IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 13 August 2023, accepted 27 August 2023, date of publication 7 September 2023, date of current version 19 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3313098

RESEARCH ARTICLE

An Efficient Post-Quantum Attribute-Based Encryption Scheme Based on Rank Metric Codes for Cloud Computing

VAHID YOUSEFIPOOR[1] **AND TARANEH EGHLIDOS**[2]
[1]Department of Electrical Engineering, Sharif University of Technology, Tehran 14588-89694, Iran
[2]Electronics Research Institute, Sharif University of Technology, Tehran 14588-89694, Iran

Corresponding author: Taraneh Eghlidos (teghlidos@sharif.edu)

ABSTRACT Attribute-based encryption is a valuable technique for ensuring data privacy and confidentiality in the realm of cloud computing. Using this cryptographic primitive, the data owner can securely store and share data within the cloud environment. On the other hand, in recent years, extensive advances have been made in quantum processors, which have raised hopes of solving certain mathematical problems includes factoring integers and computing discrete logarithms of large numbers. The advent of quantum computers has posed a significant security threat to existing cryptographic protocols. The existing post-quantum attribute-based encryption schemes have not satisfied the essential features such as verifiability, user privacy and user revocability, simultaneously. In this paper, we present the first secure, practical and post-quantum attribute-based encryption scheme based on rank metric codes. Our scheme enjoys all mentioned features due to utilization of low rank parity check codes. The proposed scheme provides security against chosen plaintext attacks in the standard model, as well as resistance against reaction attacks as a kind of chosen ciphertext attacks. Moreover, at the 256-bit security level, the key size is about 16.5 KB, with an execution time of around 31.2 ms on a desktop. Our implementation results confirm that the proposed scheme is more efficient than the existing post-quantum and classical schemes.

INDEX TERMS Attribute based encryption, completeness of the search results, rank metric codes, user privacy, user revocation.

I. INTRODUCTION

In a computer network using public key encryption schemes, it is essential to exchange the public key of each user for identification. Therefore, the existence of an authority becomes essential to generate a pair of public and private keys for applicants or revoke it, if necessary [1]. For this purpose, the public key infrastructure has been created as a framework for managing certificates and verifying user identities to provide a secure environment. To avoid the need for generating a certificate on the public key, each user's public key can be generated based on their identity. This idea was first proposed by Shamir in 1984 and called identity-based encryption (IBE) [2]. As a trusted authority issues each user's private key based on the their ID, users are not required to extract public key from the certificate issued by a trusted authority. The first practical scheme for IBE was proposed by Boneh and Franklin in 2001 [3]. The salient advantage of IBE is that the private key of the trusted authority is removed after registering all users in the system. Afterwards, a key distribution center will no longer be needed [4].

Inspired by IBE, the first attribute-based encryption (ABE) scheme was developed by Sahai and Waters [5]. They called their scheme fuzzy ID-based encryption. In the original scheme proposed by Sahai and Waters, the system policy for decryption is the proximity of two sets of features. Despite its simplicity, this policy may need to be more convenient [6]. Subsequent research determined how the policy was defined [7]. There are generally two types of policies in ABE: key policy ABE (KP-ABE) [8] and ciphertext policy ABE

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

(CP-ABE) [9]. In KP-ABE, the private key of the recipient of the message is defined based on the desired access structure, and the sender encrypts the message using the intended attributes. In CP-ABE, the reverse happens, the data sender encrypts the data using its desired access structure, and the recipient's private key is generated based on his attributes.

The utilization of cloud servers to enhance computing speed, expand data storage capabilities, and reduce hardware maintenance costs, has been increasingly considered in recent years [10], [11]. Data is usually encrypted when stored in the cloud, because users cannot trust the cloud server completely. Therefore, one of the significant problems in this regard is how to securely share data based on a policy between multiple users. A convenient solution to this problem is provided by ABE.

In recent years, features such as the ability to revoke user access [12] and verify the outsourced computation results [13] have been added to ABE. These schemes are based on elliptic curve pairing, which reduces the efficiency. This problem has already been solved in [14] and [15], and the efficiency is significantly increased.

Nowadays, tremendous advances have been made in quantum processors. Using Shor's quantum algorithm [16] and assuming to build strong enough quantum processors, the security of many existing cryptographic algorithms such as RSA and DSA is compromised. To encounter the threats of quantum processors, the National Institute of Standards and Technology (NIST) launched a process to standardize asymmetric encryption, key exchange, and digital signature schemes [17].

Many ABE schemes, which rely on number theoretic problems, have been proposed to enhance data security in cloud computing environments [18], [19]. Unfortunately, none of them are secure against quantum computers. Recently, ABE schemes based on lattices have been proposed to withstand the threats of quantum processors [20], [21]. However, these schemes have large key lengths and need to be more efficient.

We have proposed a post-quantum ABE scheme based on rank metric codes, which is semantically secure and efficient in terms of key length and encryption/decryption time complexity according to the number of attributes. In recent years, rank metric codes have gained significant attention as a superior and more efficient alternative to Hamming metric codes in cryptography applications [22], [23], [24]. The proper key-length and good performance of rank metric codes motivate us to use them for designing ABE. Our proposed scheme is based on CP-ABE because it makes sense for the sender to specify the access structure in practical applications for data sharing. The access structure in the proposed scheme is based on Bloom filter [25]. Bloom filter is a hash-based data structure that checks an element's membership in a set. The Bloom filter generation and membership verification processes are described in Section III-II. In our approach, we have employed low rank parity check (LRPC) codes, which offer the advantage of fast decoding algorithms and requires minimal memory to store the parity check matrix.

The proposed ABE schemes has several advantages, including:

- **Post-quantum security in standard model:** We have proved the semantic security of our scheme by assuming that it is challenging to solve the ideal rank syndrome decoding problem for LRPC codes and LRPC indistinguishability. So far, neither classical nor quantum algorithm has been proposed to solve this problems.
- **User revocation:** In the proposed scheme, whenever the user access is revoked, all cloud data matching the user's attributes is re-encrypted so that the revoked user cannot access it.
- **Completeness**: After receiving the data from the cloud, the user can quickly verify whether the whole data stored in the cloud has been searched.
- **User privacy**: When a user submits a data search request to the cloud server, it is encrypted using the user's key and then transmitted to the cloud server. Therefore, the cloud server conducts the search for the requested data without having access to the user's attributes.
- **Efficient implementation and appropriate key length**: Almost all calculations of the proposed scheme are based on linear operations on LRPC codes or applying hash functions. Therefore, the proposed scheme is efficient compared to the existing post-quantum and classical ones. The length of the key of the proposed scheme for the security level of 256 bits is about 16.5 KB, which is acceptable according to the latest technology advancements and enables the use of the proposed scheme in various applications.

The rest of this paper is organized as follows. Section II provides problem statements including ABE, user revocation and completeness. In Section III, we review some prerequisites for the scheme. We describe the proposed ABE scheme in Section IV. In Section V, we analyze the security of the scheme under chosen plaintext and reaction attacks. The performance of the proposed scheme, in terms of the key length and execution time, is discussed in Section VI. Finally, we conclude the paper in Section VII.

II. PROBLEM STATEMENT

A. ATTRIBUTE BASED ENCRYPTION

Attribute-based encryption consists of three entities, the trusted authority (TA), the sender and the receiver. TA owns the master secret key, which is used to generate users' private keys. It also generates public system parameters and provides users with them. The sender uses public parameters and encrypts the message m under the access structure x to generate the encrypted message, ct_x. The receiver decrypts ct_x, using the private key sk_y, as a function of the attribute y, received from TA. For correct decryption, x and y need to match. We remind the readers that in the CP-ABE, the sender encrypts the message using its desired access structure, x, and the receiver's private key, y, is generated based on his attributes. Conversely, In KP-ABE, the private key of the recipient of the message, y, is defined based on the desired access structure and the sender encrypts the message using the intended set of attributes, x [26]. As mentioned in the previous section, we focus on CP-ABE, which generally consists of four main algorithms, Setup, KeyGen, Enc, Dec as follows [27].

- Setup: A probabilistic polynomial time (PPT) algorithm run by TA. The algorithm input is the security parameter, and its output is the master secret key along with the public parameters. The trusted authority keeps a master secret key confidential and publishes the public parameters.
- KeyGen: A PPT algorithm run by TA. Using the set of receiver's attributes and the master secret key, TA generates and delivers the private key, sk_y, to the receiver.
- Enc: A PPT algorithm run by the sender. The sender encrypts a message m under an access structure x and sends the ciphertext to the receiver.
- Dec: Is a deterministic algorithm, executed by the receiver. The receiver receives the ciphertext, ct_x, and intends to decrypt it using the private key, sk_y. If the private key and the ciphertext match, then the decryption is successful.

In a sense, IBE and ABE are similar. This means that in ABE, there is no need to generate a certificate on the user's public key by the trusted authority. Similarly, the user's private key is made based on his information. However, there are some differences between them. In ABE, the trusted authority generates each user's private key not only based on the user's ID, but also based on a set of user's attributes. These attributes are specified for all users in the setting up of the system. For example, attributes can include ID, age, gender, level of education, etc. Another difference is in the type of users' access control. In ABE, by defining specific policies based on predefined attributes, it is possible to determine which users can decrypt the data. These two key features increase the flexibility of ABE compared to IBE.

B. USER REVOCATION

Preventing access to data by unauthorized users of the system is an essential feature required for an ABE scheme [18]. By establishing this feature, the system administrator can expel from the system any offending user or user whose data access time has expired.

In the scheme defined in [18], this problem is well solved, using re-encryptions. The technique used in [18] has the advantage that there is no need to change the user's key after re-encrypting the data by the cloud server. Guo et al. proposed a blockchain-based scheme for user revocation, which the blockchain cannot independently receive the whole key from the data user [28]. Instead, the keys are produced between the blockchain and the data user using a secure key issuance protocol. Additionally, the scheme uses a group manager to update the group keys of users who have not been recovered and produces re-encryption keys, as well as the decryption cloud server to plan pre-decryption tasks in the cloud. A major drawback of the schemes [18] and [28] is that they are based on different types of Diffie-Hellman problems which are susceptible to being broken by the Shor's quantum algorithm.

TABLE 1. Common symbols used in the manuscript.

Symbol	Description
TA	Trusted authority
DO	Data owner
DU	Data user
CS	Cloud server
$\mathbf{v} = (v_0, \ldots, v_{n-1})$	A vector in $\mathbb{F}_{q^m}^n$
$\mathbf{V}$	a matrix with entries in $\mathbb{F}_{q^m}$
$\lVert\mathbf{v}\rVert$	Rank of $\mathbf{v}$, *see Definition 2*
$\mathbf{b} = (b_1, \ldots, b_m)$	A basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ as a vector space
$\mathsf{Supp}(\mathbf{v})$	Support of $\mathbf{v}$, *see Definition 3*
$rot(\mathbf{v})$	Circulant matrix corresponding to $\mathbf{v}$
$\mathbf{uv}$	The corresponding vector of the polynomial $(\sum_{i=0}^{n-1} u_i X^i)(\sum_{j=0}^{n-1} v_j X^j) \bmod (X^n - 1)$
$F = \langle F_1, \ldots, F_d \rangle$	A vector subspace of $\mathbb{F}_{q^m}$ with dimension d
$F.E$	$\langle F_1, \ldots, F_d, E_1, \ldots, E_r \rangle$
$\mathcal{C}$	A linear code over $\mathbb{F}_{q^m}$
$\mathbf{G}$	The generator matrix of $\mathcal{C}$
$\mathbf{H}$	The parity check matrix of $\mathcal{C}$
PF	Probability of failure of RSR

C. COMPLETENESS OF THE RECEIVED DATA

One of the essential features required for ABE schemes is the completeness of the results received from the cloud server. Users do not fully trust the cloud server. The cloud server may not return all data whose access structure matches the user's attribute set. Therefore, the user must be able to verify the completeness of the received data from the cloud server [29].

A solution to check the correct execution of computations by the cloud server is provided in [13]. Despite the efficiency of the technique used in this scheme, the user cannot check the completeness of the received data. In other words, the cloud server can perform computations correctly but only some part of the stored data. Li et al. proposed a scheme with verifiable outsourced decryption that allows both authorized users and unauthorized users to concurrently verify the accuracy of ciphertext [30]. But the scheme is pairing-based. Therefore, it is neither efficient nor resistant to Shor's algorithm.

III. PRELIMINARIES

In this section, we review some necessary definitions related to rank metric codes and Bloom filter. Table 1 contains common symbols used in the manuscript.

A. RANK METRIC CODES

In this section, we go over a few helpful rank metric linear coding topics. We use bold lower-case and upper-case alphabets to designate vectors and matrices, respectively.

Rank Support Recover (RSR) [31]

```
Inputs: $r$, the dimension of error vector support $E = \langle E_1, \ldots, E_r \rangle$, $F = \langle F_1, \ldots, F_d \rangle$, $\mathbf{s} = (\mathbf{s_1}, \ldots, \mathbf{s_n})$
Outputs: Error vector support $E = \langle E_1, \ldots, E_r \rangle$

1- Compute vector space $S = \langle s_1, \ldots, s_n \rangle$
2- For $i \in \{1, 2, \ldots, d\}$ compute $S_i = F_i^{-1}.S$
3- For $i \in \{1, 2, \ldots, d-1\}$ compute $S_{i,i+1} = S_i \bigcap S_{i+1}$
4- For $i \in \{1, 2, \ldots, d-2\}$ do
5-     $temp = S + F.(S_{i,i+1} \oplus S_{i+1,i+2} \oplus S_{i,i+2})$
6-     If $dim(temp) \leq rd$ then:
7-         $S = S \bigcup temp$
8-     End If
9- End For
10- $E = F_1^{-1}.S \bigcap \ldots \bigcap F_d^{-1}.S$
11- return $E$
```

A vector's weight in the rank metric is indicated by $\|\mathbf{v}\|$ [23]. Recall that $dist(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|$ defines the distance between two vectors, $\mathbf{x}$ and $\mathbf{y}$.

Definition 1 (Rank Metric Over $\mathbb{F}_{q^m}^n$ [22]): Let $\mathbf{b} = (b_1, \ldots, b_m) \in \mathbb{F}_{q^m}$ be a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ as a vector space and $\mathbf{v} = (v_0, \ldots, v_{n-1}) \in \mathbb{F}_{q^m}^n$, where $v_j = \sum_{i=1}^{m} \alpha_{i,j} b_i$. The rank of the related matrix $\left(\alpha_{i,j}\right)_{1 \leq j \leq n}^{1 \leq i \leq m}$, is defined as the rank weight of a vector $\mathbf{v}$, represented by $\|\mathbf{v}\|$.

Definition 2 (Support of the Vector [22]): The support of a vector $\mathbf{v}$ is represented by $\mathsf{Supp}(\mathbf{v})$, and it is a $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^m}$ which is formed by coordinates of $\mathbf{v} = (v_0, \ldots, v_{n-1}) \in \mathbb{F}_{q^m}^n$.

We can deduce from definition 2, that the dimension of the $\mathsf{Supp}(\mathbf{v})$ is $\|\mathbf{v}\|$. The number of $\mathbb{F}_{q^m}$ supports of dimension r is $\left[{m \atop r} \right]_q = \prod_{j=0}^{r-1} \frac{q^m - q^j}{q^r - q^j}$ [31]. If $r \ll m$, $\left[{m \atop r} \right]_q \approx q^{mr}$. Because each support is actually a subspace of $\mathbb{F}_{q^m}$, the number of subspaces with small dimension is exponential in terms of m. This finding is later used in the proposed ABE scheme.

A specific type of multiplication of two vectors in $\mathbb{F}_{q^m}^n$ is defined in [31]. Let $\mathbf{v} = (v_0, \ldots, v_{n-1}) \in \mathbb{F}_{q^m}^n$. The circulant matrix associated with $\mathbf{v}$ is defined as [31]:

$$rot(\mathbf{v}) = \begin{pmatrix} v_0 & v_1 & \ldots & v_{n-1} \\ v_{n-1} & v_0 & \ldots & v_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ v_1 & v_2 & \ldots & v_0 \end{pmatrix} \tag{1}$$

The set of all $n \times n$ circulant matrices over $\mathbb{F}_{q^m}$ is denoted by $\mathcal{M}_n(\mathbb{F}_{q^m})$ [31]. There is an $\mathbb{F}_{q^m}$-algebra isomorphism, such that:

$$\begin{aligned} \varphi : \mathbb{F}_{q^m}[X]/(X^n - 1) &\longrightarrow \mathcal{M}_n(\mathbb{F}_{q^m}) \\ \sum_{j=0}^{n-1} v_j X^j &\mapsto rot(\mathbf{v}) \end{aligned} \tag{2}$$

Now, for two vectors $\mathbf{u}$ and $\mathbf{v}$ the multiplication $\mathbf{vu}$ is defined as the corresponding vector of the polynomial $(\sum_{i=0}^{n-1} v_i X^i)(\sum_{j=0}^{n-1} u_j X^j) \mod (X^n - 1)$ [31]. Also, $\mathbf{vv^{-1}} = \mathbf{1} = (10 \cdots 0)$.

Definition 3 (Linear Code Over $\mathbb{F}_{q^m}$ [23]): A k-dimensional subspace of $\mathbb{F}_{q^m}^n$ is a linear code $\mathcal{C}$ over $\mathbb{F}_{q^m}$ with two parameters n and k. The variables k and n represent the dimension and length of $\mathcal{C}$, respectively. $\mathcal{C}$ is denoted by $[n, k]_{q^m}$.

$\mathcal{C}$ can be demonstrated in two equivalent ways:

- $\mathcal{C} = \{\mathbf{xG} : \mathbf{x} \in \mathbb{F}_{q^m}^k, \mathbf{G} \in \mathbb{F}_{q^m}^{k \times n}\}$, where $\mathbf{G}$ is the generator matrix of $\mathcal{C}$.
- $\mathcal{C} = \{\mathbf{y} \in \mathbb{F}_{q^m}^n : \mathbf{Hy}^T = \mathbf{0}, \mathbf{H} \in \mathbb{F}_{q^m}^{(n-k) \times n}\}$, where $\mathbf{H}$ is the parity check matrix of $\mathcal{C}$.

A $[2n, n]$ linear code $\mathcal{C}$ is said to be double circulant if its generator matrix $\mathbf{G}$ is of the form $(rot(\mathbf{u})|rot(\mathbf{v}))$ where $\mathbf{u}$ and $\mathbf{v}$ are two vectors in $\mathbb{F}_{q^m}^n$ [23]. Double circulant codes need fewer storage than that of usual linear codes. Because, we need only to save $\mathbf{u}$ and $\mathbf{v}$.

LRPC codes, which are rank-metric equivalents of LDPC codes, are a specific type of rank metric linear codes. The following defines LRPC codes:

Definition 4 (LRPC Codes [22]): An LRPC code with parameters d, n, k is a $[n, k]$ linear code over $\mathbb{F}_{q^m}$ in which the elements of its parity check matrix $\mathbf{H} \in \mathbb{F}_{q^m}^{(n-k) \times n}$ form a sub-vector space with maximum dimension d.

The ideal LRPC code is a special class of LRPC codes, which enjoys interesting properties.

Definition 5 (Ideal LRPC Codes [31]): Let $F \subset \mathbb{F}_{q^m}$ with $\dim(F) = d$, $\mathbf{h_1}, \mathbf{h_2} \in \mathbb{F}_{q^m}^n$ with $\mathsf{Supp}(\mathbf{h_1}, \mathbf{h_2}) = F$ and $P \in \mathbb{F}_q[X]$ with $\deg(P) = n$. Let, $\mathbf{H_1} = \left(X^{i-1}\mathbf{h_1} \mod P\right)_{i=1}^{i=n}$ and $\mathbf{H_2} = \left(X^{j-1}\mathbf{h_2} \mod P\right)_{j=1}^{j=n}$. The code $\mathcal{C}$ with the parity check matrix $\mathbf{H} = [\mathbf{H_1}|\mathbf{H_2}]$ is an $[2n, n]$ ideal LRPC code.

The parity check code of $\mathcal{C}$ can be rewritten as $\mathbf{H} = [\mathbf{I}|\mathbf{H'}]$, where $\mathbf{H'} = \left(X^{j-1}\mathbf{h'} \mod P\right)_{j=1}^{j=n}$ and $\mathbf{h'} = \mathbf{h_1^{-1}}\mathbf{h_2}$. The LRPC code decoding algorithm is fast and probabilistic. [31]. The error vector is obtained by first calculating its support and then solving a system of linear equations. The first phase of the LRPC decoding algorithm, known as the Rank support recover (RSR) algorithm [31], which we use in our proposed ABE scheme. Inputs of RSR are as follows:

- r, the dimension of error vector support $E = \langle E_1, \ldots, E_r \rangle$.

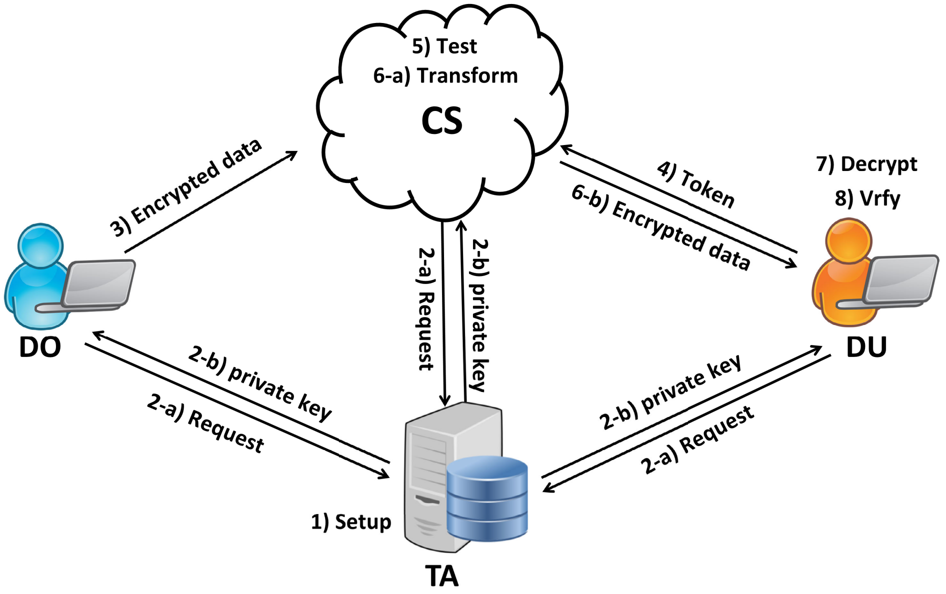

FIGURE 1. System model architecture.

- $F = \langle F_1, \ldots, F_d \rangle \subset \mathbb{F}_{q^m}$, A d-dimensional vector subspace of $\mathbb{F}_{q^m}$.
- $\mathbf{s} = (\mathbf{s_0}, \ldots, \mathbf{s_{n-1}}) \in \mathbb{F}_{q^m}^n$ a vector provided that $\langle \mathbf{s_0}, \ldots, \mathbf{s_{n-1}} \rangle \subset \langle F_1, \ldots, F_d, E_1, \ldots, E_r \rangle = F.E$

The probability of failure (PF) of the RSR is given by [31]:

$$\mathsf{PF} = \max\{q^{-n+2d+2r+3}, q^{-2n+2rd-4}\} \tag{3}$$

For $[2n, n]$ LRPC code, the RSR works properly if $r < n/2$ [22].

B. BLOOM FILTER

A probabilistic data structure called Bloom filter is used to determine if an element is a member of a set [25]. Bloom filters are m'-bit arrays that are initialized to zero and employ collision-free hash functions $H_1, \ldots, H_{k'}$ with the same range $\{0, 1, \ldots, m' - 1\}$. Inserting a member $x \in X$ in the Bloom filter BF causes the positions corresponding to $H_i(x)$ to be set to 1. Where $1 \leq i \leq k'$. As a result, if $x' \notin X$, then at least one position corresponding to $H_i(x')$ is zero. With independent and random hash functions, the false-positive rate is $(1 - (1 - \frac{1}{m'})^{k'n'})^{k'} \approx (1 - e^{-k'n'/m'})^{k'}$. As a result, assuming $k' = (\ln 2)m'/n'$, the least false-positive rate is $(0.6185)^{m'/n'}$ [25]. An m'-bit Bloom filter includes two algorithms:

- $\mathsf{BF} \leftarrow \mathsf{BFGen}(\{H_1, \ldots, H_{k'}\}, X = \{x_1, \ldots, x_{n'}\})$: This algorithm produces an m'-bit Bloom filter BF by hashing a data set X with $\{H_1, \ldots, H_{k'}\}$.
- $\{0, 1\} \leftarrow \mathsf{BFVrfy}(\{H_1, \ldots, H_{k'}\}, \mathsf{BF}, x')$: This algorithm returns 1 if $\mathsf{BF}[H_j(x')] = 1$, for $1 \leq j \leq k'$. Otherwise it returns 0, if at least one location has zero value.

IV. PROPOSED ATTRIBUTE BASED ENCRYPTION BASED ON RANK METRIC CODES

In this section, we introduce the proposed ABE scheme. Then, we describe the proposed scheme in detail, including the entities and interactions between them. We provide an intuition of the techniques and methods used in the scheme to establish the user revocation and completeness of the returned data to the users who satisfy the attributes. We also introduce the scheme and determine the threat model. Finally, we present all algorithms that are used in the scheme.

A. SYSTEM MODEL

Figure 1 shows the system model architecture for the proposed ABE scheme. The proposed scheme consists of four entities as follows.

- **Trusted authority (TA):** This institution is responsible for generating the private keys of the system users. The trusted authority first generates the master secret key and public parameters by executing Setup algorithm, publishes the public parameters, and keeps the master secret key confidential. The sender, receiver and the cloud server receive their private keys by sending a request to TA. The keys are generated by TA using KeyGen algorithm.
- **Data owner (DO) or sender:** The sender encrypts the data under his intended access structure, executes Enc algorithm, and sends it to the cloud server. In addition, the sender outsources auxiliary data along with the ciphertext so that the receiver can use it to verify the completeness of the received data.

- **Data user (DU) or receiver:** This entity generates the token corresponding to his attribute set using the $\mathsf{TokenGen}$ algorithm and sends the token to the cloud server. After receiving the data from the cloud server, DU can access the original data and verify their completeness by executing Dec and Vrfy algorithms, respectively.
- **Cloud server (CS):** This entity stores the sender's data. Whenever a user is revoked from the system, TA sends the user's attributes to CS, and CS re-encrypt only the data matching the attributes using ReEnc algorithm. Also, CS finds the data corresponding to the tokens received from the user by executing Test algorithm. If the data is re-encrypted, CS converts the re-encrypted data to encrypted data and sends it to DU using $\mathsf{Transform}$ algorithm.

B. INTUITION FOR PROPOSED CONSTRUCTION

In this section, we present the techniques used in our proposed scheme. As mentioned earlier, the proposed ABE scheme supports user revocation and completeness of the data received from CS.

- **User revocation**: In the proposed scheme, TA informs CS whenever a user is expelled from the system. Then, CS re-encrypts all the data stored in the cloud using his own key. The main advantage of the proposed technique is that, first, data re-encryption is performed using linear operations on LRPC codes, and as a result, they are fast. Second, users' keys do not need to be changed after each re-encryption. After re-encryption, if the user requests data that matches his attribute set, CS first checks whether the attribute set matches the access structure of the encrypted data in the cloud. Then, CS converts the re-encrypted data into encrypted data using its key pair and returns it to the user.
- Completeness: Whenever the sender wants to encrypt new data, P, under the access structure A, first he generates a Bloom filter including all the data $P_1, \ldots, P_z$ encrypted with this access structure, and the new data. Then, the sender obtains the hash of $P_1, \ldots, P_z$ and P using a hash function, Δ, and XORs them with a Bloom filter. Therefore, the auxiliary data Γ is obtained as $\Gamma = \mathsf{BF}_A^{enc} \oplus \Delta(P) \bigoplus_{i=1}^{i=z} \Delta(P_i)$. The sender sends the result, Γ, along with the ciphertext to CS. Upon receiving the requested authorized ciphertexts from CS, the receiver decrypts them, XORs the hash value of the corresponding plaintexts with Γ to obtain BF_X. If CS has returned all the authorized ciphertexts, BF_X is the Bloom filter corresponding to the whole data. Otherwise, BF_X is a random sequence. In this way, the authorized receiver can ensure receiving all the data shared by the sender in the cloud if the membership of each data in BF_X is confirmed. Otherwise, the authorized receiver conclude that CS has not sent at least one authorized data.

C. DEFINITION OF THE SCHEME

We define the proposed scheme as follows.

Definition 6: A CP-ABE with user revocation and completeness of received data from CS is a tuple $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{ReEnc}, \mathsf{TokenGen}, \mathsf{Test}, \mathsf{Transform}, \mathsf{Dec}, \mathsf{Vrfy})$ of nine probabilistic polynomial time algorithms as follows:

- $(\mathsf{MSK}, \mathsf{PP}) \leftarrow \mathsf{Setup}(1^\lambda)$: On input a security parameter, the algorithm generates the public parameters, PP, and the master secret key, MSK.
- $(\mathbf{sk}_{\mathsf{DU}}, \mathbf{sk}_{\mathsf{CS}}) \leftarrow \mathsf{KeyGen}(\mathsf{MSK}, \mathsf{PP}, \mathsf{Att}, \mathsf{ID}_{\mathsf{CS}})$: On inputs the master secret key, the public parameters, DU's attributes, and the ID of CS, the algorithm generates the private keys of data user and CS, respectively.
- $(\mathsf{Cipher}, \Gamma) \leftarrow \mathsf{Enc}(\mathsf{PP}, A, P \in \mathcal{P})$: On inputs the public parameters PP, the access structure A and a plaintext P, the encryption algorithm generates the ciphertext Cipher and the auxiliary data Γ.
- $\mathsf{Cipher}_{\mathsf{RE}} \leftarrow \mathsf{ReEnc}(\mathsf{PP}, \mathsf{Cipher})$: On inputs the public parameters PP and the ciphertext, the re-encryption algorithm generates the re-encrypted ciphertext of the encrypted outsourced data.
- $\mathbf{tk}_{\mathsf{Att}} \leftarrow \mathsf{TokenGen}(\mathsf{PP}, \mathsf{Att}, \mathbf{sk}_{\mathsf{DU}})$: On inputs the public parameters, data user's attribute set, and the private key of DU, the algorithm generates the token corresponding to the attributes of DU.
- $\{0, 1\} \leftarrow \mathsf{Test}(\mathsf{PP}, \mathsf{Cipher}/\mathsf{Cipher}_{\mathsf{RE}}, \mathbf{tk}_{\mathsf{Att}}, \mathbf{sk}_{\mathsf{CS}})$: On inputs the public parameters, the (re-encrypted) ciphertext, the token corresponding to the users' attributes, and the CS's private key, the algorithm checks whether the token and the (re-encrypted) ciphertext are matched.
- $\mathsf{Cipher} \leftarrow \mathsf{Transform}(\mathsf{PP}, \mathsf{Cipher}_{\mathsf{RE}}, \mathbf{sk}_{\mathsf{CS}})$: On inputs the public parameters, the re-encrypted ciphertext, and the CS's private key, the algorithm generates the corresponding ciphertext on inputs of the public parameters, which made by DO (sender).
- $(P, \Gamma) \leftarrow \mathsf{Dec}(\mathsf{PP}, \mathsf{Cipher}, \mathbf{sk}_{\mathsf{DU}})$: On inputs the public parameters, a ciphertext, and the DU's private key, the algorithm decrypts the ciphertext and generates the corresponding plaintext based on the sender's desired access structure. Also, the algorithm returns auxiliary data.
- $\{0, 1\} \leftarrow \mathsf{Vrfy}(\mathsf{PP}, \Gamma, \{P_i\}_{i=1}^{z'})$: On inputs the public parameters, the auxiliary data and the set of plaintexts encrypted under the same access structure, the algorithm checks the completeness of the returned ciphertexts to DU by CS.

D. THREAT MODEL

For the proposed ABE scheme, the security against chosen plaintext attack (CPA) can be illustrated by the following games between two entities, a PPT adversary $\mathcal{A}$ and a challenger $\mathcal{C}$. The game phases played by $\mathcal{C}$ and $\mathcal{A}$ are as follows [18]:

- **Init**: The adversary specifies an selected access structure A^* to be attacked and declares it to $\mathcal{C}$.
- **Setup**: The challenger executes $\mathsf{Setup}(1^\lambda)$ to generate $(\mathsf{MSK}, \mathsf{PP})$ and sends PP to $\mathcal{A}$.
- **KeyGenQuery**: The adversary can adaptively query on KeyGen algorithm for old private keys and the private keys corresponding to $\mathsf{Att} \notin A^*$. The challenger has two empty tables T_1 and T_2 and initializes two integers $k_1 := 0$ and $k_2 := 0$. For queries on $\{\mathsf{Att}_i \notin A^*\}_{i=1}^{q(\lambda)}$, where $q(.)$ is a polynomial function, the challenger executes $\mathsf{KeyGen}(\mathsf{MSK}, \mathsf{PP}, \mathsf{Att_i}, \mathsf{ID_{CS}})$, returns the corresponding private keys, records them in T_1 and increases k_1 by 1. For queries on old private keys which are corresponding to $\mathsf{Att} \in A^*$, the challenger runs $\mathsf{KeyGen}(\mathsf{MSK}, \mathsf{PP}, \mathsf{Att}, \mathsf{ID_{CS}})$, returns the corresponding private keys, records them in T_2 and increases k_2 by 1.
- **Challenge**: The adversary $\mathcal{A}$, outputs two messages m_0, m_1 with the same length and sends them to $\mathcal{C}$. The challenger chooses uniformly at random a bit b, encrypts m_b under A^*, and returns the corresponding ciphertext to $\mathcal{A}$. The adversary can still adaptively query on KeyGen algorithm.
- **Guess**: The adversary outputs a guess bit b' and he/she successfully win the game if $b' = b$.

Definition 7 (Security Against CPA [18]): Let $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CPA}}$ denote the advantages of the adversary $\mathcal{A}$ of successfully performing CPA against an ABE scheme. We say that the scheme is secure against CPA if for all PPT adversaries $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{CPA}} \leq \epsilon(\lambda)$, where $\epsilon(\lambda)$ is a negligible function.

E. PROPOSED SCHEME

In this section, we describe nine algorithms Setup, KeyGen, Enc, ReEnc, $\mathsf{TokenGen}$, Test, $\mathsf{Transform}$, Dec and Vrfy that are used in the proposed scheme.

$(\mathsf{MSK}, \mathsf{PP}) \leftarrow \mathsf{Setup}(1^\lambda)$: The algorithm selects nine positive integers $m, n, q, r, d_1, d_2, m', n', k'$ such that $d_1, d_2, r \ll m, n$ and $d_1 \times d_2 = d \ll n$. Then it chooses two vectors $\mathbf{z_1}, \mathbf{z_2} \in \mathbb{F}_{q^m}^n$ such that $\|\mathbf{z_1}\| = \|\mathbf{z_2}\| = d_1$ and computes $\mathbf{h} = \mathbf{z_2^{-1}}\mathbf{z_1}$. A number of collision free hash functions $H : \{0, 1\}^* \rightarrow \mathbb{F}_1^n$, $G : \mathcal{E} \rightarrow \mathcal{P}$, $J : \{0, 1\}^* \rightarrow E^n$, $H_1, \ldots, H_{k'} : \{0, 1\}^* \rightarrow \{1, \ldots, m'\}$ and $\Delta : \{0, 1\}^* \rightarrow \{0, 1\}^{m'}$ are used by the Setup algorithm, where $\mathcal{E}$ is the set of all subspaces of $\mathbb{F}_{q^m}$ with fixed dimension of r, $\mathcal{P}$ is the plaintext space and E, F_1 are two subspaces of $\mathbb{F}_{q^m}$ with $dim(E) = r$ and $dim(F_1) = d_2$. The public parameters and master secret key are $\mathsf{PP} = (m, n, q, r, d_1, d_2, m', n', k', H, G, J, H_1, \ldots, H_{k'}, \Delta, \mathbf{h}, E, F_1)$ and $\mathsf{MSK} = (\mathbf{z_1}, \mathbf{z_2})$, respectively.

$(\mathbf{sk}_{\mathsf{DU}}, \mathbf{sk}_{\mathsf{CS}}) \leftarrow \mathsf{KeyGen}(\mathsf{MSK}, \mathsf{PP}, \mathsf{Att}, \mathsf{ID_{CS}})$: This algorithm generates the private keys of DU and CS. The algorithm computes the Bloom filter corresponding to the attributes of DU, $\mathsf{Att} = \{\mathsf{Att}_1, \ldots, \mathsf{Att}_l\}$. For this purpose, the algorithm calls BFGen and computes $\mathsf{BF_{Att}}$. Then, the algorithm computes $\mathbf{s} = H(\mathsf{BF_{Att}})$, $\mathbf{s}' = \mathbf{z_2 s}$ and decode the syndrome $\mathbf{s}' = \mathbf{x_1} + \mathbf{z_1 y_1}$ for $\mathbf{x_1}$ and $\mathbf{y_1}$, such that $\|\mathbf{x_1}\| = \|\mathbf{y_1}\| = d_2$. Note that, $\|\mathbf{z_1}\| = d_1 \ll m, n$, $\|\mathbf{s}'\| = d \ll m$, n and $d_2 \ll n$. Therefore, $\mathbf{s}'$ is the syndrome corresponding to error vector $(\mathbf{x_1}, \mathbf{y_1})$ for the LRPC code whose parity check matrix is $[\mathbf{I}|rot(\mathbf{z_1})]$. Given that $\mathbf{s}' = \mathbf{z_2 s}$ we have:

$$\begin{aligned}\mathbf{s} &= \mathbf{z_2^{-1}}\mathbf{s}'\\ &= \mathbf{z_2^{-1}}\mathbf{x_1} + (\mathbf{z_2^{-1}}\mathbf{z_1})\mathbf{y_1}\\ &= \mathbf{x} + \mathbf{hy}\end{aligned} \tag{4}$$

where, $\mathbf{x} = \mathbf{z_2^{-1}}\mathbf{x_1}$ and $\mathbf{y} = \mathbf{y_1}$. Then, the algorithm computes $\mathbf{s_c} = H(\mathsf{ID_{CS}})$ and in a similar process computes $(\mathbf{x}', \mathbf{y}')$ such that $\mathbf{s_c} = \mathbf{x}' + \mathbf{hy}'$. The private keys of DU and cloud server are $\mathbf{sk}_{\mathsf{DU}} = (\mathbf{x}, \mathbf{y})$ and $\mathbf{sk}_{\mathsf{CS}} = (\mathbf{x}', \mathbf{y}')$, respectively.

$(\mathsf{Cipher}, \Gamma) \leftarrow \mathsf{Enc}(\mathsf{PP}, A, P \in \mathcal{P})$: The algorithm encrypts the plaintext P under the access structure $A = \{A_1, \ldots, A_t\}$. The algorithm calls BFGen and computes the Bloom filter of A, namely BF_A and the Bloom filter of all plaintexts, $P_1, P_2, \ldots, P_z, P$ encrypted under the access structure A, namely BF_A^{enc}. The algorithm selects randomly a subspace $E_1 \in \mathcal{E}$ and four random vectors $(\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}) \in E_1^{3n}$ and $\mathbf{r}' \in E^n$ with the same length. The algorithm computes:

$$\begin{aligned}\mathbf{s_o} &= H(\mathsf{BF}_A)\\ \mathbf{a} &= J(\mathsf{BF}_A)\end{aligned} \tag{5}$$

and

$$\begin{aligned}\mathbf{c_1} &= \mathbf{r_2 h} + \mathbf{r_1}\\ \mathbf{c_2} &= \mathbf{r_2 s_o} + \mathbf{r_3}\\ c &= P \oplus G(E_1)\\ \Lambda &= \mathbf{s_o a} + \mathbf{r}'\\ \Gamma &= \mathsf{BF}_A^{enc} \oplus \Delta(P) \bigoplus_{i=1}^{i=z} \Delta(P_i)\end{aligned} \tag{6}$$

Finally, the algorithm computes the ciphertext as $\mathsf{Cipher} = (\mathbf{c_1}, \mathbf{c_2}, c, \Lambda)$ and the auxiliary data as Γ.

$\mathsf{Cipher_{RE}} \leftarrow \mathsf{ReEnc}(\mathsf{PP}, \mathsf{Cipher})$: The algorithm re-encrypts the ciphertext, $\mathsf{Cipher} = (\mathbf{c_1}, \mathbf{c_2}, c, \Lambda)$ by selecting randomly a subspace $E_2 \in \mathcal{E}$ and three random vectors $(\mathbf{r_4}, \mathbf{r_5}, \mathbf{r_6}) \in E_2^{3n}$ with the same length. The algorithm computes:

$$\begin{aligned}\mathbf{s_c} &= H(\mathsf{ID_{CS}})\\ \mathbf{c_3} &= \mathbf{r_5 h} + \mathbf{r_4}\\ \mathbf{c_4} &= \mathbf{r_5 s_c} + \mathbf{r_6}\\ c' &= c \oplus G(E_2)\end{aligned} \tag{7}$$

The re-encrypted ciphertext is obtained as $\mathsf{Cipher_{RE}} = (\mathbf{c_1}, \mathbf{c_2}, \mathbf{c_3}, \mathbf{c_4}, c', \mathbf{\Lambda})$.

$\mathbf{tk}_{\mathsf{Att}} \leftarrow \mathsf{TokenGen}(\mathsf{PP}, \mathsf{Att}, \mathbf{sk}_{\mathsf{DU}})$: The algorithm generates the token corresponding to the attributes of DU. It calls BFGen algorithm to computes the Bloom filter corresponding to Att. Then, it computes $\mathbf{b} = J(\mathsf{BF_{Att}})$, chooses randomly $\mathbf{r_7} \in E^n$ and generates the token as $\mathbf{tk}_{\mathsf{Att}} = \mathbf{hyb} + \mathbf{r_7}$.

$\{0,1\} \leftarrow \mathsf{Test}(\mathsf{PP}, \mathsf{Cipher}/\mathsf{Cipher}_{\mathsf{RE}}, \mathbf{tk}_{\mathsf{Att}}, \mathbf{sk}_{\mathsf{CS}})$: The algorithm checks if the token and the (re-encrypted) ciphertext match. It computes $\mathbf{u} = \mathbf{\Lambda} - \mathbf{tk}_{\mathsf{Att}}$ and executes $\mathsf{RSR}(\mathbf{u}, r, \langle \mathbf{x}', \mathbf{y}' \rangle)$. If RSR outputs E, then Test returns 1, otherwise returns 0.

$\mathsf{Cipher} \leftarrow \mathsf{Transform}(\mathsf{PP}, \mathsf{Cipher}_{\mathsf{RE}}, \mathbf{sk}_{\mathsf{CS}})$: The algorithm transforms the re-encrypted data into encrypted data, using $\mathbf{sk}_{\mathsf{CS}} = (\mathbf{x}', \mathbf{y}')$ as follows:

$$\mathbf{e}' = \mathbf{c_4} - \mathbf{y}'\mathbf{c_3} \tag{8}$$

Then, the algorithm calls $\mathsf{RSR}(\mathbf{e}', r, \langle \mathbf{x}', \mathbf{y}' \rangle)$ to compute E_2, a subspace of $\mathbb{F}_{q^m}$ with a fixed dimension of r. Next, it computes $c = c' \oplus G(E_2)$ and $\mathsf{Cipher} = (\mathbf{c_1}, \mathbf{c_2}, c, \Lambda)$.

$(P, \Gamma) \leftarrow \mathsf{Dec}(\mathsf{PP}, \mathsf{Cipher}, \mathbf{sk}_{\mathsf{DU}})$: The algorithm utilizes $\mathbf{sk}_{\mathsf{DU}} = (\mathbf{x}, \mathbf{y})$ for decrypting $\mathsf{Cipher} = (\mathbf{c_1}, \mathbf{c_2}, c, \Lambda)$ and computes $\mathbf{e} = \mathbf{c_2} - \mathbf{y}\mathbf{c_1}$. Then, Dec calls $\mathsf{RSR}(\mathbf{e}, r, \langle \mathbf{x}, \mathbf{y} \rangle)$ to compute E_1. Finally, Dec computes the plaintext as $P = c \oplus G(E_1)$ and return (P, Γ) for verification algorithm.

$\{0,1\} \leftarrow \mathsf{Vrfy}(\mathsf{PP}, \Gamma, \{P_i\}_{i=1}^{z'})$: The algorithm checks the completeness of the plaintext, encrypted under the same access structure and then computes all $\{\Delta(P_i)\}_{i=1}^{z'}$ and $\mathsf{BF}_X = \Gamma \bigoplus_{i=1}^{i=z'} \Delta(P_i)$. Then, the algorithm calls $\mathsf{BFVrfy}(\{H_j\}_{j=1}^{j=k'}, \mathsf{BF}_X, P_i)$ for $1 \leqslant i \leqslant z'$. If, at least for one plaintext, BFVrfy returns 0, then Vrfy returns 0, otherwise it returns 1.

F. CORRECTNESS

In this section, we must show that in Transform and Dec algorithms, the RSR algorithm outputs E_2 and E_1, respectively. Also, we must show that the Vrfy algorithm returns 1 if CS returns all data encrypted with the same access structure.

- **correctness of the Transform and Dec algorithms**: First, we simplify the equation 8:

$$\begin{aligned}\mathbf{e}' &= \mathbf{c_4} - \mathbf{y}'\mathbf{c_3}\\ &= (\mathbf{s_c}\mathbf{r_5} + \mathbf{r_6}) - \mathbf{y}'(\mathbf{r_4} + \mathbf{r_5}\mathbf{h})\\ &= \left([\mathbf{x}' + \mathbf{h}\mathbf{y}']\mathbf{r_5} + \mathbf{r_6}\right) - \mathbf{y}'.(\mathbf{r_4} + \mathbf{r_5}\mathbf{h})\\ &= \mathbf{x}'\mathbf{r_5} - \mathbf{y}'\mathbf{r_4} + \mathbf{r_6}\end{aligned} \tag{9}$$

  We denote $\langle \mathbf{x}', \mathbf{y}' \rangle$ with F_2. Because $\mathbf{x}', \mathbf{y}' \in F_2^n$ and $\mathbf{r_4}, \mathbf{r_5}, \mathbf{r_6} \in E_2^n$, implies that $\mathbf{e}' \in (E_2.F_2)^n$. Therefore, according to the RSR algorithm, the output of $\mathsf{RSR}(\mathbf{e}', r, F_2)$ is E_2 with a probability of $1 - \mathsf{PF}$. In a similar way, the output of $\mathsf{RSR}(\langle \mathbf{x}, \mathbf{y} \rangle, r, \mathbf{e})$ is E_1 with a probability of $1 - \mathsf{PF}$. Thus, Dec works properly with a probability of $(1 - \mathsf{PF})^2$.
- **correctness of Vrfy algorithm**: If CS returns all ciphertext encrypted under the same access structure, then we have $\mathsf{BF}_X = \mathsf{BF}_A^{enc}$. Therefore, due to the properties of the Bloom filter, for all $\{P_i\}_{i=1}^{i=z'}$, $\mathsf{BFVrfy}(\{H_j\}_{j=1}^{j=k'}, \mathsf{BF}_X, P_i)$ returns 1 with a high probability. Otherwise, $\mathsf{BF}_X = \mathsf{BF}_A^{enc} \bigoplus_i \Delta(P_i)$ for some $1 \leqslant i \leqslant z'$. Therefore, in this case BF_X is a random string. Thus, there exists an index $1 \leqslant t \leqslant z'$ such that $\mathsf{BFVrfy}(\{H_j\}_{j=1}^{j=k'}, \mathsf{BF}_X, P_t)$ returns 0.

V. SECURITY ANALYSIS

In this section, we analyze the security of the proposed scheme against CPA and reaction attack.

A. HARD PROBLEMS

The security of the proposed scheme against CPA relies on the difficulty of solving two problems associated with LRPC codes: LRPC indistinguishability and the ideal rank syndrome decoding (IRSD) problem. In the following, we review the definition of the problems and provide an overview of the complexity associated with the best algorithms currently available for solving them.

Definition 8 (LRPC Indistinguishability [31]): Given a polynomial $P \in \mathbb{F}_q[X]$ with $\deg(P) = n$ and $\mathbf{h} \in \mathbb{F}_{q^m}^n$. It is hard to distinguish if $\mathbf{h}$ was sampled uniformly at random or as $\mathbf{x}^{-1}\mathbf{y} \mod P$, where $\deg(\mathsf{Supp}(\mathbf{x}, \mathbf{y})) = d \ll n$.

Definition 9 (Ideal Rank Syndrome Decoding Problem [31]): Given $\mathbf{h} \in \mathbb{F}_{q^m}^n$, a polynomial $P \in \mathbb{F}_q[X]$ with $\deg(P) = n$, a syndrome $\boldsymbol{\sigma}$ and a weight $d \ll n, m$, it is hard to sample $\mathbf{x} = (\mathbf{x_1}, \mathbf{x_2}) \in \mathbb{F}_{q^m}^{2n}$ such that $\mathbf{x_1} + \mathbf{x_2}\mathbf{h} = \boldsymbol{\sigma}$ and $\|\mathbf{x}\| = d$.

The best known attack to solve LRPC indistinguishability and IRSD problems has a complexity of $\mathcal{O}\left((nm)^\omega q^{d\lceil \frac{m(n+1)}{2n} \rceil - m}\right)$ [31].

B. SECURITY PROOF UNDER CPA

In this section, we prove that the proposed scheme is semantically secure, or simply CPA-secure. For this purpose, we formalize our proof as a sequence of security games between the adversary and a challenger. We show that the adversary cannot recognize whether the challenger responded to the queries randomly or used the parameters of the proposed scheme. It is noteworthy that if the challenger answers the queries randomly, the adversary's advantage is negligible. So, the adversary's advantage is negligible when the queries are answered using the parameters of the proposed scheme.

Theorem 1: The proposed scheme is secure against chosen plaintext attack under ideal LRPC indistinguishability and IRSD problem.

Proof: Consider a simulator (challenger) and a PPT chosen plaintext adversary, $\mathcal{A}$. The challenger randomly selects one of the following games.

- **Game** G_0: In this game, the simulator follows the proposed ABE scheme in **Setup, KeyGenQuery** and **Challenge** phases of the CPA security game.
- **Game** G_1: In this game, in the **Setup** phase, the simulator produces $\mathbf{z_1}$ and $\mathbf{z_2}$ parameters according to the Setup algorithm but instead of computing $\mathbf{h}$ as $\mathbf{z_2}^{-1}\mathbf{z_1}$, it chooses $\mathbf{h} \in \mathbb{F}_{q^m}^n$ randomly. Therefore, according to Equations 6, the first component of the ciphertext, namely $\mathbf{c_1}$, is changed and randomized in **Challenge** phase. The only difference between game G_1 and G_0 is changing the distribution of vector $\mathbf{h}$ to a random one in $\mathbb{F}_{q^m}^n$. Therefore, according to definition 8, the advantage difference of the adversary in these two games is,

TABLE 2. Parameter set and key length of the scheme.

Security level (in bits)	q	n	m	d_1, d_2	r	sk (in bits)	PF
128	2	193	89	5	6	34354	2^{-128}
192	2	275	137	6	8	75350	2^{-192}
256	2	365	181	7	8	132130	2^{-256}

TABLE 3. The required time in millisecond of the implementation for different security levels.

Security level (in bits)	KeyGen	Enc	TokenGen	Test	Transform	Dec	Vrfy
128	0.90	1.93	0.26	2.07	2.27	2.27	0.94
192	1.87	2.77	0.34	4.12	4.30	4.30	2.11
256	3.02	4.02	0.41	6.64	6.89	6.89	3.31

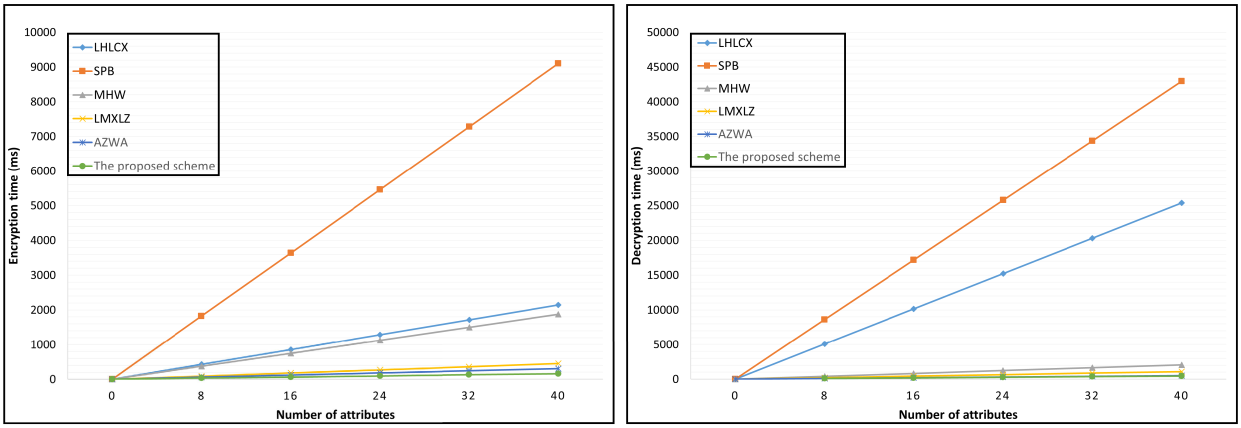

FIGURE 2. Encryption and decryption costs comparison at the 128-bit security level.

TABLE 4. Functional comparision.

Scheme	Post-quantum	Completeness/Checkability	User revocation
LHLCX [13]	×	✓	✓
SPB [32]	×	×	×
MHW [15]	×	×	✓
LMXLZ [33]	✓	×	×
AZWA [34]	✓	×	×
The proposed scheme	✓	✓	✓

at most, equal to solving the LRPC indistinguishability problem. Thus we have:

$$\mathsf{Adv}^{G_0}_{\mathcal{A}} \leq \mathsf{Adv}^{G_1}_{\mathcal{A}} + \mathsf{Adv}^{\mathsf{LRPC}}_{\mathcal{A}} \tag{10}$$

- **Game** G_2: In this game, the simulator answers all KeyGen queries randomly. In this way, for each query, he chooses a random $(\mathbf{x}, \mathbf{y}) \in \mathbb{F}^{2n}_{q^m}$ and returns it to the adversary. Note that in G_1, despite the randomness of $\mathbf{h}$, according to Equation 4 and Definition 9, $(\mathbf{x}, \mathbf{y})$ in **KeyGenQuery** is an instance of IRSD. But in G_2, it is uniformly at random. Therefore, according to Definition 9 and similar to the previous case, the difference between the adversary's advantage in the games G_1 and G_2 is at most equal to the adversary's advantage in solving the IRSD problem. Therefore:

$$\mathsf{Adv}^{G_1}_{\mathcal{A}} \leq \mathsf{Adv}^{G_2}_{\mathcal{A}} + \mathsf{Adv}^{\mathsf{IRSD}}_{\mathcal{A}} \tag{11}$$

- **Game** G_3: In this game, the simulator also picks $\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}, \mathbf{r}'$ uniformly at random in $\mathbb{F}^{n}_{q^m}$. Therefore, according to Equation 6, $\mathbf{c_2}$ and Λ randomized in the **Challenge** phase. While according to Definition 9, these parameters were an IRSD instances in the previous games. This results in the following relation:

$$\mathsf{Adv}^{G_2}_{\mathcal{A}} \leq \mathsf{Adv}^{G_3}_{\mathcal{A}} + \mathsf{Adv}^{\mathsf{IRSD}}_{\mathcal{A}} \tag{12}$$

In G_3, the simulator launches a random scheme. Because it randomly selects all necessary parameters for **Setup**, **KeyGenQuery** and **Challenge** phases. Therefore, $\mathsf{Adv}^{G_3}_{\mathcal{A}}$ is negligible. On the other hand, we assume that LRPC indistinguishability and IRSD are hard problems. Thus, $\mathsf{Adv}^{\mathsf{LRPC}}_{\mathcal{A}}$ and $\mathsf{Adv}^{\mathsf{IRSD}}_{\mathcal{A}}$ are negligible. Thus $\mathsf{Adv}^{G_2}_{\mathcal{A}}$ is negligible too. Therefore, using equations 10-12, we conclude that $\mathsf{Adv}^{G_0}_{\mathcal{A}}$ is negligible. Thus, the proposed scheme is secure against CPA. ■

C. REACTION ATTACK

Another attack that can potentially threaten code-base encryption systems having an iterative decoding algorithm, is the reaction attack [35], regardless of whether it is based on the Hamming metric [35] or the rank [36]. It is noteworthy that, the reaction attack is a kind of chosen ciphertext attack. Usually, in code-based cryptosystems, the decryption algorithm decodes the underlying code. In the reaction attack, the attacker tries to generate ciphertexts that fail to decode. Using these ciphertexts and the corresponding code structure, the adversary can recover the private key, which is usually the parity check matrix or the generator matrix of the underlying code.

Our countermeasures to resist the proposed ABE scheme against reaction attack is to adjust the code parameters so that the probability of decryption failure is negligible. Because the primary condition of a reaction attack is the high probability of decryption failure, If we set this probability on the order of the security level of the underlying scheme, such attack is computationally infeasible. In the proposed scheme, we have used this technique. The main challenge is that usually, by reducing the probability of decryption failure, the dimensions of the code matrix and, thus, the key length increase. According to Equation 3, the probability of decryption failure has an inverse relationship with n and a direct relationship with the parameters r and d. An increase in n leads to a decrease in the probability of decryption failure, but it results in a linear increase in the key length. Therefore, with a slight increase in n, the probability of decryption failure can be significantly reduced. Based on the experiments, we found that the best performance is obtained when n is a prime number. In this case, the efficiency is improved if m is approximately twice as large as n ($m \approx 2n$).

In the next section, we show that the key size of the proposed scheme, and efficiency are suitable for practical applications.

VI. PERFORMANCE ANALYSIS

In this section we first present the set of parameters of the proposed scheme for different security levels. Then, we compare the proposed ABE scheme with some existing protocols. Finally, we provide the implementation results and show that the proposed scheme is more efficient compared to the existing classical and post-quantum ABE ones.

A. PARAMETER SETS

Based on the recommendation of the national institute of standards and technology (NIST), there are 128-, 192-, and 256-bit security levels that are acceptable for encryption [17]. Therefore, the proposed scheme has all three security levels of 128-, 192-, and 256-bits. On the other hand, to strengthen the cryptosystem against reaction attacks, we set the parameters such that the failure probability of Test, Transform and Dec algorithms are equal to $2^{-\lambda}$ for λ-bit security. The parameter set of the proposed scheme is given in Table 2.

Assuming security against reaction attack, Table 2 shows that the key length at the highest security level is about 16.5 KB. In applications where the adversary cannot perform a reaction attack, the key size is reduced by reducing the decryption failure probability. In other words, the data in Table 2 shows the upper bound of the key length.

B. THE RESULTS OF THE EXPERIMENTS

We have implemented the proposed scheme on a desktop computer with Intel®Core™i7-2630QM CPU at 2.00 GHz, 8GB of RAM, and Linux Debian 8. We have shown the performance of the proposed scheme for different security levels in Table 3.

According to Table 3, It takes about 31.2 milliseconds to execute the scheme at the 256-bit security level. Therefore, comparing to the efficiency of the existing number theoretic and post-quantum ABE schemes, the proposed scheme not only ensures security but also provides superior performance in terms of efficiency. The details of the encryption and decryption complexities are compared with those of the existing ABE schemes in Figure 2. Since LHLCX [13] and SPB [32] are based on pairing, they are not efficient in both encryption and decryption compared to other scheme. On the other hand, MHW [15] is a pairing-free scheme that is based on elliptic curve cryptography, which results in improved efficiency compared to the aforementioned schemes. All of the three algorithms are subject to failure against quantum attacks based on Shor's quantum algorithm. LMXLZ [33] and AZWA [34] are based on lattices and they are both efficient, compared to the proposed scheme, and secure against quantum algorithms. However, the proposed scheme offers several additional desirable properties, such as user revocation and the assurance of data completeness when received from the CS, in addition to its post-quantum security. Table 4 compares the functionalities of the proposed scheme with those of the existing protocols.

VII. CONCLUSION

To the best of our knowledge, we have proposed the first attribute-based encryption scheme based on rank metric codes. The proposed scheme is based on low rank parity check code. As a result, the proposed scheme not only exhibits efficiency and a small key size, but also has provable security against chosen keyword attack. Its security is based on the difficulty of the LRPC indistinguishability and ideal rank syndrome decoding problems, for which there has so far been neither classical nor quantum algorithm to solve them in polynomial time. Also, it is secure against reaction attack, which is a kind of chosen cipheretext attack. The proposed scheme guarantees user revocation. The cloud server re-encrypts the whole database to revoke a user's access to the database. The re-encryption process is fast, that does not require to change the user's key. In the proposed scheme, the user can verify the correctness and the completeness of data received from the cloud server using Bloom filter. At the 256-bit security level, the key size and the execution time of the scheme on the desktop are about 16.5KB and 31.2 ms, respectively. Our implementation results show that the efficiency of the proposed scheme is superior compared to the existing classical and post-quantum ones. Note that, we have analyzed the security of the proposed scheme against a specific type of chosen ciphertext attack known as reaction attacks. However, proving security of the proposed scheme against chosen ciphertext attacks remains as a future work.

REFERENCES

[1] S. Choudhury, K. Bhatnagar, and W. Haque, *Public Key Infrastructure Implementation and Design*. Hoboken, NJ, USA: Wiley, 2002.

[2] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. Workshop Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 1984, pp. 47–53.

[3] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 2001, pp. 213–229.

[4] Z. Lu, G. Qu, and Z. Liu, "A survey on recent advances in vehicular network security, trust, and privacy," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 2, pp. 760–776, Feb. 2019.

[5] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Proc. Annu. Int. Conf. Theory Appl. Cryptograph. Techn.* Springer, 2005, pp. 457–473.

[6] J. Li, Y. Zhang, X. Chen, and Y. Xiang, "Secure attribute-based data sharing for resource-limited users in cloud computing," *Comput. Secur.*, vol. 72, pp. 1–12, Jan. 2018.

[7] M. Ali, S. U. Khan, and A. V. Vasilakos, "Security in cloud computing: Opportunities and challenges," *Inf. Sci.*, vol. 305, pp. 357–383, Jun. 2015.

[8] Y. Yu, J. Shi, H. Li, Y. Li, X. Du, and M. Guizani, "Key-policy attribute-based encryption with keyword search in virtualized environments," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 6, pp. 1242–1251, Jun. 2020.

[9] T. V. X. Phuong, G. Yang, and W. Susilo, "Hidden ciphertext policy attribute-based encryption under standard assumptions," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 1, pp. 35–45, Jan. 2016.

[10] V. Yousefipoor, M. H. Ameri, J. Mohajeri, and T. Eghlidos, "A secure attribute-based keyword search scheme against keyword guessing and chosen keyword attacks," *Int. J. Inf. Commun. Technol. Res.*, vol. 10, no. 1, pp. 48–55, 2018.

[11] T. Alam, "Cloud computing and its role in the information technology," *IAIC Trans. Sustain. Digit. Innov.*, vol. 1, no. 2, pp. 108–115, 2020.

[12] K. Fan, T. Liu, K. Zhang, H. Li, and Y. Yang, "A secure and efficient outsourced computation on data sharing scheme for privacy computing," *J. Parallel Distrib. Comput.*, vol. 135, pp. 169–176, Jan. 2020.

[13] J. Li, X. Huang, J. Li, X. Chen, and Y. Xiang, "Securely outsourcing attribute-based encryption with checkability," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 8, pp. 2201–2210, Aug. 2014.

[14] Y. Wang, B. Chen, L. Li, Q. Ma, H. Li, and D. He, "Efficient and secure ciphertext-policy attribute-based encryption without pairing for cloud-assisted smart grid," *IEEE Access*, vol. 8, pp. 40704–40713, 2020.

[15] Y. Ming, B. He, and C. Wang, "Efficient revocable multi-authority attribute-based encryption for cloud storage," *IEEE Access*, vol. 9, pp. 42593–42603, 2021.

[16] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proc. 35th Annu. Symp. Found. Comput. Sci.*, 1994, pp. 124–134.

[17] L. Chen, L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, "Report on post-quantum cryptography," U.S. Dept. Commerce, Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. NISTIR 8105, 2016, vol. 12.

[18] H. Deng, Z. Qin, Q. Wu, Z. Guan, and H. Yin, "Revocable attribute-based data storage in mobile clouds," *IEEE Trans. Services Comput.*, vol. 15, no. 2, pp. 1130–1142, Mar. 2022.

[19] K. Yang, X. Jia, K. Ren, B. Zhang, and R. Xie, "DAC-MACS: Effective data access control for multiauthority cloud storage systems," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 11, pp. 1790–1801, Nov. 2013.

[20] S. Zhao, R. Jiang, and B. Bhargava, "RL-ABE: A revocable lattice attribute based encryption scheme based on R-LWE problem in cloud storage," *IEEE Trans. Services Comput.*, vol. 15, no. 2, pp. 1026–1035, Mar. 2022.

[21] R. Chaudhary, G. S. Aujla, N. Kumar, and S. Zeadally, "Lattice-based public key cryptosystem for Internet of Things environment: Challenges and solutions," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4897–4909, Jun. 2019.

[22] P. Gaborit, G. Murat, O. Ruatta, and G. Zémor, "Low rank parity check codes and their application to cryptography," in *Proc. Workshop Coding Cryptogr. (WCC)*, 2013, pp. 1–14.

[23] C. Aguilar-Melchor, O. Blazy, J.-C. Deneuville, P. Gaborit, and G. Zémor, "Efficient encryption from random quasi-cyclic codes," *IEEE Trans. Inf. Theory*, vol. 64, no. 5, pp. 3927–3943, May 2018.

[24] V. Yousefipoor and T. Eghlidos, "An efficient, secure and verifiable conjunctive keyword search scheme based on rank metric codes over encrypted outsourced cloud data," *Comput. Electr. Eng.*, vol. 105, Jan. 2023, Art. no. 108523.

[25] Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren, "Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 12, pp. 2706–2716, Dec. 2016.

[26] C.-C. Lee, P.-S. Chung, and M.-S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *Int. J. Netw. Secur.*, vol. 15, no. 4, pp. 231–240, Jul. 2013.

[27] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2007, pp. 321–334.

[28] Y. Guo, Z. Lu, H. Ge, and J. Li, "Revocable blockchain-aided attribute-based encryption with escrow-free in cloud storage," *IEEE Trans. Comput.*, vol. 72, no. 7, pp. 1901–1912, Jul. 2023.

[29] T. K. Dang, "Ensuring correctness, completeness, and freshness for outsourced tree-indexed data," *Database Technologies: Concepts, Methodologies*. Hershey, PA, USA: IGI Global, 2009, p. 2204.

[30] J. Li, Y. Wang, Y. Zhang, and J. Han, "Full verifiability for outsourced decryption in attribute based encryption," *IEEE Trans. Services Comput.*, vol. 13, no. 3, pp. 478–487, May 2020.

[31] C. A. Melchor, N. Aragon, M. Bardet, S. Bettaieb, L. Bidoux, O. Blazy, J.-C. Deneuville, P. Gaborit, A. Hauteville, A. Otmani, O. Ruatta, J.-P. Tillich, and G. Zémor, "ROLLO—Rank-ouroboros, LAKE & LOCKER," Round-2 Submission NIST PQC Project, Nat. Inst. Standards Technol. (NIST), MD, USA, Tech. Rep. 3, 2019.

[32] K. Sethi, A. Pradhan, and P. Bera, "Practical traceable multi-authority CP-ABE with outsourcing decryption and access policy updation," *J. Inf. Secur. Appl.*, vol. 51, Apr. 2020, Art. no. 102435.

[33] X. Liu, J. Ma, J. Xiong, Q. Li, T. Zhang, and H. Zhu, "Threshold attribute-based encryption with attribute hierarchy for lattices in the standard model," *IET Inf. Secur.*, vol. 8, no. 4, pp. 217–223, Jul. 2014.

[34] E. Affum, X. Zhang, X. Wang, and J. B. Ansuura, "Efficient lattice CP-ABE AC scheme supporting reduced-OBDD structure for CCN/NDN," *Symmetry*, vol. 12, no. 1, p. 166, Jan. 2020.

[35] Q. Guo, T. Johansson, and P. Stankovski, "A key recovery attack on MDPC with CCA security using decoding errors," in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.* Berlin, Germany: Springer, 2016, pp. 789–815.

[36] S. Samardjiska, P. Santini, E. Persichetti, and G. Banegas, "A reaction attack against cryptosystems based on LRPC codes," in *Proc. Int. Conf. Cryptol. Inf. Secur. Latin Amer.* Berlin, Germany: Springer, 2019, pp. 197–216.

VAHID YOUSEFIPOOR received the B.Sc. degree in electrical engineering from the Amirkabir University of Technology, Tehran, Iran, in 2014, and the M.Sc. degree in electrical engineering from the Sharif University of Technology, Tehran, in 2016, where he is currently pursuing the Ph.D. degree in electrical engineering. His major research interests include post-quantum cryptography, cloud security, functional encryption, provable security, and network security. He is also interested in coding theory and its applications in cryptography.

TARANEH EGHLIDOS received the B.Sc. degree in mathematics from the University of Shahid Beheshti, Tehran, Iran, in 1986, the M.Sc. degree in industrial mathematics from the University of Kaiserslautern, Germany, in 1991, and the Ph.D. degree in mathematics from the University of Giessen, Germany, in 2000. She joined the Sharif University of Technology (SUT), as a Faculty Member, in 2002, where she is currently an Associate Professor with the Electronics Research Institute. Her research interests include symmetric and asymmetric cryptography, applications of coding theory in cryptography, mathematical modeling for representing and solving real-world problems, and lattice-based and code-based cryptography.