IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received October 14, 2017, accepted November 15, 2017, date of publication November 24, 2017,
date of current version February 14, 2018.

Digital Object Identifier 10.1109/ACCESS.2017.2775741

Ciphertext-Policy Attribute-Based Encryption With Delegated Equality Test in Cloud Computing

QIANG WANG[1], LI PENG[1], HU XIONG[1,2], JIANFEI SUN[1], AND ZHIGUANG QIN[1]
[1]School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
[2]State Key Laboratory of Cryptology, Beijing 100878, China

Corresponding author: Hu Xiong (xionghu.uestc@gmail.com)

This work was supported in part by the National Science Foundation of China under Grant 61370026 and Grant U1401257, in part by the Science and Technology Project of Guangdong Province under Grant 2016A010101002, in part by the 13th Five-Year Plan of National Cryptography Development Fund for Cryptographic Theory of China under Grant MMJJ20170204, and in part by the Fundamental Research Funds for the Central Universities under Grant ZYGX2016J091.

ABSTRACT Public key encryption supporting equality test (referred to as PKE-ET) provides the capability of testing the equivalence between two messages encrypted under different public keys. Ciphertext-policy attribute-based encryption (CP-ABE) is a promising primitive to achieve versatile and secure data sharing in the cloud computing by providing flexible one-to-many encryption. In this paper, we first initialize the concept of CP-ABE with equality test (CP-ABE-ET) by combining the notions of PKE-ET and CP-ABE. Using ABE-ET primitive, the receiver can delegate a cloud server to perform an equivalence test between two messages, which are encrypted under different access policies. During the delegated equivalence test, the cloud server is unable to obtain any knowledge of the message encrypted under either access policy. We propose a concrete CP-ABE-ET scheme using bilinear pairing and Viète's formulas, and give the security proof of the proposed scheme formally in the standard model. Moreover, the theoretic analysis and experimental simulation reveal that the proposed scheme is efficient and practical.

INDEX TERMS Attribute based encryption with equality test, equivalence test, standard model.

I. INTRODUCTION

The popularity and pervasiveness of cloud computing have brought a revolutionary innovation to data sharing [1], [2]. With cloud computing, cloud users can not only acquire useful data more effortlessly, but can offer noteworthy benefits to society as well by sharing their own data with other users or organizations. In this way, the cost for cloud users to share data can be saved significantly. Taking the personal health record (PHR) system for example [3], [4]. Patients in PHR system can measure and gather their sensitive PHR information by using medical sensors. To share their PHR data with physicians in the hospital or other patients with similar symptoms, patients can upload their PHR data to a cloud server. Based on the collected PHR data from various patients featured with similar symptoms, one can evaluate his/herown health status accurately. Moreover, the physicians can treat such kind of disease more precisely by analyzing the PHR data from a group of patients.

No matter how favorable the cloud computing is, the unauthorized access of the sharing data should be prevented prior to the practical deployment of cloud computing to ensure the security of these data. When these data, such as e-mails, personal health records, financial transactions, are accessed by illegal entities including the cloud server itself, the data owner may suffer incalculable economic and reputational losses. Therefore, every data owner should take measures to ensure the efficient access control of their data before uploading them to clouds. Attribute-based encryption (shorten as ABE) [5], [6] is commonly considered as a flexible and versatile solution to enforce access control with fine-granularity over encrypted data in the cloud computing. So far, there are two types of ABE schemes, i.e., the ciphertext-policy ABE (CP-ABE) and key-policy ABE (KP-ABE). In CP-ABE, any user is labeled with a set of attributes and can obtain a secret key according to these attributes. And the ciphertext is generated under a given access policy. One secret key can be used to decipher a specific ciphertext only if the attributes related to this secret key satisfy the policy embedded into the ciphertext. Different from CP-ABE, the access policy and the attributes are attached to secret keys and ciphertexts of the user in a reverse order in KP-ABE. Apparently, the encryptor in the KP-ABE is unable to decide who ought to or ought not to access the data and thus CP-ABE is more suitable for achieving flexible access control over sharing

data in the environment of cloud computing. So, in this paper, we only focus on CP-ABE. By leveraging CP-ABE, the fine-grained access control for PHR system can be achieved as follows. Suppose one patient, say Alice, intends to share her PHR data m with medical researchers and attending physicians in the Massachusetts General Hospital. To enforce access control over her PHR data, Alice specifies the access policy pol = {(''Massachusetts General Hospital'') AND (''Medical Researchers'' OR ''Attending Physicians'')} and generates the ciphertext according to pol by using CP-ABE scheme. After uploading the ciphertext to the cloud server, the secure and flexible data sharing can be realized such that only the specified users can access m by using their own secret keys.

However, the standard ABE alone may hinder search functionality over encrypted data outsourced in the cloud server. Suppose $(\mathrm{Enc}(m_1, \mathsf{pol}_1), \mathrm{Enc}(m_2, \mathsf{pol}_2), \ldots, \mathrm{Enc}(m_n, \mathsf{pol}_n))$ is a set of encrypted medical data contributed by anonymous donators for research purpose. Here, each medical data m_i is encrypted under the corresponding policy pol_i such that m_i can only be accessed by cloud users who satisfy pol_i. To obtain intended information from this set of encryption, cloud user needs to download all ciphertexts and then decrypt these ciphertexts. It is easy to observe that this naive approach is inefficient and impractical. To solve this problem, the idea of ABE with keyword search (ABE-KS) [7], [8] was invented as the combination of ABE and public key encryption with keyword search (PKE-KS) [9]–[11]. In ABE-KS scheme, a receiver can delegate the searching capability to the cloud server. With a trapdoor issued by the receiver, the cloud server is able to search the stored ABE-type ciphertext once the attributes related to the trapdoor match the access structure of these ciphertexts. Meanwhile, the ciphertext is unable to be decrypted by the cloud server who owns the trapdoor. Although ABE-KS seems to be a promising solution to provide search functionality in the ABE-based access control system, it is still far from satisfactory since the trapdoor can be used to search ciphertexts only if the attributes of the trapdoor satisfy the policies of the ciphertexts. For instance, if the attributes of Bob, match policies pol_1 and pol_2, then only the encryptions of $(\mathrm{Enc}(m_1, \mathsf{pol}_1)$ and $(\mathrm{Enc}(m_2, \mathsf{pol}_2)$ can be searched by the cloud server on behalf of Bob. To obtain more flexibility about ciphertext searching, a desirable solution is to allow the cloud server to perform search functionality on ciphertexts associated with different access policies. This practical requirement naturally motivates us to design a novel attribute based encryption system with equality test (ABE-ET), which enables cloud user to search over the ABE-type ciphertexts associated with different access policies.

An example of ABE-ET is illustrated in Fig. 1, suppose the receiver (say Alice) intends to search the ABE-type ciphertexts stored in the cloud server with another receiver (say Bob). It is desirable that the searching capability can be delegated to the cloud server by Alice. Inspired by the primitive of ABE-ET, Alice first delegates her trapdoor to

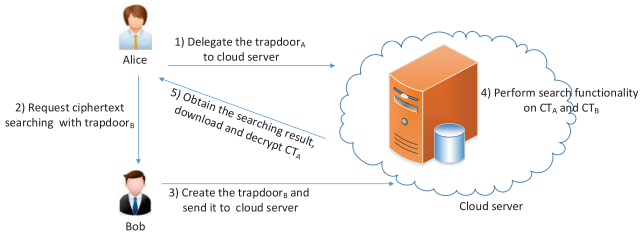

FIGURE 1. Ciphertext Searching with ABE-ET in cloud environment.

the untrusted cloud server. After receiving the request of keyword searching from Alice, Bob then creates his trapdoor using his own secret key and delivers his trapdoor to the cloud server. Equipped with the trapdoor of Alice, the cloud server could be authorized to perform search functionality on messages encrypted under different access policies. By using the ABE-ET primitive, the ABE-type ciphertexts can be searched only if the attributes related to the trapdoor match the access structure of these ciphertexts, whereas any useful information about the plaintext or secret keys of Alice or Bob can not obtained by the cloud server. Finally, Alice receives the returned searching result from the cloud server and then decrypts the ciphertext with her own secret key. In this way, the overburden of ciphertext searching could be offloaded to the cloud server with sufficient resources.

A. RELATED WORK

Public key encryption with equality test (PKE-ET), initiated by Yang *et al.* [12], enables any entity to perform an equivalence test between two messages encrypted distinct public keys. This primitive can be used to achieve flexible search functionality over ciphertexts under different public keys. To equip this primitive with authorization mechanism, a novel PKE-ET was suggested by Tang [13] to designate the entity who can carry out equality test. In [13], the authorization needs to be realized by performing an interactive protocol between the delegating users. It is easy to observe this authorization mechanism is not scalable since each user needs to interact with other users in the system to delegate the capability of equivalence test power. Thus, the notion of PKE-ET scheme with delegated equality test (PKE-DET) was introduced by Tang [14] and Ma *et al.* [16] respectively in which each user can issue the delegation token independently to the cloud server. After that, Tang [15] formulated an enhanced PKE-ET scheme by allowing two proxies jointly to execute the equality test and impede off-line message recovery attacks. Subsequently, Huang *et al.* [17] introduced a novel PKE with authorized equality test (PKE-AET) such that a user can authorize warrants on all of his/her ciphertexts or a specified ciphertext. To feature the authorization with more flexibility. An efficient PKE-ET scheme was proposed by Ma *et al.* [18] in which four kinds of authorization are contained. As a special kind of PKE, identity-based encryption (IBE) has attracted a huge amount of interest by simplifying public key certificate management [19]–[21]. Subsequently, an identity-based encryption scheme with

outsourced equality test (IBE-ET) was formulated by Ma [22] by incorporating the concept of IBE and PKE-ET scheme. Following Ma's work [22], a semi-generic construction of IBE-ET scheme was introduced by Lee *al et.* [23] to strengthen the security requirement. Very recently, to improve the efficiency of Ma's IBE-ET scheme, Wu *et al.* [24] proposed a novel IBE-ET scheme which is more fitting for mobile cloud environment.

PKE with keyword search (PKE-KS), firstly formulated in [9], achieves the functionality to perform an equivalence test between keywords embedded in a ciphertext or a tag. IBE with keyword search (IBE-KS), initially introduced in [25], is an extension of PKE-KS to enjoy the merits of IBE scheme and PKE-KS scheme. As an extension of IBE, ABE has also attracted a lot of concern since it can provide fine-grained access control [26]–[28]. Similarly, the attribute-based encryption with keyword search (ABE-KS) [7], [8] has been proposed as the best-of-two-worlds to enjoy the merits of ABE scheme and PKE-KS scheme. However, the above three primitives only allow performing an equivalence test on ciphertexts under a fixed public key, a fixed identity and a fixed access policy. Recently, Zhu *et al.* proposed a KP-ABE with ET scheme [29] that allows testing whether the ciphertexts contain the same information under different attribute sets. However, it only supports monotonic access structure which limits the express of access policy. Besides, it only achieves one-way against chosen-ciphertext attack (OW-CCA) in the random oracle model. As far as we know, CP-ABE with equality test has not been treated to support the functionality to perform an equivalence test on ciphertexts under different access policies in the literature so far.

B. OUR CONTRIBUTION

To provide search functionality on CP-ABE-type ciphertext flexibly, the construction of CP-ABE with equality test (CP-ABE-ET) has been proposed in this paper. To summarize, the contributions are three-fold as follows:

1) We, for the first time, introduce the idea of PKE-ET into the CP-ABE-based setting to enjoy the best-of-the-two-worlds. Specifically, a semi-trusted entity (such as cloud server) in ABE-ET can be delegated to execute an equivalence test on CP-ABE-type ciphertexts encrypted under different access policies. Meanwhile, this delegated entity cannot learn any information about the plaintext.
2) We then propose a concrete CP-ABE-ET scheme using the bilinear pairing and Viète's formulas technique, which features with constant-size ciphertext. Compared with the scheme in [29], our scheme supports more expressive access policy that includes positive, negative as well as wildcard attributes. And it has been proved to be selective security against a chosen plaintext attack in the standard model under Decisional n-Bilinear Diffie-Hellman Exponent (n-DBHE) assumption.
3) Finally, both the theoretic analysis and experimental simulation indicate our suggested scheme is efficient and practical.

C. ORGANIZATION

In section II, some preliminaries are presented such as Viète's formulas, bilinear map, AND-Gate access structure, the underlying assumption, the formal definition of our ABE-ET and security model. The construction of our suggested ABE-ET scheme is concretely described in section III. We introduce the rigorous security proof of the formulated ABE-ET under n-Decision Bilinear Diffie-Hellman Exponent assumption (n-DBHE) in the standard model in section IV. In section V, the performance comparison of existing IBE-ETs and our ABE-ET scheme are described. A conclusion for our paper is summarized in section VI.

II. PRELIMINARIES

This section briefly reviews Viète's formulas, bilinear map, AND-Gate access structure, the underlying assumption, the formal definition of our ABE-ET and security model, which will be used throughout the whole paper.

A. THE Viète's FORMULAS

Let $\overrightarrow{w} = (w_1, w_2, \ldots, w_L)$ and $\overrightarrow{u} = (u_1, u_2, \ldots, u_L)$ be two vectors such that the former vector includes both alphabets and wildcards, whereas the latter vector only contains alphabets. A set $S = \{k_1, \ldots, k_n\} \subset \{1, \ldots, L\}$ stands for the wildcards positions of the former vector $\overrightarrow{w}$. According to the statement $((w_i = u_i) \vee (w_i = *)$ for $i = 1, \ldots, L)$, it is easy to have

$$\sum_{i=1, i\notin S}^{L} w_i \prod_{k\in S}(i-k) = \sum_{i=1}^{L} u_i \prod_{k\in S}(i-k) \tag{1}$$

Expand $\prod_{k\in S}(i-k) = \prod_{j=1}^{n} a_j i^j$ such that the coefficients a_j are generated based on the set S. Then, the following equation can be derived from (1):

$$\sum_{i=1, i\notin S}^{L} w_i \prod_{k\in S}(i-k) = \sum_{j=0}^{n} a_j \sum_{i=1}^{L} u_i i^j \tag{2}$$

In order to hide the calculation in (2), we randomly pick a group element A_i and regard w_i, u_i as the exponents of A_i, then the following equation can be derived from (2)

$$\prod_{i=1, i\notin S}^{L} A_i^{w_i \prod_{k\in S}(i-k)} = \prod_{j=0}^{n} \left(\prod_{i=1}^{L} A_i^{u_i i^j}\right)^{a_j} \tag{3}$$

According to the Viète's formulas [26], [30], [31], [33], the coefficients a_j in (2) can be reconstructed by

$$a_{n-j} = (-1)^j \sum_{1\le i_1 < i_2 < \ldots < i_j \le n} k_{i_1} k_{i_2} \ldots k_{i_j}, 0 \le j \le n = |S|. \tag{4}$$

If taking $S = \{k_1, k_2, k_3, k_4\}$ as an example, then the polynomial can be represented as $(i-k_1)(i-k_2)(i-k_3)(i-k_4)$, therefore, we can obtain the coefficient values as $a_4 = 1$, $a_3 = -(k_1+k_2+k_3+k_4)$, $a_2 = (k_1k_2+k_1k_3+k_1k_4+k_2k_3+k_2k_4+k_3k_4)$, $a_1 = -(k_1k_2k_3+k_1k_2k_4+k_1k_3k_4+k_2k_3k_4)$, $a_0 = k_1k_2k_3k_4$.

B. BILINEAR MAP

Definition 1 (Bilinear Map): $(\mathbb{G}_1, \mathbb{G}_2)$ is a bilinear group pair in case there exists a computable map $\mathrm{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ equipped with the following features:

- Cyclic multiplicative groups $\mathbb{G}_1$ and $\mathbb{G}_2$ own the same prime order p.
- Bilinearity: For any $g, h \in \mathbb{G}_1$ and any $u, v \in \mathbb{Z}_p$, $e(g^u, h^v) = e(g, h)^{uv}$.
- Non-degeneracy: Let g denote one generator of $\mathbb{G}$, $e(g, g) \neq 1$.

C. AND-GATE ACCESS STRUCTURE

Let $Att = \{Att_1, Att_2, \ldots, Att_L\}$ represent the universe of attributes, where each Att_i, for $i \in \{1, \ldots, L\}$, contains two potential values, i.e., positive value " + " and negative value " − ". Each user in the system is labeled with an attribute list $U = \{U_1, \ldots, U_L\}$, where each attribute $U_i \in \{\text{" + "}, \text{" − "}\}$ for $i \in \{1, \ldots, L\}$. On the other hand, we denote an AND-gate access policy by $W = \{W_1', \ldots, W_L'\}$, where each $W_i' \in \{\text{" + "}, \text{" − "}, \text{" ∗ "}\}$. The wildcard " ∗ " in the access policy demonstrates that the attribute value U_i does not make sense (both " + " and " − " are accepted for U_i) in case access policy value $W_i = *$. In addition, we denote a specified attribute list U matches (or mismatches) a particular policy W by $U \models W$ (or $U \not\models W$).

D. HARD ASSUMPTION

Definition 2 (DLIN Assumption): Given a tuple $(g, g^a, g^b, g^{ac}, g^d, T) \in \mathbb{G}_1^6$, where g is a generator of $\mathbb{G}_1$ and $a, b, c, d \in \mathbb{Z}_p$, the Decisional Linear (DLIN) problem is to determine whether T is randomly selected in $\mathbb{G}_1$ or $T = g^{b(c+d)}$. The advantage of DLIN problem-solving for adversary $\mathcal{A}$ can be defined as follow: $Adv^{DLIN} = |\mathsf{Pr}[T = g^{b(c+d)}] - \mathsf{Pr}[T \overset{R}{\leftarrow} \mathbb{G}_1]|$.

E. FORMAL DEFINITION OF OUR ABE-ET SCHEME

Our proposed ABE-ET scheme is comprised of six algorithms: **Setup**, **KeyGen**, **Trapdoor**, **Encrypt**, **Decrypt**, **Test**. These algorithms are defined as follows:

- **Setup(1^λ):** Produce the master key **MSK** and the public parameter **PP** based on a security parameter 1^λ.
- **KeyGen(PP, MSK, AL):** Create the decryption secret key **SK** for users based on the public parameter **PP**, the master key **MSK** and an attribute list **AL**.
- **Trapdoor(PP, AL):** Generate the trapdoor **TD** for users based on the public parameter **PP**, an attribute list **AL** and the secret key **SK**.
- **Encrypt(PP, M, W):** Produce the ciphertext based on the public parameter **PP**, a plaintext message M and the predefined access structure W.
- **Decrypt(CT,SK):** Decipher the ciphertext **CT** using the decryption secret key **SK**.
- **Test($\mathsf{CT_A}$,$\mathsf{TD_A}$, $\mathsf{CT_B}$,$\mathsf{TD_B}$):** Decide whether M_A in $\mathsf{CT_A}$ is the same with M_B in $\mathsf{CT_B}$ using the trapdoor $\mathsf{TD_A}$ and the trapdoor $\mathsf{TD_B}$.

F. SECURITY MODEL

Definition 3: An ABE-ET scheme is secure against selective chosen-plaintext attack via the security game between an adversary $\mathcal{A}$ and a challenger $\mathcal{B}$.

- **Init**: The targeted challenge attribute list **AL** are picked by the adversary $\mathcal{A}$.
- **Setup**: The security parameter 1^λ is first given and then the **Setup** algorithm is executed by $\mathcal{B}$ to create the master key **MSK** and the public parameter **PP** which is delivered to $\mathcal{A}$.
- **Phase 1&2**: $\mathcal{A}$ selects an access structure W and makes secret key queries to produce a secret key **SK** and a trapdoor **TD**. For **AL** $\models W$ or **AL** $\not\models W$, the $\mathcal{B}$ creates corresponding secret key **SK** and corresponding trapdoor **TD** for the adversary $\mathcal{A}$.
- **Challenge**: After obtaining M_0 and M_1 with equal length from $\mathcal{A}$, $\mathcal{B}$ replies the $\mathcal{A}$ with the challenge ciphertext **CT** by running **Encrypt**(**PP**, W, M_ζ), where $\zeta \in \{0, 1\}$.
- **Guess**: A guess $\zeta' \in \{0, 1\}$ on ζ is replied by $\mathcal{A}$ and $\mathcal{A}$ wins the security game if $\zeta = \zeta'$.

III. CONCRETE CONSTRUCTION

Our formulated scheme is composed of the following various procedures: **Setup**, **KeyGen**, **Trapdoor**, **Encrypt**, **Decrypt**, **Test**. The detailed description of our construction is elaborated as followed.

A. SETUP(1^λ)

Produce the master key **MSK** and the public parameter **PP** as follows based on a security parameter 1^λ.

1) Choose a bilinear map group $\mathbb{BM} = (\mathbb{G}_1, \mathbb{G}_2, p, g, e)$ and two hash functions $\mathcal{H}_1 : \mathbb{G}_2 \to \mathbb{G}_1 \times \mathbb{Z}_p$, $\mathcal{H}_2 : \mathbb{G}_2 \to \mathbb{G}_1$.
2) Choose random generators $r_1, \ldots, r_N \in_R \mathbb{Z}_p$ and compute $R_i = g^{r_i}$, where N denotes the number of system attributes and i ranges from 1 to N.
3) Pick $\alpha, \alpha', \gamma_1, \gamma_2, \gamma_3 \in \mathbb{Z}_p$ and $W_1, W_2 \in \mathbb{G}_1$ randomly and compute

$$\begin{aligned} u_1 &= e(g, W_1)^{\alpha\gamma_1} e(g, W_2)^{\alpha\gamma_1}, \\ v_1 &= e(g, W_1)^{\alpha\gamma_2} e(g, W_2)^{\alpha\gamma_2}, \\ u_2 &= e(g, W_1)^{\alpha'\gamma_1} e(g, W_2)^{\alpha'\gamma_1}, \\ v_2 &= e(g, W_1)^{\alpha'\gamma_3} e(g, W_2)^{\alpha'\gamma_3}. \end{aligned}$$

4) Publish $\mathsf{PP} = (\mathbb{BM}, g^\alpha, g^{\alpha'}, u_1, v_1, u_2, v_2, R_1, \ldots, R_N, \mathcal{H}_1, \mathcal{H}_2)$ and keep $\mathsf{MSK} = (\alpha, \alpha', \gamma_1, \gamma_2, \gamma_3, r_1, \ldots, r_N)$ secure.

B. ENCRYPT(PP, M, W)

Taking as input public parameter PP, a cleartext $M \in \mathbb{G}_1$ and one access policy W, which contains: $l_1 \leqslant L_1$ wildcards occur at positions $J = (\omega_1, \ldots, \omega_{l_1})$; $l_2 \leqslant L_2$ positive attributes occur at positions $X = (x_1, \ldots, x_{l_2})$; $l_3 \leqslant L_3$ negative attributes occur at positions $Y = (y_1, \ldots, y_{l_3})$; By means of the Viète's formulas, for the wildcard positions $\{\omega_k\}_{k=1,\ldots,l_1}$ in access structure, compute $\{a_{\omega_k}\}$ and set $t_\omega = \sum_{k=0}^{l_1} a_{\omega_k}$. This algorithm creates the ciphertext CT as follows:

1) Choose $z_1, z_2, z \in \mathbb{Z}_p$ and compute

$$C_0 = M \| z \oplus \mathcal{H}_1(u_1^{z_1} v_1^{z_2}), C_1 = M^z \cdot \mathcal{H}_2(u_2^{z_1} v_2^{z_2}),$$
$$C_2 = g^{\frac{\alpha z_1}{t_\omega}}, C_3 = g^{\frac{z_2}{t_\omega}}, C_2' = g^{\frac{\alpha' z_1}{t_\omega}}, C_3' = g^z$$
$$C_4 = (W_1 \prod_{i \in X} R_i^{\frac{\prod_{k=0}^{l_1}(i-\omega_k)}{t_\omega}})^{z_1+z_2},$$
$$C_5 = (W_2 \prod_{i \in Y} R_i^{\frac{\prod_{k=0}^{l_1}(i-\omega_k)}{t_\omega}})^{z_1+z_2}.$$

2) Return $\mathsf{CT} = (C_0, C_1, C_2, C_2', C_3, C_3', C_4, C_5, J)$.

C. KEYGEN(PP, MSK, AL)

Taking as input public parameter PP, the master secret key MSK and a list of attributes AL which contains: $l_2 \leqslant L_2$ positive attributes appear at positions $X' = (x_1', \ldots, x_{l_2}')$; $l_3 \leqslant L_3$ negative attributes appear at positions $Y' = (y_1', \ldots, y_{l_3}')$; By means of the Viète's formulas, for all positive positions $\{x_i'\}_{i\in\{1,\ldots,l_2\}}$ and negative positions $\{y_j'\}_{j\in\{1,\ldots,l_2\}}$, calculate $\{a_{x_i'}\}$,$\{a_{y_j'}\}$ and set $t_x' = \sum_{i=0}^{l_2} a_{x_i'}$, $t_y' = \sum_{j=0}^{l_2} a_{y_j'}$. This algorithm produces the decryption secret key SK as follows:

1) Randomly choose s, compute $s_1 = \gamma_1 + s$, $s_2 = \gamma_2 + s$, $s_3 = \gamma_3 + s$ and create the secret key as follows:

$$sk_1 = g^{\frac{\alpha s}{t_x'}}, sk_2 = g^{\frac{\alpha s}{t_y'}}, sk_1' = g^{\frac{\alpha' s}{t_x'}}, sk_2' = g^{\frac{\alpha' s}{t_y'}}$$
$$sk_3 = \{sk_{3,0}, sk_{3,1}, \ldots, sk_{3,L_1}\}$$
$$= \{W_1^{s_1} \prod_{i\in X'} g^{sr_i}, W_1^{s_1} \prod_{i\in X'} g^{sr_i i}, \ldots, W_1^{s_1} \prod_{i\in X'} g^{sr_i i^{L_1}}\},$$
$$sk_3' = \{sk_{3,0}', sk_{3,1}', \ldots, sk_{3,L_1}'\}$$
$$= \{W_1^{\alpha s_2} \prod_{i\in X'} g^{s\alpha r_i}, W_1^{\alpha s_2} \prod_{i\in X'} g^{s\alpha r_i i}, \ldots, W_1^{\alpha s_2} \prod_{i\in X'} g^{s\alpha r_i i^{L_1}}\},$$
$$sk_3'' = \{sk_{3,0}'', sk_{3,1}'', \ldots, sk_{3,L_1}''\}$$
$$= \{W_1^{\alpha' s_3} \prod_{i\in X'} g^{s\alpha' r_i}, W_1^{\alpha' s_3} \prod_{i\in X'} g^{s\alpha' r_i i}, \ldots, W_1^{\alpha' s_3} \prod_{i\in X'} g^{s\alpha' r_i i^{L_1}}\},$$
$$sk_4 = \{sk_{4,0}, sk_{4,1}, \ldots, sk_{4,L_1}\}$$
$$= \{W_2^{s_1} \prod_{i\in Y'} g^{sr_i}, W_2^{s_1} \prod_{i\in Y'} g^{sr_i i}, \ldots, W_2^{s_1} \prod_{i\in Y'} g^{sr_i i^{L_1}}\},$$
$$sk_4' = \{sk_{4,0}', sk_{4,1}', \ldots, sk_{4,L_1}'\}$$
$$= \{W_2^{\alpha s_2} \prod_{i\in Y'} g^{s\alpha r_i}, W_2^{\alpha s_2} \prod_{i\in Y'} g^{s\alpha r_i i}, \ldots, W_2^{\alpha s_2} \prod_{i\in Y'} g^{s\alpha r_i i^{L_1}}\},$$
$$sk_4'' = \{sk_{4,0}'', sk_{4,1}'', \ldots, sk_{4,L_1}''\}$$
$$= \{W_2^{\alpha' s_3} \prod_{i\in Y'} g^{s\alpha' r_i}, W_2^{\alpha' s_3} \prod_{i\in Y'} g^{s\alpha' r_i i}, \ldots, W_2^{\alpha' s_3} \prod_{i\in Y'} g^{s\alpha' r_i i^{L_1}}\}.$$

2) Set $\mathsf{SK} = (sk_1, sk_2, sk_1', sk_2', sk_3, sk_3', sk_3'', sk_4, sk_4', sk_4'')$.

D. TRAPDOOR(PP,AL,SK)

Taking as input public parameter PP, a list of attributes AL and the secret key SK, this algorithm produces trapdoor TD as follows:

1) Set $td_1 = sk_1', td_2 = sk_2', td_{3,i} = sk_{3,i}, td_{3,i}' = sk_{3,i}'', td_{4,i} = sk_{4,i}, td_{4,i}' = sk_{4,i}''$.
2) Output $\mathsf{TD} = (td_1, td_2, (td_{3,i}, td_{3,i}', td_{4,i}, td_{4,i}')_{i\in[0,L_1]})$.

E. DECRYPT(CT,SK)

Taking as input the ciphertext CT, the decryption secret key SK, this algorithm recovers the plaintext message M by executing the following step:

$$V_1 = \frac{e(\prod_{j=1}^{l_1} sk_{3,j}^{a_{\omega_j}}, C_2) e(\prod_{j=1}^{l_1} sk_{3,j}'^{a_{\omega_j}}, C_3)}{e(sk_1, C_4)^{t_{x'}}} \times \frac{e(\prod_{j=1}^{l_1} sk_{4,j}^{a_{\omega_j}}, C_2) e(\prod_{j=1}^{l_1} sk_{4,j}'^{a_{\omega_j}}, C_3)}{e(sk_2, C_5)^{t_{y'}}},$$
$$V_2 = \frac{e(\prod_{j=1}^{l_1} sk_{3,j}^{a_{\omega_j}}, C_2') e(\prod_{j=1}^{l_1} sk_{3,j}''^{a_{\omega_j}}, C_3)}{e(sk_1', C_4)^{t_{x'}}} \times \frac{e(\prod_{j=1}^{l_1} sk_{4,j}^{a_{\omega_j}}, C_2') e(\prod_{j=1}^{l_1} sk_{4,j}''^{a_{\omega_j}}, C_3)}{e(sk_2', C_5)^{t_{y'}}},$$
$$M \| z = C_0 \oplus \mathcal{H}_1(V_1).$$

If $C_3' = g^z$ and $C_1 / M^z = \mathcal{H}_2(V_2)$, it recovers the plaintext M. Here the above-mentioned a_k are coefficients in the unfolding polynomial $\prod_{k=0}^{l_1}(i - \omega_k)$.

F. TEST(CT_A,TD_A,CT_B, TD_B)

Taking as input A's ciphertext $\mathsf{CT_A}$, A's trapdoor $\mathsf{TD_A}$ and B's ciphertext $\mathsf{CT_B}$, B's trapdoor $\mathsf{TD_B}$, this algorithm compute as

follows to decide whether $M_A = M_B$:

$$Q'_A = \frac{e(\prod_{j=1}^{l_1} td_{\{3,j\},A}^{a_{\omega_j,A}}, C'_{2,A})e(\prod_{j=1}^{l_1} td'^{\,a_{\omega_j,A}}_{\{3,j\},A}, C_{3,A})}{e(td_{1,A}, C_{4,A})^{t_{x'_A}}}$$
$$\times \frac{e(\prod_{j=1}^{l_1} td_{\{4,j\},A}^{a_{\omega_j,A}}, C'_{2,A})e(\prod_{j=1}^{l_1} td'^{\,a_{\omega_j,A}}_{4,j}, C_{3,A})}{e(td_{2,A}, C_{5,A})^{t_{y'_A}}},$$
$$Q_A = C_1/\mathcal{H}_2(Q'_A).$$
$$Q'_B = \frac{e(\prod_{j=1}^{l_1} td_{\{3,j\},B}^{a_{\omega_j,B}}, C'_{2,B})e(\prod_{j=1}^{l_1} td'^{\,a_{\omega_j,B}}_{\{3,j\},B}, C_{3,B})}{e(td_{1,B}, C_{4,B})^{t_{x'_B}}}$$
$$\times \frac{e(\prod_{j=1}^{l_1} td_{\{4,j\},B}^{a_{\omega_j,B}}, C'_{2,A})e(\prod_{j=1}^{l_1} td'^{\,a_{\omega_j,B}}_{4,j}, C_{3,B})}{e(td_{2,B}, C_{5,B})^{t_{y'_B}}},$$
$$Q_B = C_1/\mathcal{H}_2(Q'_B).$$

and returns 1 if there holds $e(Q_A, C'_{3,B}) = e(Q_B, C'_{3,A})$. Otherwise, it outputs 0.

G. CORRECTNESS OF DECRYPTION

$$V_1 = \frac{e(\prod_{j=1}^{l_1} sk_{3,j}^{a_{\omega_j}}, C_2)e(\prod_{j=1}^{l_1} sk'^{\,a_{\omega_j}}_{3,j}, C_3)}{e(sk_1, C_4)^{t_{x'}}}$$
$$\times \frac{e(\prod_{j=1}^{l_1} sk_{4,j}^{a_{\omega_j}}, C_2)e(\prod_{j=1}^{l_1} sk'^{\,a_{\omega_j}}_{4,j}, C_3)}{e(sk_2, C_5)^{t_{y'}}}$$
$$= \frac{e(\prod_{j=1}^{l_1}(\prod_{i\in X'} g^{sr_i i^j})^{a_{\omega_j}}, g^{\frac{\alpha}{t_\omega}})^{z_1+z_2} e(W_1, g)^{\alpha(s_1z_1+s_2z_2)}}{e(g^{\frac{\alpha s}{t'_x}}, (W_1 \prod_{i\in X} R_i^{\frac{\prod_{k=0}^{l_1}(i-\omega_k)}{t_\omega}})^{z_1+z_2})^{t_{x'}}}$$
$$\times \frac{e(\prod_{j=1}^{l_1}(\prod_{i\in Y'} g^{sr_i i^j})^{a_{\omega_j}}, g^{\frac{\alpha}{t_\omega}})^{z_1+z_2} e(W_2, g)^{\alpha(s_1z_1+s_2z_2)}}{e(g^{\frac{\alpha s}{t'_y}}, (W_2 \prod_{i\in Y} R_i^{\frac{\prod_{k=0}^{l_1}(i-\omega_k)}{t_\omega}})^{z_1+z_2})^{t_{y'}}}$$
$$= e(g, W_1)^{\alpha\gamma_1 z_1+\alpha\gamma_2 z_2} e(g, W_2)^{\alpha\gamma_1 z_1+\alpha\gamma_2 z_2}$$
$$= u_1^{z_1} v_1^{z_2}$$

Similar to the above calculation process, $V_2 = u_2^{z_1} v_2^{z_2}$ can be computed. Subsequently, $M\|z = C_0 \oplus \mathcal{H}_1(V_1)$ can be obtained and the plaintext M can be recovered if $C'_3 = g^z$ and $C_1/M^z = \mathcal{H}_2(V_2)$. Meanwhile, the computation process of **Test** is similar to the **Decrypt** and here we omit it.

IV. SECURITY PROOF

Lemma 1 (Selective IND-CPA Security for ABE-ET Scheme): Suppose an adversary $\mathcal{A}$ can win the IND-CPA security game, then the challenger $\mathcal{B}$ can solve the decisional DLIN problem by interacting with $\mathcal{A}$.

Proof: Assume that our ABE-ET scheme can be broken by the adversary $\mathcal{A}$ with non-negligible advantage, then another algorithm $\mathcal{B}$ can be created to solve the DLIN problem by interacting games with $\mathcal{A}$ with non-negligible advantage. Based on input a tuple$(g, g^a, g^b, g^{ac}, g^{a'}, g^{a'c}, g^d, T) \in \mathbb{G}_1$, the $\mathcal{B}$ calls $\mathcal{A}$ and simulates the game to determine whether $T = g^{b(c+d)}$ or T is a random number in $\mathbb{G}_1$.

To improve the readability of the security proof, we briefly demonstrate the basic principle of the reduction as follows. In the beginning of the security game, $\mathcal{B}$ generates (W_1, W_2), (u_1, v_1) and (u_2, v_2) by embedding g^b, g^a and $g^{a'}$ into the public parameters respectively. In this manner, the master secret key is implicitly set as $\alpha = a, \alpha' = a', \gamma_1 = \sigma_1 - \sigma_2, \gamma_2 = \frac{\sigma_3}{a} - \sigma_2, \gamma_3 = \frac{\sigma_3}{a'} - \sigma_2$, where $\sigma_1, \sigma_2, \sigma_3$ are randomly chosen by $\mathcal{B}$ from $\mathbb{Z}_p$. (Please refer to our proof for the detail of the setting.) During the simulation, $\mathcal{B}$, who has no knowledge of the master secret key, can generate the secret key for $\mathcal{A}$ by utilizing the public parameters W_1, W_2 and g^a directly. The trick about the generation of secret key is to assign σ_2 and σ_3 as s and $\alpha(\gamma_2 + \sigma_2)$, respectively. In the challenge phase, $\mathcal{B}$ generates the challenging ciphertext by using T and public parameters such that z_1 and z_2 are implicitly set as c and d respectively. At the end of the security game, $\mathcal{B}$ is able to solve the DLIN problem successfully iff $\mathcal{A}$ can output the correct guess of ζ.

Now, we begin to describe the concrete security proof in detail.

Init: In this phase, a challenge access structure $W^* = \{W_1^*, \ldots, W_L^*\}$ is picked by the adversary $\mathcal{A}$ such that W^* contains $l_1 \le L_1$ wildcards which appear in positions $J^* = (\omega_1^*, \ldots, \omega_{l_1}^*)$, $l_2 \le L_2$ positive attributes which appear in positions $X^* = (x_1^*, \ldots, x_{l_2}^*)$ and $l_3 \le L_3$ negative attributes which appear in positions $Y^* = (y_1^*, \ldots, y_{l_2}^*)$.

Setup: Algorithm $\mathcal{B}$ chooses an upper bound $l_1 \le L_1 \le N$ for the number of wildcards in an access structure, then produces **MSK** and **PP** by randomly picking $\sigma_1, \sigma_2, \sigma_3 \in_R \mathbb{Z}_p$, $\beta_0, \beta_1, \{r'_i\}_{1\le i\le N} \in_R \mathbb{Z}_p$, $\mathcal{H}_1 : \mathbb{G}_2 \to \mathbb{G}_1 \times \mathbb{Z}_p$, $\mathcal{H}_2 : \mathbb{G}_2 \to \mathbb{G}_1$. $\mathcal{B}$ also computes by means of the Viète's formulas $\{a_{\omega_j}\}_{\omega_j\in J}$, sets $t_\omega = \sum_{j=0}^{l_1} a_{\omega_j}$, and then simulates the public parameters, the master secret key as follows:

$$W_1 = (g^b)^{\beta_0} g^{-\sum_{att_i\in W_i^*, i\in X} \frac{r'_i \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}},$$
$$W_2 = (g^b)^{\beta_1} g^{-\sum_{att_i\in W_i^*, i\in Y} \frac{r'_i \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}},$$

$$R_i = g^{r_i} = \begin{cases} g^{r_i'} & att_i = W_i^* \\ g^{\frac{r_i'}{\sum_{att_m \in W_i^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}} & att_i \neq W_i^*, \end{cases}$$

$$u_1 = e(g^a, W_1)^{\sigma_1-\sigma_2} e(g^a, W_2)^{\sigma_1-\sigma_2},$$
$$v_1 = e(g^{\sigma_3}(g^a)^{-\sigma_2}, W_1) e(g^{\sigma_3}(g^a)^{-\sigma_2}, W_2),$$
$$u_2 = e(g^{a'}, W_1)^{\sigma_1-\sigma_2} e(g^{a'}, W_2)^{\sigma_1-\sigma_2},$$
$$v_2 = e(g^{\sigma_3}(g^{a'})^{-\sigma_2}, W_1) e(g^{\sigma_3}(g^{a'})^{-\sigma_2}, W_2).$$

After that, the $\mathcal{B}$ delivers $\mathsf{PP} = (\mathbb{BM}, R_1, \ldots, R_N, u_1, v_1, u_2, v_2, W_1, W_2, g^a, g^{a'}, \mathcal{H}_1, \mathcal{H}_2)$ to the adversary $\mathcal{A}$. The responding master secret key is $\mathsf{MSK} = (\alpha = a, \alpha' = a', \gamma_1 = \sigma_1 - \sigma_2, \gamma_2 = \frac{\sigma_3}{a} - \sigma_2, \gamma_3 = \frac{\sigma_3}{a'} - \sigma_2, r_1, \ldots, r_N)$.

Phase 1&2: After receiving the attribute list AL, $\mathcal{B}$ creates corresponding secret key for $\mathcal{A}$. Assume the attribute list AL includes: $l_2 \leq L_2$ positive attributes which appear at positions $X = (x_1, \ldots, x_{l_2})$ and $l_3 \leq L_3$ positive attributes which appear at positions $Y = (y_1, \ldots, y_{l_3})$. By means of the Viète's formulas, for all positive positions $\{x_i'\}_{i\in\{1,\ldots,l_2\}}$ and negative positions $\{y_j'\}_{j\in\{1,\ldots,l_2\}}$, calculate $\{a_{x_i'}\}$,$\{a_{y_j'}\}$ and set $t_x' = \sum_{i=0}^{l_2} a_{x_i'}$, $t_y' = \sum_{j=0}^{l_2} a_{y_j'}$.

$$sk_1 = (g^a)^{\frac{\sigma_2}{t_x'}}, sk_2 = (g^a)^{\frac{\sigma_2}{t_y'}}, sk_1' = (g^{a'})^{\frac{\sigma_2}{t_x'}}, sk_2' = (g^{a'})^{\frac{\sigma_2}{t_y'}},$$

$$sk_3 = \{sk_{3,0}, sk_{3,1}, \ldots, sk_{3,L_1}\}$$

$$= \{W_1^{\sigma_1} \prod_{att_i\in W^*, i\in X} g^{\sigma_2 r_i'} \prod_{att_i\notin W^*, i\in X} g^{\frac{\sigma_2 r_i'}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$W_1^{\sigma_1} \prod_{att_i\in W^*, i\in X} g^{\sigma_2 r_i' i} \prod_{att_i\notin W^*, i\in X} g^{\frac{\sigma_2 r_i' i}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$\ldots,$$

$$W_1^{\sigma_1} \prod_{att_i\in W^*, i\in X} g^{\sigma_2 r_i' i^{L_1}} \prod_{att_i\notin W^*, i\in X} g^{\frac{\sigma_2 r_i' i^{L_1}}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}}\},$$

$$sk_3' = \{sk_{3,0}', sk_{3,1}', \ldots, sk_{3,L_1}'\}$$

$$= \{(g^b)^{\sigma_3\beta_0} g^{-\sigma_3 \frac{\sum_{att_i\in W_i^*, i\in X} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i\in W^*, i\in X} (g^a)^{\sigma_2 r_i'}$$

$$\cdot \prod_{att_i\notin W^*, i\in X} (g^a)^{\frac{\sigma_2 r_i'}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$(g^b)^{\sigma_3\beta_0} g^{-\sigma_3 \frac{\sum_{att_i\in w_i^*, i\in X} r_i' \prod_{j=1}^{l_1}(i-w_j)}{t_w}} \prod_{att_i\in W^*, i\in X} (g^a)^{\sigma_2 r_i' i}$$

$$\cdot \prod_{att_i\notin W^*, i\in X} (g^a)^{\frac{\sigma_2 r_i' i}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$\ldots,$$

$$(g^b)^{\sigma_3\beta_0} g^{-\sigma_3 \frac{\sum_{att_i\in W_i^*, i\in X} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i\in W^*, i\in X} (g^a)^{\sigma_2 r_i' i^{L_1}}$$

$$\cdot \prod_{att_i\notin W^*, i\in X} (g^a)^{\frac{\sigma_2 r_i' i^{L_1}}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}}\},$$

$$sk_3'' = \{sk_{3,0}'', sk_{3,1}'', \ldots, sk_{3,L_1}''\}$$

$$= \{(g^b)^{\sigma_3\beta_0} g^{-\sigma_3 \frac{\sum_{att_i\in W_i^*, i\in X} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i\in W^*, i\in X} (g^{a'})^{\sigma_2 r_i'}$$

$$\cdot \prod_{att_i\notin W^*, i\in X} (g^{a'})^{\frac{\sigma_2 r_i'}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$(g^b)^{\sigma_3\beta_0} g^{-\sigma_3 \frac{\sum_{att_i\in W_i^*, i\in X} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i\in W^*, i\in X} (g^{a'})^{\sigma_2 r_i' i}$$

$$\cdot \prod_{att_i\notin W^*, i\in X} (g^{a'})^{\frac{\sigma_2 r_i' i}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$\ldots,$$

$$(g^b)^{\sigma_3\beta_0} g^{-\sigma_3 \frac{\sum_{att_i\in W_i^*, i\in X} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i\in W^*, i\in X} (g^{a'})^{\sigma_2 r_i' i^{L_1}}$$

$$\cdot \prod_{att_i\notin W^*, i\in X} (g^{a'})^{\frac{\sigma_2 r_i' i^{L_1}}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}}\},$$

$$sk_4 = \{sk_{4,0}, sk_{4,1}, \ldots, sk_{4,L_1}\}$$

$$= \{W_2^{\sigma_1} \prod_{att_i\in W^*, i\in Y} g^{\sigma_2 r_i'} \prod_{att_i\notin W^*, i\in Y} g^{\frac{\sigma_2 r_i'}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$W_2^{\sigma_1} \prod_{att_i\in W^*, i\in Y} g^{\sigma_2 r_i' i} \prod_{att_i\notin W^*, i\in Y} g^{\frac{\sigma_2 r_i' i}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$\ldots,$$

$$W_2^{\sigma_1} \prod_{att_i\in W^*, i\in Y} g^{\sigma_2 r_i' i^{L_1}} \prod_{att_i\notin W^*, i\in Y} g^{\frac{\sigma_2 r_i' i^{L_1}}{\sum_{att_m\in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}}\},$$

$$sk_4' = \{sk_{4,0}', sk_{4,1}', \ldots, sk_{4,L_1}'\}$$

$$= \{(g^b)^{\sigma_3\beta_1} g^{-\sigma_3 \frac{\sum_{att_i\in W_i^*, i\in Y} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i\in W^*, i\in Y} (g^a)^{\sigma_2 r_i'}$$

$$\cdot \prod_{att_i \notin W^*, i \in Y} (g^a)^{\frac{\sigma_2 r_i'}{\sum_{att_m \in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$(g^b)^{\sigma_3\beta_1} g^{-\sigma_3 \frac{\sum_{att_i \in W_i^*, i \in Y} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i \in W^*, i \in Y} (g^a)^{\sigma_2 r_i' i}$$

$$\cdot \prod_{att_i \notin W^*, i \in Y} (g^a)^{\frac{\sigma_2 r_i' i}{\sum_{att_m \in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$\cdots,$$

$$(g^b)^{\sigma_3\beta_1} g^{-\sigma_3 \frac{\sum_{att_i \in W_i^*, i \in X} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i \in W^*, i \in Y} (g^a)^{\sigma_2 r_i' i^{L_1}}$$

$$\cdot \prod_{att_i \notin W^*, i \in Y} (g^a)^{\frac{\sigma_2 r_i' i^{L_1}}{\sum_{att_m \in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}}\},$$

$$sk_4'' = \{sk_{4,0}'', sk_{4,1}'', ..., sk_{4,L_1}''\}$$

$$= \{(g^b)^{\sigma_3\beta_1} g^{-\sigma_3 \frac{\sum_{att_i \in W_i^*, i \in Y} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i \in W^*, i \in Y} (g^{a'})^{\sigma_2 r_i'}$$

$$\cdot \prod_{att_i \notin W^*, i \in Y} (g^{a'})^{\frac{\sigma_2 r_i'}{\sum_{att_m \in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$(g^b)^{\sigma_3\beta_1} g^{-\sigma_3 \frac{\sum_{att_i \in W_i^*, i \in Y} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i \in W^*, i \in Y} (g^{a'})^{\sigma_2 r_i' i}$$

$$\cdot \prod_{att_i \notin W^*, i \in Y} (g^{a'})^{\frac{\sigma_2 r_i' i}{\sum_{att_m \in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}},$$

$$\cdots,$$

$$(g^b)^{\sigma_3\beta_1} g^{-\sigma_3 \frac{\sum_{att_i \in W_i^*, i \in X} r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \prod_{att_i \in W^*, i \in Y} (g^{a'})^{\sigma_2 r_i', i^{L_1}}$$

$$\cdot \prod_{att_i \notin W^*, i \in Y} (g^{a'})^{\frac{\sigma_2 r_i' i^{L_1}}{\sum_{att_m \in W^*} r_m' \prod_{j=1}^{l_1}(m-\omega_j)}}\}.$$

Which implicity sets $s = \sigma_2$.

Thus, the trapdoor can be created as $\mathsf{TD} = (td_1, td_2, (td_{3,i}, td_{3,i}', td_{4,i}, td_{4,i}')_{i\in[0,L_1]})$ as follows: $td_1 = sk_1'$, $td_2 = sk_2'$, $td_{3,i} = sk_{3,i}$, $td_{3,i}' = sk_{3,i}''$, $td_{4,i} = sk_{4,i}$, $td_{4,i}' = sk_{4,i}''$.

Challenge: After obtaining M_0 and M_1 with equal length from $\mathcal{A}$, algorithm $\mathcal{B}$ replies $\mathcal{A}$ with the challenge ciphertext $\mathsf{CT}^* = (C_0, C_1, C_2, C_2', C_3, C_3', C_4, C_5)$. Pick random number $z' \in \mathbb{Z}_p$, set $z = z'$ and compute as follows:

$$T_1 = e(g^{ac}, g^b)^{\sigma_1\beta_0} \cdot e(g^{ac}, g)^{(\sigma_1-\sigma_2)\sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}}$$

$$\cdot e(g^a, g^d)^{\sigma_2 \sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^b, g^d)^{\sigma_3\beta_0}$$

$$\cdot e(g^{ac}, g^b)^{\sigma_1\beta_1} \cdot e(g^{ac}, g)^{(\sigma_1-\sigma_2)\sum_{att_i \in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}}$$

$$\cdot e(g^a, g^d)^{\sigma_2 \sum_{att_i \in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^b, g^d)^{\sigma_3\beta_1}$$

$$\cdot e(g^d, g)^{\sigma_3 \sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^a, T)^{\sigma_2\beta_0}$$

$$\cdot e(g^d, g)^{\sigma_3 \sum_{att_i \in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^a, T)^{\sigma_2\beta_1},$$

$$T_2 = e(g^{a'c}, g^b)^{\sigma_1\beta_0} \cdot e(g^{a'c}, g)^{(\sigma_1-\sigma_2)\sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}}$$

$$\cdot e(g^{a'}, g^d)^{\sigma_2 \sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^b, g^d)^{\sigma_3\beta_0}$$

$$\cdot e(g^{a'c}, g^b)^{\sigma_1\beta_1} \cdot e(g^{a'c}, g)^{(\sigma_1-\sigma_2)\sum_{att_i \in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}}$$

$$\cdot e(g^{a'}, g^d)^{\sigma_2 \sum_{att_i \in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^b, g^d)^{\sigma_3\beta_1}$$

$$\cdot e(g^d, g)^{\sigma_3 \sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^{a'}, T)^{\sigma_2\beta_0}$$

$$\cdot e(g^d, g)^{\sigma_3 \sum_{att_i \in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-\omega_j)}{t_\omega}} \cdot e(g^{a'}, T)^{\sigma_2\beta_1},$$

$$C_0 = M_\zeta \| z \oplus \mathcal{H}_1(T_1), \quad C_1 = M_\zeta^z \cdot \mathcal{H}_2(T_2),$$

$$C_2 = (g^{ac})^{\frac{1}{t_w}}, \quad C_3 = (g^d)^{\frac{1}{t_w}}, \quad C_2' = (g^{a'c})^{\frac{1}{t_w}}, C_3' = g^z,$$

$$C_4 = (W_1 \prod_{i\in X} R_i^{\frac{\prod_{j=0}^{l_1}(i-w_j)}{t_w}})^{z_1+z_2}$$

$$= ((g^b)^{\beta_0} g^{-\sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-w_j)}{t_w}}$$

$$\cdot g^{\sum_{att_i \in W_i^*, i\in X} \frac{r_i' \prod_{j=1}^{l_1}(i-w_j)}{t_w}})^{c+d}$$

$$= T^{\beta_0},$$

TABLE 1. Comparison of the existing IBE-ETs, ABE-KSs and our ABE-ET.

		IBE-ET ([22])	IBE-ET ([23])	ABE-KS ([7])	ABE-KS ([8])	KP-ABE-ET ([29])	CP-ABE-ET (Our proposed scheme)
Comp of	Enc	6Exp_1	6Exp_1	$\text{P}+(s+5)\text{Exp}_1+2\text{Exp}_2$	$(2N+4)\text{Exp}_1$	$(2s'+3)\text{Exp}_1$	$(2N+11)\text{Exp}_1$
	Dec	$2\text{P}+2\text{Exp}_1$	$3\text{P}+2\text{Exp}_1$	$2\text{P}+2s\text{Exp}_1$	-	$2s'\text{P}+(2s'+2)\text{Exp}_1$	$(8L_1+1)\text{Exp}_1+4\text{Exp}_2+12\text{P}$
	Test	4P	$2\text{P}+2\text{Exp}_1$	$2\text{P}+2s\text{Exp}_1$	$2\text{P}+\text{Exp}_1$	$2s'\text{P}+2s'\text{Exp}_1$	$8L_1\text{Exp}_1+4\text{Exp}_2+14\text{P}$
Size of	PP	$2\lvert\mathbb{G}_1\rvert$	$4\lvert\mathbb{G}_1\rvert$	$(2N+10)\lvert\mathbb{G}_1\rvert+3\lvert\mathbb{G}_2\rvert$	$8\lvert\mathbb{G}_1\rvert$	$N\lvert\mathbb{G}_1\rvert+2\lvert\mathbb{G}_2\rvert$	$(N+7)\lvert\mathbb{G}_1\rvert+6\lvert\mathbb{G}_2\rvert$
	CT	$4\lvert\mathbb{G}_1\rvert+\lvert\mathbb{Z}_p\rvert$	$2\lvert\mathbb{G}_1\rvert+5\lvert\mathcal{H}\rvert$	$(s+6)\lvert\mathbb{G}_1\rvert$	$4\lvert\mathbb{G}_1\rvert$	$(2s'+4)\lvert\mathbb{G}_1\rvert+2\lvert\mathbb{Z}_p\rvert$	$8\lvert\mathbb{G}_1\rvert+\lvert\mathbb{Z}_p\rvert$
	SK	$2\lvert\mathbb{G}_1\rvert$	$2\lvert\mathbb{G}_1\rvert$	$3s\lvert\mathbb{G}_1\rvert$	$3\lvert\mathbb{G}_1\rvert$	$2s\lvert\mathbb{G}_1\rvert$	$4\lvert\mathbb{G}_1\rvert+6l_1\lvert\mathbb{G}_2\rvert$
	TD	$\lvert\mathbb{G}_1\rvert$	$\lvert\mathbb{G}_1\rvert$	$3s\lvert\mathbb{G}_1\rvert$	$5\lvert\mathbb{G}_1\rvert$	$s\lvert\mathbb{G}_1\rvert$	$6\lvert\mathbb{G}_1\rvert$
Fun	KS	✓	✓	✓	✓	✓	✓
	ET	✓	✓	×	×	✓	✓
	FAC	×	×	✓	✓	✓	✓
SL		OW-ID-CCA	IND-ID-CCA	IND-ID-CCA	IND-ID-CPA	OW-CCA	IND-ID-CPA
SM		×	×	×	×	×	✓
Assumption		BDH	BDH	n-DBHE	DLIN	BDH	DLIN

‡ IBE-ET: Identity based encryption with equality test, ABE-KS: Attribute based encryption with keyword search, ABE-ET: Attribute based encryption with equality test, Comp: Computational complexity, Fun: functionality, SL: Security level, SM: Standard model, Enc: Encryption algorithm, Dec: Decryption algorithm, Test: Test algorithm, PP: Public parameter, CT: Ciphertext, SK: Secret key, TD: Trapdoor, KS: Keyword search, ET: Equality test, FAC: Fine-grained access control, n : System parameter, N: the amount of attributes in the system, L_1: the amount of wildcards in access policy, s: the amount of attributes in access policy, s': the amount of attributes in a attribute set, Exp_1: Exponentiation in group $\mathbb{G}_1$, Exp_2: Exponentiation in group $\mathbb{G}_2$, P: Pairing, $|\mathbb{G}_1|$, $|\mathbb{G}_2|$: Length of one element in $\mathbb{G}_1$, $\mathbb{G}_2$, $|\mathbb{Z}_p|$: Length of one random number in $\mathbb{Z}_p$, $|\mathcal{H}|$: Length of output of hash function, BDH: Bilinear Diffie-Hellman, DBDH: Decisional Bilinear Diffie-Hellman, DLIN: Decisional Linear, n-DBHE: n-Decisional bilinear Diffie-Hellman Exponent.

$$C_5 = (W_2 \prod_{i\in Y} R_i^{\frac{\prod_{j=0}^{l_1}(i-w_j)}{t_w}})^{z_1+z_2}$$
$$= ((g^b)^{\beta_1} g^{-\sum_{att_i\in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-w_j)}{t_w}}$$
$$\cdot g^{\sum_{att_i\in W_i^*, i\in Y} \frac{r_i' \prod_{j=1}^{l_1}(i-w_j)}{t_w}})^{c+d}$$
$$= T^{\beta_1}.$$

Which implies $z_1 = c, z_2 = d$. $\mathcal{B}$ then delivers the generated ciphertext CT^* to $\mathcal{A}$ in the following:

$$\mathsf{CT}^* = (C_0, C_1, C_2, C_3, C_2', C_3', C_4, C_5).$$

Guess: A guess $\zeta' \in \{0, 1\}$ on ζ is replied by $\mathcal{A}$. If $\zeta = \zeta'$, then $\mathcal{B}$ returns 1 to guess $T = g^{b(c+d)}$. Otherwise, $\mathcal{B}$ returns 0 to guess T is one random numbers in $\mathbb{G}_1$.

If $T = g^{b(c+d)}$, the simulator $\mathcal{B}$ gives a perfect simulation so we have: $\Pr[\mathcal{B}(g, g^a, g^b, g^{ac}, g^{a'}, g^{a'c}, g^d, T = g^{b(c+d)}) = 1 | T = g^{b(c+d)}] = \frac{1}{2} + Adv_A(k)$. If T is a random group element the message M_b is completely hidden from the adversary and we have: $\Pr[\mathcal{B}(g, g^a, g^b, g^{ac}, g^{a'}, g^{a'c}, g^d, T = g^{b(c+d)}) = 1 | T = g^r] = \frac{1}{2}$, where $r \in_R \mathbb{Z}_p$. $\mathcal{B}$ can solve DLIN with non-negligible advantage if $Adv_A(k)$ is non-negligible.

V. PERFORMANCE ANALYSIS

In this section, we present the comparisons of the existing IBE-ET, ABE-KS, KP-ABE-ET schemes and our proposed CP-ABE-ET scheme in terms of computational complexity, size, functionality, security level, security model and hardness assumption. Besides, the simulations are also given to demonstrate the practicality of our scheme.

In Table 1, the comparisons of computational overheads for encryption algorithm, decryption algorithm, test algorithm are listed in the third, fourth, fifth rows respectively. The comparisons of size for public parameter, ciphertext, decryption secret key, trapdoor are located in sixth, seventh, eighth, ninth rows respectively. The tenth and eleventh, twelfth rows indicate whether the table-listed schemes support the functionality of keyword searchable, equivalence test, fine-grained access control, respectively. The thirteenth row is used to indicate the security levels that can be attained by the above-listed schemes. The fourteenth row suggests whether the security proof can be proven in standard model. The hardness assumptions are presented in the last row.

As presented in Table 1, the computational complexities of encryption, decryption and test algorithms in Ma's IBE-ET scheme [22], Lee *et al.*'s IBE-ET scheme [23], Liang and Susilo's ABE-KS scheme [7] and Zheng *et al.*'s ABE-KS scheme [8] are indeed more lightweight than KP-ABE-ET [29] and our suggested CP-ABE-ET scheme. Similarly, the sizes of public parameter, ciphertext, decryption secret key, trapdoor in our proposed scheme are obviously much costly than the previous IBE-ET schemes.

With regard to the functionality of table-listed schemes, the IBE-ET schemes, KP-ABE-ET scheme and our proposed scheme in Table 1 support keyword search as well as equivalence test between two messages. However, the two referenced IBE-ET schemes could not achieve fine-grained access control over the encrypted data. And the two ABE-KS schemes realizes key search and fine-grained access control but not achieve equality test. Only the KP-ABE-ET scheme and our proposed scheme could provide the above properties because of the advantages of ABE scheme. Furthermore, due to the introduced CP-ABE scheme, our scheme is more flexible than the KP-ABE-ET scheme so that a data owner has greater freedom deciding who ought to or ought not to access the shared data. Moreover, our scheme could

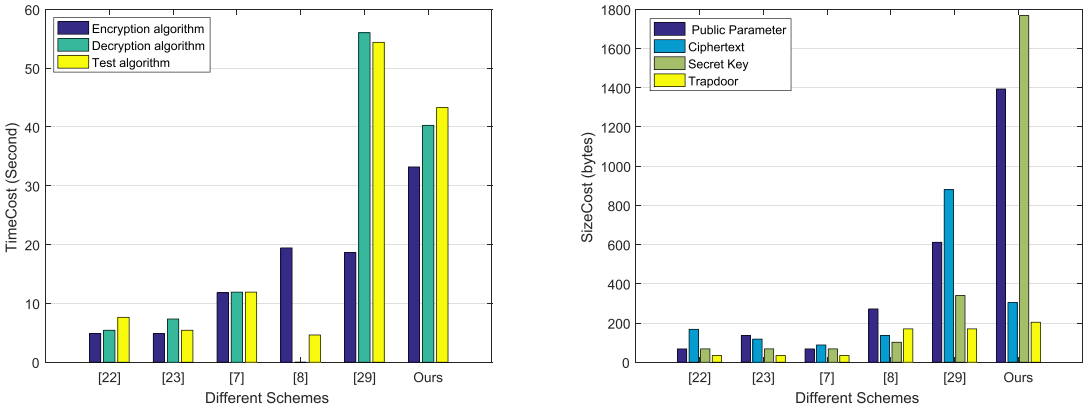

FIGURE 2. (a) Comparison of the time-cost for encryption, decryption and test algorithm. (b) Comparison of the storage-cost for public parameter, ciphertext, secret key and trapdoor.

provide more expressive access policy because our scheme supports not only positive and negative attributes but also wildcard attributes. In terms of security, the security in Ma 's IBE-ET scheme [22] could be one-way secure against the OW-CCA attack. Meanwhile, its security proof is reduced to BDH assumption in the random oracle model. We could also obtain that the IND-CCA security can be achieved in Lee *et al.*'s IBE-ET scheme [23] under the BDH assumption in the random oracle model. Liang *et al.*'s ABE-KS scheme [7] achieves IND-ID-CCA security under n-DBHE assumption and Zheng *et al.*'s ABE-KS scheme [8] meets IND-ID-CPA security under DL assumption. And Zhu *et al.*' scheme [29] achieves one-way secure against the OW-CCA attack under BDH assumption. Compare with the above schemes, our proposed scheme cloud achieve the IND-CPA security under DLIN assumption in standard model.

In order to present practical performance evaluations, we simulated our scheme and other related schemes [7], [8], [22], [23], [29] based on cpabe toolkit and Pairing-Based Cryptography (PBC) library [34]. Specifically, these experiments are executed on an i5-4460 CPU @3.2 GHz and 4G ROM running Windows 7 64 bit system and VC++ 6.0. To attain the 80-bit security level, our simulation is executed based on a 20-byte elliptic curve group constructed on the curve $z^2 = x^3 + x$ over a 64-byte finite field. Here we further set the system parameter n as 20, the amount of attributes in the system N as 15, the amount of wildcards in access structure l_1 as 2 and the amount of attributes in access policy s as 5. So that, we can get the result in Table 2. The $|\mathcal{H}|$ denotes the length of output of hash function, and the $|\mathbb{G}_1|$, *the*$|\mathbb{G}_2|$ denotes the length of one element in $\mathbb{G}_1$, $\mathbb{G}_2$, the $|\mathbb{Z}_p|$ denotes the length of one random number in $\mathbb{Z}_p$. In addition, the Exp_1 denotes the time-costs of a exponentiation in group $\mathbb{G}_1$, the Exp_2 denotes the time-cost of a exponentiation in group $\mathbb{G}_2$, and the P denotes the time-cost a pairing operation.

In Fig. 2(a), we compare the overheads of the encryption, decryption and test algorithms in [7], [8], [22], [23], and [29] and our scheme. We can see that ours and [29] are

TABLE 2. Size-costs (byte) and time-costs (s).

$\lvert\mathcal{H}\rvert$	$\lvert\mathbb{Z}_p\rvert$	$\lvert\mathbb{G}_1\rvert$	$\lvert\mathbb{G}_2\rvert$	P	Exp_1	Exp_2
10	32	34	136	1.91	0.81	0.90

based on attributes to design, the time-costs of encryption, decryption and test algorithms in ours and [29] are relatively costly, because the time-costs are related to the number of attributes or wildcards involved in corresponding algorithms. So our time-cost is higher than other schemes. But compared with [29], our scheme is more efficient than [29]. Moreover, in Fig. 2(b), we can observe that the size of ciphertext in our scheme is constant. No matter how many attributes are involved in our scheme, the size of ciphertext in our scheme remains unchanging. That means that our scheme has more scalability than [7], [8], [22], [23], and [29]. Overall, although our scheme is less efficient than other schemes, our scheme provides more functionality and more expressiveness. In general, our scheme is versatile and practical. In addition, great efforts have already been made to outsource the decryption or encryption in ABE systems to the cloud server without the cloud being able to access any part of userąś messages. It is reasonable to directly adopt these approaches in our CP-ABE-ET scheme to outsource the heavy computational workload to the cloud server. Simultaneously, this is our future work.

VI. CONCLUSION

In our paper, a novel CP-ABE-ET cryptosystem named ciphertext-policy attribute based encryption with equality test is introduced to provide users with searching capability on ciphertexts and fine-grained access control. With our proposed CP-ABE-ET scheme, each user featured with attributes delegates a cloud server to test the equivalence between two messages under different access policies. Meanwhile, the cloud server cannot access the plaintext during the delegated equivalence test. Finally, the rigorous security proof is given to show the IND-CPA security in the standard model under

DLIN assumption. Additionally, we present performance and simulation comparisons of existing IBE-ET, ABE-KS and KP-ABE-ET with our CP-ABE-ET scheme to demonstrate that our scheme is practical. Future work contains seeking to build CP-ABE-ET scheme to achieve the security level of IND-CCA2 in standard model.

REFERENCES

[1] M. Armbrust *et al.*, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, 2010.
[2] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, A. Ghalsasi, "Cloud computing—The business perspective," *Decision support Syst.*, vol. 51, no. 1, pp. 176–189, 2011.
[3] C. Pagliari, D. Detmer, and P. Singleton, "Potential of electronic personal health records," *Brit. Med. J.*, vol. 335, no. 7615, pp. 330–333, 2007.
[4] D. Kaelber *et al.*, "A research agenda for personal health records (PHRs)," *J. Amer. Med. Informat. Assoc.*, vol. 15, no. 6, pp. 729–736, 2008.
[5] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. 13th ACM Conf. Comput. Commun. Secur. (CCS)*, 2006, pp. 89–98.
[6] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy (SP)*, 2007, pp. 321–334.
[7] K. Liang and W. Susilo, "Searchable attribute-based mechanism with efficient data sharing for secure cloud storage," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 9, pp. 1981–1992, Sep. 2015.
[8] Q. Zheng, S. Xu, and G. Ateniese, "VABKS: Verifiable attribute-based keyword search over outsourced encrypted data," in *Proc. 33rd Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr./May 2014, pp. 522–530.
[9] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in *Advances in Cryptology* (Lecture Notes in Computer Science), vol. 3027, C. Cachin and J. Camenisch, Eds. Berlin, Germany: Springer-Verlag, 2004, pp. 506–522.
[10] J. Baek, R. Safavi-Naini, and W. Susilo, "Public key encryption with keyword search revisited," in *Proc. Int. Conf. ICCSA*, vol. 5072. Perugia, Italy, Jun./Jul. 2008, pp. 1249–1259.
[11] Y. H. Hwang and P. J. Lee, "Public key encryption with conjunctive keyword search and its extension to a multi-user system," in *Pairing-Based Cryptography—Pairing* (Lecture Notes in Computer Science), vol. 4575. Berlin, Germany: Springer, 2007, pp. 2–22.
[12] G. Yang, C. H. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Proc. Int. Conf. Topics Cryptol. (CT-RSA)*, vol. 5985. 2010, pp. 119–131.
[13] Q. Tang, "Towards public key encryption scheme supporting equality test with fine-grained authorization," in *Proc. Austral. Conf. Inf. Secur. Privacy*, vol. 6812. 2011, pp. 389–406.
[14] Q. Tang, "Public key encryption supporting plaintext equality test and user-specified authorization," *Secur. Commun. Netw.*, vol. 5, no. 12, pp. 1351–1362, 2012.
[15] Q. Tang, "Public key encryption schemes supporting equality test with authorisation of different granularity," *Int. J. Appl. Cryptogr.*, vol. 2, no. 4, pp. 304–321, 2012.
[16] S. Ma, M. Zhang, Q. Huang, and B. Yang, "Public key encryption with delegated equality test in a multi-user setting," *Comput. J.*, vol. 58, no. 4, pp. 986–1002, 2014.
[17] K. Huang, R. Tso, Y.-C. Chen, S. M. M. Rahman, A. Almogren, and A. Alamri, "PKE-AET: Public key encryption with authorized equality test," *Comput. J.*, vol. 58, no. 10, pp. 2686–2697, 2015.
[18] S. Ma, Q. Huang, M. Zhang, and B. Yang, "Efficient public key encryption with equality test supporting flexible authorization," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 3, pp. 458–470, Mar. 2015.
[19] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology—CRYPTO*. Berlin, Germany: Springer, 2001, pp. 213–229.
[20] C. Gentry and S. Halevi, "Hierarchical identity based encryption with polynomially many levels," in *Proc. Theory Cryptogr. Conf.*, 2009, pp. 437–456.
[21] S. Luo and Z. Chen, "Hierarchical identity-based encryption without key delegation in decryption," *Int. J. Grid Utility Comput.*, vol. 5, no. 2, pp. 71–79, 2014.
[22] S. Ma, "Identity-based encryption with outsourced equality test in cloud computing," *Inf. Sci.*, vol. 328, pp. 389–402, Jan. 2016.
[23] H. T. Lee, S. Ling, J. H. Seo, and H. Wang, "Semi-generic construction of public key encryption and identity-based encryption with equality test," *Inf. Sci.*, vol. 373, pp. 419–440, Dec. 2016.
[24] L. Wu, Y. Zhang, K.-K. R. Choo, and D. He, "Efficient and secure identity-based encryption scheme with equality test in cloud computing," *Future Generat. Comput. Syst.*, vol. 73, pp. 22–31, Aug. 2017.
[25] M. Abdalla *et al.*, "Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions," in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 3621. Santa Barbara, CA, USA: Springer, 2005, pp. 205–222.
[26] T. V. X. Phuong, G. Yang, and W. Susilo, "POSTER: Efficient ciphertext policy attribute based encryption under decisional linear assumption," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2014, pp. 1490–1492.
[27] S. Zhu and X. Yang, "Protecting data in cloud environment with attribute-based encryption," *Int. J. Grid Utility Comput.*, vol. 6, no. 2, pp. 91–97, 2015.
[28] Z. Liu, J. Luo, and L. Xu, "A fine-grained attribute-based authentication for sensitive data stored in cloud computing," *Int. J. Grid Utility Comput.*, vol. 7, no. 4, pp. 237–244, 2016.
[29] H. Zhu, L. Wang, H. Ahmad, and X. Niu, "Key-policy attribute-based encryption with equality test in cloud computing," *IEEE Access*, vol. 5, pp. 20428–20439, 2016, doi: 10.1109/ACCESS.2017.2756070.
[30] T. V. X. Phuong, G. Yang, W. Susilo, and X. Chen, "Attribute based broadcast encryption with short ciphertext and decryption key," in *Proc. Eur. Symp. Res. Comput. Secur. (ESORICS)*, vol. 9372. 2005, pp. 252–269.
[31] S. Sedghi, P. van Liesdonk, S. Nikova, P. Hartel, and W. Jonker, "Searching keywords with wildcards on encrypted data," in *Security and Cryptography for Networks* (Lecture Notes in Computer Science), vol. 6280. Berlin, Germany: Springer-Verlag, 2010, pp. 138–153.
[32] Z. Zhou, D. Huang, and Z. Wang, "Efficient privacy-preserving ciphertext-policy attribute based-encryption and broadcast encryption," *IEEE Trans. Comput.*, vol. 64, no. 1, pp. 126–138, Jan. 2015.
[33] F. Viète, *Opera Mathematica*. Leiden, The Netherlands: Officina Bonaventurae & Abrahami Elzeviriorum, 1646.
[34] B. Lynn. *The Stanford Pairing Based Crypto Library*. Accessed: Nov. 2017. [Online]. Available: http://crypto.stanford.edu/pbc/

QIANG WANG received the M.S. degree from the School of Computer Science and Engineering, University of Electronic Science and Technology of China, in 2012, where he is currently pursuing the Ph.D. degree. His research interests include public key cryptography and network security.

LI PENG received the B.S. degree from Guangxi University. He is currently pursuing the M.S. degree with the School of Information and Software Engineering, University of Electronic Science and Technology of China. His research interests include attribute-based encryption and malicious code detection.

HU XIONG received the Ph.D. degree from the University of Electronic Science and Technology of China (UESTC) in 2009. He is currently an Associate Professor with the UESTC. His research interests include cryptography and ad hoc networks security.

JIANFEI SUN received the B.S. degree from the School of Computer and Information Engineering, Chaohu University, in 2013. He is currently pursuing the Ph.D. degree with the School of Information and Software Engineering, UESTC. His research interests include public key cryptography and network security.

ZHIGUANG QIN received the Ph.D. degree from UESTC in 1996. He is currently a Full Professor and the President with the School of Computer Science and Engineering, UESTC. His research interests include information security and wireless networks.