A Construction of Group Oriented Attribute Based Encryption Scheme from Lattices

Maharage Nisansala Sevwandi Perera[1,a)] Toru Nakamura[1,b)]
Masayuki Hashimoto[1,c)] Hiroyuki Yokoyama[1,d)] Kouichi Sakurai[1,2,e)]

Abstract: In this paper, we present an attribute-based encryption scheme from lattices called Group Oriented Attribute-Based Encryption Scheme (GO-ABE), which facilitates users from the same group to pool their attributes to match the decryption policy of a given ciphertext. This scheme is applicable when no single user can read the message as he fails to satisfy the given policy. The idea is first presented by Li et al. in NSS 2015. However, their scheme is not secure once the quantum computers become a reality as their construction is based on bilinear mappings. To ensure the security of the scheme against quantum computers, we construct the scheme using lattices.

Keywords: Attribute-based Encryption, Lattice cryptography, group-oriented, secure the privacy

1. Introduction

Attribute-based Encryption (ABE) was first introduced by Sahai and Waters [26] at EUROCRYPT 2005. In their scheme, both ciphertexts and user private keys are constructed based on sets of attributes. Thus, a message sender encrypts his message with a specific attribute set w. Again each user has a decryption (private) key based on the possessing attribute set α. A user can decrypt the ciphertext only when at least t (threshold value) attributes that he possesses match with the given policy w. For instance, Alice encrypts a message to the attribute set $\{A, B, C\}$ and set threshold value $t = 2$. Thus Bob with attributes $\{A, B\}$ can decrypt the message.

The scheme of Sahai and Waters satisfies the threshold access structure, and they presented Fuzzy Identity-based Encryption (FIBE) in their work. Later more works ([14], [15], [16], [17], [24]) were delivered using the threshold ABE technique. On the other hand, attribute-based encryption schemes can be seen as a generalization of the identity-based encryption ([7], [8], [25], [29]). Again Goyal, Pandey, Sahai, and Waters [12] categorized ABE into two ABE types, namely, *Key-Policy Attribute-based Encryption* (KP-ABE) and Ciphertext-Policy Attribute-based Encryption (CP-ABE).

In KP-ABE ([2], [22], [23], [27], [28], [32]), each data has attributes, i.e., a ciphertext is associated with a set of attributes, and a user private key is associated to a policy (an access tree). Thus the ciphertext can be decrypted when it satisfies the user access tree. KP-ABE is used when need to decide which data can be accessed by a user. However, since the user key has the access tree (policy), KP-ABE has no authority to check access data. KP-ABE is widely used in applications like purchased (subscribe) broadcasting, structured organizations, and secure forensic analysis, which control data access. For instance, Alice can access data that only she paid to watch in a broadcasting channel. Many KP-ABE schemes were presented [22], [27], [32] to secure outsourced data in cloud.

In CP-ABE ([1], [6], [10], [11], [20], [21], [30]), a message is encrypted with an access tree (policy) selected by the message sender (or encrypter), and a trusted authority generates private keys for users based on their set of attributes. Thus, only users whose private key matches the access policy can decrypt the message. CP-ABE control who can access data. For instance, if a data is encrypted with attributes A, B, C, then a user only with those attributes can decrypt the message. CP-ABE is applied more than KP-ABE as it is more suitable in real-life applications.

However, all the schemes mentioned above are in danger once the quantum computers become a reality as their works are not based on quantum resists primitives. Some schemes successfully realized using lattices. In [13], the authors deliver the construction of Lattice based IBE scheme in the random model by using trapdoor functions, while in [3], [4], authors present an (H)IBE scheme in the standard model.

[1] Adaptive Communications Research Laboratories,
Advanced Telecommunications Research Institute International (ATR),
Kyoto, Japan
[2] Faculty of Information Science and Electrical Engineering,
Kyushu University, Fukuoka, Japan
[a)] perera.nisansala@atr.jp
[b)] tr-nakamura@atr.jp
[c)] masayuki.hashimoto@atr.jp
[d)] hr-yokoyama@atr.jp
[e)] sakurai@inf.kyushu-u.ac.jp

In [5], Boyen suggested an attribute-based functional encryption scheme from lattices. Again Yongtao Wang [31] presented the CP-ABE scheme in the standard model from lattices.

In 2015, Li et al. [18] presented a new variant of attribute-based encryption called Group-Oriented Attribute-Based Encryption (GO-ABE), enabling users from the same group to pool their attributes and private keys to generate a decryption key. Thus if the union of attributes matches the access policy, then they can retrieve the message. This scheme seems advantageous in an emergency when no user alone possesses an attribute set that matches the access tree. In their scheme, they guarantee no user from different groups cannot generate a valid decryption key pooling their attributes. Again, the scheme ensure no user reveal their private keys. However, their scheme is not quantum-safe as they form their scheme using bilinear mappings. In this paper, we construct their idea with lattice cryptography.

1.1 Contribution

Since the existing GO-ABE scheme is not quantum-safe, we construct GO-ABE scheme from lattices. For the construction of the new scheme, we employ Yongtao Wang's scheme [31] and the group signature scheme [19] from lattices. Thus our new GO-ABE scheme ensures security against quantum computers. The existing GO-ABE scheme is a threshold policy encryption scheme. That is, a user (set of users pooling their attributes) can decrypt the ciphertext if they can satisfy at least t attributes, where t is the threshold value declared in the policy. On the other hand, Wang's scheme concern policies that achieve in AND-gates on multi-valued attributes. In our scheme, we also concern AND-gate achieving policy, and we assume no set of attributes are shared by two users as considered in [31]. However, the existing schemes ([18], [31]), and the new scheme have limitations that should be realized in the future. At the end of the paper, we discuss those limitations.

2. Preliminaries

2.1 Notation

We denote matrices by upper-case bold letters and vectors by lower-case bold letters. For any integer $k \geq 1$, a set of integers $\{1, 2, \ldots, k\}$ is denoted by $[k]$. If S is a finite set, $|S|$ is its size. $S(k)$ indicates its permutations of k elements and $b \hookleftarrow D$ denotes that b is sampled from a uniformly random distribution D. The encoding function with full rank differences (FRD) $\mathcal{H} : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$ described in [3].

2.2 Lattices

Let q be a prime and $\mathbf{B} = [\mathbf{b}_1|\cdots|\mathbf{b}_m] \in \mathbb{Z}_q^{r\times m}$ be linearly independent vectors in $\mathbb{Z}_q^r$. The r-dimensional lattice $\Lambda(\mathbf{B})$ for $\mathbf{B}$ is defined as

$$\Lambda(\mathbf{B}) = \{\mathbf{y} \in \mathbb{Z}^r \mid \mathbf{y} \equiv \mathbf{B}\mathbf{x} \bmod q \text{ for some } \mathbf{x} \in \mathbb{Z}_q^m\},$$

which is the set of all linear combinations of columns of $\mathbf{B}$. The value m is the rank of $\mathbf{B}$.

Definition 1 (Learning With Errors (LWE)) For integers $n, m \geq 1$, and $q \geq 2$, a vector $\mathbf{s} \in \mathbb{Z}_q^n$, and the Gaussian error distribution χ, the distribution $A_{\mathbf{s},\chi}$ is obtained by sampling $\mathbf{a} \in \mathbb{Z}_q^n$ and $e \leftarrow \chi$, and outputting the pair $(\mathbf{a}, \mathbf{a}^T \cdot \mathbf{s} + e)$. LWE problem (decision-LWE problem) requires to distinguish LWE samples from truly random samples $\leftarrow \mathbb{Z}_q^n \times \mathbb{Z}_q$.

For a prime power q, $b \geq \sqrt{n}\omega(\log n)$, and distribution χ, solving $LWE_{n,q,\chi}$ problem is at least as hard as solving $SIVP_\gamma$ (*Shortest Independent Vector Problem*), where $\gamma = \tilde{\mathcal{O}}(nq/b)$ [13].

Definition 2 (Small Integer Solution (SIS)) Given uniformly random matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, find non-zero vector $\mathbf{x} \in \mathbb{Z}^m$, such that $\mathbf{A}\cdot\mathbf{x} = 0 \mod q$ and $\|\mathbf{x}\|_\infty \leq \beta$.

For any m, $\beta = \mathsf{poly}(n)$, and $q > \sqrt{n}\beta$, solving $SIS_{n,m,q,\beta}$ problem with non-negligible probability is at least as hard as solving $SIVP_\gamma$ problem, for some $\gamma = \beta \cdot \tilde{\mathcal{O}}(\sqrt{nm})$ [13], [19].

2.3 Lattice Related Algorithms

Lemma 1 (TrapGen[31]) For a odd integer $q \geq 3$ and $m = \lceil 6n\log q \rceil$ this algorithm outputs a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a basis $\mathbf{T_A} \in \mathbb{Z}_q^{m\times m}$ for $\Gamma_q^\perp(\mathbf{A})$ such that $||\tilde{\mathbf{T}_\mathbf{A}}|| \leq O(\sqrt{n\log q})$ and $||S|| \leq O(n\log q)$ with all but negligible probability in n.

Lemma 2 (SamplePre [13]) On input a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, a trapdoor basis $\mathbf{R}$, a target image $\mathbf{u} \in \mathbb{Z}_q^n$, and the standard deviation $\sigma \geq \omega(\sqrt{\log m})$, the PPT algorithm $\mathsf{SamplePre}(\mathbf{A}, \mathbf{R}, \mathbf{u}, \sigma)$ outputs a sample $\mathbf{e} \in \mathbb{Z}^m$ from a distribution that is within negligible statistical distance of $D_{\Lambda_q^{\mathbf{u}}(A),\sigma}$.

Lemma 3 (ExtBasis [9]) ExtBasis is a PPT algorithm that takes a matrix $\mathbf{B} \in \mathbb{Z}_q^{n\times m'}$, whose first m columns span $\mathbb{Z}_q^n$, and a basis $\mathbf{T_A}$ of $\Lambda_q^\perp(\mathbf{A})$, where $\mathbf{A}$ is the left $n\times m$ submatrix of $\mathbf{B}$, as inputs, and outputs a basis $\mathbf{T_B}$ of $\Lambda_q^\perp(\mathbf{B})$ with $||\widetilde{\mathbf{T_B}}|| \leq ||\widetilde{\mathbf{T_A}}||$.

Lemma 4 (SampleLeft [31]) On input a $n-$rank matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, a matrix $\mathbf{M}_1 \in \mathbb{Z}_q^{n\times m_1}$, a short basis $\mathbf{T_A} \in \mathbb{Z}_q^{m\times m}$ of $\Lambda_q^\perp(\mathbf{A})$, a vector $\mathbf{u} \in \mathbb{Z}_q^n$, and a gaussian parameter $\sigma > ||\tilde{\mathbf{T}_\mathbf{A}}|| \cdot \omega(\sqrt{\log(m + m_1)})$, outputs a vector $\mathbf{x} \in \Lambda_q^u(\mathbf{F}_1)$ satisfying $\mathbf{F}_1 \cdot \mathbf{x} = \mathbf{u}$, where $\mathbf{F}_1 = (\mathbf{A}|\mathbf{M}_1)$.

2.4 Attribute Based Encryption (ABE)

Setup: This algorithm takes the security parameter λ as inputs, and generates a public parameter $\mathbf{PK}$ and a master secret key $\mathbf{MK}$.

KeyGen: For a given public parameter $\mathbf{PK}$, a master secret key $\mathbf{MK}$, and an attribute set $\mathcal{S}$ for a user, this algorithm outputs a user private key $\mathbf{SK}$ associated with $\mathcal{S}$.

Encrypt: On input the public parameter $\mathbf{PK}$, and an access tree (policy) $\mathcal{W}$, and a message m, this algorithm outputs a ciphertext C.

Decrypt: On input a user private key $\mathbf{SK}$ and a ciphertext C for a message m, this algorithm output the message m, if the user attribute set $\mathcal{S}$ can satisfy the given policy.

3. Group Oriented Attribute Based Encryption Scheme

In the GO-ABE scheme suggested in [18], the users are belonged to a specific group and only users from the same group can pool their attributes to satisfy the access tree. However, no user will reveal their private keys.

3.1 GO-ABE

Definition 3 (GO-ABE) [18] A group-oriented attribute-based encryption scheme is parameterized by a universal set of possible attributes $\mathbb{A}$, a space of group identities $\mathbb{G} = g_1, g_2, \ldots, g_n$, and a message space $\mathbb{M}$, and has the following algorithms.

Setup: This randomized algorithm takes inputs as only the security parameter, outputs a public parameter **PK** and a master secret key **MK**.

Encryption: On input, a message $m \in \mathbb{M}$, the public parameter **PK**, and a set of attributes (access structure) $\mathcal{W}$, this algorithm outputs the ciphertext C.

KeyGen: On input, an attribute set $\mathcal{S}$ of a user, a group id g, the master secret key **MK**, and the public parameter **PK**, this algorithm outputs a decryption key $\mathbf{SK}^g_{\mathcal{S}}$.

Decryption: On input, the ciphertext C, that was encrypted under a set of attribute $\mathcal{W}$, the public key **PK**, and a set of users from same group g, this algorithm pools the user attribute sets as $U = \mathcal{S}_1 \cup \mathcal{S}_2 \cup \ldots, \mathcal{S}_N$ to generate a decryption key $\mathbf{SK}^g_U$ and outputs message m if $|\mathcal{W} \cap U| \geq t$, where t is the threshold value.

3.2 Security Definition: Selective-Set Model for GO-ABE

Int: The adversary declares the attribute set $\mathcal{W}$ that he wishes to be challenged upon.

Setup: The challenger generates a public parameter **PK** and a master secret key **MK** executing Setup and sends **PK** to the adversary.

Phase 1: The adversary queries the private secret keys $\mathbf{SK}^g_{\mathcal{S}_i}$ fro different attribute sets $\mathcal{S}_i$ with a group id $g \in \mathbb{G}$, where $|\mathcal{S}_j \cap \mathcal{W}| < t$ for all i.

At the end of Phase 1, $|U_i \cap \mathcal{W}| < t$, where $U_i = \mathcal{S}_1 \cup \mathcal{S}_2, \ldots, \mathcal{S}_N$ is the union of attribute sets all from the group g.

Challenge: The adversary sends two messages M_0 and M_1 whose lengths are the same. The challenger selects $b \leftarrow \{0, 1\}$ and encrypts M_b with $\mathcal{W}$. Then he passes the generates ciphertext C to the adversary.

Phase 2: Phase 1 is repeated with the same conditions.

Guess: The adversary outputs a guess b'.

The advantage of the adversary winning the game is $Pr[b' = b] - 1/2$.

Definition 4 The GO-ABE scheme is secure in the Selective-set model of security if all polynomial time and adversaries have at most negligible advantage in the above Selective-set game.

4. CP-ABE Scheme from Lattices

In [31], Yongtao Wang presents a lattice ciphertext policy attribute-based encryption scheme and gives two corresponding constructions. In that scheme, the policy is AND-gates on multi-valued attributes. We define this access structure below.

4.1 Lattice CP-ABE in the Standard Model

Definition 5 [31]

Let $U = \{att_1, att_2, \ldots, att_n\}$ be a set of attributes. For $att_i \in U$, $V_i - \{v_{i,1}, v_{i,2}, \ldots, v_{i,n_i}\}$ is a set of possible values, where n_i is the number of possible values for att_i. Let $\mathcal{S} = [\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n], \mathcal{S}_i \in V_i$ be an attribute list for a user, and $\mathcal{W} = [\mathcal{W}_1, \mathcal{W}_2, \ldots, \mathcal{W}_n], \mathcal{W}_i \in V_i$ be an access structure. $\mathcal{S} \models \mathcal{W}$ indicates that an attribute list $\mathcal{S}$ satisfies an access structure $\mathcal{W}$, namely $\mathcal{S}_i = \mathcal{W}_i (i = 1, 2, \ldots, n)$.

Setup: This algorithm takes as input the security parameter λ and generates a public parameter **PK** and a master secret key **MK**.

KeyGen: For a given public parameter **PK**, a master secret key **MK**, and an attribute set $\mathcal{S}$ for a user, this algorithm outputs a user private key **SK** associated with $\mathcal{S}$.

Encrypt: On input the public parameter **PK**, and an access tree (policy) $\mathcal{W}$, and a message m, this algorithm outputs a ciphertext C.

Decrypt: On input a user private key **SK** and a ciphertext C for a message m, this algorithm output the message m, if the user attribute set $\mathcal{S} \models \mathcal{W}$.

In [31], two constructions are given using [3] and [4]. For the development of our scheme we employ the construction built using [3]. Again the scheme used the encoding function with full rank differences (FRD) $\mathcal{H} : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n \times n}$ described in [3].

We provide the construction given in [31] using [3] in below.

4.2 Construction

Setup(1^λ): Take a security parameter λ as inputs, and output the public parameter **PK** and the master key **MK**.

- Set parameters n, m, q, σ and $\mathbb{A}$ as in [3].
- Execute TrapGen(n,m, q) to obtain a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a short basis $\mathbf{T_A} \in \mathbb{Z}_q^{m \times m}$.
- Select uniformly random matrices $\mathbf{B}, \mathbf{A}_1 \in \mathbb{Z}_q^{n \times m}$ and a uniformly random vector $\mathbf{u} \in \mathbb{Z}_q^n$.
- For each $v_{i,j}$ select a random vector $\mathbf{u}_{i,j}$, where $i \in [n], j \in [n_i]$.
- Output $\mathbf{PK} = (\mathbf{A}, \mathbf{B}, \mathbf{A}_1, \mathbf{u}, \{\mathbf{u}_{i,j}\}_{i \in [n], j \in [n_i]})$ and $\mathbf{MK} = \mathbf{T_A}$.

KeyGen($\mathbf{PK}, \mathbf{MK}, \mathcal{S}$): On input the public parameter **PK**, the master key **MK**, and the attribute set $\mathcal{S}$ of a user, this algorithm output a private key **SK** for the user.

- Set $\mathbf{F}_{\mathcal{S}} = \mathbf{A}|\mathbf{A}_1 + (\sum_{v_{i,j} \in \mathcal{S}} \mathcal{H}(\mathbf{u}_{i,j})) \cdot \mathbf{B}$.
  Assumption: $\sum_{v_{i,j} \in \mathcal{S}} \mathcal{H}(\mathbf{u}_{i,j}) \neq \sum_{v_{i,j} \in \mathcal{S}'} \mathcal{H}(\mathbf{u}_{i,j})$ for all $\mathcal{S} \neq \mathcal{S}'$.

- Get $\mathbf{x}_{\mathcal{S}} \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{A}_1 + (\sum_{v_{i,j}\in\mathcal{S}} \mathcal{H}(\mathbf{u}_{i,j})) \cdot \mathbf{B}, \mathbf{T}_{\mathbf{A}}, \mathbf{u}, \sigma)$, where $\mathbf{F}_{\mathcal{S}} \cdot \mathbf{x}_{\mathcal{S}} = \mathbf{u}$.
- Outputs $\mathbf{x}_{\mathcal{S}}$.

$\mathsf{Encrypt}(\mathbf{PK}, b, \mathcal{W})$: This algorithm takes the public parameter $\mathbf{PK}$, a message bit $b \in \{0,1\}$, and a policy $\mathcal{W}$, and output the ciphertext C as below.

- Set $\mathbf{F}_{\mathcal{W}} = [\mathbf{A}|\mathbf{A}_1 + (\sum_{v_{i,j}\in\mathcal{S}} \mathcal{H}(\mathbf{u}_{i,j})) \cdot \mathbf{B}]$.
- Select a uniformly random $s \in \mathbb{Z}_q^n$ and a uniformly random matrix $\mathbf{R} \in \{-1,1\}^{m\times m}$.
- Select noise vectors $\mathbf{e}_1 \in \mathbb{Z}_q$ and $\mathbf{y} \in \mathbb{Z}_q^m$, and set $\mathbf{z} \leftarrow \mathbf{R}^T\mathbf{y} \in \mathbb{Z}_q^m$.
- Set $\mathbf{c}_1 = \mathbf{F}_{\mathcal{W}}^T\mathbf{s} + \begin{bmatrix}\mathbf{y}\\ \mathbf{z}\end{bmatrix} \in \mathbb{Z}_q^{2m}$,
  $\mathbf{c}_2 = \mathbf{u}^T\mathbf{s} + \mathbf{e}_1 + b\lfloor q/2 \rfloor$.
- Output $C = (\mathcal{W}, \mathbf{c}_1, \mathbf{c}_2)$.

$\mathsf{Decrypt}(\mathbf{PK}, C, \mathbf{x}_{\mathcal{S}})$: On input the public parameter $\mathbf{PK}$, the ciphertext C, and the private key $\mathbf{x}_{\mathcal{S}}$ this algorithm executes as below and returns a message m if the attributes satisfies the policy, i.e $\mathcal{S} \models \mathcal{W}$.

- Compute $w \leftarrow \mathbf{c}_2 - \mathbf{x}_{\mathcal{S}}^T\mathbf{c}_1 \in \mathbb{Z}_q$.
- If $|w - \lfloor q/2 \rfloor| < \lfloor q/4 \rfloor$ in $\mathbb{Z}$ then outputs 1, otherwise outputs 0.

5. Our Scheme

5.1 Overview

For the universal attribute set $\mathbb{A}$, each attribute is associated with a random vector $\mathbf{u}$. At the setup stage, a trusted authority generates public parameter $\mathbf{PK}$ and a master key $\mathbf{MK}$. This master key is used when generating private keys for users. A message sender encrypts the message by selecting an attribute set. In this construction, we consider a message bit b. We take a group with N users, where $N = 2^\ell$. Different groups have a different number of users. Thus each group gets ℓ-bit id. For a user with a set of attributes, the trusted authority with master key $\mathbf{MK}$ generates a private key. First, the trusted authority, selects a vector $\mathbf{z}$ for each attribute that the user possesses, and he creates a secret key $\mathbf{x}$. Again based on the possessing attributes the user gets a trapdoor $\mathbf{T}_{\mathbf{D}_{\mathcal{S}}}$. Thus a user's private key is $(\mathbf{T}_{\mathbf{D}_{\mathcal{S}}}, \mathbf{x})$. When decrypting a message, no user reveals their private keys. They generate $\mathbf{y}$ using their private keys and pass only $\mathbf{y}$. Anyone can combine all $\mathbf{y}$ and decrypt the message. Since we assume $\mathcal{S} \neq \mathcal{S}'$, each user outputting $\mathbf{y}$ is unique.

5.2 Construction

We assume a group has N no of users such that $N = 2^\ell$.

$\mathsf{Setup}(1^\lambda)$: Take inputs a security parameter λ and outputs the public parameter $\mathbf{PK}$ and the master key $\mathbf{MK}$.

- Set parameters n, m, q, σ and $\mathbb{A}$ as in [3].
- Execute $\mathsf{TrapGen}$(n,m, q) to obtain $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a short basis $\mathbf{T}_{\mathbf{A}} \in \mathbb{Z}_q^{m\times m}$.
- Select uniformly random matrices $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$.
- Select $\mathbf{A}_1, \mathbf{A}_2, \dots \mathbf{A}_\ell \in \mathbb{Z}_q^{n\times m}$.
- For each attribute in $\mathbb{A}$ select a random vector $\mathbf{u}_i$.
- Output $\mathbf{PK} = (\mathbf{A}, \mathbf{B}, \mathbf{A}_1, \mathbf{A}_2, \dots \mathbf{A}_\ell, \{\mathbf{u}_i\}_{i\in||\mathbb{A}||})$ and $\mathbf{MK} = \mathbf{T}_{\mathbf{A}}$.

$\mathsf{Encrypt}(\mathbf{PK}, b, \mathcal{W})$: This algorithm takes the public parameter $\mathbf{PK}$, a message bit $b \in \{0,1\}$, and a policy $\mathcal{W}$, and outputs the ciphertext C as below.

- Set $\mathbf{F}_{\mathcal{W}} = \mathbf{A}|\mathbf{A}_0 + (\sum_{i=1}^{|\mathcal{W}|} \mathcal{H}(\mathbf{u}_i)) \cdot \mathbf{B}$.
- Select a uniformly random $s \in \mathbb{Z}_q^n$, $e_1 \in \mathbb{Z}_q^{2m}$, and $e_2 \in \mathbb{Z}_q$.
- Set $\mathbf{u} \leftarrow \sum_{i=1}^{|\mathcal{W}|} \mathbf{u}_i$
- Set $\mathbf{c}_1 = \mathbf{F}_{\mathcal{W}}^T\mathbf{s} + e_1$,
  $\mathbf{c}_2 = \mathbf{u}^T\mathbf{s} + \mathbf{e}_2 + b\lfloor q/2 \rfloor$.
- Output $C = (\mathcal{W}, \mathbf{c}_1, \mathbf{c}_2)$.

$\mathsf{KeyGen}(\mathbf{PK}, \mathbf{MK}, \mathcal{S})$: On input the public parameter $\mathbf{PK}$, the master key $\mathbf{MK}$, and the attribute set $\mathcal{S}$ of a user, this algorithm output a private key $\mathbf{SK}$ for the user.

- Select ℓ-bit string d for group id g.
- Generate $\mathbf{A}_g = [\mathbf{A}|\mathbf{A}_0 + \sum_{i=1}^{\ell} d_i\mathbf{A}_i]$.
- For a user with attribute set $\mathcal{S}$:
  - Select $\mathbf{z}_i \in \mathbb{Z}^{2m}$ for all attributes in $\mathcal{S} = \{\mathbf{u}_1', \mathbf{u}_2', \dots, \mathbf{u}_{|\mathcal{S}|}'\}$, such that $\mathbf{A}g \cdot \mathbf{z}_i = \mathbf{u}_i'$.
  - Compute $\mathbf{x} = \sum_{i=1}^{|\mathcal{S}|} \mathbf{A}_g \cdot \mathbf{z}_i$.
  - Get $\mathbf{D}_{\mathcal{S}} = [\mathbf{A}|\mathbf{A}_0 + (\sum_{i=1}^{|\mathcal{S}|} \mathbf{u}_i')\mathbf{B}]$.
  - Get $\mathbf{T}_{\mathbf{D}_{\mathcal{S}}} = \mathsf{ExtBasis}(\mathbf{T}_{\mathbf{A}}, \mathbf{D}_{\mathcal{S}})$.
- Outputs user private key $(\mathbf{T}_{\mathbf{D}_{\mathcal{S}}}, \mathbf{x})$.

$\mathsf{Decrypt}(\mathbf{PK}, C, \mathbf{x}_{\mathcal{S}})$: On input the public parameter $\mathbf{PK}$, the ciphertext C, and the private key $\mathbf{x}_{\mathcal{S}}$ this algorithm executes as below and returns a message m if the attributes satisfies the policy, i.e $\mathcal{S} \models \mathcal{W}$.

- Each user obtains and passes $\mathbf{y}_i \leftarrow \mathsf{SamplePre}(\mathbf{T}_{\mathbf{D}_{\mathcal{S}}}, \mathbf{D}_{\mathcal{S}}, \mathbf{x}, \sigma)$.
- Compute $w \leftarrow \mathbf{c}_2 - \mathbf{Y}^T\mathbf{c}_1 \in \mathbb{Z}_q$.
- If $|w - \lfloor q/2 \rfloor| < \lfloor q/4 \rfloor$ in $\mathbb{Z}$ then outputs 1, otherwise outputs 0.

6. Discussion of Security and Limitations

6.1 Security of the Scheme

Theorem 1 The advantage of an adversary on selective-set model is negligible under the hardness of LWE problem

Suppose there is a PPT adversary $\mathcal{A}$ against selective-set model with advantage ϵ. We build a simulator $\mathcal{B}$ that can solve LWE problem. The simulator $\mathcal{B}$ uses $\mathcal{A}$ to solve LWE.

First, $\mathcal{A}$ publishes the access structure $\mathcal{W}^*$ that he wants to use for challenging stage. Next $\mathcal{B}$ honestly generates $\mathbf{PK}$ and gives to $\mathcal{A}$. When $\mathcal{A}$ queries private keys for a different set of attribute $\mathcal{S}$, where $\mathcal{S} \neq \mathcal{W}^*$, he queries his own oracle and responses. At the challenge phase $\mathcal{B}$ receives a message bit $b^* \in \{0,1\}$ from $\mathcal{A}$, and he encrypts b^* using the parameters he selected honestly. Then he selects a random $r \in \{0,1\}$ and if he gets $r = 0$ then he sends $C^* = (\mathcal{W}^*, c_1^*, c_2^*)$, where c_1^*, c_2^* are honest values. If he gets $r = 1$, then he selects randomly $c_1 \in \mathbb{Z}^{2m}$ and $c_2 \in \mathbb{Z}$, and passes to $\mathcal{A}$. Next $\mathcal{A}$ guesses r' for r, which is the solution for LWE.

6.2 Limitations

The scheme given in [31] assumes two users share no same set of attributes. Our scheme also assumes the same. However, in the real world, this assumption is not practical. Again, GO-ABE scheme in [18] and our scheme users who are pooled are not tracked. Thus users who miss used the decrypted data cannot be punished. This problem is a significant concern that should need attention. Again, even though the GO-ABE scheme is advantageous in emergencies such as decrypting a patient's data to analyze his medical condition, users can use this scheme even when it is not in an emergency. Thus anyone can misuse this scheme. We note that it is required to control the situation that GO-ABE applies. The existing schemes and the new GO-ABE scheme from lattices have these issues that are opened to discuss in the future.

7. Conclusion

In this paper, we provide a construction of GO-ABE scheme from lattices that supports users to pool their attributes anonymously (without revealing their secret keys) to satisfy a given access tree. Since we used lattice cryptography, our scheme is quantum resistant. However, some limitations need to discuss in the future.

References

[1] Attrapadung, N., Imai, H.: Dual-Policy Attribute Based Encryption. In: ACNS 2009. Lecture Notes in Computer Science, vol 5536, pp 168-185, (2009).

[2] Attrapadung, N., Libert, B., de Panafieu E.: Expressive Key-Policy Attribute-Based Encryption with Constant-Size Ciphertexts. In: PKC 2011. Lecture Notes in Computer Science, vol 6571, pp 90-108, (2011).

[3] Agrawal, S., Boneh, D., Boyen X.: Efficient Lattice (H)IBE in the Standard Model. In: EUROCRYPT 2010 Lecture Notes in Computer Science, vol 6110, pp 553-572, (2010).

[4] Agrawal, S., Boneh, D., Boyen X.: Lattice Basis Delegation in Fixed Dimension and Shorter-Ciphertext Hierarchical IBE. In: CRYPTO 2010. Lecture Notes in Computer Science, vol 6223,pp 98-115, (2010).

[5] Boyen, Xavier.: Attribute-Based functional encryption on lattices. In Proceedings of the 10th theory of cryptography conference on Theory of Cryptography (TCC'13). Springer-Verlag, Berlin, Heidelberg, pp 122-142 (2013)

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

[7] Boneh, D., Franklin, M.: Identity-Based Encryption from the Weil Pairing. In: CRYPTO 2001. Lecture Notes in Computer Science, vol 2139, pp 213-229 (2001).

[8] Cocks, C.: An Identity Based Encryption Scheme Based on Quadratic Residues. In: Cryptography and Coding 2001. Lecture Notes in Computer Science, vol 2260, pp 360-363 (2001).

[9] Cash, D., Hofheinz, D., Kiltz, E., Peikert, C.: Bonsai Trees, or How to Delegate a Lattice Basis. In: EUROCRYPT 2010. Lecture Notes in Computer Science, vol 6110, pp 523-552, (2010).

[10] Cheung, L., Newport, C.: Provably secure ciphertext policy ABE. In: Proceedings of the 14th ACM conference on Computer and communications security, pp. 456-465, (2007).

[11] Emura, K., Miyaji, A., Nomura, A., Omote, K., Soshi, M.: A Ciphertext-Policy Attribute-Based Encryption Scheme with Constant Ciphertext Length. In: ISPEC 2009. Lecture Notes in Computer Science, vol 5451, pp 13-23, (2009).

[12] Goyal, V., Pandey, O., Sahai, A., Waters, B. : 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, (2006).

[13] Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In Proceedings of the fortieth annual ACM symposium on Theory of computing (STOC '08), (2008)

[14] Ge, A., Zhang, R., Chen, C., Ma, C., Zhang, Z.:Threshold Ciphertext Policy Attribute-Based Encryption with Constant Size Ciphertexts. In: ACISP 2012. Lecture Notes in Computer Science, vol 7372,pp 336-349, (2012).

[15] Herranz, J., Laguillaumie, F., Rafols, C.: Constant Size Ciphertexts in Threshold Attribute-Based Encryption. In: PKC 2010. Lecture Notes in Computer Science, vol 6056, pp 19-34, (2010).

[16] Ibraimi, L., Petkovic, M., Nikova, S., Hartel, P., Jonker, W.:Ciphertext-policy attribute-based threshold decryption with flexible delegation and revocation of user attributes. University of Twente, Tech. Rep (2009).

[17] Lin, H., Cao, Z., Liang, X., Shao, J.: Secure Threshold Multi Authority Attribute Based Encryption without a Central Authority. In: Chowdhury D.R., Rijmen V., Das A. (eds) Progress in Cryptology - INDOCRYPT 2008. INDOCRYPT 2008. Lecture Notes in Computer Science, vol 5365, pp 426-436, (2008).

[18] Li, M., Huang, X., Liu, J.K., Xu, L.: GO-ABE: Group-Oriented Attribute-Based Encryption. In: NSS 2015. Lecture Notes in Computer Science, vol 8792, pp 260-270, (2014).

[19] Ling, S., Nguyen, K., Wang, H.: Group Signatures from Lattices: Simpler, Tighter, Shorter, Ring-Based. In: PKC 2015. Lecture Notes in Computer Science, vol 9020, pp 427-449, (2015)

[20] Lewko, A., Okamoto, T., Sahai, A., Takashima, K., Waters, B.: Fully Secure Functional Encryption: Attribute-Based Encryption and (Hierarchical) Inner Product Encryption. In: EUROCRYPT 2010. Lecture Notes in Computer Science, vol 6110, pp 62-91, (2010).

[21] Lewko, A., Waters, B.: Decentralizing Attribute-Based Encryption. In: EUROCRYPT 2011. Lecture Notes in Computer Science, vol 6632, pp 568-588, (2011).

[22] Li, M., Yu, S., Zheng, Y., Ren, K., Lou, W.: Scalable and Secure Sharing of Personal Health Records in Cloud Computing Using Attribute-Based Encryption. In: IEEE Transactions on Parallel and Distributed Systems, vol. 24, no. 1, pp 131-143, (2013).

[23] Li, Q., Xiong, H., Zhang, F., Zeng, S.: An expressive decentralizing kp-abe scheme with constant-size ciphertext. In: IJ Network Security, 15(3), pp 161-170 (2013).

[24] Nali, D., Adams, C. M., Miri, A.: Using Threshold Attribute-based Encryption for Practical Biometric-based Access Control. IJ Network Security, 1(3), pp 173-182 (2005).

[25] Shamir, A.: Identity-Based Cryptosystems and Signature Schemes. In: CRYPTO 1984. Lecture Notes in Computer Science, vol 196, pp 47-53 (1985).

[26] Sahai, A., Waters, B.: Fuzzy Identity-Based Encryption. In: EUROCRYPT 2005. Lecture Notes in Computer Science, vol 3494, pp 457-473, (2005).

[27] Tu, S., Wang, C., Ren, K., Lou, W.: Attribute based data sharing with attribute revocation. In Proceedings of the 5th ACM symposium on information, computer and communications security, pp 261-270, (2010).

[28] Wang, Y., Chen, K., Long, Y., Liu, Z. (2012).: Accountable authority key policy attribute-based encryption. In: Science China Information Sciences, 55(7), pp 1631-1638, (2012).

[29] Waters, B.: Efficient Identity-Based Encryption Without Random Oracles. In: EUROCRYPT 2005. EUROCRYPT 2005. Lecture Notes in Computer Science, vol 3494, pp 114-127(2005).

[30] Waters, B.: Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization. In: PKC 2011. Lecture Notes in Computer Science, vol 6571, pp 53-70, (2011).

[31] Wang, Y.: Lattice Ciphertext Policy Attribute-based Encryption in the Standard Model. Int. J. Netw. Secur., 16, pp 444-451, (2014).

[32] Yu, S., Wang, C., Ren, K., Lou, W.: Achieving secure, scalable, and fine-grained data access control in cloud computing. In 2010 Proceedings IEEE INFOCOM , pp 1-9, (2010).