Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 1701675, 12 pages
https://doi.org/10.1155/2018/1701675

Research Article

Traceable Ciphertext-Policy Attribute-Based Encryption with Verifiable Outsourced Decryption in eHealth Cloud

Qi Li,[1,2,3] **Hongbo Zhu**,[3] **Zuobin Ying**,[4] **and Tao Zhang**[5]

[1] *School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China*
[2] *Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, Nanjing University of Posts and Telecommunications, Nanjing 210023, China*
[3] *Jiangsu Innovative Coordination Center of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China*
[4] *School of Computer Science and Technology, Anhui University, Hefei 230601, China*
[5] *School of Computer Science and Technology, Xidian University, Xian 710071, China*

Correspondence should be addressed to Qi Li; liqics@njupt.edu.cn

Received 8 March 2018; Revised 19 April 2018; Accepted 7 May 2018; Published 6 June 2018

Academic Editor: Kim-Kwang Raymond Choo

In cloud-assisted electronic health care (eHealth) systems, a patient can enforce access control on his/her personal health information (PHI) in a cryptographic way by employing ciphertext-policy attribute-based encryption (CP-ABE) mechanism. There are two features worthy of consideration in real eHealth applications. On the one hand, although the outsourced decryption technique can significantly reduce the decryption cost of a physician, the correctness of the returned result should be guaranteed. On the other hand, the malicious physician who leaks the private key intentionally should be caught. Existing systems mostly aim to provide only one of the above properties. In this work, we present a verifiable and traceable CP-ABE scheme (VTCP-ABE) in eHealth cloud, which simultaneously supports the properties of verifiable outsourced decryption and white-box traceability without compromising the physician's identity privacy. An authorized physician can obtain an ElGamal-type partial decrypted ciphertext (PDC) element of original ciphertext from the eHealth cloud decryption server (CDS) and then verify the correctness of returned PDC. Moreover, the illegal behaviour of malicious physician can be precisely (white-box) traced. We further exploit a delegation method to help the resource-limited physician authorize someone else to interact with the CDS. The formal security proof and extensive simulations illustrate that our VTCP-ABE scheme is secure, efficient, and practical.

1. Introduction

Electronic health care (eHealth) system is regarded as an outstanding approach to provide well health care service through various emerging technologies, including Internet of Things, cloud computing, mobile computing, and wireless sensor networks. In cloud-assisted eHealth systems, an individual patient integrates his/her personal health information (PHI) collected via various wearable and embedded sensors, stores the PHI in the cloud, and receives real-time and high-quality medical treatment. Unfortunately, when the patient enjoys convenient storage services provided by cloud server, the risk of privacy exposure also raises. The sensitive PHI may be exposed to the cloud server which can not be fully trusted. Even worse, the PHI may be widely propagated to unauthorized parties for commerce benefit or other purposes. Thus, the PHI must be encrypted before hosted to the eHealth cloud. Meanwhile, an access policy must be specified to point out who are authorized to access the PHI.

Aiming to realize access control on encrypted message, attribute-based encryption (ABE) [1] was presented to provide an efficient solution to this kind of applications. According to the place where the access policy is embedded, the ABE schemes are divided into two forms, key-policy ABE (KP-ABE) [2] and another type of ABE named ciphertext-policy ABE (CP-ABE) [3]. In the former framework, every user's key is labeled with an access policy while the ciphertexts are annotated with chosen sets of attributes. On the contrary, the

user's key in CP-ABE is issued according to his/her attributes while the ciphertext is encrypted under an access policy. Since that ABE is a feasible mechanism which preserves the security and privacy of patients' PHI, a series of attribute-based access control systems [4–8] have been proposed, aiming at expressive policies, security, or efficiency. In particular, there remain two significant features to be considered in utilizing ABE technique in eHealth systems.

The first feature is verifiability of outsourced decryption. In most ABE systems [1–3, 9–12], the decryption overhead is linear to the scale of involved attributes and expensive for energy-constrained terminals. The decryption outsourcing technique [13] is proposed to reduce the number of exponential operations and bilinear pairing operations on user side by offloading the heavy decryption computation to a third-party server, e.g., the cloud server. The user then recovers the plaintext by executing only one exponential operation over ElGamal-style partial decrypted ciphertext element generated by the third-party server. However, such outsourced scheme can not guarantee the correctness of returned ElGamal-style element. Lai *et al.* [14] presented the verifiable approach in ABE to check whether the third-party server has honestly executed the decryption service. They also bring redundant overhead in both encryption computation and ciphertext size. Qin *et al.* [15] provided an efficient verifiable ABE scheme which significantly reduces the computation cost in encryption and the decryption overhead for users.

Another considerable feature is traceability. We take CP-ABE as an instance; the private key is generated from some descriptive attributes rather than from a unique identity. Each attribute may be possessed by multiple users. It could be impossible to distinguish who is the original owner of a given private key. Imagine two physicians in eHealth systems, Tomas and Jack. They have the attribute set '{orthopedics department, chief physician}' which is not possessed by any other users. By the key delegate technique [3], both Tomas and Jack can regenerate a private key responding to the set '{orthopedics department, chief physician}', if there is a third user who can decrypt the ciphertext labeled by access policy '{'orthopedics department' *AND* 'chief physician'}'. Where did the key come from? Tomas or Jack? To solve the problem above, Liu *et al.* [16] extended an adaptively secure CP-ABE scheme [9] to support 'white-box' traceability, where the malicious user directly leaks his/her private key. Subsequently, Ning *et al.* [17] constructed a large attribute universe and traceable CP-ABE scheme. On the contrary to the 'small universe' in [3, 10, 14–16], the 'large universe' means that the scale of attribute universe is unbounded [18].

However, existing works mostly aimed to support the property of verifiable outsourced decryption or traceability separately. There is no CP-ABE scheme with both verifiable outsourced decryption and white-box traceability in practice: (1) the CP-ABE schemes [16, 17] support the traceability well, but the user's decryption cost grow with the attribute number; (2) these CP-ABE schemes [14, 15, 19, 20] provide decryption assistance for users, and the correctness of returned PDC element is guaranteed; however, the traceability property is not addressed.

In this work, we propose a novel verifiable and traceable CP-ABE scheme named VTCP-ABE for eHealth cloud applications. The VTCP-ABE scheme is the first scheme which simultaneously achieves white-box traceability and verifiable outsourced decryption without exposing the physician's identity information. Since we take the 'large universe' scheme [18] as the basis, the attribute universe in our scheme is inherently unbounded. We further extend the VTCP-ABE to support another delegation property. We also provide the formal proof of the selective CPA security, verifiability, and traceability for VTCP-ABE. The comparison and simulation results show that our VTCP-ABE is applicable for practical eHealth cloud applications. In particularly, we make the following contributions:

(1) We propose a new VTCP-ABE scheme which simultaneously achieves the properties of verifiable outsourced decryption, white-box traceability, and large universe. An authorized physician can check the correctness of partial decrypted ciphertext (PDC) which is requested from the eHealth CDS. Given a private key, the original owner can be precisely tracked. The attribute universe can be exponentially large and the number of public parameter elements is constant no matter how many attributes are chosen.

(2) We present an efficient approach to prevent the CDS from knowing the fixed identification information of physician during offering decryption service. The original ciphertext and the transmission private key will be preprocessed before being sent to the CDS. This method is acceptable since only two additional exponential operations for each decryption request are added.

(3) We exploit an additional property of delegation for our VTCP-ABE, with which a resource-constrained physician can delegate someone to obtain a PDC element without compromising the privacy of PHI.

1.1. Related Works. ABE was first introduced in [1]. The first KP-ABE scheme with threshold tree access structures was presented in [2]. The first CP-ABE scheme with the same structures was presented in [3]. Waters [21] presented several CP-ABE schemes to support the access policy defined as Linear Secret Sharing Schemes (LSSS). Yu *et al.* [22] demonstrated the deployment of ABE technique in cloud computing. In [4], Li *et al.* presented a personal health record (PHR) secure sharing scheme in cloud computing. Subsequently, various constructions of ABE schemes were presented in [9, 23–29].

Green *et al.* [13] constructed the first decryption outsourcing ABE, where the most decryption overhead is hosted to a third party. With the returned partial decrypted ciphertext, a user could recover the plaintext message by executing only one exponential operation. Based on the outsourced method [13], Li *et al.* [7] presented a PHR data sharing scheme for cloud storage applications in the multi-authority settings. In both [7, 13], the correctness of returned PDC is not guaranteed. Lai *et al.* [14] presented an approach to check whether the partial decrypted ciphertext element (transformed ciphertext element) is correctly calculated. Their technique incurred noticeable overhead in both decryption and encryption. Based on key encapsulation mechanism, Lin

Table 1: Comparison between ours and some related works.

Systems	CP/KP	AU[1]	OD[2]	Verifiability	Traceability	Delegation
Rouselakis *et al.* [18]	CP,KP	Large	No	No	No	No
Green *et al.* [13]	CP,KP	Small	Yes	No	No	No
Lai *et al.* [14]	CP	Small	Yes	Yes	No	No
Qin *et al.* [15]	CP	Small	Yes	Yes	No	No
Liu *et al.* [16]	CP	Small	No	No	Yes	No
Ning *et al.* [17]	CP	Large	No	No	Yes	No
Our VTCP-ABE	CP	Large	Yes	Yes	Yes	Yes

[1]AU is the abbreviation of attribute universe.
[2]OD is the abbreviation of outsourced decryption.

et al. [19] and Qin *et al.* [15] separately proposed a fascinating method to support verifiable outsourced decryption in ABE. The difference between [19] and [15] is that, in [19], the hash value of a random group element R is set as the symmetric key to encrypt the original data, then R is encrypted by a ABE scheme to obtain a ABE-type ciphertext, which will be used to generate the verification key. In [15], the original data M is encrypted along with a randomly chosen bit string r, while the verification key is set by executing exponential operations in the group by taking the hash values of M and r as exponents.

Liu *et al.* presented the first adaptively secure and white-box traceable CP-ABE scheme in [16], where any monotonic LSSS access structure is supported. They further constructed another CP-ABE scheme with black-box traceability in [30]. Based on the scheme [31], Ning *et al.* [17] exploited the white-box traceability for CP-ABE in large universe settings. From then on, many traceable ABE constructions are proposed in [6, 32, 33]. However, in these traceable schemes [6, 16, 17, 30, 32, 33], the decryption overhead grows with the scale of attribute set adopted in decryption.

Table 1 compares the characteristics between some related works and our VTCP-ABE. From Table 1, our VTCP-ABE scheme is the only practical scheme to simultaneously support the properties of large universe, verifiable outsourced decryption, white-box traceability, and delegation in CP-ABE.

2. Preliminaries

2.1. Bilinear Maps. Denote $\mathbb{G}$ and $\mathbb{G}_1$ as two multiplicative cyclic groups with prime order p. g is a generator of group $\mathbb{G}$. The bilinear map $e : \mathbb{G}\times\mathbb{G} \to \mathbb{G}_1$ has the following properties:

(1) Bilinearity: $\forall \varsigma, \eta \in \mathbb{G}$ and $\iota, \upsilon \in \mathbb{Z}_p$, $e(\varsigma^\iota, \eta^\upsilon) = e(\varsigma, \eta)^{\iota\upsilon}$.

(2) Non-degeneracy: $e(g, g) \neq 1$.

(3) Computability: for all $\varsigma, \eta \in \mathbb{G}$, $e(\varsigma, \eta)$ are efficiently computable.

Since that $e(g^\iota, g^\upsilon) = e(g, g)^{\iota\upsilon} = e(g^\upsilon, g^\iota)$, e is symmetric.

2.2. Linear Secret Sharing Schemes (LSSS)

Definition 1. Linear Secret Sharing Schemes [21, 34]: let $\mathbb{P}$ denote a set of attributes, and let p be a chosen prime. Let $T \in \mathbb{Z}_p^{m\times n}$ be a matrix. For all $i = 1, \ldots, m$, a function ρ labels the i-th row of T with an attribute (*i.e.,* $\rho \in \mathscr{F}([m] \to \mathbb{P})$). A secret sharing scheme Π over the attribute universe $\mathbb{P}$ is linear if one has the following:

(1) The shares for each attribute make a vector over $\mathbb{Z}_p$.

(2) In order to generate the shares of a secret $s \in \mathbb{Z}_p$, we select the column vector $\overrightarrow{\varrho} = (s, \varrho_2, \ldots, \varrho_n)^\top$, where $\varrho_2, \ldots, \varrho_n$ are randomly selected from $\mathbb{Z}_p$, then $T\overrightarrow{\varrho}$ is the shares of s according to Π. The share $(T\overrightarrow{\varrho})_i$ belongs to the attribute $\rho(i)$.

As demonstrated in [34], the linear reconstruction property of LSSS is defined as follows: Suppose (T, ρ) is the access structure $\mathbb{T}$ and S is an authorized set. Let $I = \{i : \rho(i) \in S\}$ be the index set of rows which are linked with the attributes in S. There exist constants $\{\psi_i \in \mathbb{Z}_p\}_{i\in I}$ which satisfy that if $\{\lambda_i = (T\overrightarrow{\varrho})_i\}$ are valid, then we have $\sum_{i\in I} \psi_i\lambda_i = s$.

2.3. φ-Type Assumption. The security of VTCP-ABE is reduced to a φ-type assumption [18].

Suppose $\mathbb{G}$ is a cyclic group and prime p is the group order. Randomly pick $g \in \mathbb{G}$ and choose $\iota, s, \upsilon_1, \upsilon_2, \ldots, \upsilon_\varphi \in \mathbb{Z}_p$. If an adversary $\mathscr{A}$ is given the group description $(p, \mathbb{G}, \mathbb{G}_1, e)$ and Ξ including all of the following terms:

$$\Xi =$$

$$g, g^s$$

$$g^{\iota^i}, g^{\upsilon_j}, g^{s\upsilon_j}, g^{\iota^i\upsilon_j}, g^{\iota^i/\upsilon_j^2}, \forall (i, j) \in [\varphi, \varphi]$$

$$g^{\iota^i/\upsilon_j}, \forall (i, j) \in [2\varphi, \varphi] \text{ with } i \neq \varphi + 1$$

$$g^{\iota^i\upsilon_j/\upsilon_{j'}^2}, \forall (i, j, j') \in [2\varphi, \varphi, \varphi] \text{ with } j \neq j'$$

$$g^{s\iota^i\upsilon_j/\upsilon_{j'}}, g^{s\iota^i\upsilon_j/\upsilon_{j'}^2} \ \forall (i, j, j') \in [\varphi, \varphi, \varphi] \text{ with } j \neq j'$$

It must be hard for $\mathscr{A}$ to distinguish the element $e(g, g)^{\iota^{\varphi+1}s} \in \mathbb{G}_1$ from a randomly chosen element $F \in \mathbb{G}_1$.

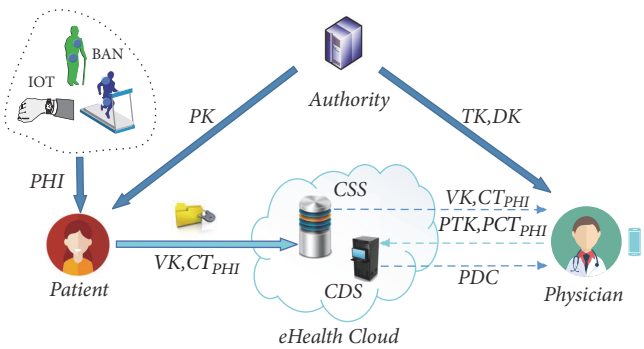

Figure 1: Architecture of VTCP-ABE in eHealth cloud.

The advantage of an algorithm $\mathscr{A}$ which solves the above q-type problem is

$$Adv_{\mathscr{A}(\lambda)} = \left|\Pr\left[\mathscr{A}\left(\Xi, W = e\left(g,g\right)^{t^{\varphi+1}s}\right) = 0\right] - \Pr\left[\mathscr{A}\left(\Xi, W = F\right) = 0\right]\right| \quad (1)$$

Definition 2. We claim that the φ-type assumption holds if the advantage of all polynomial time adversaries is negligible in the above φ-type game.

2.4. ϑ-Strong Diffie-Hellman Assumption (ϑ-SDH). The ϑ-SDH problem [35, 36]: suppose $\mathbb{G}$ is a cyclic group. Let prime p be the group order. g is randomly selected from $\mathbb{G}$. Given a $(\vartheta+1)$-tuple $(g, g^{\varpi}, g^{\varpi^2}, \ldots, g^{\varpi^{\vartheta}})$, output a pair $(\delta, g^{1/(\varpi+\delta)}) \in \mathbb{Z}_p \times \mathbb{G}$. An algorithm $\mathscr{A}$ has advantage ϵ in solving ϑ-SDH problem if $\Pr[\mathscr{A}(g, g^{\varpi}, g^{\varpi^2}, \ldots, g^{\varpi^{\vartheta}}) = (\delta, g^{1/(\varpi+\delta)})] \geq \epsilon$, where the probability is over the random bits consumed by $\mathscr{A}$ and the randomness of $\varpi \in \mathbb{Z}_p$.

Definition 3. We claim that the (ϑ, t, ϵ)-SDH assumption holds if the advantage of all t-time adversaries is at most ϵ in solving the above ϑ-SDH problem.

3. System Architecture and Security Model

3.1. System Description. As shown in Figure 1, our VTCP-ABE framework in the eHealth cloud mainly consists of the following components.

The authority: the authority produces the system parameters and generates private keys for the legal physicians depending on their attributes. It is also in charge of tracing the malicious physicians.

The patient: with the help of IOT techniques, the patient integrates and then encrypts his/her PHI under appropriate access policy and further uploads the ciphertext to the eHealth cloud storage server.

The eHealth cloud storage server (CSS): the eHealth CSS provides storage service for the patient. If necessary, the patient can call CSS to delete his/her PHI data.

The eHealth cloud decryption server (CDS): the eHealth CDS provides pre-decryption service of the encrypted PHI and returns the partial decrypted ciphertext to the authorized physician.

The physician: the physician takes responsibility of medical treatment for the patient whose access policy accepts his/her attributes. The physician is also enabled to check the correctness of returned pre-decryption results from the CDS. The malicious physician may leak his private key for economic benefit or some malignant purpose.

We note that the eHealth CSS and CDS are assumed to be semi-trusted as in [22]. That is, the CSS and CDS honestly execute the pre-set algorithms. But they attempts to get useful information of the encrypted PHI as much as possible. In addition, the eHealth CDS may want to obtain the identification information of physician.

As one of the important applications in IOT environments, the eHealth cloud system enables the patient to collect his PHI via wearable devices, physiologic sensor nodes and body area networks, etc. Before uploading the PHI to the cloud sever to get real-time health care services, the patient can define expressive access policy of his PHI over descriptive attributes by VTCP-ABE. According to the assigned attributes, the individual physicians have differential flexible access rights. They can provide various (free or paid) health care services by smart devices on condition that their attributes match the access policy of patient's PHI. Our VTCP-ABE also offers the traceability to prevent the key abuse problem and the verifiable outsourced decryption technique to offload most decryption cost to the cloud server and guarantee the returned results.

3.2. Definition of VTCP-ABE. Our VTCP-ABE is comprised by the following seven algorithms.

Setup$(\kappa, U) \rightarrow (PK, MSK)$: this algorithm takes in a security parameter κ and the system attribute universe U. It then outputs the system public parameters PK and the master secret key MSK. Besides, it initializes an identity table $IT = \varnothing$.

Encrypt $(M, PK, \mathbb{T}) \rightarrow (CT, VK)$. This algorithm takes in a message M, PK, and an access structure $\mathbb{T}$. It then outputs a ciphertext CT and a verification key VK.

KeyGen $(PK, MSK, id, S) \rightarrow (TK, DK)$. This algorithm takes in PK, MSK, an identity information id and an attribute set S. It then outputs a transmission private key TK and a user decryption key DK.

Pre-Process $(PK, CT, TK) \rightarrow (PCT, PTK)$. This algorithm takes in CT and TK. It then outputs a pre-processed ciphertext PCT and a pre-processed private key PTK.

Pre-Decrypt $(PCT, PTK) \rightarrow (PDC)$. This algorithm takes in PCT and PTK. If S matches $\mathbb{T}$, the algorithm outputs a partial decrypted ciphertext PDC. Otherwise, it outputs $\perp$.

Decrypt $(PDC, DK, VK) \rightarrow (M)$. This algorithm takes in PDC, DK, and VK. If PDC is not valid, it outputs $\perp$. Otherwise, it outputs a message M.

Trace $(IT, PK, TK, DK) \rightarrow id$ or $\top$. This algorithm takes in IT, PK, TK, and DK. It first verifies whether TK and DK are well-formed. If so, this algorithm outputs the id annotated with TK and DK. Otherwise, it outputs $\top$ implying that TK and DK are not required to be traced. If TK and DK can pass

a "key sanity check" which means that they can be used in the normal decryption phase, they are called well-formed [16].

3.3. CPA Security Model. Similar to [17, 18], the definition of selective security model of VTCP-ABE against chosen plaintext attack (CPA) is given as follows:

Init. The adversary $\mathscr{A}$ gives the simulator $\mathscr{B}$ the challenge access policy $\mathbb{T}^*$.

Setup. $\mathscr{B}$ runs **Setup** to produce (PK, MSK) and passes PK to $\mathscr{A}$.

Phase 1. $\mathscr{A}$ can ask $\mathscr{B}$ to produce the private keys for $((id_1, S_1), (id_2, S_2), \ldots, (id_{Q_1}, S_{Q_1}))$. For each (id_i, S_i), $\mathscr{B}$ returns by the corresponding private key pairs (TK_i, DK_i). Note that, for each $i \in [Q_1]$, S_i can not match $\mathbb{T}^*$.

Challenge Phase. $\mathscr{A}$ submits two messages M_0 and M_1 of equal length. $\mathscr{B}$ encrypts M_μ under $\mathbb{T}^*$ to obtain CT^* and VK^*, where μ is randomly chosen from $\{0, 1\}$. It then gives CT^* and VK^* to $\mathscr{A}$.

Phase 2. As in **Phase 1**, $\mathscr{B}$ is asked to produce the private keys of $((id_{Q_1+1}, S_{Q_1+1}), \ldots, (id_Q, S_Q))$.

Guess. $\mathscr{A}$ guesses μ' for μ. $\mathscr{A}$'s advantage is defined as $\Pr[\mu' = \mu] - 1/2$.

Definition 4. We claim that a VTCP-ABE scheme is selectively CPA secure if the advantage is negligible for all PPT adversaries in the above selective security game.

3.4. Security Game for Verifiability. Based on the replayable chosen ciphertext attack (RCCA) security model [13, 15], we briefly introduce the verifiability game as follows.

Setup. The challenger $\mathscr{B}$ generates (PK, MSK) and sends PK to the attacker $\mathscr{A}$.

Phase 1. $\mathscr{B}$ queries the results from the *Create*, *Corrupt*, and *Decrypt* oracles as in [15].

Challenge Phase. The attacker $\mathscr{A}$ submits an access policy $\mathbb{T}^*$ and a message M^*. $\mathscr{B}$ encrypts M_* under $\mathbb{T}^*$ to obtain (CT^*, VK^*) and sends them to $\mathscr{A}$.

Phase 2. $\mathscr{A}$ repeats the key queries as in **Phase 1**.

Output. $\mathscr{A}$ gives $\mathscr{B}$ PDC^* and an attribute set S^* which satisfies $\mathbb{T}^*$.

The attacker $\mathscr{A}$ wins the above game if **Decrypt** $(PDC^*, DK^*, VK^*) \notin \{M, \perp\}$. $\mathscr{A}$'s advantage in this game is defined as $ADV_{VTCP-ABE}\mathscr{A}$.

Definition 5. We claim that a VTCP-ABE scheme is verifiable if $ADV_{VTCP-ABE}\mathscr{A}$ is negligible for all PPT attackers in the above game.

3.5. Security Game for Traceability. The traceability game of our VTCP-ABE is defined as follows.

Setup. The challenger $\mathscr{B}$ generates (PK, MSK) and sends PK to the attacker $\mathscr{A}$. It keeps MSK as a secret key.

Key Query. $\mathscr{A}$ submits the tuples $(id_1, S_1), (id_2, S_2), \ldots, (id_q, S_q)$ to $\mathscr{B}$, where q refers to the query number that $\mathscr{A}$ can make.

Key Forgery. $\mathscr{A}$ outputs $DK_\star$ and $TK_\star$. $\mathscr{A}$ wins if **Trace** $(IT, PK, DK_\star, TK_\star) \neq \top$ and **Trace**$(IT, PK, DK_\star, TK_\star) \notin \{id_1, id_2, \ldots, id_q\}$. $\mathscr{A}$'s advantage is defined as $\Pr[\textbf{Trace}(IT, PK, DK_\star, TK_\star) \neq \{\top\} \cup \{id_1, id_2, \ldots, id_q\}]$.

Definition 6. We claim that a VTCP-ABE scheme is fully traceable if the advantage is negligible for all PPT attackers in the above game.

4. The Proposed VTCP-ABE

In this section, we first briefly introduce the techniques of constructing a verifiable and traceable CP-ABE scheme and then give the details of VTCP-ABE construction.

4.1. Technical Overview. To achieve the traceability in [17], each private key is associated with a unique fixed number δ so that the key owner cannot re-randomize his own private key to get a completely new key. In the verifiable CP-ABE scheme with outsourced decryption [15], the private key is composed of a transmission key and a user decryption key. The transmission key is sent to a third party to get the partial decryption result and the user decryption key is used to decrypt the partial decryption result and check its correctness.

Our goal is to achieve the efficient user decryption and traceability without compromising the security and privacy. However, if we combine the traceable CP-ABE [17] and the verifiable outsourced decryption approach [15] in a naive way, the fixed identifier number δ will be exposed to the eHealth CDS. Even worse, the CDS may use δ and the transmission private key to fabricate a key which could pass the check in the traceable algorithm of [17]. That is, a legal physician may be framed to be malicious and further revoked from the system. To prevent the CDS from knowing δ, we process the transmission private key and original ciphertext before submitting them to the eHealth CDS. Meanwhile, we add the user decryption key as input of the traceable algorithm. Finally, we add the property of verifiable outsourced decryption into the traceable CP-ABE scheme [17] at a very low cost on the physician side (one additional element in private key, two additional exponential operations in pre-processing)

4.2. Detailed Construction. We now give the detailed construction of the VTCP-ABE.

Setup. Given a group description $G = (p, \mathbb{G}, \mathbb{G}_1, e)$, where prime order p is the order of groups $(\mathbb{G}, \mathbb{G}_1)$ and e denotes a map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_1$. The system attribute universe is set as $U = \mathbb{Z}_p$. Then randomly pick $g, u, \hbar, w, v \in \mathbb{G}$ and $\alpha, a \in \mathbb{Z}_p$.

Select two collision-resistant hash functions $HA_1 : \mathbb{G}_1 \to \{0, 1\}^{\ell_{HA_1}}$ and $HA_2 : \{0, 1\}^* \to \{0, 1\}^{\ell_{HA_2}}$. $SE =$ (**SE-Keygen**, **SE-Encrypt**, **SE-Decrypt**) refers to a one-time symmetric encryption scheme and the key space is defined as $\{0, 1\}^{\ell_{SE}}$. Select $HA_3 : \mathbb{G}_1 \to \{0, 1\}^{\ell_{SE}}$ from $\mathscr{H}$, which denotes a party of pairwise independent hash functions.

It sets $(G, g, u, \hbar, w, v, e(g, g)^\alpha, g^a, SE, HA_1, HA_2, HA_3)$ as PK and (α, a) as MSK. It also initializes $IT = \varnothing$.

Encrypt. Given the PHI data $M \in \aleph$ and a LSSS policy $\mathbb{T} = (T, \rho) \in (\mathbb{Z}_p^{m\times n}, \rho : [m] \to \mathbb{Z}_p)$, the encryption algorithm acts as follows.

Randomly select $\chi \in \mathbb{G}_1$ and $\overrightarrow{\varrho} = (s, \varrho_2, \ldots, \varrho_n)^\top \in \mathbb{Z}_p^{n\times 1}$. Calculate $\overrightarrow{\lambda} = T \cdot \overrightarrow{\varrho} = (\lambda_1, \lambda_2, \ldots, \lambda_n)^\top$. Choose $t_1, t_2, \ldots, t_m$ randomly from $\mathbb{Z}_p$ and compute

$$C = \chi \cdot e(g,g)^{\alpha s},$$
$$C_1 = g^s,$$
$$C_2 = g^{as}.$$

For each $i \in [m]$, compute $C_{i,1} = w^{\lambda_i} v^{t_i}$, $C_{i,2} = (u^{\rho(i)}\hbar)^{-t_i}$ and $C_{i,3} = g^{t_i}$.

The ciphertext of χ is $CT_\chi = (\mathbb{T}, C, C_1, C_2, \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i\in[m]})$.

After that, this algorithm sets $TAG_1 = HA_1(\chi)$ and computes a symmetric key $SEK = HA_3(\chi)$. Then it calls SE to create a ciphertext $CT_M = \textbf{SE-Encrypt}(M, SEK)$ and the verification key $VK = TAG_2 = HA_2(TAG_1 \parallel CT_M)$.

Finally, the ciphertext of PHI data $CT = (CT_\chi, CT_M)$ is uploaded to the eHealth CSS as well as VK.

KeyGen. Given a tuple $(id, S = \{AT_1, AT_2, \ldots, AT_k\} \subseteq \mathbb{Z}_p)$, this algorithm randomly selects $b, \delta, \beta, \beta_1, \ldots, \beta_k \in \mathbb{Z}_p$ and then calculates

$$KK = g^{\alpha/b(a+\delta)} w^\beta,$$
$$KK_1 = \delta,$$
$$L_1 = g^\beta,$$
$$L_2 = g^{a\beta}.$$

For each $\tau \in [k]$, it computes $KK_{\tau,1} = g^{\beta_\tau}$ and $KK_{\tau,2} = (u^{AT_\tau}\hbar)^{\beta_\tau} v^{-(a+\delta)\beta}$.

Finally, it outputs the private key for (id, S) as $TK = (S, KK, KK_1, L_1, L_2, \{KK_{\tau,1}, KK_{\tau,2}\}_{\tau\in[k]})$ and $DK = b$. Simultaneously, the tuple (id, δ) is added to IT.

Pre-Process. The physician can request the PHI ciphertext $CT = (CT_\chi, CT_M)$ and VK from the eHealth CSS, which will response by the elements C_1, C_2, CT_M, and VK while the other elements will be sent to the eHealth CDS.

Before calling the pre-decryption service, he/she processes the C_1, C_2, and TK by calculating $C_3 = C_1^{KK_1} C_2 = g^{(a+\delta)s}$ and $L_3 = L_1^{KK_1} L_2 = g^{(a+\delta)\beta}$.

Then $PCT_\chi = C_3$ and $PTK = (KK, L_3, \{KK_{\tau,1}, KK_{\tau,2}\}_{\tau\in[k]})$ are sent to the eHealth CDS.

Pre-Decrypt. Once receiving PCT_χ and PTK, this algorithm works as follows.

If S does not match $\mathbb{T}$, this algorithm aborts. Otherwise, it sets $I = \{i : \rho(i) \in S\}$ and calculates constants $\{\psi_i \in \mathbb{Z}_p\}_{i\in I}$ such that $\sum_{i\in I} \psi_i T_i = (1, 0, \ldots, 0)$, where T_i refers to the i-th row of T. Then it calculates

$$\begin{aligned} C' &= \frac{e(KK, C_3)}{\prod_{i\in I}\left(e(L_3, C_{i,1})\, e(KK_{\tau,1}, C_{i,2})\, e(KK_{\tau,2}, C_{i,3})\right)^{\psi_i}} \\ &= \frac{e(g,g)^{\alpha s/b}\, e(w,g)^{(a+\delta)\beta s}}{\prod_{i\in I}\left(e(g,w)^{(a+\delta)\beta\lambda_i}\, e(g,v)^{(a+\delta)\beta t_i}\, e(g,u)^{-t_i\rho(i)\beta_\tau}\, e(g,\hbar)^{-t_i\beta_\tau}\, e(u,g)^{t_i AT_\tau \beta_\tau}\, e(\hbar,g)^{t_i\beta_\tau}\, e(v,g)^{-(a+\delta)\beta t_i}\right)^{\psi_i}} \\ &= \frac{e(g,g)^{\alpha s/b}\, e(w,g)^{(a+\delta)\beta s}}{\prod_{i\in I} e(g,w)^{(a+\delta)\beta\lambda_i\psi_i}} = \frac{e(g,g)^{\alpha s/b}\, e(w,g)^{(a+\delta)\beta s}}{e(g,w)^{(a+\delta)\beta s}} = e(g,g)^{\alpha s/b} \end{aligned} \tag{2}$$

Finally, it outputs $PDC = C'$.

Decrypt. This algorithm first computes $\chi = C/(C')^b$. Then it calculates $TAG_1 = HA_1(\chi)$. If $HA_2(TAG_1 \parallel CT_M) \neq VK$, it aborts immediately. Otherwise, it calculates $SEK = HA_3(\chi)$ and recovers $M := \textbf{SE-Decrypt}(CT_M, SEK)$.

Trace. This algorithm first verifies whether TK and DK are well-formed by the following checks:

(1) TK is expressed as $(S, KK, KK_1, L_1, L_2, \{KK_{\tau,1}, KK_{\tau,2}\}_{\tau\in[k]})$, where $KK_1 \in \mathbb{Z}_p$ and $KK, L_1, L_2, KK_{\tau,1}, KK_{\tau,2} \in \mathbb{G}$.

(2) $DK = b \in \mathbb{Z}_p$.

(3) $e(g, L_2) = e(g^a, L_1)$.

(4) $e(KK, g^a \cdot g^{KK_1}) = e(g,g)^{\alpha/b} e(L_1^{KK_1} L_2, w)$.

(5) $\exists \tau \in [k]$, *s.t.* $e(KK_{\tau,2}, g) e(L_1^{KK_1} L_2, v) = e(KK_{\tau,1}, \hbar) e(KK_{\tau,1}, u)^{AT_\tau}$.

If TK and DK fail to pass the above five checks, it outputs $\top$. Otherwise, it searches KK_1 in IT: if KK_1 exists, it outputs the corresponding id. If KK_1 does not exist, it aborts.

5. Security Proof

5.1. CPA Security. For simplicity, the security of the presented VTCP-ABE scheme is reduced to that of the traceable scheme [17] which is proved under the φ-type assumption. We let $\sum_{TCP-ABE}$ and $\sum_{VTCP-ABE}$ denote the traceable scheme [17] and our VTCP-ABE scheme, respectively.

Theorem 7. *Suppose that $\sum_{TCP-ABE}$ is selectively secure, the one-time symmetric encryption scheme SE is semantically secure, HA_3 is chosen from a party of pairwise independent hash functions, and the parameters satisfy $0 < \ell_{SE} \leq (\log|\aleph| - \ell_{HA_1}) - 2\log(1/\epsilon_{HA_2})$. Then, the proposed $\sum_{VTCP-ABE}$ is selectively secure.*

Proof. Similar to the proof in [15], we define a series of hybrid argument of games as in [37].

Game$_0$. Identical to the original security game as defined in Section 3.3.

Game$_1$. Identical to **Game**$_0$, except that TAG_1^* and SEK^* are computed by selecting another random key R^* rather than χ^* in CT_χ.

Game$_2$. Identical to **Game**$_1$, except that we replace SEK^* by a randomly selected string $SEK_R^* \in \{0,1\}^{\ell_{SE}}$.

Let SPB_i be the success probability of the attacker in **Game**$_i$. □

Lemma 8. *If $\sum_{TCP-ABE}$ is selectively secure, then the attacker can not distinguish **Game**$_0$ from **Game**$_1$ with a non-negligible advantage.*

Proof. Suppose that an attacker $\mathscr{A}$ can distinguish **Game**$_0$ from **Game**$_1$, then we can build a PPT algorithm $\mathscr{B}$ to break $\sum_{TCP-ABE}$.

Init. The attacker $\mathscr{A}$ submits the challenge access policy $\mathbb{T}^*$ to $\mathscr{B}$. $\mathscr{B}$ then sends $\mathbb{T}^*$ to $\sum_{TCP-ABE}$.

Setup. Based on $\mathbb{T}^*$, $\sum_{TCP-ABE}$ gives $\mathscr{B}$ the parameter $PK_{TCP-ABE} = (G, g, u, \hbar, w, v, e(g,g)^\alpha, g^a)$ as in [17]. After that, $\mathscr{B}$ chooses SE^* and sets HA_1^*, HA_2^* and HA_3^* as random oracles. It also sets IT^*. Finally, it sends $PK^* = (PK_{TCP-ABE}, SE^*, HA_1^*, HA_2^*, HA_3^*)$ to $\mathscr{A}$.

Phase 1. To reply the key query of (id, S) from $\mathscr{A}$, $\mathscr{B}$ transmits (id, S) to $\sum_{TCP-ABE}$ and obtains $SK_{TCP-ABE} = (S, \widehat{KK}, \widehat{KK_1}, \widehat{L_1}, \widehat{L_2}, \{\widehat{KK_{\tau,1}}, \widehat{KK_{\tau,2}}\}_{\tau\in[k]})$, where

$$\widehat{KK} = g^{\alpha/(a+\delta)} w^{\hat{\beta}},$$
$$\widehat{KK_1} = \delta,$$
$$\widehat{L_1} = g^{\hat{\beta}},$$
$$\widehat{L_2} = g^{a\hat{\beta}}.$$
$$\forall \tau \in [k], \widehat{KK_{\tau,1}} = g^{\widehat{\beta_\tau}} \text{ and } \widehat{KK_{\tau,2}} = (u^{AT_\tau}\hbar)^{\widehat{\beta_\tau}} v^{-(a+\delta)\hat{\beta}}.$$

$\mathscr{B}$ randomly picks $b \in \mathbb{Z}_p$ and sets

$$KK = (\widehat{KK})^{1/b} = g^{\alpha/b(a+\delta)} w^{\beta},$$
$$KK_1 = \widehat{KK_1} = \delta,$$
$$L_1 = (\widehat{L_1})^{1/b} = g^{\beta},$$
$$L_2 = (\widehat{L_2})^{1/b} = g^{a\beta}.$$

□

For each $\tau \in [k]$, it computes $KK_{\tau,1} = (\widehat{KK_{\tau,1}})^{1/b} = g^{\beta_\tau}$ and $KK_{\tau,2} = (\widehat{KK_{\tau,2}})^{1/b} = (u^{AT_\tau}\hbar)^{\beta_\tau} v^{-(a+\delta)\beta}$.

$\mathscr{B}$ implicitly sets $\beta = (\hat{\beta})^{1/b}$ and $\beta_\tau = (\widehat{\beta_\tau})^{1/b}$.

Finally, $\mathscr{B}$ sends $TK = (S, KK, KK_1, L_1, L_2, \{KK_{\tau,1}, KK_{\tau,2}\}_{\tau\in[k]})$ and $DK = b$ to $\mathscr{A}$. Simultaneously, it adds (id, δ) to IT^*.

Challenge. $\mathscr{A}$ submits two equal-length messages M_0 and M_1, and $\mathscr{B}$ first picks two independent random keys χ^* and R^* from $\mathbb{G}_1$. It sends $((M_0^* = \chi^*, M_1^* = R^*), \mathbb{T}^*)$ to $\sum_{TCP-ABE}$. $\sum_{TCP-ABE}$ responds by a challenge ciphertext $CT_{M_\mu^*} = (\mathbb{T}^*, C, C_1, C_2, \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i\in[m]})$. Then, $\mathscr{B}$ computes $SEK^* = HA_3(R^*)$ and $TAG_1^* = HA_1^*(R^*)$. It randomly picks $\mu \in \{0,1\}$ and calculates $CT_{M_\mu} =$ **SE-Encrypt**(M_μ, SEK^*). It also computes $VK^* = TAG_2^* = HA_2^*(TAG_1^* \parallel CT_{M_\mu})$.

Finally, it sends $CT^* = (CT_{M_\mu^*}, CT_{M_\mu})$ and VK^* to the attacker.

Note that, if the key encrypted under $\mathbb{T}^*$ in $CT_{M_\mu^*}$ is R^*, CT^* is regarded as a challenge ciphertext in **Game**$_0$. Otherwise, CT^* can be regarded as a challenge ciphertext in **Game**$_1$.

Phase 2. Similar to **Phase 1**.

Finally, $\mathscr{A}$ gives $\mathscr{B}$ a μ'. $\mathscr{B}$ then sends μ' to $\sum_{TCP-ABE}$. From the above game, we have $|\Pr[SPB_0] - \Pr[SPB_1]| \le 2ADV_{\mathscr{B}} \sum_{TCP-ABE}$.

Lemma 9. *Suppose that $\mathscr{H}$ is a family of pairwise independent hash functions, then **Game**$_1$ can not be distinguished from **Game**$_2$ with a non-negligible advantage.*

Proof. The key R^* is completely independent of PK, $CT_{M_\mu^*}$, and HA_3^* in both **Game**$_1$ and **Game**$_2$. Moreover, the number of possible values of $TAG_1^* = HA_1^*(R^*)$ is at most $2^{\ell_{HA_1}}$. According to the analysis in [15] and $0 < \ell_{SE} \le (\log|\aleph| - \ell_{HA_1}) - 2\log(1/\epsilon_{HA_2})$, the $SEK^* = HA_3(R^*)$ is ϵ_{HA_2}-statistically indistinguishable from the randomly selected $R_{SE}^* \in \{0,1\}^{\ell_{SE}}$. Hence, we have $|\Pr[SPB_1] - \Pr[SPB_2]| \le \epsilon_{HA_2}$. □

Lemma 10. *Suppose that SE is a semantically secure symmetric encryption scheme, then the attacker can not win **Game**$_2$ with a non-negligible advantage.*

Proof. In **Game**$_2$, $R_{SE}^* \in \{0,1\}^{\ell_{SE}}$ is a truly random symmetric key. An algorithm $\mathscr{B}$ can be directly constructed from $\mathscr{A}$ to break the semantic security of SE^*. Therefore, we have $|\Pr[SPB_1] - 1/2| \le ADV_{\mathscr{B}} SE^*$. □

Remark that **Game**$_0$ is identical to the selective security game for our proposed VTCP-ABE scheme. The advantage is $|\Pr[SPB_0] - 1/2|$. Thus, the security of our $\sum_{VTCP-ABE}$ follows.

5.2. Verifiability

Theorem 11. *Suppose that these two hash functions HA_1 and HA_2 are collision-resistant, our proposed VTCP-ABE scheme is privately verifiable.*

Proof. Suppose that an attacker $\mathscr{A}$ can win the verifiability game, we can employ $\mathscr{A}$ to build an algorithm $\mathscr{B}$ to break the collision-resistance of HA_1 and HA_2.

Given the challenge hash functions HA_1^* and HA_2^*, $\mathscr{B}$ processes as follows.

$\mathscr{B}$ runs **Setup** to generate PK and MSK, except for HA_1^* and HA_2^*. To answer the key queries, $\mathscr{B}$ acts as in **Phase 1** and **Phase 2**.

In the **Challenge** phase, $\mathscr{B}$ invokes the **Encrypt** to obtain the CT_{χ^*}. Then, it computes $TAG_1^* = HA_1^*(\chi^*)$ and $SEK^* = HA_3^*(\chi^*)$. It also calculates $CT_{M^*} =$ **SE-Encrypt**(M^*, SEK^*) and $VK^* = TAG_2^* = HA_2^*(TAG_1^* \parallel CT_{M^*})$. It sends $CT^* = (CT_{\chi^*}, CT_{M^*})$ and VK^* to $\mathscr{A}$.

$\mathscr{A}$ outputs an attribute set S^* which satisfies $\mathbb{T}^*$ and a partially decrypted ciphertext $PDC^* = C'$ and CT_M.

If $\mathscr{A}$ wins the verifiability game, $\mathscr{B}$ will get a message $M \notin \{M^*, \perp\}$. Note that the **Decrypt** algorithm outputs $\perp$ if $HA_2^*(TAG_1 \parallel CT_M) \neq TAG_2^*$, where $TAG_1 = HA_1^*(\chi)$ and χ is recovered from PDC^* and CT_M.

We now analyze the success probability of $\mathscr{A}$ by considering the following cases:

(1) $(TAG_1, CT_M) \neq (TAG_1^*, CT_{M^*})$. If this case happens, $\mathscr{B}$ gets a collision of HA_2^* immediately.

(2) $(TAG_1, CT_M) = (TAG_1^*, CT_{M^*})$, but $\chi \neq \chi^*$. Note that $HA_1^*(\chi^*) = TAG_1^* = TAG_1 = HA_1^*(\chi)$. Thus, $\mathscr{B}$ gets a collision of HA_1^*. □

5.3. Traceability

Theorem 12. *If the ϑ-SDH assumption holds, then our proposed VTCP-ABE scheme is fully traceable on condition that $q < \vartheta$, where q is the number of key queries made by the attacker $\mathscr{A}$.*

Proof. We here briefly introduce the traceability proof. Given $(G = (p, \mathbb{G}, \mathbb{G}_1, e), \overline{g}, \overline{g}^{\omega}, \overline{g}^{\omega^2}, \ldots, \overline{g}^{\omega^{\vartheta}})$, the simulator $\mathscr{B}$ has to generate a pair $(\delta, \overline{g}^{1/(\omega+\delta)}) \in \mathbb{Z}_p \times \mathbb{G}$ to solve the ϑ-SDH problem.

Setup. Assuming $\vartheta = q + 1$, $\mathscr{B}$ sets $D_i = \overline{g}^{\omega^i}$ for each $i \in [\vartheta]$ and randomly selects q distinct numbers $\delta_1, \ldots, \delta_q$ from $\mathbb{Z}_p^*$. It then sets $f(z) = \prod_{i=1}^{q}(z + \delta_i) = \sum_{i=0}^{q} a_i z^i$, where $\{a_i\}$ are the coefficients of $f(z)$. The simulator computes $g \leftarrow \prod_{i=0}^{q}(D_i)^{a_i} = \overline{g}^{f(\omega)}$ and $g^{\omega} \leftarrow \prod_{i=1}^{q+1}(D_i)^{a_{i-1}} = \overline{g}^{\omega \cdot f(\omega)}$. It then randomly picks $u, \hbar \in \mathbb{G}$ and $\alpha, \theta_1, \theta_2 \in \mathbb{Z}_p$. Finally, $\mathscr{B}$ sets $(G, g, u, \hbar, w = g^{\theta_1}, v = g^{\theta_2}, e(g,g)^{\alpha}, g^{\omega}, SE, HA_1, HA_2, HA_3, IT)$ as PK, where SE, HA_1, HA_2, HA_3, and IT are set as in the CPA game. It gives PK to $\mathscr{A}$.

Key Query. $\mathscr{B}$ answers the i-th query of (id_i, S_i) as follows.

$\mathscr{B}$ sets $f_i(z) = f(z)/(z+\delta_i) = \prod_{j=1, j\neq i}^{q}(z+\delta_j) = \sum_{j=0}^{q-1} b_j z^j$ and computes $\zeta_i \leftarrow \prod_{j=0}^{q-1}(D_j)^{b_j} = \overline{g}^{f_i(\omega)} = \overline{g}^{f(\omega)/(\omega+\delta_i)} = g^{1/(\omega+\delta_i)}$. Then $\mathscr{B}$ randomly selects $b, \beta, \beta_1, \ldots, \beta_k \in \mathbb{Z}_p$ and calculates TK by computing:

$$\begin{aligned}
KK &= (\zeta_i)^{\alpha/b} = g^{\alpha/b(\omega+\delta_i)} w^{\beta},\\
KK_1 &= \delta_i,\\
L_1 &= g^{\beta},\\
L_2 &= g^{\omega\beta}.
\end{aligned}$$

For each $\tau \in [k]$, it computes $KK_{\tau,1} = g^{\beta_\tau}$ and $KK_{\tau,2} = (u^{AT_\tau}\hbar)^{\beta_\tau}((g^{\omega})^{\theta_2} \cdot v^{\delta_i})^{-\beta} = (u^{AT_\tau}\hbar)^{\beta_\tau} v^{-(\omega+\delta_i)\beta}$.

It gives TK and $DK = b$ to $\mathscr{A}$ and add (id_i, δ_i) to IT.

Key Forgery. $\mathscr{A}$ submits $TK_\star = (S, KK, KK_1, L_1, L_2, \{KK_{\tau,1}, KK_{\tau,2}\}_{\tau\in[k]})$ and $DK_\star = b$ to $\mathscr{B}$. $\Psi_{\mathscr{A}}$ refers to the event that $\mathscr{A}$ wins, i.e., $TK_\star$ and $DK_\star = b$ are well-formed and $KK_1 \notin \{\delta_1, \delta_2, \ldots, \delta_q\}$.

If $\Psi_{\mathscr{A}}$ happens, $\mathscr{B}$ writes $f(z)$ as $f(z) = \phi(z)(z + KK_1) + \phi_{-1}$ for some polynomial $\phi(z) = \sum_{i=0}^{q-1} \phi_i z^i$ and some $\phi_{-1} \neq 0 \in \mathbb{Z}_p$. Note that β in $TK_\star$ is unknown to $\mathscr{B}$. $\mathscr{B}$ then computes

$$\begin{aligned}
&\sigma \leftarrow (KK/L_1^{\theta_1})^{b/\alpha} = \overline{g}^{\phi(\omega)}\overline{g}^{\phi_{-1}/(\omega+KK_1)},\\
&w_s \leftarrow (\sigma \cdot \textstyle\prod_{i=0}^{q-1} D_i^{-\phi_i})^{1/\phi_{-1}} = \overline{g}^{1/(\omega+KK_1)},\\
&\delta_s \leftarrow KK_1 \bmod p \in \mathbb{Z}_p.
\end{aligned}$$

Since $e(\overline{g}^{\omega} \cdot \overline{g}^{\delta_s}, w_s) = e(\overline{g}, \overline{g})$, (δ_s, w_s) is the solution for the ϑ-SDH problem.

If $\Psi_{\mathscr{A}}$ does not happen, $\mathscr{B}$ randomly picks $(\delta_s, w_s) \in \mathbb{Z}_p \times \mathbb{G}$ as the solution.

As analyzed in [17], $\mathscr{B}$'s advantage is non-negligible in solving the ϑ-SDH problem. □

6. Performance Comparison

We here compare the performance of the VTCP-ABE scheme with the TCP-ABE scheme [17] and the VCP-ABE scheme [15] in the setting of key encapsulation, where the PHI data is encrypted by a symmetric encryption key χ which will be encrypted under an access policy in ABE.

6.1. Numeric Result. Tables 2 and 3 show the numeric comparison between our scheme and other two schemes [15, 17]. Let P, E, and E_1 be the overhead in executing a bilinear pairing, an exponential operation in $\mathbb{G}$ and $\mathbb{G}_1$, respectively. U denotes the system attribute universe. S_C, S_K, and I refer to the set of attributes used in encryption, key generation, and decryption, respectively. Let ℓ_{H_2} be the output length of H_2.

In Table 2, we calculate the computation cost incurred in the following phases: encryption, key generation, pre-decryption, and user decryption. The user in VCP-ABE and our VTCP-ABE expends constant size computation cost of exponential operation in $\mathbb{G}_1$. Note that our VTCP-ABE requires two additional exponential operations in the user side since that the ciphertext and transmission key need to be processed before being transmitted to the eHealth CDS.

In Table 3, the length of system public parameter, private key, and ciphertext is calculated by the number of group elements. The VCP-ABE scheme requires more public parameters which are linear with the scale of system attribute universe due to the fact that all the possible attributes need to be listed during the system initialization phase. Compared with the non-outsourced TCP-ABE scheme, our VTCP-ABE requires an additional element as the user decryption key and an output of H_2 as the verification key.

6.2. Implementation. We implement VCP-ABE scheme [15], TCP-ABE scheme [17], and the proposed VTCP-ABE on a windows 7 platform of an Intel(R) Core(TM) i5-3450 CPU at 3.10 GHz with 8.00 GB Memory. A Type A elliptic curve group is chosen from the JPBC library [38] and the order is a 512-bit prime. We mainly count the computation cost incurred by ABE relevant operations. The computation time of each algorithm is the average of 20 trials.

Figure 2 illustrates the computation cost comparison among VCP-ABE scheme, TCP-ABE scheme, and our proposed VTCP-ABE scheme.

Table 2: Computation cost comparison.

	Encryption	KeyGen	Pre-Decrypt	User Decrypt
VCP-ABE [15]	$(3\|S_C\|+1)E+1E_1$	$(\|S_K\|+3)E$	$\|I\|E_1+(2\|I\|+1)P$	$1E_1$
TCP-ABE [17]	$(5\|S_C\|+2)E+1E_1$	$(4\|S_K\|+4)E$	\	$2E+\|I\|E_1$
VTCP-ABE	$(5\|S_C\|+2)E+1E_1$	$(4\|S_K\|+4)E$	$\|I\|E_1+(3\|I\|+1)P$	$+(3\|I\|+1)P$ $3E_1$

Table 3: The parameter length comparison.

	Public Parameter	Private Key	Ciphertext
VCP-ABE [15]	$(\|U\|+2)\|\mathbb{G}\|+1\|\mathbb{G}_1\|$	$(\|S_K\|+2)\|\mathbb{G}\|+1\|\mathbb{Z}_p\|$	$(2\|S_C\|+1)\|\mathbb{G}\|+1\|\mathbb{G}_1\|+\ell_{H_2}$
TCP-ABE [17]	$6\|\mathbb{G}\|+1\|\mathbb{G}_1\|$	$(2\|S_K\|+3)\|\mathbb{G}\|+1\|\mathbb{Z}_p\|$	$(3\|S_C\|+2)\|\mathbb{G}\|+1\|\mathbb{G}_1\|$
VTCP-ABE	$6\|\mathbb{G}\|+1\|\mathbb{G}_1\|$	$(2\|S_K\|+3)\|\mathbb{G}\|+2\|\mathbb{Z}_p\|$	$(3\|S_C\|+2)\|\mathbb{G}\|+1\|\mathbb{G}_1\|+\ell_{H_2}$

Figure 2(a) shows the computation time in the initialization phase. In the three schemes, the computation cost is mainly incurred by computing the parameters $e(g,g)^{\alpha}$ and g^{a}.

Figures 2(b) and 2(c) show the computation time in the key generation phase and the encryption phase, respectively. It is observed that the key generation cost and encryption overhead in three schemes are linearly with the number of used attributes. More precisely, TCP-ABE and ours require more computation operation than VCP-ABE since that the combination of parameters u and $\hbar$ is employed to indicate an attribute.

Figure 2(d) shows the computation cost in the pre-process phase of our VTCP-ABE. Two exponential and multiplicative operations in group $\mathbb{G}$ are required in computing C_3 and L_3 no matter how many attributes are involved.

Figure 2(e) illustrates the computation cost comparison in the user decryption phase among three schemes. We can find that the user decryption cost in TCP-ABE scheme increases with the number of attributes. Thanks to the efficient outsourced decryption approach, the final decryption costs on the user side in VCP-ABE scheme and ours are significantly lower than that in TCP-ABE and independent of the attribute number.

Figure 2(f) gives the computation cost comparison in tracing the malicious users between TCP-ABE and ours. We can observe that the computation cost in both scheme grows with the number of attributes and our scheme only requires one additional exponential operation in group $\mathbb{G}_1$.

7. Delegate Extension

If a physician is in trouble to connect to the eHealth CSS and CDS, he/she can delegate someone to download the PHI ciphertext from the CSS and request the partial decrypted ciphertext from the CDS. However, the access privilege of delegated user has to be restricted. Inspired by [20, 39, 40], we employ a verifiable random function to limit the access of delegated users to maximum ξ times and propose a verifiable and traceable CP-ABE scheme with key delegation (VTDCP-ABE).

Setup. Besides generating PK and MSK as in VTCP-ABE, this algorithm calculates $\Omega = e(g,g)$ and chooses a hash function $HA_4 : \{0,1\}^* \rightarrow \mathbb{Z}_p$. The public parameter is $PK_D = (PK, \Omega, HA_4)$.

The **Encrypt**, **KeyGen**, **Pre-Process**, **Pre-Decrypt**, and **Trace** algorithms are as well as that in VTCP-ABE.

Delegate KeyGen. Given a transmission key $TK = (S, KK, KK_1, L_1, L_2, \{KK_{\tau,1}, KK_{\tau,2}\}_{\tau\in[k]})$ of an id for a set S, an identity information id_D and a set $S_D = \{AT_1, AT_2, \ldots, AT_{\hat{k}}\} \subseteq S$. This algorithm generates a delegated transmission key TK_D as follows.

Randomly select $d \in \mathbb{Z}_p$ and compute $\Gamma_{pse,1} = \Omega^{1/(d+HA_4(pse))}$, $\Gamma_{pse,2} = g^{1/(d+HA_4(pse))}$ and $\Gamma_{pse,3} = g^{d}$, where pse refers to the unique and random pseudonym of a delegated user. Set ξ as the maximum number of pre-decryption request that a delegated user can make.

Then compute

$$\begin{aligned} KK_{pse} &= KK^{1/d} = g^{\alpha/bd(a+\delta)}w^{\beta/d}, \\ KK_{pse,1} &= KK_1 = \delta, \\ L_{pse,1} &= (L_1)^{1/d} = g^{\beta/d}, \\ L_{pse,2} &= (L_2)^{1/d} = g^{a\beta/d}. \end{aligned}$$

For each $\tau \in [\hat{k}]$, compute $KK_{pse,\tau,1} = (KK_{\tau,1})^{1/d} = g^{\beta_\tau/d}$ and $KK_{pse,\tau,2} = (KK_{\tau,2})^{1/d} = (u^{A_\tau}\hbar)^{\beta_\tau/d}v^{-(a+\delta)\beta/d}$.

The ξ-times delegated transmission key is set as

$$\begin{aligned} TK_D = \Big(&\xi, pse, S_D, \Gamma_{pse,1}, \Gamma_{pse,2}, \Gamma_{pse,3}, KK_{pse}, KK_{pse,1}, \\ &L_{pse,1}, L_{pse,2}, \left\{KK_{pse,\tau,1}, KK_{pse,\tau,2}\right\}_{\tau\in[\hat{k}]}\Big). \end{aligned} \tag{3}$$

Delegate Pre-Process. The same as **Pre-Process**, the delegated user requests $CT = (CT_\chi, CT_M)$ from eHealth CSS and computes $C_{pse,3} = C_1^{KK_{pse,1}}C_2 = g^{(a+\delta)s}$ and $L_{pse,3} = (L_{pse,1})^{KK_{pse,1}}L_{pse,2} = g^{(a+\delta)r/d}$.

The delegated user then sends PCT_D and PTK_D to the eHealth CDS, where

$$\begin{aligned} &PCT_D = (\mathbb{T}, C_{pse,3}, \{C_{i,1}, C_{i,2}, C_{i,3}\}_{i\in[m]}) \\ &\text{and } PTK_D = \Big(\xi, pse, S_D, \Gamma_{pse,1}, \Gamma_{pse,2}, \Gamma_{pse,3}, KK_{pse}, L_{pse,3}, \\ &\quad \left\{KK_{pse,\tau,1}, KK_{pse,\tau,2}\right\}_{\tau\in[\hat{k}]}\Big). \end{aligned} \tag{4}$$

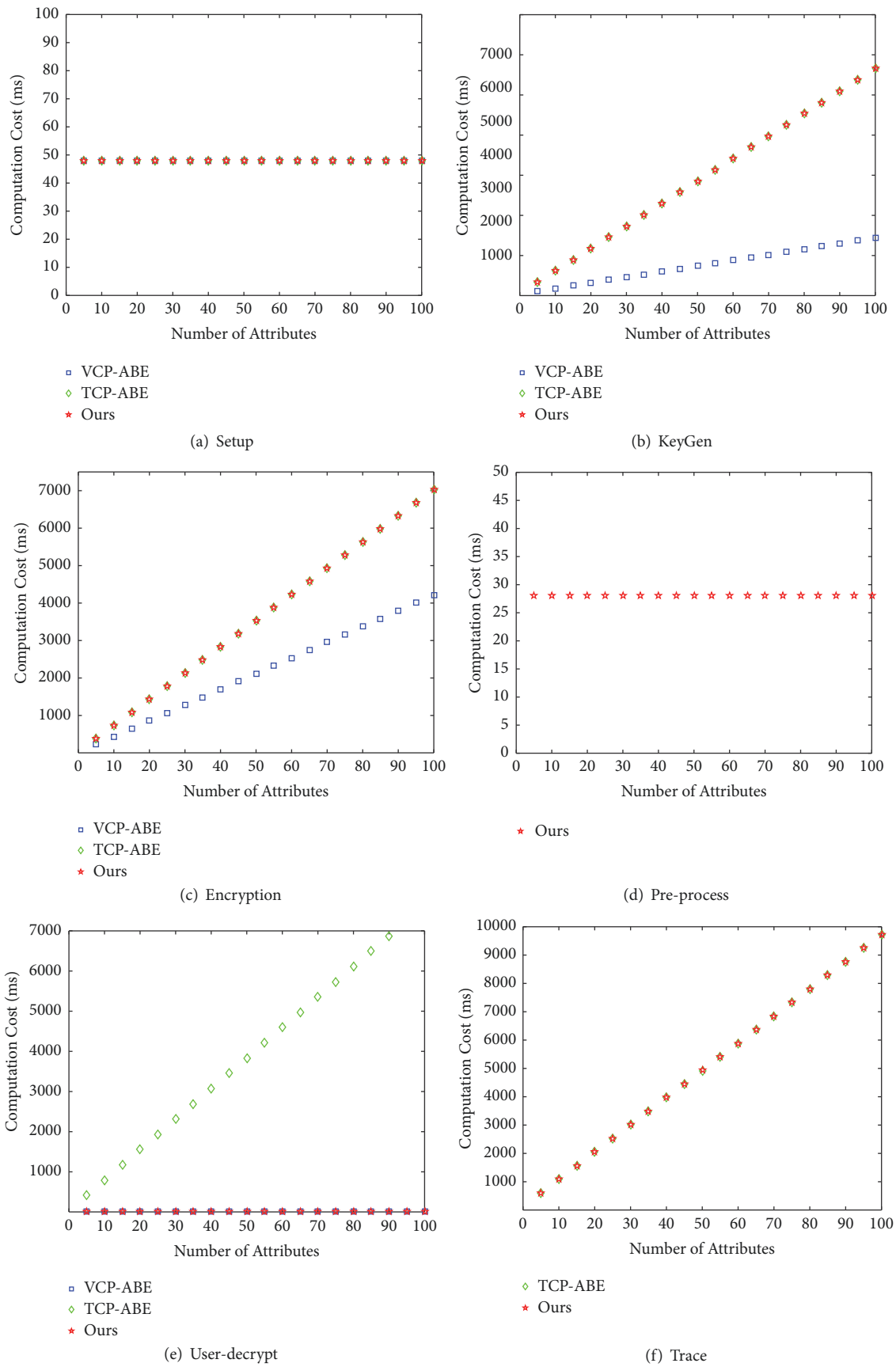

Figure 2: Comparisons of computation cost.

Delegate Pre-Decrypt. The eHealth CDS first initializes a counter $cou = 0$ and a set $S_{pse} = \{\Gamma_{pse,1}\}$ for each delegated user and stores the tuple (cou, S_{pse}) in a delegation list DL. Once receiving the Pre-Decrypt request from a delegated user, the CDS responds by the following way.

If S does not match $\mathbb{T}$, it outputs $\perp$.

Otherwise, it searches (cou, S_{pse}) in DL related to PTK_D and checks

(1) $e(g^{HA_4(pse)} \cdot \Gamma_{pse,3}, \Gamma_{pse,2}) = \Omega$ and $\Gamma_{pse,1} = e(g, \Gamma_{pse,2})$;

(2) $cou + 1 \leq \xi$;

(3) $\Gamma_{pse,1} \in S_{pse}$.

If the above three conditions do not hold, it aborts. Otherwise, it updates $cou \leftarrow cou + 1$ and computes the partial decryption ciphertext as

$$C'_D = \frac{e\left(KK_{pse}, C_{pse,3}\right)}{\prod_{i \in I}\left(e\left(L_{pse,3}, C_{i,1}\right) e\left(KK_{pse,\tau,1}, C_{i,2}\right) e\left(KK_{pse,\tau,q}, C_{i,3}\right)\right)^{\psi_i}} = e(g, \mathrm{g})^{\alpha s/bd} \quad (5)$$

Finally, the CDS responds the delegated user by $PDC_D = C'_D$. Then the delegated user gives PDC_D and CT_M to the physician.

Decrypt. If the physician interacts with the CSS and CDS directly, this algorithm acts exactly as in the **Decrypt** algorithm of VTCP-ABE. If the physician asks a delegated user to get the ciphertext and request the outsourced decryption service, χ is recovered by $\chi = C/(C'_D)^{bd}$. The verification and PHI decryption operations are identical to that of VTCP-ABE.

Since that the DK of physician and d are kept secretly, the delegated user can not obtain any content of the PHI ciphertext except a partial decrypted ciphertext.

8. Conclusion

In this paper, we have constructed a verifiable and traceable CP-ABE (VTCP-ABE) scheme for eHealth cloud applications, which also achieves the properties of large universe and delegation. With VTCP-ABE, the patient can enforce fine-grained access control over his/her PHI in a cryptographical way. Before submitting the encrypted PHI to the eHealth cloud decryption server, a pre-process on the ciphertext and transmission key is employed to preserve the identity privacy of the physician. The correctness of returned ciphertext can be efficiently verified. Moreover, the malicious physician who leaks the private key can be precisely tracked. Besides, we extend the proposed VTCP-ABE to support the delegation property, with which a resource-limited physician can authorize someone else to obtain a partial decrypted ciphertext without exposing the PHI content. The security of VTCP-ABE is proved in the selective model. The extensive experiments illustrate that our VTCP-ABE scheme efficiently achieves verifiability, traceability, and large attribute universe.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research is supported by the National Natural Science Foundation of China under Grants nos. 61502248, 61427801, u1405255, and 61602365, China Postdoctoral Science Foundation (Grant no. 2018M632350), and NUPTSF (Grant no. NY215008).

References

[1] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology—EUROCRYPT 2005*, R. Cramer, Ed., vol. 3494 of *Lecture Notes in Computer Science*, pp. 457–473, Springer Berlin Heidelberg, 2005.

[2] V. Goyal, O. Pandey, A. Sahai, 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 (CCS '06)*, pp. 89–98, New York, NY, USA, November 2006.

[3] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proceedings of the IEEE Symposium on Security and Privacy (SP '07)*, pp. 321–334, May 2007.

[4] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 1, pp. 131–143, 2013.

[5] J. Zhou, Z. Cao, X. Dong, and X. Lin, "TR-MABE: white-box traceable and revocable multi-authority attribute-based encryption and its applications to multi-level privacy-preserving e-healthcare cloud computing systems," in *Proceedings of the 34th IEEE Annual Conference on Computer Communications and Networks (IEEE INFOCOM '15)*, pp. 2398–2406, April 2015.

[6] C. Hahn, H. Kwon, and J. Hur, "Efficient attribute-based secure data sharing with hidden policies and traceability in mobile health networks," *Mobile Information Systems*, vol. 2016, Article ID 6545873, 13 pages, 2016.

[7] Q. Li, J. Ma, R. Li, X. Liu, J. Xiong, and D. Chen, "Secure, efficient and revocable multi-authority access control system in cloud storage," *Computers & Security*, vol. 59, pp. 45–59, 2016.

[8] Y. Zhang, X. Chen, J. Li, D. S. Wong, H. Li, and I. You, "Ensuring attribute privacy protection and fast decryption for outsourced data security in mobile cloud computing," *Information Sciences*, vol. 379, pp. 42–61, 2017.

[9] A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters, "Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption," in *Advances in Cryptology—EUROCRYPT 2010*, H. Gilbert, Ed., vol. 6110 of *Lecture Notes in Computer Science*, pp. 62–91, Springer Berlin Heidelberg, 2010.

[10] K. Xue, Y. Xue, J. Hong et al., "RAAC: Robust and Auditable Access Control with Multiple Attribute Authorities for Public

Cloud Storage," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 953–967, 2017.

[11] W. Li, K. Xue, Y. Xue, and J. Hong, "TMACS: a robust and verifiable threshold multi-authority access control system in public cloud storage," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 5, pp. 1484–1496, 2016.

[12] S. Wang, K. Liang, J. K. Liu, J. Chen, J. Yu, and W. Xie, "Attribute-based data sharing scheme revisited in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 8, pp. 1661–1673, 2016.

[13] M. Green, S Hohenberger, and B. Waters, "Outsourcing the decryption of abe ciphertexts," in *Proceedings of the 20th USENIX Conference on Security (SEC '11)*, pp. 34-34, USENIX Association, Berkeley, CA, USA, 2011.

[14] J.-Z. Lai, R. H. Deng, C. Guan, and J. Weng, "Attribute-based encryption with verifiable outsourced decryption," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 8, pp. 1343–1354, 2013.

[15] B. Qin, R. H. Deng, S. Liu, and S. Ma, "Attribute-based encryption with efficient verifiable outsourced decryption," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1384–1393, 2015.

[16] Z. Liu, Z. Cao, and D. S. Wong, "White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 76–88, 2013.

[17] J. Ning, X. Dong, Z. Cao, L. Wei, and X. Lin, "White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 6, pp. 1274–1288, 2015.

[18] Y. Rouselakis and B. Waters, "Practical constructions and new proof methods for large universe attribute-based encryption," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security (CCS '13)*, pp. 463–474, New York, NY, USA, November 2013.

[19] S. Lin, R. Zhang, H. Ma, and M. Wang, "Revisiting attribute-based encryption with verifiable outsourced decryption," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2119–2130, 2015.

[20] J. Ning, Z. Cao, X. Dong, K. Liang, H. Ma, and L. Wei, "Auditable sigma-time outsourced attribute-based encryption for access control in cloud computing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 1, pp. 94–105, 2018.

[21] B. Waters, *Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization*, vol. 6571 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2011.

[22] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *Proceedings of the IEEE INFOCOM*, pp. 1–9, March 2010.

[23] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07)*, pp. 456–465, New York, NY, USA, November 2007.

[24] J. Li, Q. Huang, X. Chen, S. S. M. Chow, D. S. Wong, and D. Xie, "Multi-authority ciphertext-policy attribute-based encryption with accountability," in *Proceedings of the 6th International Symposium on Information, Computer and Communications Security (ASIACCS '11)*, pp. 386–390, New York, NY, USA, March 2011.

[25] F. Guo, Y. Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "CP-ABE with constant-size keys for lightweight devices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 763–771, 2014.

[26] Q. Li, J. Ma, R. Li, J. Xiong, and X. Liu, "Large universe decentralized key-policy attribute-based encryption," *Security and Communication Networks*, vol. 8, no. 3, pp. 501–509, 2015.

[27] Q. Li, J. Ma, R. Li, J. Xiong, and X. Liu, "Provably secure unbounded multi-authority ciphertext-policy attribute-based encryption," *Security and Communication Networks*, vol. 8, no. 18, pp. 4098–4109, 2015.

[28] Q. M. Malluhi, A. Shikfa, and V. C. Trinh, "A ciphertext-policy attribute-based encryption scheme with optimized ciphertext size and fast decryption," in *Proceedings of the 2017 ACM Asia Conference on Computer and Communications Security (ASIA CCS '17)*, pp. 230–240, New York, NY, USA, April 2017.

[29] J. Ning, Z. Cao, X. Dong, K. Liang, L. Wei, and K. R. Choo, "CryptCloud+: secure and expressive data access control for cloud storage," *IEEE Transactions on Services Computing*, vol. 99, 2018.

[30] Z. Liu, Z. Cao, and D. S. Wong, "Blackbox traceable CP-ABE: how to catch people leaking their keys by selling decryption devices on eBay," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS '13)*, pp. 475–486, New York, NY, USA, November 2013.

[31] Y. Rouselakis and B. Waters, "Efficient statically-secure large-universe multi-authority attribute-based encryption," in *Financial Cryptography and Data Security*, R. Böhme and T. Okamoto, Eds., vol. 8975 of *Lecture Notes in Computer Science*, pp. 315–332, Springer Berlin Heidelberg, 2015.

[32] J. Ning, Z. Cao, X. Dong, J. Gong, and J. Chen, "Traceable CP-ABE with short ciphertexts: How to catch people selling decryption devices on ebay efficiently," in *Computer Security—ESORICS 2016*, I. Askoxylakis, S. Ioannidis, S. Katsikas, and C. Meadows, Eds., vol. 9879, pp. 551–569, Springer International Publishing, 2016.

[33] G. Yu, Z. Cao, G. Zeng, and W. Han, "Accountable ciphertext-policy attribute-based encryption scheme supporting public verifiability and nonrepudiation," in *Provable Security*, vol. 10005 of *Lecture Notes in Computer Science*, pp. 3–18, Springer International Publishing, Cham, Switzerland, 2016.

[34] A. Beimel, *Secure Schemes for Secret Sharing and Key Distribution [Mater Thesis]*, 1996.

[35] D. Boneh and X. Boyen, *Short Signatures Without Random Oracles*, vol. 3027 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2004.

[36] V. Goyal, *Reducing Trust in the PKG in Identity Based Cryptosystems*, vol. 4622 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2007.

[37] V. Shoup, "Sequences of games: a tool for taming complexity in security proofs, 2004," shoup@cs.nyu.edu 13166 received 30 Nov 2004, last revised 18 Jan 2006.

[38] A. de Caro and V. Iovino, "jPBC: Java pairing based cryptography," in *Proceedings of the 16th IEEE Symposium on Computers and Communications (ISCC '11)*, pp. 850–855, Kerkyra, Corfu, Greece, July 2011.

[39] D. Yevgeniy and A. Yampolskiy, "A verifiable random function with short proofs and keys," in *Public Key Cryptography—PKC 2005*, S. Vaudenay, Ed., pp. 416–431, Springer Berlin Heidelberg, 2005.

[40] T. H. Yuen, J. K. Liu, M. H. Au, X. Huang, W. Susilo, and J. Zhou, "k-times attribute-based anonymous access control for cloud computing," *Institute of Electrical and Electronics Engineers. Transactions on Computers*, vol. 64, no. 9, pp. 2595–2608, 2015.