IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 30, 2019, accepted June 9, 2019, date of publication June 13, 2019, date of current version July 2, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2922646

KS-ABESwET: A Keyword Searchable Attribute-Based Encryption Scheme With Equality Test in the Internet of Things

SHANGPING WANG[1], LISHA YAO[1], JUANJUAN CHEN[1], AND YALING ZHANG[2]
[1]School of Science, Xi'an University of Technology, Xi'an 710054, China
[2]School of Computer Science and Engineering, Xi'an University of Technology, Xi'an 710048, China

Corresponding author: Lisha Yao (yaolishaqh@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61572019, in part by the Key Research and Development Program of Shaanxi under Grant 2019GY-028, and in part by the Start-Up Fund for Ph.D. Teachers in Xi'an University of Technology under Grant 256081502.

ABSTRACT With the widespread application of the Internet of Things (IoT) technology, it is expected to become a new data management model. But IoT devices have limited resource storage and computing power, they are highly vulnerable to hackers and leak sensitive information to third parties. The traditional attribute-based searchable encryption schemes usually require an intelligent terminal to perform complex calculations while accessing data, therefore IoT devices may not be able to withstand excessive calculation burden. In this paper, we present a keyword searchable attribute-based encryption scheme with equality test (KS-ABESwET) in the IoT by combining the notions of attribute-based searchable encryption (ABSE) with equality test. The proposed scheme adopts a keyword search algorithm based on the inverted index and equality test mechanism. If the keyword token match index is successful, the cloud server sends all ciphertexts that meet the conditions to the data user. Then, data user classifies the ciphertexts by equality test mechanism, which is executed by the authorized cloud server to determine whether the two ciphertexts encrypted by different access policies contain the same plaintext without decrypting. In this way, data user does not need to decrypt all ciphertexts, which decreases storage resource consumption of IoT devices and simplifies the complex operations generated by the traditional ABSE schemes. Using outsourcing technology, most calculations in the scheme are outsourced to the server, and IoT devices only perform a few calculations, which reduces greatly the computing and storage burden. Based on the decisional $q-1$ assumption and decisional Diffie–Hellman (DDH) assumption, the proposed scheme proves that has chosen-plaintext security and chosen-keyword security. Moreover, through comparative analysis and experimental simulation, our scheme is effective and suitable for IoT environment.

INDEX TERMS Internet of Things, keyword searchable, attribute-based encryption, equality test.

I. INTRODUCTION

A. MOTIVATION

IoT technology proposes the interconnection between different devices, such as smart phones, infinite sensors, RFID etc, achieves data collection, transmission and storage [1]. These IoT devices are widely used in companies, factories, and everyday life, thus data security and privacy protection are crucial. In particular, although IoT devices are diverse in variety, their storage and computing capabilities are limited, it is necessary to reduce their computation burden. In the application of IoT, data is always dynamically shared in a diverse distributed network [2]. In order to prevent unauthorized data users accessing data, it is important to adopt searchable encryption, access policy and equality test mechanism.

The attribute-based encryption (ABE) system [3] is an encrypted access control mechanism that effectively protects privacy and data security. It provides flexible data sharing for data users in the system, allows data owners to perform more detailed encryption operation for specified features. The decryption algorithm is a process that matches with the specified feature description value, so that data user who satisfies the condition can decrypt ciphertext. Generally speaking, ABE technology is mainly divided into two types:

The associate editor coordinating the review of this manuscript and approving it for publication was Tony Thomas.

key-policy attribute-based encryption (KP-ABE) [4] and ciphertext-policy attribute-based encryption (CP-ABE) [5]. In KP-ABE schemes, the data user's secret key is related to access policy, the ciphertext is related to attributes, CP-ABE schemes are opposite. Therefore, the difference between CP-ABE and KP-ABE scheme mainly depends on who is involved in access policy in the encryption system.

Since the cloud server stores the encrypted data, these ciphertexts no longer have the semantic features of plaintext. Their logical relationships and size rules are different from the plaintext. Hence, the plaintext search method is not suitable for ciphertext search. The searchable encryption (SE) technology realizes the search function on the ciphertext. The data user searches for ciphertexts based on the keyword of interest and then decrypt them, which improves the usability of cloud storage and cloud computing. In addition, the proposed ABES mechanism [6] implements keyword search for fine-grained access control. Particularly, the setting of file index is very important in keyword search. The inverted index is highly effective for a large dataset, because the search result directly points to relevant files when the inverted list matches with query keyword. In other words, the data user can search for multiple corresponding files based on the keyword of interest.

For many existing ABE schemes, the local computing burden is reduced by controlling the size of ciphertext. Although it decreases the storage cost, this is limited to the encryption phase. The data users still have to perform a lot of calculations during the decryption operation, which obviously does not apply to resource-constrained lightweight devices. In the outsourcing technology model, data users outsource most of the computational load to servers with strong computing power, and these servers will execute algorithms and return the results to data users. The outsourcing technology [7] is widely applied to different phases of the ABE schemes, which reduces the computation and communication burden of local devices while ensuring fine-grained access control.

For the sake of achieving data classification and accurate decryption, an equality test mechanism came into being. The initial equality test is combined with public key encryption system, so that data user could perform equality test between two messages encrypted by different public keys. Later, KP-ABE with equality test (KP-ABEwET) and CP-ABE with equality test (CP-ABEwET) are proposed, which are determined by the authorized cloud server whether two ciphertexts encrypted by different access policies contain the same plaintext without decrypting ciphertext. For the combination of ABSE with equality test, the data user first makes an equality judgment on the searched ciphertexts, excludes ciphertexts that have the same plaintext, then decrypts the remaining ciphertexts, which saves time and reduces the workload.

Therefore, our solution fully considers the practical application of ABE scheme, analyzes problems of the existing schemes, combines searchable encryption, outsourced computing and equality test mechanism to construct KS-ABESwET for the IoT environment.

B. CONTRIBUTIONS

Owing to the limited calculation, resource and energy of IoT devices, they are not sufficient to bear a large amount of computing and storage burden, traditional ABSE schemes may be difficult to realize flexible and effective data sharing in IoT environment. Thus, we propose a KS-ABESwET in IoT. Our main contributions are as follows:

(1) Our scheme combines ABSE with equality test. Under the premise of ensuring fine-grained access control, data user first searches ciphertexts according to the keyword of interest, and then classifies all ciphertexts by the equality test mechanism which could judge whether the two ciphertexts encrypted by different access policies contain the same plaintext without decryption, so as to exclude ciphertexts that have same plaintext, and finally data user decrypts the desired ciphertexts. In this way, invalid repeat operations in the decryption phase are avoided.

(2) The proposed scheme is applicable to resource-constrained devices in IoT environment. Specifically, most of the computational load during the secret key generation, encryption and decryption phases is outsourced to server, IoT devices only need to perform few operations, which greatly reduces the local computing and storage burden.

(3) On the basis of decisional $q-1$ assumption and DDH assumption, our scheme proves that has the chosen-plaintext security and chosen-keyword security. Moreover, through comparative analysis and experimental simulation, the proposed scheme is effective and practical.

The structure of this paper is laid out as follows. Section II introduces the related work. Section III describes the notation definitions, mathematical knowledge and complexity assumptions. Section IV is a framework of the paper. In Section V, The details of the paper are given. Section VI is a security proof of the paper.The performance analysis and experimental simulation are in Section VII. Finally, the conclusions and future research directions of the paper are presented.

II. RELATED WORK

A. ATTRIBUTE-BASED ENCRYPTION TECHNOLOGY

ABE is a generalization of the identity-based encryption (IBE) [8], [9], which replaces the identity in IBE with attribute, and first proposed by Sahai and Waters [10] in EUROCRYPT 2005. ABE technology plays an important role in the fine-grained access control system [11]–[14]. In 2006, Goyal *et al.* [12] proposed the first monotonous access policy in KP-ABE scheme, but the KP-ABE scheme is not as flexible as CP-ABE, because once the data user's secret key is confirmed, the access structure is determined accordingly, which leads the encryption operation to difficult, and the data owner needs to choose a suitable attribute set for ciphertext. Subsequently, Bethencourt *et al.* [15] proposed the first CP-ABE scheme, the scheme only analyzes the security under the general group model and does not achieve provable security. In 2018, Liu *et al.* [16] proposed an efficient

revocable CP-ABE scheme, the scheme adds time validity technique based on direct revocation, but has restriction for repeated attributes. In 2019, Li *et al.* [17] proposed an ABE scheme with CCA2 security for fog computing. Wu *et al.* [18] combined ABE technology with the blockchain to achieve privacy protection and track the private key of malicious users, but the decryption algorithm of the scheme needs to be further optimized. Li *et al.* [19] proposed a hierarchical ABE scheme with the help of hierarchical ideas. Considering side channel attacks, the scheme adopts leakage-resilience technology. Currently, the application of ABE technology is very extensive, including: cloud computing [20], cloud storage [21], [22] and personal health records [23], [24].

B. SEARCHABLE ENCRYPTION WITH INVERTED INDEX

SE was first proposed by Song et al. [25], the scheme supports keyword search, but does not set index, which means server must scan the entire system to find the file that meets requirements, results in significant resource loss. Subsequently, many searchable symmetric encryption schemes were proposed, including static and dynamic schemes [26], [27]. In order to improve search efficiency and save costs, some schemes [28]–[30] set different index structures, and the schemes [31], [32] established secure SE based on inverted index. In 2011, Curtmola *et al.* [31] proposed the first SE scheme based on inverted index, but this scheme dose not support dynamic encryption and index operations. In 2015, Wang et al. [33] proposed a public key SE scheme based on inverted index, which supports multi-keyword search, but cannot achieve fine-grained access control.

C. OUTSOURCING TECHNOLOGY

In ABE schemes, as the number of attributes in access policy increases, which leads to a high computation burden of encryption and decryption phase. In fact, due to the fundamental nature of bilinear mapping, ABE schemes usually have more pairing operations. Therefore, this is a huge limitation for resource-constrained devices. To solve this problem, some ABE schemes [34]–[37] generated constant-size ciphertexts during the encryption phase, or controlled the number of bilinear operations. However, the methods adopted by these schemes lacked ideal expressions. In 2011, Green *et al.* [38] proposed a new method, which is similar to the concept of proxy re-encryption [39], that is in decryption phase, the data user generates a conversion key based on the secret key and sends it to third party to execute partial decryption, which outsources a large number of operations to the server and reduces the data user's local computing burden. In 2017, the scheme that supports outsourced encryption and decryption was proposed by Shao *et al.* [40], which adopts a two-step outsourcing operation and reduces the amount of computation in both encryption and decryption phases. Subsequently, Belguith *et al.* [2] proposed a secure outsourced decryption ABE scheme for cloud-assisted IoT environment.

D. EQUALITY TEST MECHANISM

In 2010, the public key encryption with equality test scheme [41] solved the problem of whether two ciphertexts encrypted by different public keys contain the same plaintext without decryption, but this scheme allows anyone to perform equality test algorithm. Subsequently, some public key encryption with equality test schemes [42]–[44] added the concept of authorization, which improves security. In 2016, Ma *et al.* [45] first combined IBE with equality test (IBEwET). Later, Lee *et al.* [46] strengthened the security requirements of IBEwET scheme. Wu *et al.* [47] designed the IBEwET scheme for mobile cloud environment in 2017. In order to achieve fine-grained access control, many ABE with equality test schemes [48]–[50] were proposed. In 2017, Zhu *et al.* [48] proposed a KP-ABEwET scheme, which enjoys a more flexible authorization process, but this scheme is proved to be unsafe by Liao *et al.* [49]. Recently, Wang *et al.* [50] proposed a CP-ABEwET scheme based on the bilinear pairing and Viéte formula, which achieves multi-function and secure data sharing in cloud computing.

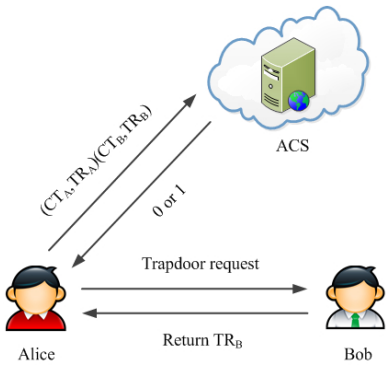

FIGURE 1. The process of equality test.

The Figure 1 is a brief introduction to the process of equality test algorithm. Alice receives two ciphertexts CT_A and CT_B, delegates the authorized cloud server (ACS) to perform equality judgment on the two ciphertexts. She first sends the trapdoor request of ciphertext test to Bob in the same system. Then Bob returns trapdoor TR_B to Alice, Alice generates its own trapdoor TR_A and transmits two pairs of ciphertexts and trapdoors: (CT_A, TR_A) , (CT_B, TR_B) to the ACS to execute equality test (two ciphertexts do not need to be decrypted during testing). If the two ciphertexts contain the same plaintext, ACS returns 1 to Alice, otherwise returns 0.

III. PRELIMINARIES

A. NOTATION DEFINITIONS

Our scheme contains many mathematical symbols, in order to improve the readability of the paper, we list these symbols and their explanations (as shown in Table 1).

TABLE 1. The explanation of symbols.

Symbol	Description
$\mathbb{G}_i\ (i \in [1,2])$	Multiplication cycle group
$\mathbb{Z}_p$	Residual ring of modulo p
κ	Security parameter
PP	Public parameter
MSK	Master secret key
$H_i\ (i \in [1,4])$	One-way hash function
S	Attribute set
SK'	Outsourced secret key
SK	Secret key
$\mathbf{M}$	Shared matrix
$(\mathbf{M}, \rho, \{x_{\rho(i)}\})$	Access policy
CT'	Outsourced ciphertext
$M_{\eta_j}\ (j \in [1,m])$	File
$H_1(M_{\eta_j})$	The hash value of M_{η_j}
$E_{H_1(M_{\eta_j})}(M_{\eta_j})$	The ciphertext of M_{η_j}
CT''_{η_j}	The ciphertext of $H_1(M_{\eta_j})$
w	Encrypted keyword
w'	Searched keyword
$Index$	The index of w
CT	Uploaded ciphertext
$Token$	The token of w'
$\perp$	Termination symbol
CT'''	Partial ciphertext
SK''	Partial secret key
CT_{in}	Intermediate ciphertext
TR	Trapdoor
$\mathcal{A}$	Adversary in the secure game
$\mathcal{B}$	Simulator in the secure game

B. BILINEAR MAPPING

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups whose order is prime p, g be a generator of $\mathbb{G}_1$. Let $e: \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a bilinear mapping when it satisfies the following characters [51]:

(1) Bilinearity: For $\forall a, b \in \mathbb{Z}_p$, exist $e(g^a, g^b) = e(g,g)^{ab}$.

(2) Non-degeneracy: $\exists g \in \mathbb{G}_1$, such that $e(g,g) \neq 1$.

(3) Computability: For $\forall u, v \in \mathbb{G}_1$, $e(u,v)$ is efficiently computed.

C. LSSS ACCESS POLICY

Let P be a set of entities, a $LSSS\,\Pi$ defined on P includes [52]: 1) The shares for each entity form a vector over $\mathbb{Z}_p$; 2) There is a shared matrix $\mathbf{M}$ of size $l \times n$ for Π and a mapping ρ from $\{1, 2, \cdots, l\}$ to P. We randomly select a vector $\mathbf{v} = \{s, v_2, v_3, \cdots, v_n\} \in \mathbb{Z}_p^n$, where s is a secret, v_i is a random value in $\mathbb{Z}_p$. Then $\mathbf{M}\mathbf{v}^T$ is a vector of l shares of s based on Π, the share $(\mathbf{M}_i\mathbf{v}^T)$ belongs to entity $\rho(i)$, and is expressed as $\lambda_i = (\mathbf{M}_i\mathbf{v}^T)$.

LSSS defined by the above method is reconstruction: Suppose that Π is a *LSSS* for access policy Λ. On the one hand, for the authorized set of data user $S \in \Lambda$, we define $I = \{i : \rho(i) \in S\} \subseteq \{1, 2, \cdots, l\}$. There is a vector $\boldsymbol{\omega} = \{\omega_i \in \mathbb{Z}_p\}_{i \in I}$, so that $\sum_{i \in I} \omega_i \mathbf{M}_i = (1, 0, \cdots, 0)$, we have $\sum_{i \in I} \omega_i \mathbf{M}_i \mathbf{v}^T = \sum_{i \in I} \omega_i \lambda_i = s$. On the other hand, for the unauthorized set of data user, there is a vector $\tilde{\boldsymbol{\omega}} \in \mathbb{Z}_p^n$, so that $\tilde{\boldsymbol{\omega}} \cdot (1, 0, \cdots, 0)^T = -1$, we have $\tilde{\boldsymbol{\omega}} \cdot \mathbf{M}_i^T = 0$, where $i \in I$.

D. INVERTED INDEX

In SE system, the inverted index [53] is used to improve the search efficiency of file, which is a more practical structure, as shown in Table 2. $Index_{w_i}$ denotes an index list of keyword w_i, M_{i,n_i} denotes a file that contains keyword w_i. The inverted index includes many index lists, such as $Index_{w_1}, Index_{w_2}, \cdots, Index_{w_t}$, The index list of a keyword is a collection of the files that contains the keyword, that is $M_{i,1}, M_{i,2}, \cdots, M_{i,n_i}$.

TABLE 2. The inverted index.

Index	Files
$Index_{w_1}$	$M_{1,1}, M_{1,2}, \cdots, M_{1,n_1}$
$Index_{w_2}$	$M_{2,1}, M_{2,2}, \cdots, M_{2,n_2}$
...	...
$Index_{w_t}$	$M_{t,1}, M_{t,2}, \cdots, M_{t,n_t}$

E. COMPLEXITY ASSUMPTION

The security of our scheme relies on the decisional $q-1$ assumption [52] and DDH assumption [54]. The description of these complexity assumptions are as follows:

Definition 1 (Decisional $q-1$ Assumption): For all probabilistic polynomial time (PPT) algorithm, let $\mathbf{y} =$

$$g, g^s$$
$$g^{a^i}, g^{b_j}, g^{sb_j}, g^{a^i b_j}, g^{a^i/b_j^2} \quad \forall (i,j) \in [q,q]$$
$$g^{a^i/b_j} \quad \forall (i,j) \in [2q,q],\ i \neq q+1$$
$$g^{a^i b_j/b_{j'}^2} \quad \forall (i,j,j') \in [2q,q,q],\ j \neq j'$$
$$g^{sa^i b_j/b_{j'}}, g^{sa^i b_j/b_{j'}^2} \quad \forall (i,j,j') \in [q,q,q],\ j \neq j'$$

It is difficult to distinguish between $\left(\mathbf{y}, e(g,g)^{sa^{q+1}}\right)$ and $(\mathbf{y}, Z)$, where $g \in \mathbb{G}_1, Z \in \mathbb{G}_2, a, s, b_1, \cdots, b_q \in \mathbb{Z}_p$.

Definition 2 (Decisional Diffie-Hellman Assumption, DDH): If any PPT adversary $\mathcal{A}$ can distinguish between tuple $(g, g^{z_1}, g^{z_2}, g^{z_1 z_2})$ and (g, g^{z_1}, g^{z_2}, Q) with a negligible advantage, the advantage $Adv_{\mathcal{A}}$ of $\mathcal{A}$ is defined as

$$Adv_{\mathcal{A}} = \left|\Pr\left[\mathcal{A}\left(g, g^{z_1}, g^{z_2}, g^{z_1 z_2}\right) = 1\right] - \Pr\left[\mathcal{A}\left(g, g^{z_1}, g^{z_2}, Q\right) = 1\right]\right|$$

where $g, Q \in \mathbb{G}_1, z_1, z_2 \in \mathbb{Z}_p$.

IV. SCHEME ARCHITECTURE

A. SCHEME MODEL

In this section, we describe the participants and the framework of scheme. Table 3 lists that each participant needs

TABLE 3. The participant and algorithm.

Participant	Algorithm
AA	$Setup, KeyGen$
CS	$Search$
OKGS	$KeyGen_{out}$
OES	$Encrypt_{out}$
ODS	$Decrypt_{out}$
ACS	$Equality\ Test$
DO	$Encrypt\,(Encrypt-files, Encrypt-keyword)$
DU	$TokenGen, Decrypt, TrapGen$

to perform algorithms. For example, AA is responsible for executing the *Setup* and *KeyGen* algorithm. The specific description is shown in the next two subsections.

1) SCHEME FRAMEWORK

The structure of scheme is shown in Figure 2, (1) to (11) are the algorithms in the scheme. It mainly contains eight parties.

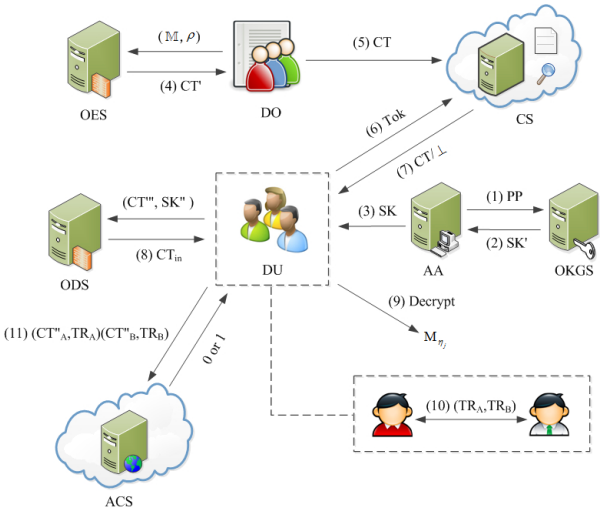

FIGURE 2. The framework of scheme.

Attribute Authority (AA). AA is in charge of system establishment. It generates public parameter and master secret key, and collaborates with OKGS to generate the data user's secret key.

Cloud Server (CS). CS takes charge of storing uploaded data and helping data user to search for desired ciphertext according to the index, then it returns corresponding result.

Outsourced Key Generation Server (OKGS). OKGS produces data user's outsourced secret key according to public parameter and sends it to AA.

Outsourced Encryption Server (OES). OES generates outsourced ciphertext based on the access structure and returns it to DO.

Outsourced Decryption Server (ODS). After receiving partial ciphertext and partial secret key, ODS performs outsourced decryption algorithm, obtains the outsourced ciphertext, and then transmits it to DU.

Authorized Cloud Server (ACS). ACS is responsible for judging the equality of two ciphertexts according to the equality test algorithm, and returns corresponding result.

Data Owner (DO). DO is in charge of encrypting data, collaborating with OES to generate ciphertext, and uploading it to CS.

Data User (DU). DU is responsible for generating a token of the keyword and sending it to CS. If the search is successful, DU will gain the ciphertext, and then collaborate with ODS to obtain the plaintext. When it is necessary to determine whether two ciphertexts contain the same plaintext, DU sends two pairs of ciphertexts and trapdoors to ACS, ACS executes equality test algorithm and returns result to DU.

2) THE DEFINITIONS OF SCHEME

The proposed scheme mainly includes 11 algorithms: *Setup*, $KeyGen_{out}$, *KeyGen*, $Encrypt_{out}$, *Encrypt*, *TokenGen*, *Search*, $Decrypt_{out}$, *Decrypt*, *TrapGen*, *Equality Test*, which are defined as follows:

$Setup\ (\kappa) \rightarrow PP, MSK$. The setup algorithm is performed by AA. Inputting security parameter κ, it outputs public parameter PP and master secret key MSK.

$KeyGen_{out}\ (PP, S) \rightarrow SK'$. The outsourced key generation algorithm is performed by OKGS. Inputting public parameter PP and attribute set S, it outputs outsourced secret key SK'.

$KeyGen\left(MSK, SK'\right) \rightarrow SK$. The key generation algorithm is carried out by AA. Inputting master secret key MSK and outsourced secret key SK', it outputs secret key SK.

$Encrypt_{out}\left(PP, \left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)\right) \rightarrow CT'$. The outsourced encryption algorithm is carried out by OES. Inputting public parameter PP and access policy $(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\})$, it outputs outsourced ciphertext CT'.

Encrypt. The encryption algorithm is executed by DO. The encryption process is divided into two parts: encrypting files and encrypting keyword.

(i) $Encrypt - files\left(PP, H_1\left(M_{\eta_j}\right), CT', \left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)\right) \rightarrow CT''_{\eta_j}$. $M_{\eta_1}, M_{\eta_2}, \cdots, M_{\eta_m}$ is a file set that contains the keyword w. First, DO calculates hash value $H_1\left(M_{\eta_j}\right)$ of the file, and symmetrically encrypts M_{η_j} with $H_1\left(M_{\eta_j}\right)$, $E_{H_1\left(M_{\eta_j}\right)}\left(M_{\eta_j}\right)$ is obtained, then $H\left(M_{\eta_j}\right)$ is encrypted by the following algorithm, where $j \in [1, m]$.

The algorithm inputs public parameter PP, hash value $H_1\left(M_{\eta_j}\right)$, outsourced ciphertext CT' and access policy $\left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)$, outputs ciphertext CT''_{η_j}.

(ii) $Encrypt - keyword\left(PP, w, CT'\right) \rightarrow Index$. The algorithm inputs public parameter PP, keyword w and outsourced ciphertext CT', outputs keyword index $Index$.

Finally, the encryption algorithm outputs ciphertext $CT = \left(\left\{E_{H_1\left(M_{\eta_j}\right)}\left(M_{\eta_j}\right), CT''_{\eta_j}\right\}_{j\in[1,m]}, Index\right)$.

$TokenGen\left(PP, w', SK\right) \rightarrow Tok$. The token generation algorithm is carried out by DU. Inputting public parameter PP, keyword w' and secret key SK, it outputs keyword token Tok.

Search $(Index, Tok) \rightarrow CT/\perp$. The search algorithm is performed by CS. Inputting keyword index *Index* and token *Tok*. If *Index* matches *Tok* successfully, the algorithm outputs ciphertext CT, otherwise outputs $\perp$.

$Decrypt_{out}\left(CT''', SK''\right) \rightarrow CT_{in}$. The outsourced decryption algorithm is performed by ODS. Inputting partial ciphertext $CT'''$and partial secret key SK'', it outputs intermediate ciphertext CT_{in}.

Decrypt $(CT, SK, CT_{in}) \rightarrow H_1\left(M_{\eta_j}\right)$. The decryption algorithm is carried out by DU. Inputting ciphertext CT, secret key SK and intermediate ciphertext CT_{in}, it outputs hash value $H_1\left(M_{\eta_j}\right)$ of the file. Then, DU symmetrically decrypts $E_{H_1\left(M_{\eta_j}\right)}\left(M_{\eta_j}\right)$ with $H_1\left(M_{\eta_j}\right)$ to get M_{η_j}, where $j \in [1, m]$.

TrapGen $(PP, S, SK) \rightarrow TR$. The trapdoor generation algorithm is performed by DU. Inputting public parameter PP, attribute set S and secret key SK, it outputs trapdoor TR.

Equality Test $\left(\left(CT''_A, TR_A\right), \left(CT''_B, TR_B\right)\right) \rightarrow \{0, 1\}$. The equality test algorithm is carried out by ACS. Inputting two pairs of ciphertexts and trapdoors: $\left(CT''_A, TR_A\right)$ and $\left(CT''_B, TR_B\right)$. If two ciphertexts contain the same plaintext, the algorithm outputs 1 and otherwise outputs 0.

B. SECURITY MODEL

In this section, we present definitions of chosen-plaintext security and chosen-keyword security for the scheme. They are described as an interactive game between the adversary $\mathcal{A}$ and the simulator $\mathcal{B}$ (as shown in Figure 3). The game process mainly includes 5 stages: Setup, Phase 1, Challenge, Phase 2 and Guess. The specific interaction process is described in the secure game.

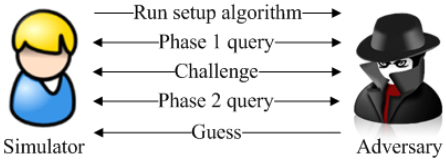

FIGURE 3. The process of secure game.

1) GAME 1: CHOSEN-PLAINTEXT SECURITY

The chosen-plaintext secure game is as follows:

Initialization. $\mathcal{A}$ submits a challenge access policy $\left(\mathbf{M}^*, \rho^*, \left\{x_{\rho^*(i)}\right\}\right)$ to $\mathcal{B}$.

Setup. $\mathcal{B}$ executes *Setup* (κ) algorithm, produces public parameter PP and master secret key MSK, and makes PP to public.

Phase 1. $\mathcal{A}$ sends an attribute set S' to $\mathcal{B}$ and issues adaptive queries. However, the restriction is that the attribute set S' cannot satisfy the access policy $\left(\mathbf{M}^*, \rho^*, \left\{x_{\rho^*(i)}\right\}\right)$ for each query.

Secret key query: $\mathcal{B}$ performs $KeyGen_{out}\left(PP, S'\right)$ algorithm and $KeyGen\left(MSK, SK'\right)$ algorithm to produce the corresponding secret key, and returns to $\mathcal{A}$.

Trapdoor query: $\mathcal{B}$ performs $TrapGen\left(PP, S', SK\right)$ algorithm according to secret key, transmits trapdoor to $\mathcal{A}$.

Challenge. $\mathcal{A}$ submits two hash values $H_1(M_0)$, $H_1(M_1)$ to $\mathcal{B}$. $\mathcal{B}$ flips a coin to choose $b \in \{0, 1\}$ and performs $Encrypt_{out}\left(PP, \left(\mathbf{M}^*, \rho^*, \left\{x_{\rho^*(i)}\right\}\right)\right)$ and $Encrypt-files\left(PP, H_1\left(M_b\right), CT', \left(\mathbf{M}^*, \rho^*, \left\{x_{\rho^*(i)}\right\}\right)\right)$ algorithm to obtain ciphertext CT''^*_b, then sends it to $\mathcal{A}$.

Phase 2. $\mathcal{A}$ repeats the inquiry of Phase 1, but the restriction is that the attribute set cannot satisfy the access policy.

Guess. $\mathcal{A}$ outputs a guess b'. If there is $b' = b$, it means that $\mathcal{A}$ winning the game. The advantage of $\mathcal{A}$ wins the game is defined as

$$Adv_{\mathcal{A}} = \left|\Pr\left[b' = b\right] - 1/2\right|$$

Definition 3. If any PPT adversary wins the above game at most with a negligible advantage, our scheme proves to be chosen-plaintext security.

2) GAME 2: CHOSEN-KEYWORD SECURITY

The chosen-keyword secure game is as follows:

Setup. $\mathcal{B}$ executes *Setup* (κ) algorithm to produce public parameter PP and master secret key MSK, while making PP to public.

Phase 1. $\mathcal{A}$ issues adaptive queries:

Token query: $\mathcal{A}$ gives a query keyword w', $\mathcal{B}$ runs $TokenGen\left(PP, w', SK\right)$ algorithm, sends *Tok* to $\mathcal{A}$.

Challenge. $\mathcal{A}$ chooses two keywords w_0, w_1 at random, $\mathcal{B}$ flips a coin to select $b \in \{0, 1\}$, executes $Encrypt-keyword\left(PP, w_b, CT'\right)$ algorithm to produce keyword index *Index*, sends it to $\mathcal{A}$.

Phase 2. $\mathcal{A}$ continues the inquiry of Phase 1, but the restriction is that $\mathcal{A}$ cannot query the keyword w_0, w_1 anymore.

Guess. $\mathcal{A}$ outputs a guess b'. If there is $b' = b$, it means that $\mathcal{A}$ winning the game.

Definition 4: If any PPT adversary wins the above game at most with a negligible advantage, the proposed scheme proves to be chosen-keyword security.

V. KS-ABESwET

Aiming at resource-constrained IoT devices, through the analysis of existing schemes, we introduce the keyword search, outsourcing, equality test algorithm based on the general ABE scheme to construct our solution, realize various functions of cryptographic scheme. The proposed scheme mainly includes 11 steps, specific structure of the algorithm is described below.

Setup $(\kappa) \rightarrow PP, MSK$: This algorithm inputs security parameter κ, it chooses a bilinear mapping $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$, where $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups whose order is prime p, g is a generator of $\mathbb{G}_1$. The algorithm selects $u, h, d, f \in \mathbb{G}_1$, $r_1, r_2, r_3, r_4, \alpha, \alpha' \in \mathbb{Z}_p$ at random, computes $g_1 = g^{r_1}, g_2 = g^{r_2}, g_3 = g^{r_3}, g_4 = g^{r_4}$, sets $H_1 : \{0, 1\}^* \rightarrow \mathbb{G}_1$, $H_2 : \mathbb{G}_2 \rightarrow \mathbb{G}_1 \times \mathbb{Z}_p$, $H_3 : \mathbb{G}_2 \rightarrow \mathbb{G}_1$, $H_4 : \{0, 1\}^* \rightarrow \mathbb{Z}_p$, where H_1, H_2, H_3, H_4 are one-way hash functions. The algorithm outputs public parameter PP and

master secret key MSK as follows:

$$PP = \left(g, u, h, d, f, g_1, g_2, g_3, g_4, H_1, H_2, H_3, H_4, e(g,g)^{\alpha}, e(g,g)^{\alpha'}\right)$$
$$MSK = \left(r_1, r_2, r_3, r_4, \alpha, \alpha'\right)$$

Making PP to public.

$KeyGen_{out}\left(PP, S\right) \rightarrow SK'$: This algorithm takes public parameter PP and attribute set S as input. Let k indicate the size of S, $x_1, \cdots, x_k \in \mathbb{Z}_p$ denote attribute value in S, namely $S = \{x_1, \cdots, x_k\} \subseteq \mathbb{Z}_p$. It randomly chooses $y, y', y_1, \cdots, y_k, y'_1, \cdots, y'_k \in \mathbb{Z}_p$, computes

$$L_{i,1} = \left(u^{x_i}h\right)^{y_i} d^{-y}, \quad L'_{i,1} = \left(u^{x_i}h\right)^{y_i},$$
$$L_{i,2} = \left(u^{x_i}h\right)^{y'_i} d^{-y'}, \quad L'_{i,2} = \left(u^{x_i}h\right)^{y'_i},$$
$$L_3 = g^{y}, \quad L'_3 = g^{y'}, \; L_{i,4} = g^{y_i},$$
$$L'_{i,4} = g^{y'_i}, \; L_5 = f^{y}, L'_5 = f^{y'}$$

The algorithm outputs outsourced secret key

$$SK' = \left(L_{i,1}, L'_{i,1}, L_{i,2}, L'_{i,2}, L_3, L'_3, L_{i,4}, L'_{i,4}, L_5, L'_5\right)_{i\in[1,k]}$$

$KeyGen\left(MSK, SK'\right) \rightarrow SK$: This algorithm takes master secret key MSK and outsourced secret key SK'. It performs the following calculation:

$$T_1 = g^{\alpha}(L_5)^{r_1 r_2}\left(L'_5\right)^{r_3 r_4} = g^{\alpha} f^{r_1 r_2 y + r_3 r_4 y'},$$
$$T'_1 = g^{\alpha'}(L_5)^{r_1 r_2}\left(L'_5\right)^{r_3 r_4} = g^{\alpha'} f^{r_1 r_2 y + r_3 r_4 y'},$$
$$T_2 = (L_3)^{r_1 r_2}\left(L'_3\right)^{r_3 r_4} = g^{r_1 r_2 y + r_3 r_4 y'},$$
$$T_{i,1} = \left(L_{i,1}\right)^{r_2} = \left(\left(u^{x_i}h\right)^{y_i} d^{-y}\right)^{r_2},$$
$$T'_{i,1} = \left(L'_{i,1}\right)^{r_2} = \left(u^{x_i}h\right)^{y_i r_2},$$
$$T_{i,2} = \left(L_{i,1}\right)^{r_1} = \left(\left(u^{x_i}h\right)^{y_i} d^{-y}\right)^{r_1},$$
$$T'_{i,2} = \left(L'_{i,1}\right)^{r_1} = \left(u^{x_i}h\right)^{y_i r_1},$$
$$T_{i,3} = \left(L_{i,4}\right)^{r_1 r_2}\left(L'_{i,4}\right)^{r_3 r_4} = g^{r_1 r_2 y_i + r_3 r_4 y'_i},$$
$$T_{i,4} = \left(L_{i,2}\right)^{r_4} = \left(\left(u^{x_i}h\right)^{y'_i} d^{-y'}\right)^{r_4},$$
$$T'_{i,4} = \left(L'_{i,2}\right)^{r_4} = \left(u^{x_i}h\right)^{y'_i r_4},$$
$$T_{i,5} = \left(L_{i,2}\right)^{r_3} = \left(\left(u^{x_i}h\right)^{y'_i} d^{-y'}\right)^{r_3},$$
$$T'_{i,5} = \left(L'_{i,2}\right)^{r_3} = \left(u^{x_i}h\right)^{y'_i r_3},$$
$$Z_1 = g^{r_1 r_2}, Z_2 = f^{r_1 r_2}$$

The algorithm outputs secret key

$$SK = (SK_1, SK_2)$$

where

$$SK_1 = \left(T_1, T'_1, T_2, \left\{T_{i,1}, T'_{i,1}, T_{i,2}, T'_{i,2}, T_{i,3}, T_{i,4}, T'_{i,4}, T_{i,5}, T'_{i,5}\right\}_{i\in[1,k]}\right)$$
$$SK_2 = (Z_1, Z_2)$$

Let SK_1 be used to decrypt ciphertext and generate the trapdoor of equality test, SK_2 be used to produce the keyword token.

$Encrypt_{out}\left(PP, \left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)\right) \rightarrow CT'$: This algorithm inputs public parameter PP and access policy $\left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)$. In the access policy $\left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)$, $\mathbf{M}$ is a shared matrix of size $l \times n$, ρ denotes a mapping from $\{1, 2, \cdots, l\}$ to P, that is map each row of $\mathbf{M}$ to an attribute, $x_{\rho(i)}$ is attribute value. If the authorized set of data user is Λ, the algorithm defines $I = \{i : \rho(i) \in \Lambda\} \subseteq \{1, 2, \cdots, l\}$. Hereafter, it randomly selects $\sigma, t, t_{1,1}, \cdots, t_{l,1}, t_{1,2}, \cdots, t_{l,2}, \sigma_1, \cdots, \sigma_l \in \mathbb{Z}_p$, computes

$$C'_i = d^{\sigma_i}, \quad D'_{i,1} = g_1^{\sigma_i - t_{i,1}}, \; D'_{i,2} = g_3^{\sigma_i - t_{i,2}},$$
$$D'_{i,3} = g_2^{t_{i,1}}, \quad D'_{i,4} = g_4^{t_{i,2}}, \; E'_i = \left(u^{x_{\rho(i)}}h\right)^{-\sigma_i},$$
$$F' = g^{\sigma}, \quad D'_1 = g_1^{\sigma - t}, D'_2 = g_2^{t}$$

The algorithm outputs outsourced ciphertext

$$CT' = \left(F', D'_1, D'_2, \left\{C'_i, D'_{i,1}, D'_{i,2}, D'_{i,3}, D'_{i,4}, E'_i\right\}_{i\in I}\right)$$

Encrypt: The encryption process is divided into two parts: encrypting files and encrypting keyword.

(i) $Encrypt - files\left(PP, H_1\left(M_{\eta_j}\right), CT', \left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)\right) \rightarrow CT''_{\eta_j}$: $M_{\eta_1}, M_{\eta_2}, \cdots, M_{\eta_m}$ is a file set that contains the keyword w. First, the hash value $H_1\left(M_{\eta_j}\right)$ of the file M_{η_j} is calculated, let $H_1\left(M_{\eta_j}\right)$ be a symmetric key to encrypt M_{η_j}, then $E_{H_1\left(M_{\eta_j}\right)}\left(M_{\eta_j}\right)$ is obtained, and finally $H_1\left(M_{\eta_j}\right)$ is encrypted by the following algorithm, where $j \in [1, m]$.

This algorithm makes public parameter PP, hash value $H_1\left(M_{\eta_j}\right) \in \mathbb{G}_1$ $(j \in [1, m])$, outsourced ciphertext CT' and access policy $\left(\mathbf{M}, \rho, \left\{x_{\rho(i)}\right\}\right)$ as input. It chooses a vector $\mathbf{v} = (s, \gamma_2, \cdots, \gamma_n) \in \mathbb{Z}_p^n$, s is a secret, $\gamma_2, \cdots, \gamma_n$ are random value. For $i \in I$, it calculates $\lambda_i = \mathbf{v} \cdot \mathbf{M}_i$, $\mathbf{M}_i$ represents the vector of i-th row in $\mathbf{M}$. It chooses $\tau \in \mathbb{Z}_p$ at random, computes

$$\hat{C}_{\eta_j} = \left(H_1\left(M_{\eta_j}\right) || \tau\right) \oplus H_2\left(e(g,g)^{\alpha s}\right),$$
$$\tilde{C}_{\eta_j} = H_1\left(M_{\eta_j}\right)^{\tau} \cdot H_3\left(e(g,g)^{\alpha'\tau}\right), \quad D = g^{s},$$
$$D' = g^{\tau}, \quad C' = f^{\tau}, \; C_i = f^{\lambda_i}\left(C'_i\right) = f^{\lambda_i} d^{\sigma_i},$$
$$D_{i,1} = D'_{i,1} = g_1^{\sigma_i - t_{i,1}}, \quad D_{i,2} = D'_{i,2} = g_3^{\sigma_i - t_{i,2}},$$
$$D_{i,3} = D'_{i,3} = g_2^{t_{i,1}}, \quad D_{i,4} = D'_{i,4} = g_4^{t_{i,2}},$$
$$E_i = E'_i = \left(u^{x_{\rho(i)}}h\right)^{-\sigma_i}$$

The algorithm outputs ciphertext

$$CT''_{\eta_j} = \left(\hat{C}_{\eta_j}, \tilde{C}_{\eta_j}, D, D', C', \left\{C_i, D_{i,1}, D_{i,2}, D_{i,3}, D_{i,4}, E_i\right\}_{i\in I}\right)$$

where $j \in [1, m]$.

(ii) $Encrypt - keyword\left(PP, w, CT'\right) \rightarrow Index$: This algorithm makes public parameter PP, keyword w and outsourced ciphertext CT' as input. It arbitrarily chooses $\bar{\tau} \in \mathbb{Z}_p$, computes

$$F = f^{\bar{\tau}} \cdot \left(F'\right)^{-H_4(w)} = f^{\bar{\tau}} g^{-H_4(w)\sigma},$$
$$D_1 = D'_1 = g_1^{\sigma - t}, \quad D_2 = D'_2 = g_2^{t}, D_3 = g^{\bar{\tau}}$$

The algorithm outputs keyword index

$$Index = (F, D_1, D_2, D_3)$$

Finally, the encryption algorithm outputs ciphertext

$$CT = \left(\left\{ E_{H_1\left(M_{\eta_j}\right)}\left(M_{\eta_j}\right), CT''_{\eta_j} \right\}_{j\in[1,m]}, Index \right)$$

TokenGen $\left(PP, w', SK\right) \rightarrow Tok$: Taking as inputs public parameter PP, keyword w' and secret key SK. This algorithm randomly selects $\varepsilon \in \mathbb{Z}_p$, computes

$$K_1 = g_2^{H_4(w')\varepsilon}, \quad K_2 = g_1^{H_4(w')\varepsilon},$$
$$K_3 = (Z_1)^{\varepsilon} = g^{r_1 r_2 \varepsilon}, \quad K_4 = (Z_2)^{\varepsilon} = f^{r_1 r_2 \varepsilon}$$

The algorithm outputs keyword token

$$Tok = (K_1, K_2, K_3, K_4)$$

Search $(Index, Tok) \rightarrow CT/\perp$: Taking as inputs keyword index $Index$ and token Tok. Whether the data user can search for the desired ciphertext is determined by whether the following equation holds. If the equation (1) holds, $Index$ matches Tok successfully, and the algorithm outputs ciphertext CT, otherwise outputs $\perp$.

$$\frac{e\left(D_3, K_4\right)}{e\left(F, K_3\right) e\left(D_1, K_1\right) e\left(D_2, K_2\right)} \stackrel{?}{=} 1 \tag{1}$$

$Decrypt_{out}\left(CT''', SK''\right) \rightarrow CT_{in}$: DU transmits partial ciphertext $CT''' = \left(C', \left\{C_i, D_{i,1}, D_{i,2}, D_{i,3}, D_{i,4}, E_i\right\}_{i\in I}\right)$ and partial secret key $SK'' = \left(T_2, \left\{T_{i,1}, T'_{i,1}, T_{i,2}, T'_{i,2}, T_{i,3}, T_{i,4}, T'_{i,4}, T_{i,5}, T'_{i,5}\right\}_{i\in[1,k]}\right)$ to ODS to perform outsourced decryption algorithm.

This algorithm inputs CT''' and SK''. If data user's attribute set satisfies access structure, there is a constant $\left\{\omega_i \in \mathbb{Z}_p\right\}_{i\in I}$, so that $\sum_{i\in I} \omega_i \mathbf{M}_i = (1, 0, \cdots, 0)$, where $\mathbf{M}_i$ indicates the vector of i-th row in $\mathbf{M}$. It performs following calculation:

$$Q_1 = \prod_{i\in I} \left(e\left(C_i, T_2\right) e\left(D_{i,1}, T_{i,1}\right) e\left(D_{i,3}, T_{i,2}\right) e\left(E_i, T_{i,3}\right) e\left(D_{i,2}, T_{i,4}\right) e\left(D_{i,4}, T_{i,5}\right)\right)^{\omega_i} \tag{2}$$

$$Q_2 = \prod_{i\in I} \left(e\left(C', T_2\right) e\left(D_{i,1}, T'_{i,1}\right) e\left(D_{i,3}, T'_{i,2}\right) e\left(E_i, T_{i,3}\right) e\left(D_{i,2}, T'_{i,4}\right) e\left(D_{i,4}, T'_{i,5}\right)\right) \tag{3}$$

The algorithm outputs intermediate ciphertext

$$CT_{in} = (Q_1, Q_2)$$

Decrypt $(CT, SK, CT_{in}) \rightarrow H_1\left(M_{\eta_j}\right)$: This algorithm takes ciphertext CT, secret key SK and intermediate ciphertext CT_{in} as input. It computes

$$Y_1 = \frac{e\left(D, T_1\right)}{Q_1} \tag{4}$$

$$Y_2 = \frac{e\left(D', T'_1\right)}{Q_2} \tag{5}$$

$$H_1\left(M_{\eta_j}\right) || \tau = \hat{C}_{\eta_j} \oplus H_2\left(Y_1\right) \tag{6}$$

If $D' = g^{\tau}$ and $\tilde{C}_{\eta_j} / H_1\left(M_{\eta_j}\right)^{\tau} = H_3\left(Y_2\right)$ hold, the algorithm outputs hash value $H_1\left(M_{\eta_j}\right)$. Then DU symmetrically decrypt $E_{H_1\left(M_{\eta_j}\right)}\left(M_{\eta_j}\right)$ with $H_1\left(M_{\eta_j}\right)$ to obtain M_{η_j}, where $j \in [1, m]$.

TrapGen $(PP, S, SK) \rightarrow TR$: This algorithm inputs public parameter PP, attribute set S and secret key SK. It selects $\bar{\varepsilon} \in \mathbb{Z}_p$ at random, and computes

$$R_1 = T'_1 = g^{\alpha} f^{r_1 r_2 y + r_3 r_4 y'}, R_2 = T_2 = g^{r_1 r_2 y + r_3 r_4 y'},$$
$$R_{i,1} = T'^{\bar{\varepsilon}}_{i,1} = \left(u^{x_i} h\right)^{y_i r_2 \bar{\varepsilon}}, \quad R_{i,2} = T'^{\bar{\varepsilon}}_{i,2} = \left(u^{x_i} h\right)^{y_i r_1 \bar{\varepsilon}},$$
$$R_{i,3} = T'^{\bar{\varepsilon}}_{i,3} = g^{r_1 r_2 y_i \bar{\varepsilon} + r_3 r_4 y'_i \bar{\varepsilon}}, \quad R_{i,4} = T'^{\bar{\varepsilon}}_{i,4} = \left(u^{x_i} h\right)^{y'_i r_4 \bar{\varepsilon}},$$
$$R_{i,5} = T'^{\bar{\varepsilon}}_{i,5} = \left(u^{x_i} h\right)^{y'_i r_3 \bar{\varepsilon}}$$

The algorithm outputs trapdoor

$$TR = \left(R_1, R_2, \left\{R_{i,1}, R_{i,2}, R_{i,3}, R_{i,4}, R_{i,5}\right\}_{i\in[1,k]}\right)$$

Equality Test $\left(\left(CT''_A, TR_A\right), \left(CT''_B, TR_B\right)\right) \rightarrow \{0, 1\}$: This algorithm inputs two pairs of ciphertexts and trapdoors: $\left(CT''_A, TR_A\right)$ and $\left(CT''_B, TR_B\right)$. It computes

$$\Phi'_A = \frac{e\left(D'_A, R_{1,A}\right)}{e\left(C'_A, R_{2,A}\right) \prod_{i\in I} e\left(D_{\{i,1\},A}, R_{\{i,1\},A}\right)} \times \frac{1}{\prod_{i\in I} e\left(D_{\{i,3\},A}, R_{\{i,2\},A}\right) e\left(E_{i,A}, R_{\{i,3\},A}\right)} \times \frac{1}{\prod_{i\in I} e\left(D_{\{i,2\},A}, R_{\{i,4\},A}\right) e\left(D_{\{i,4\},A}, R_{\{i,5\},A}\right)} \tag{7}$$

$$\Phi_A = \tilde{C}_A / H_3\left(\Phi'_A\right) \tag{8}$$

$$\Phi'_B = \frac{e\left(D'_B, R_{1,B}\right)}{e\left(C'_B, R_{2,B}\right) \prod_{i\in I} e\left(D_{\{i,1\},B}, R_{\{i,1\},B}\right)} \times \frac{1}{\prod_{i\in I} e\left(D_{\{i,3\},B}, R_{\{i,2\},B}\right) e\left(E_{i,B}, R_{\{i,3\},B}\right)} \times \frac{1}{\prod_{i\in I} e\left(D_{\{i,2\},B}, R_{\{i,4\},B}\right) e\left(D_{\{i,4\},B}, R_{\{i,5\},B}\right)} \tag{9}$$

$$\Phi_B = \tilde{C}_B / H_3\left(\Phi'_B\right) \tag{10}$$

If $e\left(\Phi_A, D'_B\right) = e\left(\Phi_B, D'_A\right)$ holds, that is $H_1\left(M_A\right) = H_1\left(M_B\right)$, it shows $M_A = M_B$, the algorithm outputs 1 and otherwise outputs 0.

Correctness: The following is the correctness verification process of the algorithm in our scheme.

1) Search: If the keyword in index is the same as the keyword in token, then the equation (1) holds. It indicates that the matching is successful, DU can obtain the desired

ciphertext.

$$\begin{aligned}
&\frac{e\left(D_3, K_4\right)}{e\left(F, K_3\right) e\left(D_1, K_1\right) e\left(D_2, K_2\right)} \\
&= \frac{e\left(g^{\bar{\tau}}, f^{r_1 r_2 \varepsilon}\right)}{e\left(f^{\bar{\tau}} g^{-H_4(w)\sigma}, g^{r_1 r_2 \varepsilon}\right) e\left(g_1^{\sigma-t}, g_2^{H_4(w')\varepsilon}\right)} \\
&\quad \times \frac{1}{e\left(g_2^t, g_1^{H_4(w')\varepsilon}\right)} \\
&= \frac{e\left(g^{\bar{\tau}}, f^{r_1 r_2 \varepsilon}\right)}{e(f, g)^{r_1 r_2 \bar{\tau} \varepsilon} e(g, g)^{-r_1 r_2 \sigma \varepsilon H_4(w)} e(g, g)^{r_1 r_2 \sigma \varepsilon H_4(w')}} \\
&\quad \times \frac{1}{e(g, g)^{-r_1 r_2 t \varepsilon H_4(w')} e(g, g)^{r_1 r_2 t \varepsilon H_4(w')}} \\
&= 1
\end{aligned}$$

2) Decryption: The decryption operation is as follows:

$$\begin{aligned}
Q_1 &= \prod_{i \in I} \left(e\left(C_i, T_2\right) e\left(D_{i,1}, T_{i,1}\right) e\left(D_{i,3}, T_{i,2}\right)\right. \\
&\quad \left. \cdot\, e\left(E_i, T_{i,3}\right) e\left(D_{i,2}, T_{i,4}\right) e\left(D_{i,4}, T_{i,5}\right)\right)^{\omega_i} \\
&= \prod_{i \in I} \left(e\left(f^{\lambda_i}, g^{r_1 r_2 y + r_3 r_4 y'}\right) e\left(d^{\sigma_i}, g^{r_1 r_2 y + r_3 r_4 y'}\right)\right. \\
&\quad \cdot e\left(g^{r_1 \sigma_i - r_1 t_{i,1}}, u^{x_i y_i r_2}\right) e\left(g^{r_1 \sigma_i - r_1 t_{i,1}}, h^{y_i r_2}\right) \\
&\quad \cdot e\left(g^{r_1 \sigma_i - r_1 t_{i,1}}, d^{-y r_2}\right) e\left(g^{r_2 t_{i,1}}, u^{x_i y_i r_1}\right) \\
&\quad \cdot e\left(g^{r_2 t_{i,1}}, h^{y_i r_1}\right) e\left(g^{r_2 t_{i,1}}, d^{-y r_1}\right) \\
&\quad \cdot e\left(u^{-x_{\rho(i)}\sigma_i}, g^{r_1 r_2 y_i + r_3 r_4 y_i'}\right) \\
&\quad \cdot e\left(h^{-\sigma_i}, g^{r_1 r_2 y_i + r_3 r_4 y_i'}\right) \\
&\quad \cdot e\left(g^{r_3 \sigma_i - r_3 t_{i,2}}, u^{x_i y_i' r_4}\right) e\left(g^{r_3 \sigma_i - r_3 t_{i,2}}, h^{y_i' r_4}\right) \\
&\quad \cdot e\left(g^{r_3 \sigma_i - r_3 t_{i,2}}, d^{-y' r_4}\right) e\left(g^{r_4 t_{i,2}}, u^{x_i y_i' r_3}\right) \\
&\quad \left. \cdot\, e\left(g^{r_4 t_{i,2}}, h^{y_i' r_3}\right) e\left(g^{r_4 t_{i,2}}, d^{-y' r_3}\right)^{\omega_i}\right) \\
&= \prod_{i \in I} e\left(f^{\lambda_i \omega_i}, g^{r_1 r_2 y + r_3 r_4 y'}\right) \\
&= e\left(f^{\sum_{i \in I} \lambda_i \omega_i}, g^{r_1 r_2 y + r_3 r_4 y'}\right) \\
&= e\left(f^s, g^{r_1 r_2 y + r_3 r_4 y'}\right) \\
Y_1 &= \frac{e\left(D, T_1\right)}{Q_1} \\
&= \frac{e\left(g^s, g^\alpha\right) e\left(g^s, f^{r_1 r_2 y + r_3 r_4 y'}\right)}{e\left(f^s, g^{r_1 r_2 y + r_3 r_4 y'}\right)} \\
&= e(g, g)^{\alpha s} \\
Q_2 &= \prod_{i \in I} \left(e\left(C', T_2\right) e\left(D_{i,1}, T_{i,1}'\right) e\left(D_{i,3}, T_{i,2}'\right)\right. \\
&\quad \left. \cdot\, e\left(E_i, T_{i,3}\right) e\left(D_{i,2}, T_{i,4}'\right) e\left(D_{i,4}, T_{i,5}'\right)\right) \\
&= e\left(f^\tau, g^{r_1 r_2 y + r_3 r_4 y'}\right) \prod_{i \in I} \left(e\left(g^{r_1 \sigma_i - r_1 t_{i,1}}, u^{x_i y_i r_2}\right)\right. \\
&\quad \left. \cdot\, e\left(g^{r_1 \sigma_i - r_1 t_{i,1}}, h^{y_i r_2}\right) e\left(g^{r_2 t_{i,1}}, u^{x_i y_i r_1}\right)\right. \\
&\quad \cdot e\left(g^{r_2 t_{i,1}}, h^{y_i r_1}\right) e\left(u^{-x_{\rho(i)}\sigma_i}, g^{r_1 r_2 y_i + r_3 r_4 y_i'}\right) \\
&\quad \cdot e\left(h^{-\sigma_i}, g^{r_1 r_2 y_i + r_3 r_4 y_i'}\right) e\left(g^{r_3 \sigma_i - r_3 t_{i,2}}, u^{x_i y_i' r_4}\right) \\
&\quad \cdot e\left(g^{r_3 \sigma_i - r_3 t_{i,2}}, h^{y_i' r_4}\right) e\left(g^{r_4 t_{i,2}}, u^{x_i y_i' r_3}\right) \\
&\quad \left. \cdot\, e\left(g^{r_4 t_{i,2}}, h^{y_i' r_3}\right)\right) \\
&= e\left(f^\tau, g^{r_1 r_2 y + r_3 r_4 y'}\right) \\
Y_2 &= \frac{e\left(D', T'_1\right)}{Q_2} \\
&= \frac{e\left(g^\tau, g^{\alpha'}\right) e\left(g^\tau, f^{r_1 r_2 y + r_3 r_4 y'}\right)}{e\left(f^\tau, g^{r_1 r_2 y + r_3 r_4 y'}\right)} \\
&= e(g, g)^{\alpha' \tau}
\end{aligned}$$

Through the above calculation of equation (2)-(5), there is $H_1\left(M_{\eta_j}\right) || \tau = \hat{C}_{\eta_j} \oplus H_2\left(Y_1\right)$. If $D' = g^\tau$ and $\tilde{C}_{\eta_j} / H_1\left(M_{\eta_j}\right)^\tau = H_3\left(Y_2\right)$ hold, the ciphertext can be decrypted, where $j \in [1, m]$.

3) Equality test: For equation (7)-(10), the following calculation is performed.

$$\begin{aligned}
\Phi_A' &= \frac{e\left(D_A', R_{1,A}\right)}{e\left(C_A', R_{2,A}\right) \prod_{i \in I} e\left(D_{\{i,1\},A}, R_{\{i,1\},A}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(D_{\{i,3\},A}, R_{\{i,2\},A}\right) e\left(E_{i,A}, R_{\{i,3\},A}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(D_{\{i,2\},A}, R_{\{i,4\},A}\right) e\left(D_{\{i,4\},A}, R_{\{i,5\},A}\right)} \\
&= \frac{e\left(g^{\tau_A}, g^{\alpha'}\right) e\left(g^{\tau_A}, f^{r_1 r_2 y_A + r_3 r_4 y_A'}\right)}{e\left(f^{\tau_A}, g^{r_1 r_2 y_A + r_3 r_4 y_A'}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(g^{r_1 \sigma_{i,A} - r_1 t_{\{i,1\},A}}, u^{x_i y_{i,A} r_2 \bar{\varepsilon}}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(g^{r_1 \sigma_{i,A} - r_1 t_{\{i,1\},A}}, h^{y_{i,A} r_2 \bar{\varepsilon}}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(g^{r_2 t_{\{i,1\},A}}, u^{x_i y_{i,A} r_1 \bar{\varepsilon}}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(g^{r_2 t_{\{i,1\},A}}, h^{y_{i,A} r_1 \bar{\varepsilon}}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(u^{-x_{\rho(i)}\sigma_{i,A}}, g^{r_1 r_2 y_{i,A} \bar{\varepsilon} + r_3 r_4 y_{i,A}' \bar{\varepsilon}}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(h^{-\sigma_{i,A}}, g^{r_1 r_2 y_{i,A} \bar{\varepsilon} + r_3 r_4 y_{i,A}' \bar{\varepsilon}}\right)} \\
&\quad \times \frac{1}{\prod_{i \in I} e\left(g^{r_3 \sigma_{i,A} - r_3 t_{\{i,2\},A}}, u^{x_i y'_{i,A} r_4 \bar{\varepsilon}}\right)}
\end{aligned}$$

$$\times \frac{1}{\prod_{i\in I} e\left(g^{r_3\sigma_{i,A}-r_3 t_{\{i,2\},A}}, h^{y'_{i,A} r_4 \bar{\varepsilon}}\right)}$$
$$\times \frac{1}{\prod_{i\in I} e\left(g^{r_4 t_{\{i,2\},A}}, u^{x_i y'_{i,A} r_3 \bar{\varepsilon}}\right) e\left(g^{r_4 t_{\{i,2\},A}}, h^{y'_{i,A} r_3 \bar{\varepsilon}}\right)}$$
$$= e(g,g)^{\alpha' \tau_A}$$
$$\Phi_A = \tilde{C}_A / H_3\left(\Phi'_A\right)$$
$$= \tilde{C}_A / H_3\left(e(g,g)^{\alpha' \tau_A}\right)$$
$$= H_1\left(M_A\right)^{\tau_A}$$

Similarly

$$\Phi'_B = e(g,g)^{\alpha' \tau_B}$$
$$\Phi_B = \tilde{C}_B / H_3\left(e(g,g)^{\alpha' \tau_B}\right) = H_1\left(M_B\right)^{\tau_B}$$

If

$$e\left(\Phi_A, D'_B\right) = e\left(H_1\left(M_A\right)^{\tau_A}, g^{\tau_B}\right) = e(H_1\left(M_A\right), g)^{\tau_A \tau_B}$$
$$e\left(\Phi_B, D'_A\right) = e\left(H_1\left(M_B\right)^{\tau_B}, g^{\tau_A}\right) = e(H_1\left(M_B\right), g)^{\tau_B \tau_A}$$

There is

$$e\left(\Phi_A, D'_B\right) = e\left(\Phi_B, D'_A\right)$$

Thus $H_1\left(M_A\right) = H_1\left(M_B\right)$, it shows that CT''_A and CT''_B contain the same plaintext.

VI. SECURITY PROOF

A. CHOSEN-PLAINTEXT SECURE GAME

To prove the security of our scheme, suppose that there is a PPT adversary $\mathcal{A}$ and a challenge matrix that satisfies the constraints, $\mathcal{A}$ breaks the proposed scheme with a non-negligible advantage on selective conditions. Based on such an attacker, we construct a PPT simulator $\mathcal{B}$ that solves the decisional $q-1$ assumption with a non-negligible advantage. For the proof of selective security, $\mathcal{A}$ needs to submit a challenge access policy before the game starts, and then through the indistinguishability process of the two ciphertexts to achieve the security proof of the scheme. Chosen-plaintext security means that $\mathcal{A}$ can select the plaintext and obtain the corresponding ciphertext.

Theorem 1: If the decisional $q-1$ assumption holds, all PPT adversaries have a challenge matrix of size $l \times n$, where $l, n \leq q$, and they break the proposed scheme with a negligible advantage on selective conditions.

Proof: The simulator performs the following chosen-plaintext secure game with adversary, we use the secure game to testify the above theorem. Figure 4 shows the interaction between the simulator and the adversary in the secure game.

Initialization. $\mathcal{A}$ commits a challenge access policy $\left(\mathbf{M}^*, \rho^*, \left\{x_{\rho^*(i)}\right\}\right)$ to $\mathcal{B}$, where $\mathbf{M}^*$ is a shared matrix of size $l \times n\,(l, n \leq q)$, $\rho^* : [l] \to \mathbb{Z}_p$, $[l]$ denotes $\{1, 2, \cdots, l\}$.

Setup. $\mathcal{B}$ randomly chooses $r_1, r_2, r_3, r_4, \tilde{\alpha}, \tilde{\alpha}', \tilde{u}, \tilde{h}, \tilde{d} \in \mathbb{Z}_p$, computes $g_1 = g^{r_1}, g_2 = g^{r_2}, g_3 = g^{r_3}, g_4 = g^{r_4}$, then selects three one-way hash functions: $H_1 : \{0,1\}^* \to \mathbb{G}_1$, $H_2 : \mathbb{G}_2 \to \mathbb{G}_1 \times \mathbb{Z}_p$, $H_3 : \mathbb{G}_2 \to \mathbb{G}_1$.

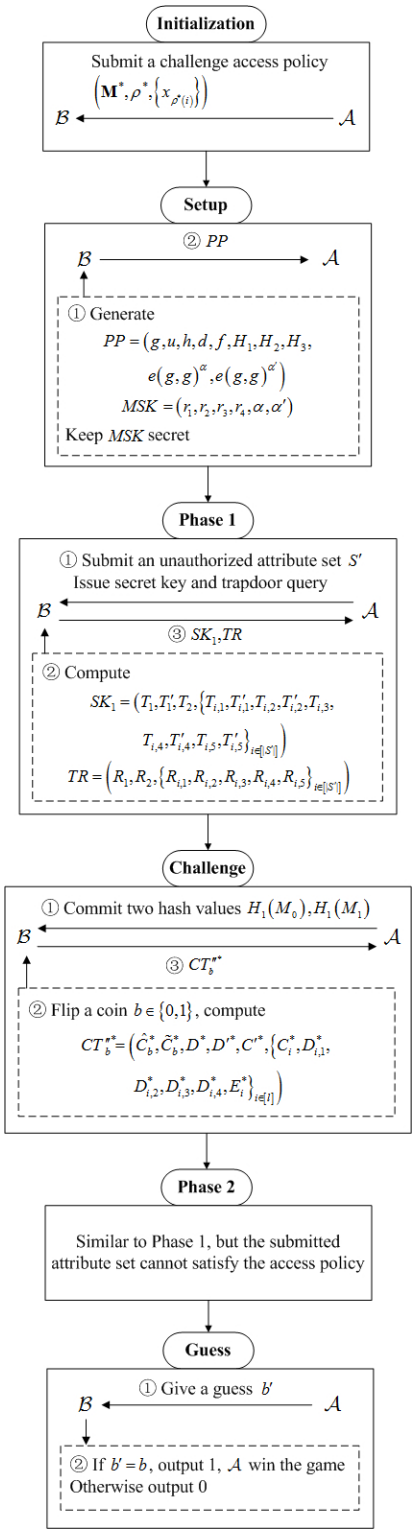

FIGURE 4. The secure game between simulator and adversary.

Additionally, $\mathcal{B}$ implicitly sets $\alpha = a^{q+1} + \tilde{\alpha}, \alpha' = a^{q+1} + \tilde{\alpha}'$, $f = g^a$. It computes

$$g = g, u = g^{\tilde{u}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{M^*_{j,j'}},$$

$$h = g^{\tilde{h}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{-x_{\rho^*(j)}M^*_{j,j'}},$$

$$d = g^{\tilde{d}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j}\right)^{M^*_{j,j'}}, f = g^a,$$

$$e(g,g)^{\alpha} = e\left(g^a, g^{a^q}\right) \cdot e(g,g)^{\tilde{\alpha}},$$

$$e(g,g)^{\alpha'} = e\left(g^a, g^{a^q}\right) \cdot e(g,g)^{\tilde{\alpha}'}$$

The public parameter and master secret key are as follows:

$$PP = (g, u, h, d, f, H_1, H_2, H_3, e(g,g)^{\alpha}, e(g,g)^{\alpha'})$$

$$MSK = (r_1, r_2, r_3, r_4, \alpha, \alpha')$$

$\mathcal{B}$ transmits PP to $\mathcal{A}$.

Phase 1. $\mathcal{A}$ sends an unauthorized attribute set to $\mathcal{B}$, $\mathcal{B}$ runs $KeyGen_{out}$, $KeyGen$ and $TrapGen$ algorithm to generate secret key and trapdoor. For convenience, let $\mathcal{A}$ submit the attribute set as $S' = \{x_1, \cdots, x_{|S'|}\}$.

Since S' is an unauthorized attribute set that does not satisfy $(\mathbf{M}^*, \rho^*, \{x_{\rho^*(i)}\})$, there is a vector $\boldsymbol{\omega} = \{\omega_1, \cdots, \omega_n\}^T \in \mathbb{Z}_p^n$, such that $\omega_1 = -1$, and for all $i \in I = \{i | i \in [l] \cap x_{\rho^*(i)} \in S'\}$, there is $(\mathbf{M}_i^*, \boldsymbol{\omega}) = 0$. $\mathcal{B}$ arbitrarily chooses $\tilde{y}, \tilde{y}' \in \mathbb{Z}_p$, implicitly sets

$$\begin{aligned} y &= \tilde{y} + \omega_1 a^q + \omega_2 a^{q-1} + \cdots + \omega_n a^{q+1-n} \\ &= \tilde{y} + \sum_{i\in[n]} \omega_i a^{q+1-i} \\ y' &= \tilde{y}' + \omega_1 a^q + \omega_2 a^{q-1} + \cdots + \omega_n a^{q+1-n} \\ &= \tilde{y}' + \sum_{i\in[n]} \omega_i a^{q+1-i} \end{aligned}$$

And computes

$$\begin{aligned} L_5 &= f^y = \left(g^a\right)^{\tilde{y}+\sum_{i\in[n]} \omega_i a^{q+1-i}} \\ L_5' &= f^{y'} = \left(g^a\right)^{\tilde{y}'+\sum_{i\in[n]} \omega_i a^{q+1-i}} \\ L_3 &= g^y = g^{\tilde{y}+\sum_{i\in[n]} \omega_i a^{q+1-i}} \\ L_3' &= g^{y'} = g^{\tilde{y}'+\sum_{i\in[n]} \omega_i a^{q+1-i}} \end{aligned}$$

Then

$$\begin{aligned} T_1 &= g^{\alpha}(L_5)^{r_1 r_2}\left(L_5'\right)^{r_3 r_4} \\ &= g^{a^{q+1}+\tilde{\alpha}}\left(g^a\right)^{r_1 r_2\left(\tilde{y}+\sum_{i\in[n]} \omega_i a^{q+1-i}\right)} \\ &\quad \cdot \left(g^a\right)^{r_3 r_4\left(\tilde{y}'+\sum_{i\in[n]} \omega_i a^{q+1-i}\right)} \\ &= \left(g^{a^{q+1}} g^{\tilde{\alpha}}\right)\left(g^{a\tilde{y}} \prod_{i\in[n]} g^{\omega_i a^{q+2-i}}\right)^{r_1 r_2} \\ &\quad \cdot \left(g^{a\tilde{y}'} \prod_{i\in[n]} g^{\omega_i a^{q+2-i}}\right)^{r_3 r_4} \end{aligned}$$

$$\begin{aligned} T_1' &= g^{\alpha'}(L_5)^{r_1 r_2}\left(L_5'\right)^{r_3 r_4} \\ &= \left(g^{a^{q+1}} g^{\tilde{\alpha}'}\right) \\ &\quad \cdot \left(g^{a\tilde{y}} \prod_{i\in[n]} g^{\omega_i a^{q+2-i}}\right)^{r_1 r_2}\left(g^{a\tilde{y}'} \prod_{i\in[n]} g^{\omega_i a^{q+2-i}}\right)^{r_3 r_4} \\ T_2 &= (L_3)^{r_1 r_2}\left(L_3'\right)^{r_3 r_4} \\ &= \left(g^{\tilde{y}} \prod_{i\in[n]} g^{\omega_i a^{q+1-i}}\right)^{r_1 r_2}\left(g^{\tilde{y}'} \prod_{i\in[n]} g^{\omega_i a^{q+1-i}}\right)^{r_3 r_4} \end{aligned}$$

Besides, for all $i \in [|S'|]$, $\mathcal{B}$ must calculate d^{-y}, $d^{-y'}$, $(u^{x_i}h)^{y_i}$, $(u^{x_i}h)^{y_i'}$ separately before calculating $T_{i,1}$, $T_{i,1}'$, $T_{i,2}$, $T_{i,2}'$, $T_{i,3}$, $T_{i,4}$, $T_{i,4}'$, $T_{i,5}$, $T_{i,5}'$. $\mathcal{B}$ selects $\tilde{y}_i, \tilde{y}_i' \in \mathbb{Z}_p$ at random, and implicitly sets

$$\begin{aligned} y_i &= \tilde{y}_i + \left(\tilde{y} \cdot \sum_{\substack{i'\in[l] \\ x_{\rho^*(i')}\notin S'}} \frac{b_{i'}}{x_i - x_{\rho^*(i')}} \right. \\ &\quad \left. + \sum_{\substack{(j,i')\in[n,l] \\ x_{\rho^*(i')}\notin S'}} \frac{\omega_j b_{i'} a^{q+1-j}}{x_i - x_{\rho^*(i')}}\right) \\ y_i' &= \tilde{y}_i' + \left(\tilde{y}' \cdot \sum_{\substack{i'\in[l] \\ x_{\rho^*(i')}\notin S'}} \frac{b_{i'}}{x_i - x_{\rho^*(i')}} \right. \\ &\quad \left. + \sum_{\substack{(j,i')\in[n,l] \\ x_{\rho^*(i')}\notin S'}} \frac{\omega_j b_{i'} a^{q+1-j}}{x_i - x_{\rho^*(i')}}\right) \end{aligned}$$

Then it computes

$$\begin{aligned} d^{-y} &= d^{-\left(\tilde{y}+\sum_{i\in[n]} \omega_i a^{q+1-i}\right)} \\ &= d^{-\tilde{y}} \cdot d^{-\sum_{i\in[n]} \omega_i a^{q+1-i}} \\ &= d^{-\tilde{y}} \cdot \prod_{i\in[n]} \left(g^{\tilde{d}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j}\right)^{M^*_{j,j'}}\right)^{-\omega_i a^{q+1-i}} \\ &= d^{-\tilde{y}} \cdot \prod_{i\in[n]} \left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i} \\ &\quad \cdot \prod_{\substack{(i,j,j')\in[n,l,n] \\ j'\neq i}} \left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'}} \\ &\quad \cdot \prod_{(i,j)\in[n,l]} \left(g^{a^{q+1}/b_j}\right)^{-\omega_i M^*_{j,i}} \\ &= g^{-\tilde{d}\tilde{y}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j}\right)^{-\tilde{y}M^*_{j,j'}} \end{aligned}$$

$$\cdot \prod_{i\in[n]} \left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i} \cdot \prod_{\substack{(i,j,j')\in[n,l,n]\\ j'\neq i}} \left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'}} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} \left(g^{a^{q+1}/b_j}\right)^{-\left(\boldsymbol{\omega},\mathbf{M}^*_j\right)}$$

$$d^{-y'} = d^{-\left(\tilde{y}'+\sum_{i\in[n]}\omega_i a^{q+1-i}\right)} = g^{-\tilde{d}\tilde{y}'} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j}\right)^{-\tilde{y}' M^*_{j,j'}} \cdot \prod_{i\in[n]} \left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i} \cdot \prod_{\substack{(i,j,j')\in[n,l,n]\\ j'\neq i}} \left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'}} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} \left(g^{a^{q+1}/b_j}\right)^{-\left(\boldsymbol{\omega},\mathbf{M}^*_j\right)}$$

To calculate $L'_{i,1}, L'_{i,2}$, $\mathcal{B}$ first computes

$$L_{i,4} = g^{y_i} = g^{\tilde{y}_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\tilde{y}}{x_i - x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\omega_{k'}}{x_i - x_{\rho^*(i')}}}$$

$$L'_{i,4} = g^{y'_i} = g^{\tilde{y}'_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\tilde{y}'}{x_i - x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\omega_{k'}}{x_i - x_{\rho^*(i')}}}$$

Then

$$T_{i,3} = \left(L_{i,4}\right)^{r_1 r_2}\left(L'_{i,4}\right)^{r_3 r_4} = \left(g^{\tilde{y}_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\tilde{y}}{x_i - x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\omega_{k'}}{x_i - x_{\rho^*(i')}}}\right)^{r_1 r_2} \cdot \left(g^{\tilde{y}'_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\tilde{y}'}{x_i - x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\omega_{k'}}{x_i - x_{\rho^*(i')}}}\right)^{r_3 r_4}$$

Afterwards, $\mathcal{B}$ computes

$$\begin{aligned}
L'_{i,1} &= \left(u^{x_i}h\right)^{y_i} = u^{x_i y_i}h^{y_i} \\
&= \left(g^{\tilde{u}}\right)^{x_i y_i} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{M^*_{j,j'}x_i y_i} \cdot \left(g^{\tilde{h}}\right)^{y_i} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{-x_{\rho^*(j)}M^*_{j,j'}y_i} \\
&= \left(g^{y_i}\right)^{\tilde{u}x_i} \cdot \prod_{(j,j')\in[l,n]} \left(g^{y_i}\right)^{x_i M^*_{j,j'}a^{j'}/b_j^2} \cdot \left(g^{y_i}\right)^{\tilde{h}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{y_i}\right)^{-x_{\rho^*(j)}M^*_{j,j'}a^{j'}/b_j^2} \\
&= \left(g^{\tilde{y}_i}\right)^{\tilde{u}x_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\tilde{y}\tilde{u}x_i}{x_i - x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\omega_{k'}\tilde{u}x_i}{x_i - x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{\tilde{y}_i}\right)^{x_i M^*_{j,j'}a^{j'}/b_j^2} \\
&\quad \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}x_i M^*_{j,j'}}{x_i - x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{\frac{\omega_{k'}x_i M^*_{j,j'}}{x_i - x_{\rho^*(i')}}} \\
&\quad \cdot \left(g^{\tilde{y}_i}\right)^{\tilde{h}} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\tilde{h}\tilde{r}}{x_i - x_{\rho^*(i')}}}
\end{aligned}$$

$$\cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\tilde{h}\omega_{k'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{\tilde{y}_i}\right)^{-x_{\rho^*(j)}M^*_{j,j'}a^{j'}/b_j^2} \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{-\tilde{y}x_{\rho^*(j)}M^*_{j,j'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{\frac{-\omega_{k'}x_{\rho^*(j)}M^*_{j,j'}}{x_i-x_{\rho^*(i')}}}$$

$$= g^{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{\tilde{y}_i\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}} \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{\frac{\omega_{k'}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} g^{\frac{\left(\boldsymbol{\omega},\mathbf{M}^*_j\right)a^{q+1}}{b_j}}$$

Then

$$T'_{i,1} = \left(L'_{i,1}\right)^{r_2} = g^{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i r_2} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}r_2}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'}r_2}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{\tilde{y}_i\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_2} \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_2}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{\Upsilon} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} g^{\frac{\left(\boldsymbol{\omega},\mathbf{M}^*_j\right)a^{q+1}r_2}{b_j}}$$

where

$$\Upsilon = \frac{\omega_{k'}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_2}{x_i-x_{\rho^*(i')}}$$

$$T'_{i,2} = \left(L'_{i,1}\right)^{r_1} = g^{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i r_1} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}r_1}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'}r_1}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{\tilde{y}_i\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_1} \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_1}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{\Omega} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} g^{\frac{\left(\boldsymbol{\omega},\mathbf{M}^*_j\right)a^{q+1}r_1}{b_j}}$$

where

$$\Omega = \frac{\omega_{k'}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_1}{x_i-x_{\rho^*(i')}}$$

It computes

$$L'_{i,2} = \left(u^{x_i}h\right)^{y'_i} = g^{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}'_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}'}{x_i-x_{\rho^*(i')}}}$$

$$\cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{(\tilde{u}x_i+\tilde{h})\omega_{k'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{\tilde{y}_i'(x_i-x_{\rho^*(j)})M^*_{j,j'}} \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}'(x_i-x_{\rho^*(j)})M^*_{j,j'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{\Gamma} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} g^{\frac{(\boldsymbol{\omega},\mathbf{M}^*_j)a^{q+1}}{b_j}}$$

where

$$\Gamma = \frac{\omega_{k'}\left(x_i - x_{\rho^*(j)}\right)M^*_{j,j'}}{x_i - x_{\rho^*(i')}}$$

Then

$$T'_{i,4} = \left(L'_{i,2}\right)^{r_4} = g^{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i'r_4} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{(\tilde{u}x_i+\tilde{h})\tilde{y}'r_4}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{(\tilde{u}x_i+\tilde{h})\omega_{k'}r_4}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{\tilde{y}_i'(x_i-x_{\rho^*(j)})M^*_{j,j'}r_4} \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}'(x_i-x_{\rho^*(j)})M^*_{j,j'}r_4}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{A} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} g^{\frac{(\boldsymbol{\omega},\mathbf{M}^*_j)a^{q+1}r_4}{b_j}}$$

where

$$A = \frac{\omega_{k'}\left(x_i - x_{\rho^*(j)}\right)M^*_{j,j'}r_4}{x_i - x_{\rho^*(i')}}$$

$$T'_{i,5} = \left(L'_{i,2}\right)^{r_3} = g^{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i'r_3} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{(\tilde{u}x_i+\tilde{h})\tilde{y}'r_3}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{(\tilde{u}x_i+\tilde{h})\omega_{k'}r_3}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j^2}\right)^{\tilde{y}_i'(x_i-x_{\rho^*(j)})M^*_{j,j'}r_3} \cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}'(x_i-x_{\rho^*(j)})M^*_{j,j'}r_3}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{B} \cdot \prod_{\substack{j\in[l]\\ x_{\rho^*(j)}\notin S'}} g^{\frac{(\boldsymbol{\omega},\mathbf{M}^*_j)a^{q+1}r_3}{b_j}}$$

where

$$B = \frac{\omega_{k'}\left(x_i - x_{\rho^*(j)}\right)M^*_{j,j'}r_3}{x_i - x_{\rho^*(i')}}$$

At last, $\mathcal{B}$ calculates

$$L_{i,1} = \left(u^{x_i}h\right)^{y_i}d^{-y} = g^{\left[\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i-\tilde{d}\tilde{y}\right]} \cdot \prod_{i\in[n]} \left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i} \cdot \prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{(\tilde{u}x_i+\tilde{h})\tilde{y}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{(\tilde{u}x_i+\tilde{h})\omega_{k'}}{x_i-x_{\rho^*(i')}}} \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}M^*_{j,j'}}\right)^{\frac{\tilde{y}_i\left(x_i-x_{\rho^*(j)}\right)}{b_j^2}-\frac{\tilde{y}}{b_j}} \cdot \prod_{(i,j,j')\in[n,l,n]} \left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'}}$$

$$\cdot \prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}}{x_i-x_{\rho^*(i')}}}$$

$$\cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{C}$$

where

$$C=\frac{\omega_{k'}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}}{x_i-x_{\rho^*(i')}}$$

Then

$$\begin{aligned}T_{i,1}&=\left(L_{i,1}\right)^{r_2}\\&=g^{\left[\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i-\tilde{d}\tilde{y}\right]r_2}\cdot\prod_{i\in[n]}\left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i r_2}\\&\cdot\prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}r_2}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'}r_2}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{(j,j')\in[l,n]}\left(g^{a^{j'}M^*_{j,j'}r_2}\right)^{\frac{\tilde{y}_i\left(x_i-x_{\rho^*(j)}\right)}{b_j^2}-\frac{\tilde{y}}{b_j}}\\&\cdot\prod_{(i,j,j')\in[n,l,n]}\left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'}r_2}\\&\cdot\prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_2}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}}\left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{D}\end{aligned}$$

where

$$D=\frac{\omega_{k'}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_2}{x_i-x_{\rho^*(i')}}$$

$$\begin{aligned}T_{i,2}&=\left(L_{i,1}\right)^{r_1}\\&=g^{\left[\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}_i-\tilde{d}\tilde{y}\right]r_1}\cdot\prod_{i\in[n]}\left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i r_1}\\&\cdot\prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}r_1}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'}r_1}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{(j,j')\in[l,n]}\left(g^{a^{j'}M^*_{j,j'}r_1}\right)^{\frac{\tilde{y}_i\left(x_i-x_{\rho^*(j)}\right)}{b_j^2}-\frac{\tilde{y}}{b_j}}\\&\cdot\prod_{(i,j,j')\in[n,l,n]}\left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'}r_1}\\&\cdot\prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_1}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}}\left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{E}\end{aligned}$$

where

$$E=\frac{\omega_{k'}\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}r_1}{x_i-x_{\rho^*(i')}}$$

Next

$$\begin{aligned}L_{i,2}&=\left(u^{x_i}h\right)^{y'_i}d^{-y'}\\&=g^{\left[\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}'_i-\tilde{d}\tilde{y}'\right]}\cdot\prod_{i\in[n]}\left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i}\\&\cdot\prod_{\substack{i'\in[l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}'}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{\substack{(k',i')\in[n,l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'}}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{(j,j')\in[l,n]}\left(g^{a^{j'}M^*_{j,j'}}\right)^{\frac{\tilde{y}'_i\left(x_i-x_{\rho^*(j)}\right)}{b_j^2}-\frac{\tilde{y}'}{b_j}}\\&\cdot\prod_{(i,j,j')\in[n,l,n]}\left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'}}\\&\cdot\prod_{\substack{(j,j',i')\in[l,n,l]\\ x_{\rho^*(i')}\notin S'}}\left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}'\left(x_i-x_{\rho^*(j)}\right)M^*_{j,j'}}{x_i-x_{\rho^*(i')}}}\\&\cdot\prod_{\substack{(j,j',k',i')\in[l,n,n,l]\\ x_{\rho^*(i')}\notin S'\\ (j'\neq k',j\neq i')}}\left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{F}\end{aligned}$$

where

$$F = \frac{\omega_{k'}\left(x_i - x_{\rho^*(j)}\right) M^*_{j,j'}}{x_i - x_{\rho^*(i')}}$$

Then

$$\begin{aligned}
T_{i,4} &= \left(L_{i,2}\right)^{r_4} \\
&= g^{\left[\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}'_i - \tilde{d}\tilde{y}'\right]r_4} \cdot \prod_{i\in[n]} \left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i r_4} \\
&\quad \cdot \prod_{\substack{i'\in[l] \\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}' r_4}{x_i - x_{\rho^*(i')}}} \\
&\quad \cdot \prod_{\substack{(k',i')\in[n,l] \\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'} r_4}{x_i - x_{\rho^*(i')}}} \\
&\quad \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}M^*_{j,j'}r_4}\right)^{\frac{\tilde{y}'_i\left(x_i - x_{\rho^*(j)}\right)}{b_j^2} - \frac{\tilde{y}'}{b_j}} \\
&\quad \cdot \prod_{(i,j,j')\in[n,l,n]} \left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'} r_4} \\
&\quad \cdot \prod_{\substack{(j,j',i')\in[l,n,l] \\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}'\left(x_i - x_{\rho^*(j)}\right)M^*_{j,j'}r_4}{x_i - x_{\rho^*(i')}}} \\
&\quad \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l] \\ x_{\rho^*(i')}\notin S' \\ (j'\neq k', j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{G}
\end{aligned}$$

where

$$G = \frac{\omega_{k'}\left(x_i - x_{\rho^*(j)}\right) M^*_{j,j'} r_4}{x_i - x_{\rho^*(i')}}$$

$$\begin{aligned}
T_{i,5} &= \left(L_{i,2}\right)^{r_3} \\
&= g^{\left[\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}'_i - \tilde{d}\tilde{y}'\right]r_3} \cdot \prod_{i\in[n]} \left(g^{a^{q+1-i}}\right)^{-\tilde{d}\omega_i r_3} \\
&\quad \cdot \prod_{\substack{i'\in[l] \\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\tilde{y}' r_3}{x_i - x_{\rho^*(i')}}} \\
&\quad \cdot \prod_{\substack{(k',i')\in[n,l] \\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{q+1-k'}}\right)^{\frac{\left(\tilde{u}x_i+\tilde{h}\right)\omega_{k'} r_3}{x_i - x_{\rho^*(i')}}} \\
&\quad \cdot \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}M^*_{j,j'}r_3}\right)^{\frac{\tilde{y}'_i\left(x_i - x_{\rho^*(j)}\right)}{b_j^2} - \frac{\tilde{y}'}{b_j}} \\
&\quad \cdot \prod_{(i,j,j')\in[n,l,n]} \left(g^{a^{q+1+j'-i}/b_j}\right)^{-\omega_i M^*_{j,j'} r_3} \\
&\quad \cdot \prod_{\substack{(j,j',i')\in[l,n,l] \\ x_{\rho^*(i')}\notin S'}} \left(g^{b_{i'}a^{j'}/b_j^2}\right)^{\frac{\tilde{y}'\left(x_i - x_{\rho^*(j)}\right)M^*_{j,j'}r_3}{x_i - x_{\rho^*(i')}}} \\
&\quad \cdot \prod_{\substack{(j,j',k',i')\in[l,n,n,l] \\ x_{\rho^*(i')}\notin S' \\ (j'\neq k', j\neq i')}} \left(g^{b_{i'}a^{q+1+j'-k'}/b_j^2}\right)^{H}
\end{aligned}$$

where

$$H = \frac{\omega_{k'}\left(x_i - x_{\rho^*(j)}\right) M^*_{j,j'} r_3}{x_i - x_{\rho^*(i')}}$$

Consequently, $\mathcal{B}$ outputs secret key

$$SK_1 = \left(T_1, T'_1, T_2, \left\{T_{i,1}, T'_{i,1}, T_{i,2}, T'_{i,2}, T_{i,3}, T_{i,4}, T'_{i,4}, T_{i,5}, T'_{i,5}\right\}_{i\in[|S'|]}\right)$$

At the same time, $\mathcal{B}$ randomly chooses $\eta \in \mathbb{Z}_p$, generates the trapdoor of equality test

$$TR = \left(R_1, R_2, \left\{R_{i,1}, R_{i,2}, R_{i,3}, R_{i,4}, R_{i,5}\right\}_{i\in[|S'|]}\right)$$

where

$$\begin{aligned}
&R_1 = T'_1, \quad R_2 = T_2, \; R_{i,1} = T'^{\eta}_{i,1}, R_{i,2} = T'^{\eta}_{i,2}, \\
&R_{i,3} = T^{\eta}_{i,3}, \quad R_{i,4} = T'^{\eta}_{i,4}, \; R_{i,5} = T'^{\eta}_{i,5}
\end{aligned}$$

$\mathcal{B}$ transmits SK_1 and TR to $\mathcal{A}$.

Challenge. $\mathcal{A}$ commits two hash values $H_1(M_0)$, $H_1(M_1)$ to $\mathcal{B}$, it runs $Encrypt_{out}$ and $Encrypt-files$ algorithm to gain the ciphertext:

$\mathcal{B}$ flips a coin to choose $b \in \{0, 1\}$, computes

$$\hat{C}^*_b = \left(H_1(M_b)||\tau\right) \oplus H_2\left(Z \cdot e\left(g, g^s\right)^{\tilde{\alpha}}\right),$$

$$\tilde{C}^*_b = H_1(M_b)^{\tau} \cdot H_3\left(Z \cdot e\left(g, g^{\tau}\right)^{\tilde{\alpha}'}\right),$$

$$D^* = g^s, D'^* = g^{\tau}, C'^* = g^{a\tau}$$

It implicitly sets $\mathbf{v} = \left(s, sa + \gamma_2, sa^2 + \gamma_3, \cdots, sa^{n-1} + \gamma_n\right)$, where $\gamma_2, \cdots, \gamma_n \in \mathbb{Z}_p$. Since $\lambda_i = \mathbf{v} \cdot \mathbf{M}_i$, then

$$\begin{aligned}
\lambda_i &= \sum_{j\in[n]} M^*_{i,j} sa^{j-1} + \sum_{j=2}^{n} M^*_{i,j}\gamma_j \\
&= \sum_{j\in[n]} M^*_{i,j} sa^{j-1} + \tilde{\lambda}_i \qquad (i \in [l])
\end{aligned}$$

For each row of matrix, $\mathcal{B}$ implicitly sets $\sigma_i = -sb_i$, and calculates

$$\begin{aligned}
C'^*_i &= d^{\sigma_i} \\
&= \left(g^{\tilde{d}} \prod_{(j,j')\in[l,n]} \left(g^{a^{j'}/b_j}\right)^{M^*_{j,j'}}\right)^{-sb_i}
\end{aligned}$$

$$= g^{-\tilde{d}sb_i} \cdot \prod_{j' \in [n]} g^{-M^*_{j,j'} a^{j'} sb_j / b_j} \cdot \prod_{\substack{(j,j') \in [l,n] \\ i \neq j}} g^{-M^*_{j,j'} a^{j'} sb_i / b_j}$$

Then

$$C_i^* = f^{\lambda_i} \left(C_i'^* \right) = g^{a\tilde{\lambda}_i - \tilde{d}sb_i} \cdot \prod_{\substack{(j,j') \in [l,n] \\ i \neq j}} g^{-M^*_{j,j'} a^{j'} sb_i / b_j}$$

Next, it computes

$$\begin{aligned}
D^*_{i,1} &= D'^*_{i,1} = g_1^{\sigma_i - t_{i,1}} = g^{-r_1 sb_i} \cdot g^{-r_1 t_{i,1}}, \\
D^*_{i,2} &= D'^*_{i,2} = g_3^{\sigma_i - t_{i,2}} = g^{-r_3 sb_i} \cdot g^{-r_3 t_{i,2}}, \\
D^*_{i,3} &= D'^*_{i,3} = g_2^{t_{i,1}} = g^{r_2 t_{i,1}}, \\
D^*_{i,4} &= D'^*_{i,4} = g_4^{t_{i,2}} = g^{r_4 t_{i,2}} \\
E_i^* &= E_i'^* \\
&= \left(u^{x_{\rho^*(i)}} h \right)^{-\sigma_i} \\
&= g^{-\tilde{u} x_{\rho^*(i)} sb_i} \cdot \prod_{(j,j') \in [l,n]} \left(g^{a^{j'} / b_j^2} \right)^{-x_{\rho^*(i)} sb_i M^*_{j,j'}} \\
&\quad \cdot g^{-\tilde{h} sb_i} \cdot \prod_{(j,j') \in [l,n]} \left(g^{a^{j'} / b_j^2} \right)^{x_{\rho^*(j)} sb_i M^*_{j,j'}} \\
&= \left(g^{sb_i} \right)^{-\left(\tilde{u} x_{\rho^*(i)} + \tilde{h} \right)} \\
&\quad \cdot \prod_{\substack{(j,j') \in [l,n] \\ i \neq j}} \left(g^{sb_i a^{j'} / b_j^2} \right)^{\left(x_{\rho^*(j)} - x_{\rho^*(i)} \right) M^*_{j,j'}}
\end{aligned}$$

where $t_{i,1}, t_{i,2} \in \mathbb{Z}_p$. $\mathcal{B}$ outputs ciphertext

$$CT_b''^* = \left(\hat{C}_b^*, \tilde{C}_b^*, D^*, D'^*, C'^*, \left\{ C_i^*, D^*_{i,1}, D^*_{i,2}, D^*_{i,3}, D^*_{i,4}, E_i^* \right\}_{i \in [l]} \right)$$

And sends it to $\mathcal{A}$.

Phase 2. Similar to Phase 1, the restriction is that attribute set cannot satisfy access policy.

Guess. $\mathcal{A}$ gives a guess b'. If $b' = b$, $\mathcal{B}$ outputs 1, it indicates that $\mathcal{A}$ gets $Z = e(g,g)^{sa^{q+1}}$ and wins the game. Otherwise $\mathcal{B}$ outputs 0, it indicates that Z is a random element in $\mathbb{G}_2$.

If $Z = e(g,g)^{sa^{q+1}}$, $\mathcal{B}$ successfully simulates the real scheme, because

$$\begin{aligned}
\hat{C}_b^* &= (H_1(M_b) || \tau) \oplus H_2 \left(Z \cdot e\left(g, g^s\right)^{\tilde{\alpha}} \right) \\
&= (H_1(M_b) || \tau) \oplus H_2 \left(e(g,g)^{\alpha s} \right) \\
\tilde{C}_b^* &= H_1(M_b)^{\tau} \cdot H_3 \left(Z \cdot e\left(g, g^{\tau}\right)^{\tilde{\alpha}'} \right) \\
&= H_1(M_b)^{\tau} \cdot H_3 \left(e(g,g)^{\alpha' \tau} \right)
\end{aligned}$$

In other words, if Z is a random element in $\mathbb{G}_2$, $H_1(M_b)$ is completely hidden in the challenge ciphertext. The advantage of $\mathcal{A}$ wins the game can be defined as

$$Adv_{\mathcal{A}} = \left| \Pr\left[b' = b \right] - 1/2 \right|$$

If $\mathcal{A}$ breaks the secure game with a non-negligible advantage, $\mathcal{B}$ can solve the decisional $q-1$ assumption with a non-negligible advantage. Because the decisional $q-1$ assumption is a hard problem, our scheme proves to be chosen-plaintext security.

B. CHOSEN-KEYWORD SECURE GAME

Theorem 2: If an adversary can win the chosen-keyword secure game with a non-negligible advantage ε, then we can construct a simulator to solve DDH assumption with advantage ε.

Proof: Given a tuple (g, g^{z_1}, g^{z_2}, Z) of DDH problem and sent it to a simulator, where $g, Z \in \mathbb{G}_1, z_1, z_2 \in \mathbb{Z}_p$. The simulator's task is to determine whether $Z = g^{z_1 z_2}$ or Z is a random element in $\mathbb{G}_1$. Therefore, simulator $\mathcal{B}$ executes the following chosen-keyword secure game with adversary $\mathcal{A}$.

Setup. $\mathcal{B}$ arbitrarily chooses $f \in \mathbb{G}_1, \beta_1, \beta_2 \in \mathbb{Z}_p$. Let $f = g^{z_1}, g_1 = g^{\beta_1}, g_2 = g^{\beta_2}, H_4 : \{0,1\}^* \to \mathbb{Z}_p$, where H_4 is a one-way hash function. It generates public parameter $PP = (g, f, g_1, g_2)$ and master secret key $MSK = (\beta_1, \beta_2)$, while making PP to public.

Phase 1. $\mathcal{A}$ issues adaptive queries:

Token query: $\mathcal{A}$ gives a query keyword w', $\mathcal{B}$ chooses $\xi \in \mathbb{Z}_p$ at random, computes

$$\begin{aligned}
Z_1 &= g^{\beta_1 \beta_2}, \quad Z_2 = f^{\beta_1 \beta_2} = g^{z_1 \beta_1 \beta_2}, \\
K_1 &= g_2^{H_4(w')\xi}, \quad K_2 = g_1^{H_4(w')\xi}, \\
K_3 &= (Z_1)^{\xi} = g^{\beta_1 \beta_2 \xi}, \quad K_4 = (Z_2)^{\xi} = g^{z_1 \beta_1 \beta_2 \xi}
\end{aligned}$$

Then, it returns $Tok = (K_1, K_2, K_3, K_4)$ to $\mathcal{A}$.

Challenge. $\mathcal{A}$ arbitrarily chooses two keywords w_0, w_1. $\mathcal{B}$ flips a coin to select $b \in \{0,1\}$, and randomly chooses $\mu, \theta, \delta \in \mathbb{Z}_p$. Let $D' = g^{\mu} = g^{z_2}, F = Z g^{-H_4(w_b)\theta}$, it implicitly sets $\mu = z_2$, computes $D_1 = g_1^{\theta - \delta}, D_2 = g_2^{\delta}$. $\mathcal{B}$ transmits keyword index $Index = \left(D', F, D_1, D_2 \right)$ to $\mathcal{A}$.

Phase 2. Similar to Phase 1, but the restriction is that $\mathcal{A}$ cannot query the keyword w_0, w_1 anymore.

Guess. $\mathcal{A}$ outputs a guess keyword $w_{b'}, b' \in \{0,1\}$ according as

$$\begin{aligned}
F' &= f^{\mu} g^{-H_4(w_{b'})\theta} \\
&= \left(g^{z_1} \right)^{\mu} g^{-H_4(w_{b'})\theta} \\
&= g^{z_1 z_2} g^{-H_4(w_{b'})\theta}
\end{aligned}$$

If $b' = b$, it means that the keyword guessed by the adversary is the same as the keyword encrypted by the simulator, there is

$$F' = F$$

where $F = Z g^{-H_4(w_b)\theta}$.

TABLE 4. Performance comparison of the scheme.

Scheme	CP-ABE/KP-ABE	Access Policy	Keyword Search	Outsourced Computing	Equality Test
[2]	CP-ABE	LSSS	No	Outsourced decryption	No
[48]	KP-ABE	Access tree	No	No	Yes
[50]	CP-ABE	AND gate	No	No	Yes
[51]	CP-ABE	LSSS	No	No	No
[55]	CP-ABE	AND gate	Yes	No	No
[56]	CP-ABE	AND gate	No	Outsourced decryption	No
[57]	CP-ABE	LSSS	No	Fully outsourced	No
[58]	No	No	No	No	Yes
Ours	CP-ABE	LSSS	Yes	Fully outsourced	Yes

Then

$$Z = g^{z_1 z_2}$$

Therefore when $b' = b$, $\mathcal{B}$ outputs 1, it shows that $\mathcal{A}$ gets $Z = g^{z_1 z_2}$ and wins the game. Otherwise $\mathcal{B}$ outputs 0, it shows that Z is a random element in $\mathbb{G}_1$.

Since the DDH problem is hard to solve, we can't construct a simulator to solve it with a non-negligible advantage, hence there is no PPT adversary wins the above game with a non-negligible advantage, the proposed scheme proves to be chosen-keyword security.

VII. PERFORMANCE AND EFFICIENCY ANALYSES

In order to analyze the performance of solution, we mainly compare the differences in function and algorithm efficiency between our scheme and other related literatures.

We list the performance comparison (Table 4) between the proposed scheme and literatures [2], [48], [50], [51], [55]–[58]. It can be seen that most of the schemes adopt CP-ABE, and the access policies used are LSSS, access tree and AND gate. Both our scheme and the literature [55] support keyword search. Although literatures [2], [56], [57] all involve outsourced computing, only [57] and our scheme support fully outsourced, [2], [56] only use outsourced decryption. In addition, the proposed scheme and literatures [48], [50], [58] have equality test judgment for ciphertext. In brief, our scheme is more comprehensive in function than other literatures.

On the basis of Pairing Based Cryptography (PBC) library [59], we mainly consider three time complexity algorithms: multiplication, exponential and pairing operation. To be specific, K indicates the number of attributes that satisfy access policy, N indicates the number of attributes owned by data user, l_1 indicates the number of wildcards in access policy, L_1 indicates the maximum number of wildcards (In our scheme, let $K = N$, $l_1 = L_1$). The environment of the hardware runtime is Intel Core i5-3470 CPU @ 3.20GHz, and RAM is 4.00GB. The software runtime environment is JDK 1.7.5, JPBC 2.0.0 and MyEclipse 10. According to the construction principle of the scheme, we choose the seven phases of secret key generation, encryption, token generation, search, decryption, trapdoor generation and equality test to compare the algorithm efficiency of our scheme with the literatures [48], [50], [51], [55].

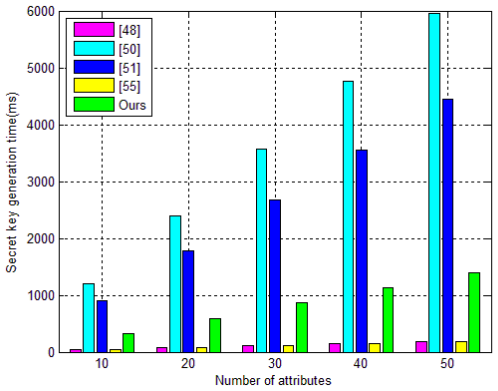

FIGURE 5. Secret key generation time.

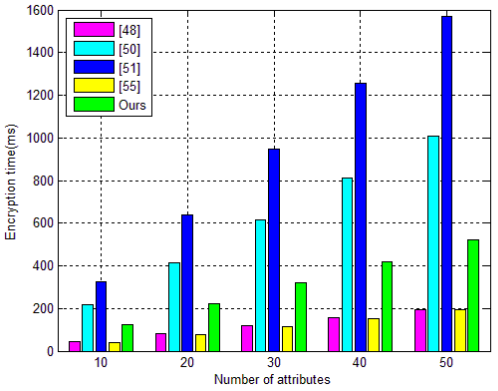

FIGURE 6. Encryption time.

First, Table 5 contains four algorithms. In the secret key generation phase, the algorithm of our scheme takes more time than the literatures [48], [50], less than the literatures [51], [55], where the time-consuming of the algorithm is mainly the exponential operation of elements in group $\mathbb{G}_1$. The encryption phase is the same as the secret key generation phase, since the encryption algorithm of proposed scheme consists of two parts and involves more exponential operations, which will take longer time, thus

TABLE 5. Efficiency comparison of the scheme.

Scheme	Secret Key	Encryption	Token	Search
[48]	$2NE_1$	$(2K+2)\,E_1+2E_2$	No	No
[50]	$(12l_1+10)\,E_1+12l_1M_1$	$(7+2L_1)\,E_1+4E_2+2L_1M_1+2M_2$	No	No
[51]	$(13N+3)\,E_1+(8N+1)\,M_1$	$(8K+3)\,E_1+E_2+(2K+1)\,M_1$	No	No
[55]	$(2N+1)\,E_1+E_2$	$(2K+1)\,E_1+E_2$	$(2N+1)\,E_1$	$E_2+2NM_2+(2N+1)\,P$
Ours	$(10+10N)\,E_1+(5+N)\,M_1$	$(K+7)\,E_1+2E_2+(K+1)\,M_1$	$4E_1$	$3M_2+4P$

E_1:An exponential operation of elements in group $\mathbb{G}_1$
E_2: An exponential operation of elements in group $\mathbb{G}_2$
M_1: A multiplication operation on group $\mathbb{G}_1$
M_2: A multiplication operation on group $\mathbb{G}_2$
P: A pairing operation

TABLE 6. Efficiency comparison of the scheme.

Scheme	Decryption	Trapdoor	Equality Test
[48]	$2E_1+KE_2+(K-1)\,M_2+KP$	NE_1	$2KE_2+(2K-2)\,M_2+(2K+2)\,P$
[50]	$(8L_1+2)\,E_1+4E_2+(8L_1-7)\,M_1+10M_2+12P$	0	$8L_1E_1+4E_2+(8L_1-6)\,M_1+10M_2+14P$
[51]	$2E_1+KE_2+M_1+(5K+1)\,M_2+(6K+1)\,P$	No	No
[55]	No	No	No
Ours	$2E_1+M_1+2M_2+2P$	$5NE_1$	$(10K+2)\,M_2+(10K+6)\,P$

E_1: An exponential operation of elements in group $\mathbb{G}_1$
E_2: An exponential operation of elements in group $\mathbb{G}_2$
M_1: A multiplication operation on group $\mathbb{G}_1$
M_2: A multiplication operation on group $\mathbb{G}_2$
P: A pairing operation

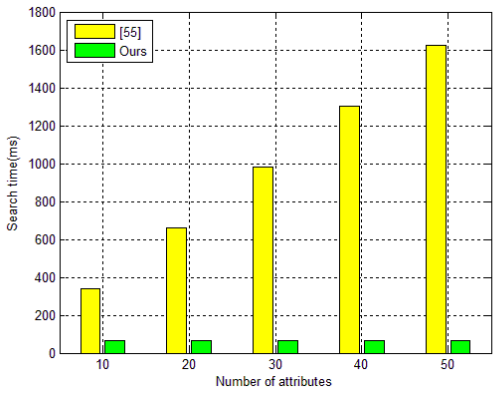

FIGURE 7. Search time.

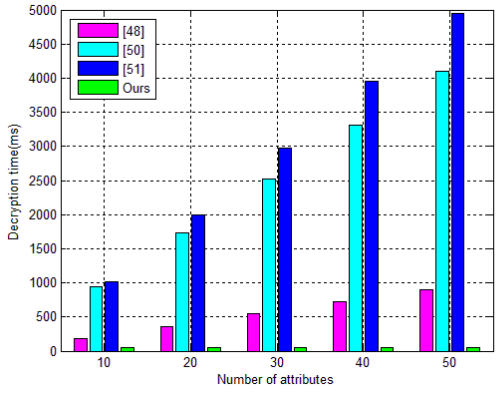

FIGURE 8. Decryption time.

our algorithm efficiency is lower than [48], [50], higher than [51], [55]. In the token generation and search phase, because literatures [48], [50], [51] do not support the keyword search function, we do not calculate the algorithm time of literatures [48], [50], [51]. Compared with the literature [55], the algorithm time of our scheme is constant in these two phases, and the algorithm efficiency is higher than [55].

Then, Table 6 contains three algorithms. In the decryption phase, the literature [55] is an ABSE scheme, which does not contain the decryption operation. Compared with literatures [48], [50], [51], our scheme has the least time-consuming and the highest efficiency, the algorithm time is constant and does not vary with the number of attributes. In the trapdoor generation and equality test phase, literatures [51], [55] do not have this function, and our scheme is mainly compared with the literatures [48], [50]. It can be seen that the algorithm time of our scheme in these two phases is higher than [48], [50], but it is better than [48], [50] in other aspects.

In order to make the efficiency comparison more intuitive, we draw the efficiency comparison figure 5-9 for the five phases of secret key generation, encryption, search, decryption and equality test. On the whole, the efficiency of our scheme is higher than other literatures.

Due to the large amount of computing of pairing operation, the general ABE schemes cannot be directly applied to resource-constrained IoT devices. Therefore, For the sake of solving this problem under the premise of ABE security,

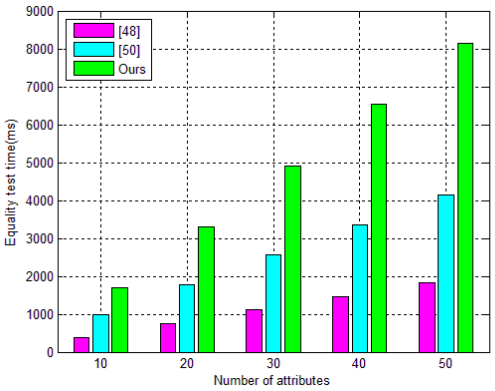

FIGURE 9. Equality test time.

we adopt outsourcing technology. In CP-ABE schemes, the size of ciphertext largely depends on the number of attributes in access policy. It can be seen from Figure 8 that the decryption operation time of our scheme is constant, and it does not change with the number of attributes. Besides outsourced decryption, the outsourcing technology in the secret key generation and encryption phase also greatly reduces the local storage and computational burden, so that the proposed scheme is suitable for IoT devices.

VIII. CONCLUSION

Currently, there are more and more devices connected to IoT. With the help of cloud server, these IoT devices reduce the burden of computing and storage while ensuring data security and privacy, which makes the IoT more widely available. In this paper, we propose KS-ABESwET in IoT, on the basis of general CP-ABE scheme, the inverted index is used to implement a more practical keyword search, which solves the data search problem of IoT devices. For the sake of reducing local load, a large number of calculations in scheme are outsourced to the server, the resource-constrained IoT terminal only needs to perform very few operations. In addition, our scheme supports the equality test, so that ACS can determine whether two ciphertexts encrypted by different access policies contain the same plaintext without decrypting ciphertext, which is beneficial to ciphertext classification and accurate decryption, decreasing storage resource consumption of IoT devices. Subsequently, based on the decisional $q-1$ assumption and DDH assumption, the proposed scheme is proved to be chosen-plaintext security and chosen-keyword security. Moreover, we compare the performance and efficiency of the solution with other relevant literatures, which indicates that our scheme is practical. In the future, how to simplify the equality test algorithm and achieve OW-CCA security will be a problem that needs to further study.

ACKNOWLEDGMENT

The authors would like to thank also to the anonymous reviewers for their useful comments.

REFERENCES

[1] I. Lee and K. Lee, "The Internet of Things (IoT): Applications, investments, and challenges for enterprises," *Bus. Horizons*, vol. 58, no. 4, pp. 431–440, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0007681315000373
[2] S. Belguith, N. Kaaniche, M. Laurent, A. Jemai, and R. Attia, "PHOABE: Securely outsourcing multi-authority attribute based encryption with policy hidden for cloud assisted IoT," *Comput. Netw.*, vol. 133, pp. 141–156, Mar. 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128618300495
[3] J. Li, W. Yao, Y. Zhang, H. Qian, and J. Han, "Flexible and fine-grained attribute-based data storage in cloud computing," *IEEE Trans. Services Comput.*, vol. 10, no. 5, pp. 785–796, Jan. 2016. doi: 10.1109/TSC.2016.2520932.
[4] J. Li, Q. Yu, and Y. Zhang, "Key-policy attribute-based encryption against continual auxiliary input leakage," *Inf. Sci.*, vol. 470, pp. 175–188, Jan. 2019. doi: 10.1016/j.ins.2018.07.077.
[5] J. Li, W. Yao, J. Han, Y. Zhang, and J. Shen, "User collusion avoidance CP-ABE with efficient attribute revocation for cloud storage," *IEEE Syst. J.*, vol. 12, no. 2, pp. 1767–1777, Jun. 2018. doi: 10.1109/JSYST.2017.2667679.
[6] H. Yin, J. Zhang, Y. Xiong, L. Ou, F. Li, S. Liao, and K. Li, "CP-ABSE: A ciphertext-policy attribute-based searchable encryption scheme," *IEEE Access*, vol. 7, pp. 5682–5694, 2019. doi: 10.1109/ACCESS.2018.2889754.
[7] J. Li, X. Lin, Y. Zhang, and J. Han, "KSF-OABE: Outsourced attribute-based encryption with keyword search function for cloud storage," *IEEE Trans. Services Comput.*, vol. 10, no. 5, pp. 715–725, Oct. 2017.
[8] S. S. M. Chow, J. K. Liu, and J. Zhou, "Identity-based online/offline key encapsulation and encryption," in *Proc. 6th ACM Symp. Inf., Comput. Commun. Secur. (ASIACCS)*, Hong Kong, Mar. 2011, pp. 52–60. doi: 10.1145/1966913.1966922.
[9] T. H. Yuen, Y. Zhang, S. M. Yiu, and J. K. Liu, "Identity-based encryption with post-challenge auxiliary inputs for secure cloud applications and sensor networks," in *Computer Security—ESORICS*, M. Kutyłowski and J. Vaidya, Eds. Cham, Switzerland: Springer, 2014, pp. 130–147.
[10] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology—EUROCRYPT*, R. Cramer, Ed. Berlin, Germany: Springer, 2005, pp. 457–473.
[11] G. Wang, Q. Liu, and J. Wu, "Hierarchical attribute-based encryption for fine-grained access control in cloud storage services," in *Proc. 17th ACM Conf. Comput. Commun. Secur. (CCS)*, Chicago, IL, USA, Oct. 2010, pp. 735–737. doi: 10.1145/1866307.1866414.
[12] 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)*, Alexandria, VA, USA, Oct./Nov. 2006, pp. 89–98. doi: 10.1145/1180405.1180418.
[13] X. Liu, J. Ma, J. Xiong, and G. Liu, "Ciphertext-policy hierarchical attribute-based encryption for fine-grained access control of encryption data," *Int. J. Netw. Secur.*, vol. 16, no. 6, pp. 437–443, Nov. 2014. [Online]. Available: http://ijns.femto.com.tw/contents/ijns-v16-n6/ijns-2014-v16-n6-p437-443.pdf
[14] S. Porwal and S. Mittal, "Implementation of ciphertext policy-attribute based encryption (CP-ABE) for fine grained access control of university data," in *Proc. 10th Int. Conf. Contemp. Comput. (IC3)*, Los Alamitos, CA, USA, Aug. 2017, pp. 1–7. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/IC3.2017.8284289
[15] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2007, pp. 321–334.
[16] J. K. Liu, T. H. Yuen, P. Zhang, and K. Liang, "Time-based direct revocable ciphertext-policy attribute-based encryption with short revocation list," in *Proc. 16th Int. Conf. Appl. Cryptogr. Netw. Secur. (ACNS)*, Leuven, Belgium, Jul. 2018, pp. 516–534. doi: 10.1007/978-3-319-93387-0_27.
[17] D. Li, J. Liu, Q. Wu, and Z. Guan, "Efficient CCA2 secure flexible and publicly-verifiable fine-grained access control in fog computing," *IEEE Access*, vol. 7, pp. 11688–11697, 2019.
[18] A. Wu, Y. Zhang, X. Zheng, R. Guo, Q. Zhao, and D. Zheng, "Efficient and privacy-preserving traceable attribute-based encryption in blockchain," *Ann. Telecommun.*, Jan. 2019. doi: 10.1007/s12243-018-00699-y.
[19] J. Li, Q. Yu, and Y. Zhang, "Hierarchical attribute based encryption with continuous leakage-resilience," *Inf. Sci.*, vol. 484, pp. 113–134, May 2019. doi: 10.1016/j.ins.2019.01.052.

[20] S. Wang, J. Zhou, J. K. Liu, J. Yu, J. Chen, and W. Xie, "An efficient file hierarchy attribute-based encryption scheme in cloud computing," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 6, pp. 1265–1277, Jun. 2016. doi: 10.1109/TIFS.2016.2523941.

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

[22] C. Zuo, J. Shao, J. K. Liu, G. Wei, and Y. Ling, "Fine-grained two-factor protection mechanism for data sharing in cloud storage," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 1, pp. 186–196, Jan. 2018.

[23] K. He, J. Weng, J. K. Liu, W. Zhou, and J.-N. Liu, "Efficient fine-grained access control for secure personal health records in cloud computing," in *Network and System Security*, J. Chen, V. Piuri, C. Su, and M. Yung, Eds. Cham, Switzerland: Springer, 2016, pp. 65–79.

[24] M. H. Au, T. H. Yuen, J. K. Liu, W. Susilo, X. Huang, Y. Xiang, and Z. L. Jiang, "A general framework for secure sharing of personal health records in cloud system," *J. Comput. Syst. Sci.*, vol. 90, pp. 46–62, Dec. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022000017300296

[25] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in *Proc. IEEE Symp. Secur. Privacy*, May 2000, pp. 44–55.

[26] X. Chen, J. Li, J. Weng, J. Ma, and W. Lou, "Verifiable computation over large database with incremental updates," *IEEE Trans. Comput.*, vol. 65, no. 10, pp. 3184–3195, Oct. 2016.

[27] X. Chen, X. Huang, J. Li, J. Ma, W. Lou, and D. S. Wong, "New algorithms for secure outsourcing of large-scale systems of linear equations," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 1, pp. 69–78, Jan. 2015.

[28] W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li, "Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," in *Proc. 8th ACM SIGSAC Symp. Inf., Comput. Commun. Secur. (ASIA CCS)*, New York, NY, USA, 2013, pp. 71–82. doi: 10.1145/2484313.2484322.

[29] W. Sun, S. Yu, W. Lou, Y. T. Hou, and H. Li, "Protecting your right: Verifiable attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 4, pp. 1187–1198, Apr. 2016.

[30] B. Wang, S. Yu, W. Lou, and Y. T. Hou, "Privacy-preserving multi-keyword fuzzy search over encrypted data in the cloud," in *Proc. IEEE INFOCOM*, May 2014, pp. 2112–2120. doi: 10.1109/INFOCOM.2014.6848153.

[31] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: Improved definitions and efficient constructions," *J. Comput. Secur.*, vol. 19, no. 5, pp. 895–934, Jan. 2011. [Online]. Available: http://dl.acm.org/citation.cfm?id=2590701.2590705

[32] M. Naveed, M. Prabhakaran, and C. A. Gunter, "Dynamic searchable encryption via blind storage," in *Proc. IEEE Symp. Secur. Privacy (SP)*, Berkeley, CA, USA, May 2014, pp. 639–654. doi: 10.1109/SP.2014.47.

[33] B. Wang, W. Song, W. Lou, and Y. T. Hou, "Inverted index based multi-keyword public-key searchable encryption with strong privacy guarantee," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, Kowloon, Hong Kong, Apr./May 2015, pp. 2092–2100. doi: 10.1109/INFOCOM.2015.7218594.

[34] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS)*, New York, NY, USA, 2007, pp. 456–465. doi: 10.1145/1315245.1315302.

[35] V. Odelu, A. K. Das, M. K. Khan, K. R. Choo, and M. Jo, "Expressive CP-ABE scheme for mobile devices in IoT satisfying constant-size keys and ciphertexts," *IEEE Access*, vol. 5, pp. 3273–3283, 2017.

[36] S. Belguith, N. Kaaniche, M. Laurent, A. Jemai, and R. Attia, "Constant-size threshold attribute based signcryption for cloud applications," in *Proc. 14th Int. Joint Conf. e-Bus. Telecommun. (ICETE)*, Madrid, Spain, vol. 4, Jul. 2017, pp. 212–225. doi: 10.5220/0006469202120225.

[37] V. Odelu, A. K. Das, Y. S. Rao, S. Kumari, M. K. Khan, and K.-K. R. Choo, "Pairing-based CP-ABE with constant-size ciphertexts and secret keys for cloud environment," *Comput. Stand. Interfaces*, vol. 54, no. P1, pp. 3–9, Nov. 2017. doi: 10.1016/j.csi.2016.05.002.

[38] M. Green, S. Hohenberger, and B. Waters, "Outsourcing the decryption of ABE ciphertexts," in *Proc. 20th USENIX Conf. Secur. (SEC)*, Berkeley, CA, USA, 2011, p. 34. [Online]. Available: http://dl.acm.org/citation.cfm?id=2028067.2028101

[39] Y. Yang, H. Zhu, H. Lu, J. Weng, Y. Zhang, and K.-K. R. Choo, "Cloud based data sharing with fine-grained proxy re-encryption," *Pervasive Mobile Comput.*, vol. 28, pp. 122–134, Jun. 2016. doi: 10.1016/j.pmcj.2015.06.017.

[40] J. Shao, Y. Zhu, and Q. Ji, "Efficient decentralized attribute-based encryption with outsourced computation for mobile cloud computing," in *Proc. IEEE Int. Symp. Parallel Distrib. Process. Appl., IEEE Int. Conf. Ubiquitous Comput. Commun. (ISPA/IUCC)*, Los Alamitos, CA, USA, Dec. 2017, pp. 417–422. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/ISPA/IUCC.2017.00067

[41] G. Yang, C. H. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Topics in Cryptology—CT-RSA*, J. Pieprzyk, Ed. Berlin, Germany: Springer, 2010, pp. 119–131.

[42] Q. Tang, "Towards public key encryption scheme supporting equality test with fine-grained authorization," in *Information Security and Privacy*, U. Parampalli and P. Hawkes, Eds. Berlin, Germany: Springer, 2011, pp. 389–406.

[43] 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.

[44] 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. doi: 10.1504/IJACT.2012.048079.

[45] S. Ma, "Identity-based encryption with outsourced equality test in cloud computing," *Inf. Sci.*, vol. 328, pp. 389–402, Jan. 2016. doi: 10.1016/j.ins.2015.08.053.

[46] H. 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. doi: 10.1016/j.ins.2016.09.013.

[47] 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. doi: 10.1016/j.future.2017.03.007.

[48] 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, 2017.

[49] Y. Liao, H. Chen, F. Li, S. Jiang, S. Zhou, and R. Mohammed, "Insecurity of a key-policy attribute based encryption scheme with equality test," *IEEE Access*, vol. 6, pp. 10189–10196, 2018.

[50] Q. Wang, L. Peng, H. Xiong, J. Sun, and Z. Qin, "Ciphertext-policy attribute-based encryption with delegated equality test in cloud computing," *IEEE Access*, vol. 6, pp. 760–771, 2018.

[51] H. Cui, R. H. Deng, G. Wu, and J. Lai, "An efficient and expressive ciphertext-policy attribute-based encryption scheme with partially hidden access structures," in *Provable Security*, L. Chen and J. Han, Eds. Cham, Switzerland: Springer, 2016, pp. 19–38.

[52] Y. Rouselakis and B. Waters, "Practical constructions and new proof methods for large universe attribute-based encryption," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS)*, Berlin, Germany, Nov. 2013, pp. 463–474. doi: 10.1145/2508859.2516672.

[53] W. Guo, X. Dong, Z. Cao, and S. Jiachen, "Efficient attribute-based searchable encryption on the cloud storage," *IACR Cryptol. ePrint Arch.*, vol. 2017, p. 782, 2017. [Online]. Available: https://eprint.iacr.org/2017/782.pdf

[54] I. E. Shparlinski, "Computational Diffie–Hellman problem," in *Encyclopedia Cryptography Security*, 2nd ed. H. C. A. van Tilborg and S. Jajodia, Eds. Boston, MA, USA: Springer, 2011, pp. 240–244. doi: 10.1007/978-1-4419-5906-5_882.

[55] S. Qiu, J. Liu, Y. Shi, and R. Zhang, "Hidden policy ciphertext-policy attribute-based encryption with keyword search against keyword guessing attack," *Sci. China Inf.*, vol. 60, no. 5, p. 052105, Sep. 2016. doi: 10.1007/s11432-015-5449-9.

[56] J. Li, F. Sha, Y. Zhang, X. Huang, and J. Shen, "Verifiable outsourced decryption of attribute-based encryption with constant ciphertext length," *Secur. Commun. Netw.*, vol. 2017, pp. 3 596 205:1–3 596 205:11, 2017. doi: 10.1155/2017/3596205.

[57] R. Zhang, H. Ma, and Y. Lu, "Fine-grained access control system based on fully outsourced attribute-based encryption," *J. Syst. Softw.*, vol. 125, pp. 344–353, Mar. 2017. doi: 10.1016/j.jss.2016.12.018.

[58] L. Wu, Y. Zhang, K. R. Choo, and D. He, "Efficient identity-based encryption scheme with equality test in smart city," *IEEE Trans. Sustain. Comput.*, vol. 3, no. 1, pp. 44–55, Jan. 2018.

[59] S. Duquesne and T. Lange, "Pairing-based cryptography," in *Handbook of Elliptic and Hyperelliptic Curve Cryptography*, vol. 22, 2005, pp. 573–590. doi: 10.1201/9781420034981.ch24.

SHANGPING WANG received the B.S. degree in mathematics from the Xi'an University of Technology, Xi'an, China, in 1982, the M.S. degree in applied mathematics from Xi'an Jiaotong University, Xi'an, in 1989, and the Ph.D. degree in cryptology from Xidian University, Xi'an, in 2003. He is currently a Professor with the Xi'an University of Technology. His current research interests include cryptography and information security.

LISHA YAO is currently pursuing the M.S. degree with the Xi'an University of Technology, Xi'an, China. Her research interests include cryptography and information security.

JUANJUAN CHEN received the Ph.D. degree from Shaanxi Normal University, Xi'an, Shaanxi, China, in 2014. She is currently a Lecturer with the Xi'an University of Technology. Her main research interest includes cryptography.

YALING ZHANG received the B.S. degree in computer science from Northwest University, Xi'an, China, in 1988, and the M.S. degree in computer science and the Ph.D. degree in mechanism electron engineering from the Xi'an University of Technology, Xi'an, in 2001 and 2008, respectively. She is currently a Professor with the Xi'an University of Technology. Her current research interests include cryptography and network security.