied Cryptography and Network Security. Springer, Berlin, Heidelberg, 2008: 94-110.

2327-4662 (c) 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: University of New Brunswick. Downloaded on May 21,2021 at 17:33:22 UTC from IEEE Xplore. Restrictions apply.