Ciphertext Policy Attribute Based Encryption for Arithmetic circuits

Mahdi Mahdavi Oliaee and Zahra Ahmadian

***Abstract*—We present the first Ciphertext Policy Attribute Based Encryption (CP-ABE) scheme with arithmetic circuit access policy. The idea is first introduced as a basic design which is based on multilinear maps. Then, two improved versions of that, with or without the property of hidden attributes, are introduced. We also define the concept of Hidden Result Attribute Based Encryption (HR-ABE) which means that the result of the arithmetic function will not be revealed to the users. We prove that the proposed schemes have adaptive security, under the $(k-1)$-Distance Decisional Diffie-Hellman assumption.**

***Index Terms*—Ciphertext Policy Attribute Based Encryption (CP-ABE), Arithmetic circuit, Multilinear map, Adaptive security, Hidden attributes, Hidden Result.**

I. Introduction

Nowadays, there is a considerable demand for fine-grained data sharing in cloud based communication systems, where access to data is supposed to be limited to specific eligible users. This type of data sharing requires a decentralized, flexible and dynamic access control over a service provider which is not necessarily trusted-enough. Based on the traditional public key encryption solutions, the sender must identify all the potential qualified users and encrypt the message separately for each of which; an extremely inefficient solution. Attribute Based Encryption (ABE) addresses this demand by providing a decentralized access control based on the user's set of attributes. The access structure, which is itself protected by encryption, can be embedded either into the key (KP-ABE) or ciphertext (CP-ABE). The flexibility of ABE makes it applicable to many different aspects of recent technologies, such as Internet of Things [?], Personal Healthcare Records [?], and vehicular networks [?].

Related work. The concept of Attribute Based Encryption (ABE) was first invented by Sahai, Waters, et al. [1], though under the title of fuzzy Identity Based Encryption. In their scheme, each user has a set of attributes and a set of secret keys associated with these attributes. The message is encrypted by the sender based on the attributes and if the intersection of the sender and receiver attribute sets are greater than a TTP-chosen threshold value, the message can be decrypted by the receiver. Goyal et al. [2] defined the concept of Key Policy Attribute Based Encryption (KP-ABE). In this type of ABE scheme, the ciphertext is labeled with a set of attributes, and the user's secret key is associated with an access structure. The ciphertext is decryptable only by the users whose secret key access structure is satisfied by the set of attributes attached to the ciphertext.

The concept of Ciphertext Policy Attribute Based Encryption (CP-ABE) was introduced by Bethencourt et al. [3], contrary to KP-ABE. In this type of ABE, the ciphertext is constructed according to access structure and the secret keys of the receiver are constructed according to the user's attributes. In these schemes, the set of attributes of the decryptor must satisfy the access structure defined in the ciphertext. Due to the possibility of choosing the access structure by the sender, this scheme is more flexible than KP-ABE. Moreover, KP-ABE is less efficient than CP-ABE in view of the ciphertext size. Bethencourt proved the security of his scheme in the generic group model. Waters in [4] proposed a CP-ABE scheme and demonstrated the security of his scheme under standard assumptions. All of these schemes support the monotone circuit access structures. Ostrofsky et al. [5] presented the first schemes for non-monotone circuits. Green et al. [6] proposed the idea of outsourcing the heavy computations to cloud service, in order to reduce the computational overhead for users.

One challenge in this domain is revoking the attributes (keys) and users. Some schemes, like [7] and [8], focus on resolving this problem. Chase in [9] proposed the multi authority ABE as a solution for the key escrow problem. In [10], Attrapondong and Imai present the Dual Policy ABE, which is a kind of ABE with simultaneous key and ciphertext policies. In [11] the Hierarchical Attribute Based Encryption (HABE) was presented. In HABE, users possessing an attribute with a higher level can decrypt the messages encrypted for those with lower level ones. For example, a commander can decrypt messages that are encrypted for soldiers. Some articles have focused on increasing the efficiency, security, and size of the ciphertext and keys [12],[13], and [14].

Two levels of security have been defined for ABE schemes: selective security and adaptive security. In the selective security game, the attacker selects the challenge attribute vector (or function) at the beginning of the setup phase and sends it to the simulator. Then, the simulator constructs the public parameters according to the received vector. The attacker can request the secret keys, adaptively. These secret keys should not satisfy the challenge vector (or function). On the other hand, in the adaptive security game, public parameters are defined by the simulator and are sent to the attacker, at the beginning. Then, the attacker defines the challenge attribute vector (or function) and sends it to the simulator. Then the attacker requests the secret keys, adaptively. These secret keys should not satisfy the challenge vector (or function). Adaptive security is known as complete security. However, there is

The authors are with the Electrical Engineering Department, Shahid Beheshti University, Tehran, Iran. e-mail: m_mahdavioliaee@mail.sbu.ac.ir, z_ahmadian@sbu.ac.ir

another security level, called semi-adaptive security [15], that lies between these two levels of security. In semi-adaptive security, the simulator defines public parameters and sends them to the attacker. Then, the attacker selects the challenge vector (or function) and sends it to the simulator, and requests the secret keys. The simulator constructs secret keys according to request and challenge vector (or function) and sends these secret keys to the attacker. These secret keys should not satisfy the challenge vector (or function).

Garg et al. in [16] presented a backtracking attack for pairing-based ABE with circuits with fan-out bigger than one. Garg presented KP-ABE for all circuits using multilinear maps, though the underlying assumptions for proving its security are non-standard ones. Hard problems related to the multilinear maps are nonstandard cryptographic assumptions. However, his scheme works for any circuits with arbitrary fanout.

All the above schemes are constructed based on the bilinear pairing and their security rely on pairing-related hard problems. Therefore, they can not be regarded as the post quantum ABE schemes. Contrary to pairing based ABE schemes, lattice based ABE scheme are proposed, where security rely on the Learning With Error (LWE) assumption. Agrawal et al. [17] presented the Fuzzy ABE based on lattice for the first time. Boyen et al. [18] and Zhang et al. [19] presented the first lattice-based KP-ABE and CP-ABE, respectively. Gorbunov et al. [20] presented the lattice based KP-ABE for circuits with arbitrary fanout. This scheme is the first ABE scheme that works for any boolean function with standard assumptions. The technique used in this scheme is called two to one Recoding (TOR). Also, this scheme supports gates with fan-in two. The first work which supports the arithmetic circuit as the access structure is Boneh's scheme [21], where a fully key homomorphic encryption for constructing KP-ABE is proposed. In this scheme, addition and multiplication gates are used instead of the conventional AND and OR gates; a more general approach.

To reduce the complexity of LWE, in [22],[23], and [24], the use of Ring-LWE was proposed for designing ABE schemes. Schemes based on R-LWE have less computational complexity and memory required. Recently, an adaptively secure ABE based on LWE is proposed [25].

If the attribute vector or policy in the ciphertext is hidden, the ABE is called Predicate Encryption [26]. Predicate Encryption is a special case of functional encryption [27], in which the receiver can obtain a function of the encrypted data. Finally, some ABE schemes, such as [28] and [29], the policy is hidden.

Our contribution. In this paper, we propose the first CP-ABE schemes for arithmetic functions with arbitrary results. The proposed schemes are designed based on multilinear maps. We introduce the new concept of hidden result ABE, which means that the result of the arithmetic function remains unknown to the user.

The proposed schemes are described in three variants. A basic scheme is first introduced by which the platform of our idea is demonstrated. In this scheme, the result and attribute vector is hidden and it covers simple arithmetic functions. Then, the first improved version is proposed, in which the arithmetic function is more general than the basic one and the attribute vector, as well as the result value, are unknown to the users. Finally, the second improved version is described, in which the attribute vector of each user is disclosed to himself. The result value will be known for the eligible users, e.g. those with a set of attributes satisfying the access structure function. The adaptive security for all of these schemes is proved based on a new-defined hard problem, which we call $k-1$-distance Diffie-Hellman problem. This problem is at least as hard as the k-multilinear Diffie Hellman problem.

Comparing to [21], which is the only existing ABE work for arithmetic circuits, the proposed scheme has significant advantages. Our schemes are CP-ABE. They have adaptive security. The result can take any arbitrary value. The variant with hidden attribute, can be used for predicate encryption. It supports the exponentiation gate and does not have any constraint over the attribute values. However, scheme [21] is lattice based which makes it a post quantum scheme, despite ours.

Paper structure. The structure of the rest of the paper is as follows. In Sec. II, the preliminaries for the paper are reviewed. In Sec. III, the proposed basic scheme is detailed and its security is proved. Sections IV and V describe the two improved versions of the basic scheme, which are with or without the property of hidden attribute and result, respectively. A comparison of the proposed scheme with Boneh's scheme is brought in Sec. VI. Finally Sec. VII concludes our work.

II. Preliminaries

In this section, we provide preliminaries that are necessary for the rest of the paper.

Definition 1. *k-Multilinear map. The multilinear map is defined over k groups of the same order $\boldsymbol{G}_1, \boldsymbol{G}_2, \dots, \boldsymbol{G}_k$. Assume that g_i is the generator of $\boldsymbol{G}_i$ for $i \in \{1, 2, \dots, k\}$. The function $e_{i,j}$ is defined as below:*

$$\begin{aligned} e_{i,j} : \boldsymbol{G}_i \times \boldsymbol{G}_j \longrightarrow \boldsymbol{G}_{i+j};\ 1 \le i, j, i+j \le k \\ e_{i,j}(g_i^a, g_j^b) = g_{i+j}^{ab} \end{aligned} \tag{1}$$

We can summerize the consecutive computations of several bilinear maps (1) into the following formula.

$$e(g_{i_1}^{x_1}, g_{i_2}^{x_2}, \dots, g_{i_m}^{x_m}) = g_n^{\prod_{i=1}^m x_i} \tag{2}$$

where $n = \sum_{j=1}^m i_j \le k$. There is a polynomial-time algorithm for computing the above equations. The bilinear map (or pairing) is a special case of this map for $k = 2$.

Definition 2. *k-Multilinear Diffie-Hellman problem. Given the vector $\Big\{\{g_1, g_2, \dots, g_k\}, g^s, g^{c_1}, g^{c_2}, \dots, g^{c_k}\Big\}$, where $g = g_1$, computing the amount of $T = g_k^{s.\prod_{i=1}^k c_i}$ is known as the k-Multilinear Diffie-Hellman (k-MDH) problem.*

Definition 3. *k-Multilinear Decisional Diffie-Hellman problem. Assume $g = g_1$, given the vector $\Big\{\{g_1, g_2, \dots, g_k\}, g^s, g^{c_1},$*

$g^{c_2}, \dots, g^{c_k}, g_k^z \}$, *deciding if* $z = \prod_{i=1}^{k} c_i$ *or not is known as the k-Multilinear Decisional Diffie-Hellman (k-MDDH) problem.*

Definition 4. $(k-1)$*-Distance Diffie-Hellman problem. Given a k-multilinear map over groups* $\mathbf{G}_1, \dots, \mathbf{G}_k$*, and* $\left\{g^x, g_k^y\right\}$*, we define the problem of computing* $T = g_k^{x.y}$ *as* $(k-1)$*-Distance Diffie-Hellman (*$(k-1)$*-DsDH) problem.*

This problem is at least as hard as k-MDH problem, i.e. given access to oracle O that solves $(k-1)$-DsDH problem, one can solve k-MDH problem. For demonstrating this claim, assume that we are given $\left\{\{g_1, g_2, \dots, g_k\}, g^x, g^{c_1}, g^{c_2}, \dots, g^{c_k}\right\}$ to compute $g_k^{s.\prod_{i=1}^{k} c_i}$. We first compute $g_k^y = e(g^{c_1}, g^{c_2}, \dots, g^{c_k})$, then we query O by $\left\{g^x, g_k^y\right\}$.

Definition 5. $(k-1)$*-Distance Decisional Diffie-Hellman problem. Assume that we have a k-Multilinear map over groups* $\mathbf{G}_1, \dots, \mathbf{G}_k$ *and are given vector* $\left\{g^x, g_k^y, g_k^z\right\}$*. We define the problem of deciding if* $z = x.y$ *or not as* $(k-1)$*-Distance Decisional Diffie-Hellman (*$(k-1)$*-DsDDH) problem.*

This problem is at least as hard as the (k-MDDH) problem. This claim can be proved similar to the hardness proof of $(k-1)$-DsDH.

III. The proposed CP-ABE Scheme, Basic version

CP-ABE schemes for arithmetic circuits aim to realize the access policies consistent with all or a class of arithmetic functions $f(\mathbf{x}) = f(x_1, x_2, \dots, x_n)$, $deg(f) \leq k$ where k is the number of groups in the underlying multilinear map. Each x_i , $i = 1, 2, \dots, n$ corresponds to one attribute and $\mathbf{x} = [x_1, x_2, \dots, x_n]$ is the attribute vector. Note that n is the number of attributes and k is called the *depth* of function (circuit). The encryptor of the message can encrypt the ciphertext in a way that only the users whose attribute vectors satisfy $f(\mathbf{x}) = y$ can decrypt the ciphertext, where y is an encryptor-chosen value and is called the *result*. $f(\cdot)$ is chosen by the encryptor, as well, conditioned that it meets the limitations of the functions supported by the design, if any.

A. Limitations and Specifications

In this section, we propose a CP-ABE scheme which can be realized for access structures with arithmetic circuits of the following form.

$$f(\mathbf{x}) = \sum_{i=1}^{|S|} \big(a_i \prod_{j \in P_i} x_j\big) \tag{3}$$

where $P_i, i = 1, \dots, 2^k$ is a subgroup of $\{1, 2, \dots, k\}$. S is defined as the set of all P_i that a_i is nonzero. The cardinality of S is denoted by $|S|$.

for the CP-ABE proposed in this section, we restrict $f(\mathbf{x})$ to the functions that $k = n$, and $\forall P_i, P_j \in S, i \neq j, P_i \cap P_j = \emptyset$. However, the proposed scheme works for any result value $y \in Z_q$. Moreover, in this scheme the user does not know the value of his/her own attribute vector as well as the value of result. Some of these constraints will be relaxed in the schemes proposed in next sections.

B. The CP-ABE Scheme

The proposed CP-ABE scheme is a quadruple (Setup, KeyGen, Enc, Dec) of probabilistic polynomial time algorithms, which are described in the following.

Setup$(\lambda, 1^k)$: This algorithm takes security parameter λ and the multilinear map paratmeter k as input. Then, it outputs the public parameters of the scheme, the public key, and the master secret key.

The k groups of $\mathbf{G}_1, \mathbf{G}_2, \dots, \mathbf{G}_k$ with generators $g_1, g_2, \dots, g_k$ respectively, all with the same prime order q, are selected as the public parameters of the scheme. For simplicity g_1 is denoted by g.

$$PP = \{\mathbf{G}_1, \mathbf{G}_2, \dots, \mathbf{G}_k, g, g_2, \dots, g_k\} \tag{4}$$

A multilinear map $\{e_{i,j}; i, j \in \{1, \dots, k-1\}\}$ which is defined over these groups is also public. A number of $2k$ random values $t_1, t_2, \dots, t_k, s_1, s_2, \dots, s_k \in Z_q$ are selected. Then, the public key PK and the master secret key MSK are generated, as below.

$$PK = \left\{[g^{t_1}, g^{t_2}, \dots, g^{t_k}], [g^{\frac{1}{s_1}}, g^{\frac{1}{s_2}}, \dots, g^{\frac{1}{s_k}}], [g^{\frac{t_1}{s_1}}, g^{\frac{t_2}{s_2}}, \dots, g^{\frac{t_k}{s_k}}]\right\}$$

$$MSK = \{[t_1, t_2, \dots, t_k], [s_1, s_2, \dots, s_k]\} \tag{5}$$

KeyGen(MSK): This algorithm takes the master secret key MSK as input. Then, it outputs the user's secret key SK. For generating the user's secret keys, the values of $x_1, x_2, \dots, x_k \in Z_q$ are set according to the value of the user's attributes. Then secret keys SK are generated as below.

$$SK = [sk_1, sk_2, \dots, sk_k] = [s_1 x_1, s_2 x_2, \dots, s_k x_k] \tag{6}$$

Note that the user, who is the owner of secret keys, does not know the value of its own attributes.

Enc(PK, f, y, m): This algorithm takes public key PK, arithmetic function f consistent with the specification given in Sec. III-A, result y, and message m which is encoded to an element of G_k, as input. It outputs the ciphertext Ctx which can be decrypted only by the users whose attribute vector $\mathbf{x}$ satisfies $f(\mathbf{x}) = y$.

Then, the encryptor chooses random numbers $r_1, r_2, \dots, r_k \in Z_q$ such that $\forall P_j \in S$, $\prod_{i \in P_j} r_i = R$. Note that since P_js are disjoint, such a set of $r_1, r_2, \dots, r_n$ always exists. Then, he computes $C_1, C_2, \dots, C_k$ as follows.

$$C_1 = g^{\frac{r_1 t_1}{s_1}}, C_2 = g^{\frac{r_2 t_2}{s_2}}, \dots, C_k = g^{\frac{r_k t_k}{s_k}} \tag{7}$$

and C_0 and $Check$ are also computed as

$$\begin{aligned} C_0 &= m \cdot (g_k^{\prod_{v=1}^{k} t_v})^{y.R} \\ Check &= g_k^y \end{aligned} \tag{8}$$

Finally, the ciphertext is generated as below:

$$Ctx = [f, C_0, C_1, C_2, \dots, C_k, Check] \tag{9}$$

The value of $Check$ is used for checking the result of the function. The value of $g_k^{\prod_{i=1}^{k} t_i}$ can be easily computed by applying a multilinear map as follows.

$$\begin{aligned} e_k(g^{t_1}, g^{t_2}, \ldots, g^{t_k}) &= e_{k-1,1}(\ldots e_{21}(e_{11}(g^{t_1}, g^{t_2}), g^{t_3}) \ldots, g^{t_k}) \\ &= g_k^{\prod_{v=1}^{k} t_v} \qquad (10) \end{aligned}$$

The above computation can be done in the KeyGen algorithm by TTP and be defined as a piece of the public key.

Dec(Ctx, PK, SK): This algorithm is a deterministic algorithm that takes the ciphertext Ctx, public key PK and the secret key SK as input and outputs message m only if Ctx is an encryption of m under the public key PK and $f(\mathbf{x}) = y$.

The decryptor first computes $I_{P_i}, i = 1, \ldots, |S|$ as follows.

$$I_{P_i} = e(C_{i_1}, C_{i_2}, \ldots, C_{i_w}, g^{t_{j_1}}, g^{t_{j_2}}, \ldots, g^{t_{j_{(k-w)}}}) \qquad (11)$$

where $P_i = \{i_1, \ldots, i_w\}$, $w = |P_i|$ and $\{1, \ldots, k\} \setminus P_i = \{j_1, \ldots, j_{k-w}\}$. Then, he computes $Mask$, and decrypts the ciphertext Ctx into message m' as follows.

$$\begin{aligned} Mask &= \prod_{i=1}^{|S|} (I_{P_i})^{a_i \prod_{j \in P_i} sk_j} \\ m' &= \frac{C_0}{Mask} \qquad (12) \end{aligned}$$

The correctness of equation (12) is as follows. We first simplify (11) according to the following.

$$\begin{aligned} I_{P_i} &= g_k^{\prod_{j \in P_i} (\frac{r_j . t_j}{s_j}) \cdot \prod_{v \notin P_i} t_v} \\ &= g_k^{\frac{\prod_{j \in P_i} (r_j)}{\prod_{j \in P_i} (s_j)} \cdot \prod_{v=1}^{k} t_v} \\ &= g_k^{\frac{R}{\prod_{j \in P_i} (s_j)} \cdot \prod_{v=1}^{k} t_v} \qquad (13) \end{aligned}$$

So, the value of $Mask$ is equal to

$$\begin{aligned} Mask &= \prod_{i=1}^{|S|} \left(I_{P_i}\right)^{a_i \prod_{j \in P_i} sk_j} \\ &= \prod_{i=1}^{|S|} \left(g_k^{\frac{R}{\prod_{j \in P_i} (s_j)} \prod_{v=1}^{k} t_v} \right)^{a_i \prod_{j \in P_i} s_j x_j} \\ &= \prod_{i=1}^{|S|} g_k^{R \cdot a_i (\prod_{j \in P_i} (x_j)) \prod_{v=1}^{k} t_v} \\ &= g_k^{\left(\sum_{i=1}^{|S|} \left(a_i . \prod_{j \in P_i} x_j\right)\right) R . \prod_{v=1}^{k} t_v} \\ &= g_k^{f(\mathbf{x}) . R \prod_{v=1}^{k} t_v} \qquad (14) \end{aligned}$$

Finally, equations (14) along with (8) yeilds (12).

For example, assume that $S = \{P_1, P_2\}$ where $P_1 = \{1, 3\}$ and $P_2 = \{2\}$. Here, $k = n = 3$ and $f(\mathbf{x}) = a_1 x_1 x_3 + a_2 x_2$. So, the value of $Mask$ is as follows.

$$\begin{aligned} Mask &= \prod_{i=1}^{2} \left(I_{P_i}\right)^{a_i \prod_{j \in P_i} sk_j} \\ &= (I_{P_1})^{a_1 \prod_{j \in \{1,3\}} sk_j} \cdot (I_{P_2})^{a_2 \prod_{j \in \{2\}} sk_j} \\ &= (I_{P_1})^{a_1 (s_1 x_1 . s_3 x_3)} \cdot (I_{P_2})^{a_2 (s_2 x_2)} \\ &= g_3^{R \cdot a_1 x_1 x_3 \prod_{v=1}^{3} t_v} \cdot g_3^{R \cdot a_2 x_2 \prod_{v=1}^{3} t_v} \\ &= g_3^{R(a_1 x_1 x_3 + a_2 x_2) \prod_{v=1}^{3} t_v} \\ &= g_k^{f(\mathbf{x}) . R \prod_{v=1}^{k} t_v} \qquad (15) \end{aligned}$$

Since the attribute vector and the result are hidden for the decryptor, it should check if $Check = g_k^{f(\mathbf{x})}$ to make sure that the decryption is correct and $m' = m$. The decryptor computes $g_k^{f(\mathbf{x})}$ by computing the following.

$$\begin{aligned} Check' &= \prod_{P_i \in S} e\left((g^{\frac{1}{s_{i_1}}})^{sk_{i_1}}, \ldots, (g^{\frac{1}{s_{i_{|P_i|}}}})^{sk_{i_{|P_i|}}} \right)^{a_i} \\ &= \prod_{P_i \in S} e\left(g^{x_{i_1}}, \ldots, g^{x_{i_{|P_i|}}} \right)^{a_i} \\ &= \prod_{P_i \in S} g_k^{a_i \prod_{j \in P_i} x_j} \\ &= g_k^{f(\mathbf{x})} \qquad (16) \end{aligned}$$

If the $Check' = Check$, the receiver will conclude that he is an authorized user for decryptyng Ctx.

C. Security Proof

In this section we prove that the proposed scheme in Sec. III-B acheieves adaptive security. Suppose that there exist a polynomial-time attacker $\mathcal{A}$ for the proposed basic ABE system for arithmetic circuit in the adaptive security game, which can distinguish between the ciphertexts of two messages m_0 and m_1 with a probability of $\frac{1}{2} + \epsilon$, where ϵ is non-negligible. Having this assumption, We prove that there is a polynomial-time challenger C that can solve $(k-1)$-DsDDH problem with a probability nonnegligibly greater than $\frac{1}{2}$.

In this model, the challenger C gets the $(k-1)$-DsDDH parameters then simulates/ the above scheme parameters to attacker $\mathcal{A}$. The attacker $\mathcal{A}$ adaptively requests for secret keys. Then, the challenger generates secret keys to the attacker. In the next step, the attacker chooses two messages m_0 and m_1 and sends the to the challenger. The challenger randomly chooses one of these messages and simulates Enc algorithm to receive Ctx. Then challenger sends it to $\mathcal{A}$. The attacker $\mathcal{A}$ should decide which message was encrypted and sends the result to the challenger. The challenger can solve to k-MDDH problem according to the received result.

Theorem 1. *The proposed basic ABE scheme (section III-B) achieves adaptive security for arithmetic functions of the form (20) with k variables under $(k-1)$-DsDDH assumption*

Proof. We follow the adaptive security game and conclude that if there exist the polynomial-time attacker $\mathcal{A}$ that distinguishes between two encrypted messages in the proposed scheme, with nonnegligible advantage, then the challenger C can construct a polynomial-time algorithm for solving $(k-1)$-DsDDH problem with nonnegligible advantage. The security game for our scheme is as follows.

1) The challenger is given the $(k-1)$-DsDDH parameters as below.

$$\{\mathbf{G}_1, \mathbf{G}_2, \ldots, \mathbf{G}_k, g, g_2, \ldots, g_k, g^x, g_k^y, g_k^z\}$$

   The challenger must distinguish if $z = x \cdot y$ or it is a random value.

2) The challenger C chooses $t_1, \ldots, t_{k-1}, s_1, \ldots s_k \in Z_q$ randomly, and computes g^{t_i} and $g^{\frac{t_i}{s_i}}$ for $i = 1, \ldots, k-1$. Then, it sets $g^{t_k} = g^{x \cdot \prod_{i=1}^{k-1} t_i^{-1}}$, and simulates the public parameters PP according to (4) and public key PK for the attacker $\mathcal{A}$ as follows.

$$\begin{aligned} PK \quad = \quad & \{[g^{t_1}, \ldots, g^{t_{k-1}}, g^{x \cdot \prod_{i=1}^{k-1} t_i^{-1}}], \\ & [g^{\frac{1}{s_1}}, \ldots, g^{\frac{1}{s_{k-1}}}, g^{\frac{1}{s_k}}], \\ & [g^{\frac{t_1}{s_1}}, \ldots, g^{\frac{t_{k-1}}{s_{k-1}}}, g^{x s_k^{-1} \cdot \prod_{i=1}^{k-1} t_i^{-1}}]\} \end{aligned} \tag{17}$$

3) After receiving public parameters and public keys, $\mathcal{A}$ requests the challenger for secret keys SK. The challenger C randomely chooses a k-tuple $(x_1, \ldots, x_k)$ as attribute values and generates the secret key according to (6). Then, it sends them to the attacker $\mathcal{A}$, upon any secret key request by attacker $\mathcal{A}$.
4) $\mathcal{A}$ chooses the challenge function $f(\mathbf{x})$ and two messages m_0 and m_1, as well. Then, it sends $f(\mathbf{x})$, m_0, and m_1 to the challenger.
5) The challenger C randomly chooses one of the two messages m_0 and m_1. Then, C runs algorithm Enc to simulate the ciphertext of m_b where $b \in_r \{0, 1\}$. The ciphertext Ctx is as below.

$$\begin{aligned} Ctx \quad = \quad & [f,\ C_0 = m_b.(g_k^z)^R, \\ & C_1 = g^{\frac{r_1 t_1}{s_1}}, \ldots, C_{k-1} = g^{\frac{r_{k-1} t_{k-1}}{s_{k-1}}}, \\ & C_k = g^{x r_k s_k^{-1} \cdot \prod_{i=1}^{k-1} t_i^{-1}}, \\ & Chek = g_k^y] \end{aligned} \tag{18}$$

   The challenger sends Ctx to the attacker.
6) The attacker can request secret keys adaptively after receiving Ctx. The challenger solves these requests similar to Step 3.
7) The attacker sends the the guessed value b' to the challenger.

The probability of success of challenger for distinguishing the $(k-1)$–DsDDH problem is as follows.

$$\begin{aligned} Pr[C_{(k-1)\text{–DsDDH}} = \text{success}] \quad & = \quad \frac{1}{2} \cdot (\frac{1}{2} + \epsilon) + \frac{1}{2} \cdot \frac{1}{2} \\ & = \quad \frac{1}{2} + \frac{\epsilon}{2} \end{aligned} \tag{19}$$

In the above equation, the probability of resolving $(k-1)$–DsDDH problem is non-negligibly greater than $\frac{1}{2}$. So, the attacker $\mathcal{A}$ does not exist because $(k-1)$–DsDDH problem is assumed to be hard. □

IV. The Improved Scheme I, hidden result and attributes

In this section, we propose an improved version of the basic CP-ABE scheme for arithmetic circuits, proposed in Sec. III, in which some of the limitations of the basic scheme is relaxed. This scheme has the property that the attribute vector and result value are both hidden to the user.

A. Specifications

The arithmetic function that this scheme can realize as access structure is of the following form:

$$f(\mathbf{x}) = \sum_{i=1}^{|S|} (a_i \prod_{j \in P_i} x_j^{u_{i_j}}) \tag{20}$$

where P_i, S and a_i are defined as previous. Since $deg(f(x)) \leq k$, it holds that $\sum_{j \in P_i} u_{i_j} \leq k$ for all i. In this scheme, $n \geq k$ and the constraint $P_i \cap P_j = \emptyset$ is relaxed, though a more slight constraint on P_i should be met that is characterized in details in Appendix A. Moreover, the value of attribute vector as well as the result value are hidden to the user.

B. Ciphertext Policy Attribute Based Encryption Scheme

This version of proposed CP-ABE scheme, is similar to the basic scheme, introduced in Sec. III-B, with the following modifications in the quadruple (Setup, KeyGen, Enc, Dec).

Setup($\lambda, n, 1^k$). The only changes are on the public key and master secret key which are as below.

$$PK = \{[g^{t_1}, g^{t_2}, \cdots, g^{t_k}], [g^{\frac{1}{s_1}}, g^{\frac{1}{s_2}}, \cdots, g^{\frac{1}{s_n}}],$$

$$\begin{bmatrix} g^{\frac{t_1}{s_1}} & g^{\frac{t_2}{s_1}} & \cdots & g^{\frac{t_k}{s_1}} \\ g^{\frac{t_1}{s_2}} & g^{\frac{t_2}{s_2}} & \cdots & g^{\frac{t_k}{s_2}} \\ \vdots & \vdots & \ddots & \vdots \\ g^{\frac{t_1}{s_n}} & g^{\frac{t_2}{s_n}} & \cdots & g^{\frac{t_k}{s_n}} \end{bmatrix}\}$$

$$MSK = \{[t_1, t_2, \ldots, t_k], [s_1, s_2, \ldots, s_n]\} \tag{21}$$

KeyGen(MSK). The secret keys of users are generated similar to the basic scheme (6).

Enc(PK, f, m). The encryptor chooses random numbers $r_j^{(i)} \in Z_q$, $j = 1, \ldots, |P_i|$ and $i = 1, \ldots, |S|$ in a way that for

Algorithm 1: Computing I_{P_i}

```
Input: $P_i$, $u_j, j \in P_i$ and $I^0_{P_i}$
Output: $I_{P_i}$
$B \leftarrow I^0_{P_i}$;
$T \leftarrow \{1, \ldots, k\} \setminus P_i$;
for $j \leftarrow 1$ to $|P_i|$ do
    for $k \leftarrow 1$ to $u_j - 1$ do
        select $i' \in T$;
        $B \leftarrow e(B, g^{\frac{t_{i'}}{s_j}})$;
        $T \leftarrow T \setminus \{i'\}$;
while $T \neq \emptyset$ do
    select $i' \in T$;
    $B \leftarrow e(B, g^{t_{i'}})$;
    $T \leftarrow T \setminus \{i'\}$;
return $B$;
```

all i it holds $\prod_{j \in P_i} r^{(i)}_j = R$. The ciphertext is then computed according to the following equation.

$$\begin{aligned} Ctx &= [f,\ C_0 = m.(g_k^{\prod_{v=1}^{k} t_v})^{y.R}, Check = g_k^y, \\ &\qquad \mathbf{C}_{P_1}, \mathbf{C}_{P_2}, \cdots, \mathbf{C}_{P_{|S|}}] \end{aligned} \tag{22}$$

where

$$\mathbf{C}_{P_i} = [C^{(i)}_1, C^{(i)}_2, \ldots, C^{(i)}_{|P_i|}], \quad \forall P_i \in S \tag{23}$$

and $C^{(i)}_j = g^{\frac{r^{(i)}_j t_{i_j}}{s_{i_j}}}$.

Dec(Ctx, PK, SK): Only the computation of $I_{P_i}, i = 1, \ldots, |S|$ changes in decryption algorithm. It is more convenient to present the way of this computation in an algorithm format rather than the closed-form expression. To compute I_{P_i}, The decryptor first computes $I^0_{P_i}$ based on the given ciphertext.

$$I^0_{P_i} = e(C^{(i)}_1, C^{(i)}_2, \ldots, C^{(i)}_{|P_i|}) \tag{24}$$

Then, It runs Algorithm 1 to get I_{P_i}. Based on Algorithm 1, it would be computed as $I_{P_i} = g_k^{\frac{R}{\prod_{j \in P_i} s_j^{u_j}} \prod_{v=1}^{k} t_v}$. The rest of the Dec algorithm is exactly similar to the basic scheme.

We bring an example here to show how Algorithm 1 works. Suppose that $k = 7$ and the i^{th} monomial of $f(x)$ is $x_1^3 x_2^2 x_4$. So, $P_i = \{1, 2, 4\}$ and $u_1 = 3, u_2 = 2$ and $u_4 = 1$. Algorithm 1 computes I_{P_i} as follows.

$$\begin{aligned} I_{P_i} &= e(C^{(i)}_1, C^{(i)}_2, C^{(i)}_4, g^{\frac{t_3}{s_1}}, g^{\frac{t_5}{s_1}}, g^{\frac{t_6}{s_2}}, g^{t_7}) \\ &= (g^{\frac{r^{(i)}_1 t_1}{s_1}}, g^{\frac{r^{(i)}_2 t_2}{s_2}}, g^{\frac{r^{(i)}_4 t_4}{s_4}}, g^{\frac{t_3}{s_1}}, g^{\frac{t_5}{s_1}}, g^{\frac{t_6}{s_2}}, g^{t_7}) \\ &= g_7^{\frac{r^{(i)}_1 r^{(i)}_2 r^{(i)}_4}{s_1^3 s_2^2 s_4} \prod_{v=1}^{k} t_v} \\ &= g_7^{\frac{R}{s_1^3 s_2^2 s_4} \prod_{v=1}^{k} t_v} \end{aligned} \tag{25}$$

C. Security proof

The security proof of this scheme is completely similar to the security proof of the basic scheme brought in Sec. III-C.

V. The Improved Scheme II, Disclosed attributes, Hidden result

In the two previous schemes the attribute vector is hidden to its owner. Depending on the application, such a property may be desired or not. In this section, we present a variant of the proposed scheme in which the values of the attributes are known to the attribute-owner.

A. Limitations and specifications

The function $f(\mathbf{x})$ which can be supported by this scheme as access structure is the same as that of the improved scheme I, characterized in Sec. IV-A. The only difference is that, the value of result y is hidden to the user prior to the decryption, but the attribute vector is known to its owner. However, the eligible user who can successfully decrypt the ciphertext can obtain the value of result after decryption.

This scheme is based on the $2k$-multilinear map where $deg(f) \leq k$. This increases the size of public parameters and secret keys as well as the computational complexity of decryption algorithm.

B. The scheme

In this section we highlight only those part of algorithms (Setup, KeyGen, Enc, Dec) that have changed comparing to the improved scheme I, in Sec. IV-B.

Setup$(\lambda, n, 1^k)$. This algorithm outputs a number of $2k$ groups $\mathbf{G}_1, \ldots, \mathbf{G}_{2k}$ over which the multilinear map $\{e_{i,j} : i, j \in \{1, \ldots, 2k - 1\}\}$ is defined. This means that the public parameters are twice of the previous schemes but the Public key and Master Secret Key are the same as the previous one.

KeyGen(MSK). In this variant, the secret key, SK, is generated as below:

$$\begin{aligned} SK &= \begin{bmatrix} sk_{11} & sk_{12} & \cdots & sk_{1n} \\ sk_{21} & sk_{22} & \cdots & sk_{2n} \end{bmatrix} \\ &= \begin{bmatrix} s_1 x_1 (x_1)^{\alpha} & s_2 x_2 (x_2)^{\alpha} & \cdots & s_n x_n (x_n)^{\alpha} \\ g^{x_1^{-\alpha}} & g^{x_2^{-\alpha}} & \cdots & g^{x_n^{-\alpha}} \end{bmatrix} \end{aligned} \tag{26}$$

where α is a randomly-chosen user-specific parameter.

Enc(PK, f, m). The only change in this algorithm is as follows.

$$\begin{aligned} C_0 &= m \cdot (g_{2k}^{\prod_{v=1}^{k} t_v})^{y \cdot R} \\ Check &= g_{2k}^{y} \end{aligned} \tag{27}$$

$\mathsf{Dec}(Ctx, PK, SK)$. This algorithm changes as follows. The receiver first computes $I_{P_i}, i = 1, \ldots, |S|$ according to Algorithm 1. Then, it computes $J_{P_i}, i = 1, \ldots, |S|$ as follows.

$$\begin{aligned} J_{P_i} &= e(\underbrace{sk_{2,j_1}, \ldots, sk_{2,j_1}}_{u_{j_1} \text{ times}}, \ldots, \underbrace{sk_{2,j_{|P_i|}}, \ldots, sk_{2,j_{|P_i|}}}_{u_{j_{|P_i|}} \text{ times}}, \underbrace{g, \ldots, g}_{k-k_i \text{ times}}) \\ &= g^{\prod_{j \in P_i} x_j^{-u_j \alpha}} \end{aligned} \tag{28}$$

where $P_i = \{j_1, \ldots, j_{|P_i|}\}$, and $k_i = \sum_{j \in |P_i|} u_j$. Finally, $Mask'$ is computed according to the following.

$$\begin{aligned} Mask' &= \prod_{i=1}^{|S|} e_{k,k}(I_{P_i}^{a_i \prod_{j \in P_i} sk_{1,j}^{u_j}}, J_{P_i}) \\ &= \prod_{i=1}^{|S|} e_{k,k}(g_k^{R \cdot a_i \prod_{j \in P_i} x_j^{u_j} (x_j)^{u_j \alpha} \prod_{v=1}^{k} t_v}, g_k^{\prod_{j \in P_i} x_j^{-u_j \alpha}}) \\ &= \prod_{i=1}^{|S|} g_{2k}^{R \cdot a_i \prod_{j \in P_i} x_j^{u_j} \prod_{v=1}^{k} t_v} \\ &= g_{2k}^{R \cdot \sum_{i=1}^{|S|} a_i \prod_{j \in P_i} x_j^{u_j} \prod_{v=1}^{k} t_v} \\ &= g_{2k}^{R \cdot f(\mathbf{x}) \prod_{v=1}^{k} t_v} \end{aligned} \tag{29}$$

C. Security Proof

The security proof of this scheme is similar to the security proof of the basic scheme brought in III-C, tough with some modifications. The adaptive security game for the improved scheme II, is as follows.

1) The challenger C receives the $(2k-1)$-DsDDH parameters as follows.

$$\{\mathbf{G}_1, \mathbf{G}_2, ..., \mathbf{G}_{2k}, g_1, g_2, ..., g_{2k}, g^x, g_{2k}^y, g_{2k}^z\}$$

where $g_1 = g$. It must distinguish if $z = x \cdot y$ or it is a random element of Z_p.

2) The challenger C randomly chooses $(k-1)$ values $t_1, t_2, ..., t_{k-1}$ and computes $g^{t_i}, i = 1, \ldots, k-1$. Then, it sets $g^{t_k} = g^{x \prod_{i=1}^{k-1} r_i^{-1}}$. The challenger also selects random values s_j and computes $g^{\frac{t_i}{s_j}}$, $i = 1, \ldots, k, j = 1, \ldots, n$. Then challenger runs Setup algorithm for simulating the public parameters PP and public keys PK. Then, challenger sends the public parameters to the attacker $\mathcal{A}$ as below.

$$PP = \{\mathbf{G}_1, \mathbf{G}_2, ..., \mathbf{G}_{2k}, g = g_1, g_2, ..., g_{2k}\}$$

$$PK = \{[g^{t_1}, g^{t_2}, \cdots, g^{t_k}], [g^{\frac{1}{s_1}}, g^{\frac{1}{s_2}}, \cdots, g^{\frac{1}{s_n}}], \begin{bmatrix} g^{\frac{t_1}{s_1}} & g^{\frac{t_2}{s_1}} & \cdots & g^{\frac{t_k}{s_1}} \\ g^{\frac{t_1}{s_2}} & g^{\frac{t_2}{s_2}} & \cdots & g^{\frac{t_k}{s_2}} \\ \vdots & \vdots & \ddots & \vdots \\ g^{\frac{t_1}{s_n}} & g^{\frac{t_2}{s_n}} & \cdots & g^{\frac{t_k}{s_n}} \end{bmatrix}\} \tag{30}$$

3) The attacker $\mathcal{A}$ requests the secret keys SK corresponding to his selected attribute vector $\mathbf{x} = [x_1, x_2, ..., x_n]$ from the challenger. Having received the public parameters and public keys, the challenger C chooses the random number α and computes $[s_i.x_i.(x_i)^{\alpha}, g^{x_i^{-\alpha}}]; i = 1, \ldots, n$ and sends them to $\mathcal{A}$ as secret keys. This item can repeat adaptively to simulate collusion of users.
4) The attacker chooses the challenge function $f(\mathbf{x})$. It also chooses two messages m_0 and m_1. Then, the attacker sends $f(\mathbf{x}), m_0, m_1$ to the challenger.
5) The challenger randomly chooses one of the two messages m_0 and m_1. Then the challenger C runs the algorithm Enc to simulate the ciphertext of m_b where $b \in_r \{0, 1\}$. The ciphertext Ctx is simulated as follows and sends it to the attacker.

$$\begin{aligned} Ctx = \; & [f, \; C_0 = m \cdot (g_{2k}^z)^R, Check = g_{2k}^y, \\ & \mathbf{C}_1, \mathbf{C}_{P_2}, \ldots, \mathbf{C}_{P_{|S|}}] \end{aligned} \tag{31}$$

6) The attacker $\mathcal{A}$ can request more secret keys for adaptively chosen attribute vectors, after receiving Ctx. The challenger responses to these requests similar to Step 3.
7) the attacker sends the value of guessed b' to the challenger.

The probability of success of challenger to distinguish the $(2k-1)$-DsDDH problem is as follows.

$$\begin{aligned} Pr[C_{(2k-1)-\mathrm{DsDDH}} = \text{success}] &= \frac{1}{2} \cdot (\frac{1}{2} + \epsilon) + \frac{1}{2} \cdot \frac{1}{2} \\ &= \frac{1}{2} + \frac{\epsilon}{2} \end{aligned} \tag{32}$$

which is greater than $\frac{1}{2}$, since $\frac{\epsilon}{2}$ has been considered non-negligible. So, we conclude that the attacker $\mathcal{A}$ does not exist because the $(2k-1)$-DsDDH problem is hard.

VI. Comparison with Boneh's scheme

Comparing to the only ABE scheme for arithmetic functions, proposed by Boneh [21], the proposed schemes in this paper have several advantages, which are listed in the following.

1) The proposed schemes are CP-ABE which is more flexible than KP-ABE.
2) The value of result in our scheme is arbitrary. But, Boneh's scheme just supports $y = 0$, though it can be modified to work for any arbitrary result.
3) The proposed schemes can support both hidden or disclosed attribute values. However, in Boneh's scheme the values of attributes can not be kept hidden, so this scheme can not be used as predicate encryption.
4) In Boneh's scheme, the values of attributes must be in $[-p, p]$, where p is less than the group order q, for **Mult** gates. However, our scheme does not put any constraint on the values of attributes.
5) Despite Boneh's scheme which has selective security, the proposed schemes have adaptive security which is stronger.

6) Despite [21], our scheme can support the exponentiation gate, tough it seems that this feature can be added to Boneh's scheme.
7) since [21] is a lattice-based scheme, the copmutational complexity and key sizes of the keys are larger than our scheme.

However, the disadvantage of our scheme comparing to Boneh's scheme is that our scheme is not post-quantum.

VII. Conclusion

We proposed some CP-ABE schemes for arithmetic circuit access structures. The proposed scheme relies on multilinear maps. We defined the new concept of hidden results ABE which refers to the ABE scheme for arithmetic functions in which the result value for the function is unknown.

In the first proposed scheme, the attribute vector and the result value are hidden to the users. It relies on a k-multilinear map and supports a number of $n = k$ attributes. The improved scheme I works for any number of $n \geq k$ attributes, conditioned that the degree of the function is at most k. In this scheme, the attribute vector and the result value are hidden to the users, too. Finally, we proposed the improved scheme II, where the attribute vector is not hidden to the users and the result value, would become disclosed to users who can decrypt the ciphertext. However, the order of groups must be greater the first two schemes.

We proved that these schemes are adaptively secure under a new defined hardness assumption, called k-Distance Decisional Diffie-Hellman problem, which is at least as hard as the well known k-multilinear decisional Diffie-Hellman problem. Finally, we compared our schemes with Boneh et al.'s scheme and described the advantages of ours.

References

[1] Sahai, A. and Waters, B., 2005, May. Fuzzy identity-based encryption. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 457-473). Springer, Berlin, Heidelberg.

[2] Goyal, V., Pandey, O., Sahai, A. and Waters, B., 2006, October. Attribute-based encryption for fine-grained access control of encrypted data. In Proceedings of the 13th ACM conference on Computer and communications security (pp. 89-98).

[3] Bethencourt, J., Sahai, A. and Waters, B., 2007, May. Ciphertext-policy attribute-based encryption. In 2007 IEEE symposium on security and privacy (SP'07) (pp. 321-334). IEEE.

[4] Waters, B., 2011, March. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In International Workshop on Public Key Cryptography (pp. 53-70). Springer, Berlin, Heidelberg.

[5] Ostrovsky, R., Sahai, A. and Waters, B., 2007, October. Attribute-based encryption with non-monotonic access structures. In Proceedings of the 14th ACM conference on Computer and communications security (pp. 195-203).

[6] Green, M., Hohenberger, S. and Waters, B., 2011, August. Outsourcing the decryption of abe ciphertexts. In USENIX security symposium (Vol. 2011, No. 3).

[7] Lewko, A., Sahai, A. and Waters, B., 2010, May. Revocation systems with very small private keys. In 2010 IEEE Symposium on Security and Privacy (pp. 273-285). IEEE.

[8] Hur, J. and Noh, D.K., 2010. Attribute-based access control with efficient revocation in data outsourcing systems. IEEE Transactions on Parallel and Distributed Systems, 22(7), pp.1214-1221.

[9] Chase, M., 2007, February. Multi-authority attribute based encryption. In Theory of cryptography conference (pp. 515-534). Springer, Berlin, Heidelberg.

[10] Attrapadung, N. and Imai, H., 2009, June. Dual-policy attribute based encryption. In International Conference on Applied Cryptography and Network Security (pp. 168-185). Springer, Berlin, Heidelberg.

[11] Zou, X., 2013. A hierarchical attribute-based encryption scheme. Wuhan University Journal of Natural Sciences, 18(3), pp.259-264.

[12] Attrapadung, N., Libert, B. and De Panafieu, E., 2011, March. Expressive key-policy attribute-based encryption with constant-size ciphertexts. In International Workshop on Public Key Cryptography (pp. 90-108). Springer, Berlin, Heidelberg.

[13] Li, J., Lin, X., Zhang, Y. and Han, J., 2016. KSF-OABE: Outsourced attribute-based encryption with keyword search function for cloud storage. IEEE Transactions on Services Computing, 10(5), pp.715-725.

[14] Koppula, V. and Waters, B., 2019, August. Realizing chosen ciphertext security generically in attribute-based encryption and predicate encryption. In Annual International Cryptology Conference (pp. 671-700). Springer, Cham.

[15] Brakerski, Z. and Vaikuntanathan, V., 2016, August. Circuit-ABE from LWE: unbounded attributes and semi-adaptive security. In Annual International Cryptology Conference (pp. 363-384). Springer, Berlin, Heidelberg.

[16] Garg, S., Gentry, C., Halevi, S., Sahai, A. and Waters, B., 2013, August. Attribute-based encryption for circuits from multilinear maps. In Annual Cryptology Conference (pp. 479-499). Springer, Berlin, Heidelberg.

[17] Agrawal, S., Boyen, X., Vaikuntanathan, V., Voulgaris, P. and Wee, H., 2011. Fuzzy Identity Based Encryption from Lattices. IACR Cryptol. ePrint Arch., 2011, p.414.

[18] Boyen, X., 2013, March. Attribute-based functional encryption on lattices. In Theory of Cryptography Conference (pp. 122-142). Springer, Berlin, Heidelberg.

[19] Zhang, J. and Zhang, Z., 2011, November. A ciphertext policy attribute-based encryption scheme without pairings. In International Conference on Information Security and Cryptology (pp. 324-340). Springer, Berlin, Heidelberg.

[20] Gorbunov, S., Vaikuntanathan, V. and Wee, H., 2015. Attribute-based encryption for circuits. Journal of the ACM (JACM), 62(6), pp.1-33.

[21] Boneh, D., Gentry, C., Gorbunov, S., Halevi, S., Nikolaenko, V., Segev, G., Vaikuntanathan, V. and Vinayagamurthy, D., 2014, May. Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 533-556). Springer, Berlin, Heidelberg.

[22] Zhu, W., Yu, J., Wang, T., Zhang, P. and Xie, W., 2014. Efficient attribute-based encryption from R-LWE. Chin. J. Electron, 23(4), pp.778-782.

[23] Fun, T.S. and Samsudin, A., 2015, August. Lattice ciphertext-policy attribute-based encryption from ring-LWE. In 2015 International Symposium on Technology Management and Emerging Technologies (ISTMET) (pp. 258-262). IEEE.

[24] Chen, Z., Zhang, P., Zhang, F. and Huang, J., 2017. Ciphertext policy attribute-based encryption supporting unbounded attribute space from R-LWE. TIIS, 11(4), pp.2292-2309.

[25] Tsabary, R., 2019, August. Fully secure attribute-based encryption for t-CNF from LWE. In Annual International Cryptology Conference (pp. 62-85). Springer, Cham.

[26] Katz, J., Sahai, A. and Waters, B., 2008, April. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In annual international conference on the theory and applications of cryptographic techniques (pp. 146-162). Springer, Berlin, Heidelberg.

[27] Boneh, D., Sahai, A. and Waters, B., 2011, March. Functional encryption: Definitions and challenges. In Theory of Cryptography Conference (pp. 253-273). Springer, Berlin, Heidelberg.

[28] Belguith, S., Kaaniche, N., Laurent, M., Jemai, A. and Attia, R., 2018. Phoabe: Securely outsourcing multi-authority attribute based encryption with policy hidden for cloud assisted IoT. Computer Networks, 133, pp.141-156.

[29] Xiong, H., Zhao, Y., Peng, L., Zhang, H. and Yeh, K.H., 2019. Partially policy-hidden attribute-based broadcast encryption with secure delegation in edge computing. Future Generation Computer Systems, 97, pp.453-461.

[30] Pohlig, S. and Hellman, M., 1978. An improved algorithm for computing logarithms over GF (p) and its cryptographic significance (Corresp.). IEEE Transactions on information Theory, 24(1), pp.106-110.