Published in IET Information Security
Received on 13th March 2013
Revised on 28th May 2013
Accepted on 19th June 2013
doi: 10.1049/iet-ifs.2013.0111

ISSN 1751-8709

Threshold attribute-based encryption with attribute hierarchy for lattices in the standard model

Ximeng Liu[1], Jianfeng Ma[2], Jinbo Xiong[2], Qi Li[2], Tao Zhang[2], Hui Zhu[1]

[1]*School of Telecommunications Engineering, Xidian University, Xi'an 710071, People's Republic of China*
[2]*School of Computer Science and Technology, Xidian University, Xi'an 710071, People's Republic of China*
E-mail: snbnix@gmail.com

Abstract: Attribute-based encryption (ABE) has been considered as a promising cryptographic primitive for realising information security and flexible access control. However, the characteristic of attributes is treated as the identical level in most proposed schemes. Lattice-based cryptography has been attracted much attention because of that it can resist to quantum cryptanalysis. In this study, lattice-based threshold hierarchical ABE (lattice-based *t*-HABE) scheme without random oracles is constructed and proved to be secure against selective attribute set and chosen plaintext attacks under the standard hardness assumption of the learning with errors problem. The notion of the HABE scheme can be considered as the generalisation of traditional ABE scheme where all attributes have the same level.

1 Introduction

Attribute-based cryptography has attracted much attention as a new public key primitive in recent years. Attribute based encryption (ABE) [1, 2] has significant advantages over the traditional PKC because of that it can achieve both information security and fine-grained access control. When a data provider wants to share some information with a user, the provider must know exactly the one with whom she/he wants to share. Although in the ABE scheme, it achieves flexible one-to-many encryption instead of one-to-one. The ciphertext is labelled with sets of descriptive attributes defined for the system users or access structure. The particular user holds private key can decrypt the particular ciphertext only if the two match. For example, if the headmaster wants to encrypt a document to the entire professors with age above 45 in the computer science department, the document would be encrypted with access structure ('Professor' AND 'CS department' AND 'Age above 45'), and only the users who hold the private key containing these three attributes can obtain the document while others cannot. However, for instance, soldiers have different military ranks according to different positions, and the attribute 'military rank' can be classified into different types (such as 'general', 'colonel', 'captain' and so on) according to their hierarchy. Once the headquarters wants to send a document to the soldier whose ranks is above colonel, and if we use the traditional ABE method to encrypt, we shall use all the attribute ranks above colonel to create the ciphertext. The length of the ciphertext is long and the amount of computation for encryption and decryption is huge. In order to solve this problem, we categorised the attribute into different hierarchy. The headquarters could only encrypt the document by using rank colonel, and soldiers with rank general and colonel can decrypt the document while soldiers with rank captain cannot. It could greatly decrease the number of attributes used to encrypt the document and the length of the ciphertext.

Most existing ABE constructions are based on bilinear groups. Since it is hard even for quantum computing, some lattice problems are exploited to construct ABE schemes. However, there is no ABE scheme using lattice to construct ABE scheme with attribute hierarchy. In real scenarios, the attributes are not always in the same level, and different attributes have different important degrees in the system. In this paper, we propose a scheme called lattice-based threshold hierarchical ABE (lattice-based *t*-HABE) in order to accord with the practical scenario. We use lattice to construct the *t*-HABE scheme which can provide better protect than bilinear pairing construction. Each user in the system possesses a set of attributes with different hierarchies and user's private key is associated with these attributes. The data owner encrypts a data with his/her own hierarchical attribute set. In order to decrypt the message successfully, the level of users attributes positions in the higher level of the private keys must be higher than the corresponding ones that in the lower level, and the total number of this difference must exceed the threshold *t*. In the HABE, the universal attributes have different levels according to their important degrees defined in the access control system. The notion of the HABE scheme can be considered as the generalisation of traditional ABE scheme where all attributes have the same level. Our scheme can be hard even for quantum computers because we use lattice to construct *t*-HABE secheme.

1.1 Related work

In basic ABE, an important application of the fuzzy identity-based encryption, a user encrypts the plaintext with

a subset of their attributes and the receiver successfully decrypts the ciphertext with any set of attributes that has at least t common attributes. We call this scheme as threshold attribute-based encryption (t-ABE) for description simplicity. Later, Goyal *et al*. [1] extended basic ABE scheme to richer kinds of ABE, where decryption is permitted when the attribute set satisfies a more complex boolean formula specified by an access structure. The existing ABE scheme can be classified into two flavours: key-policy ABE (KP-ABE) [1] and ciphertext-policy ABE (CP-ABE) [3, 4]. In CP-ABE scheme: the ciphertext is associated with the access structure while the private key is related to a set of attribute. The first CP-ABE scheme was proposed by Bethencourt *et al*. [3] used threshold secret sharing to enforce the policy during the encryption phase. Although KP-ABE proceeds in the reversed way, the ciphertext is associated with a set of attributes and the private key is related to an access structure. A construction of a KP-ABE scheme was first provided in [1]. In their scheme, when users made a secret request, the trusted authority determined which combination of attributes must appear in the ciphertext for the users to decrypt. Ostrovsky *et al*. [5] presented the first KP-ABE system which supports the expression of non-monotone formulas in key policies. In order to realise both KP-ABE and hidden CP-ABE, Katz *et al*. [6] proposed a predicate encryption scheme supporting inner product predicates and their scheme. Waters [7] presents a new methodology for realising ABE systems from a general set of access structures in the standard model. They express access control by a linear secret sharing scheme matrix over the attributes in the system. Li *et al*. [8] proposed a scheme called enhancing ABE with attribute hierarchy. In this scheme, the universal attributes were classified into trees according to their relationship. All ABE schemes introduced above constructing by using bilinear groups.

Owing to its simple implement and provable security reductions, lattice-based cryptography have been brought to the forefront these days. As a post-quantum cryptosystem, lattice-based cryptography is considered hard for quantum computers to compute [9]. It can implement efficient and highly parallel which are potentially quite practical. Ajtai [10] lattices were widely used and it can construct one-way functions and collision-resistant hash functions [10, 11], public-key encryption [12–14], identity-based encryption schemes [15–17], trapdoor functions [18] and even fully homomorphic encryption. Agrawal *et al*. [19] used lattices to constructe a fuzzy identity-based encryption (lattice-based fuzzy IBE) scheme. Lattice-based fuzzy IBE scheme is like lattice-based-IBE scheme except that the identity used to decrypt the message is 'close enough' identity containing in the ciphertext. Agrawal *et al*. [15] gave an efficient IBE scheme in the standard model rather than bit-by-bit resulting in lattices. Agrawal *et al*. [16] also proposed another efficient HIBE scheme with a new delegation mechanism which is not increasing the dimension of the lattices involved. Boyen [20] constructed (key-policy) attribute-based encryption and reduce its security from learning with errors (LWE) in the standard model.

1.2 Our contributions

In this work, our main result is the construction of lattice-based threshold HABE scheme and reduce its security from LWE. Our threshold HABE scheme likes the lattice-based fuzzy IBE scheme proposed in [19] except that attributes in ours is not in the same level. We use delegation mechanism to achieve the hierarchical attributes without increasing the dimension of the lattices. We also formalise the model of lattice-based t-HABE and give security model for lattice-based t-HABE. We prove our lattice-based t-HABE is selective security to against choose plaintext attack in the standard model by using the LWE.

1.3 Organisation

The rest of paper is organised as follows: In section 2, we introduce the formal models and and its security model of HABE scheme. In section 3, we review some concept about interger lattices, algorithm *TrapGen*, discrete Gaussians, learning with errors problem and basis delegation without dimension increase. In section 4, we give the specific construction about the lattice-based HABE scheme. In section 5, we prove our scheme under the standard model and gives parameters constraints for lattice-based t-HABE scheme and compares it with the existing schemes. Finally, we conclude this paper in Section 6.

2 HABE scheme and security model

In this section, we introduce HABE scheme and its security model.

2.1 HABE scheme

A HABE scheme consists of four fundamental algorithms: Setup, Encrypt, Key Generation and Decrypt.

Setup(λ, l, d): The setup algorithm inputs a security parameter λ, the maximum number of attributes l and the maximum deep for all attribute d. It outputs the public parameters PP and master key MK.

KeyGen(PP, MK, Att_d, k): The key generation algorithm inputs the public parameters PP, the master key MK, the attribute set Att_d with attribute with hierarchy and threshold $k \leq l$. It outputs a private key $\mathrm{SK}_{\mathrm{Att}_d}$.

Encrypt(PP, m, Att'_d): The encryption algorithm inputs the public parameters PP, the message bit m which the sender want to encryption which the sender wants to encryption and a set of attributes Att'_d with hierarchy. It outputs the ciphertext $\mathrm{CT}_{\mathrm{Att}'_d}$.

Decrypt($\mathrm{CT}_{\mathrm{Att}'_d}$, $\mathrm{SK}_{\mathrm{Att}_d}$, PP): The decryption algorithm inputs the ciphertext $\mathrm{CT}_{\mathrm{Att}'_d}$, the private key $\mathrm{SK}_{\mathrm{Att}_d}$ and the public parameter PP. Every attribute in the attribute set J satisfies that attributes in private key have higher hierarchy than the corresponding attributes in the ciphertext. It can decrypts the ciphertext and return message m if $|J| \geq k$.

2.2 Security model for the HABE

In our security model, the adversary will choose challenged attribute set Att* and ask for any private key SK containing hierarchical attribute set Att where $|\mathrm{Att}_j \cap \mathrm{Att}^*| < k$. We now give the formal security game for HABE below.

Init. The adversary declares the set of the attribute with hierarchy Att*, which he want to be challenged upon.

Setup. The challenger runs the Setup algorithm and outputs the public parameter PP. Challenger gives parameter PP to the adversary.

Phase 1. The adversary makes repeated polynomially private key queries for many attribute set Att_j, which a number of

doi: 10.1049/iet-ifs.2013.0111

attributes in Att_j have higher hierarchy than the corresponding attributes in Att* should less than k. That is $|\mathrm{Att}_j \cap \mathrm{Att}^*| < k$ for all j.

Challenge. Once decides adversary $\mathcal{A}$ that Phase 1 is over, $\mathcal{A}$ submits message m with the challenge attribute set Att*. The challenger flips a random coin β. If the $\beta = 1$, the ciphertext CT* is given to the adversary $\mathcal{A}$. Otherwise a random element of ciphertext space is returned.

Phase 2. Same as Phase 1.

Guess. The adversary outputs a guess β' of β.

Adversary $\mathcal{A}$ wins this game if $\beta' = \beta$. The advantage of $\mathcal{A}$ in this game is defined as $\mathrm{Adv}_{\mathcal{A}} = Pr[\beta' = \beta] - \frac{1}{2}$.

Definition 1: Our HABE scheme is IND-sAtr-CPA secure if no polynomial time adversaries win the above game with non-negligible advantage.

3 Preliminaries

3.1 Interger lattices

Definition 2: Let $\boldsymbol{B} = [\boldsymbol{b}_1 | \cdots | \boldsymbol{b}_m] \in \mathbb{R}^{m \times m}$ be an $m \times m$ matrix where columns vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_m \in \mathbb{R}^{m \times m}$ are linearly independent. The m-dimensional full-rank lattice $\boldsymbol{\Lambda}$ generated by $\boldsymbol{b}$ is the set

$$\boldsymbol{\Lambda} = L(\boldsymbol{B}) = \left\{ \boldsymbol{y} \in \mathbb{R}^m, \text{ s.t. } \exists \boldsymbol{y} \in \mathbb{R}^m, \ \boldsymbol{y} = B\boldsymbol{s} = \sum_{i=1}^{m} s_i \boldsymbol{b}_i \right\}$$

Here, we are interested in integer lattices, that is, when L is subset of $\mathbb{Z}^m$. We let $\det(\boldsymbol{\Lambda})$ denotes the determinant of $\boldsymbol{\Lambda}$.

Definition 3: For q prime, $\boldsymbol{A} \in \mathbb{Z}_q^{n \times m}$ and $\boldsymbol{u} \in \mathbb{Z}_q^{n \times m}$, define

$$\boldsymbol{\Lambda}_q(\boldsymbol{A}) = \{\boldsymbol{e} \in \mathbb{Z}^m \text{ s.t. } \exists \boldsymbol{s} \in \mathbb{Z}_q^n \text{ where } \boldsymbol{A}^{\mathrm{T}}\boldsymbol{s} = \boldsymbol{e} (\mathrm{mod}\ q)\}$$

$$\boldsymbol{\Lambda}_q^{\perp}(\boldsymbol{A}) = \{\boldsymbol{e} \in \mathbb{Z}^m \text{ s.t. } \boldsymbol{A}\boldsymbol{e} = 0 (\mathrm{mod}\ q)\}$$

$$\boldsymbol{\Lambda}_q^{u}(\boldsymbol{A}) = \{\boldsymbol{e} \in \mathbb{Z}^m \text{ s.t. } \boldsymbol{A}\boldsymbol{e} = \boldsymbol{u} (\mathrm{mod}\ q)\}$$

It is easy to find that if $t \in \boldsymbol{\Lambda}_q^{u}(\boldsymbol{A})$ then $\boldsymbol{\Lambda}_q^{u}(\boldsymbol{A}) = \boldsymbol{\Lambda}_q^{\perp}(\boldsymbol{A}) + t$ and hence $\boldsymbol{\Lambda}_q^{u}(\boldsymbol{A})$ is a shift of $\boldsymbol{\Lambda}_q^{\perp}(\boldsymbol{A})$.

3.2 Algorithm TrapGen

Ajtai [21] shows how to sample an essentially uniform matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n \times m}$ with an associated full rank set $\boldsymbol{T}_{\boldsymbol{A}} \subset \boldsymbol{\Lambda}_q^{\perp}(\boldsymbol{A})$ of low-norm vectors. Alwen and Peikert [22] improved the Ajtai's basis sampling algorithm. In this paper, we use the improved version presented in [22].

Proposition 1: Let $n = n(\lambda)$, $q = q(\lambda)$, $m = m(\lambda)$ be positive integers with $q \geq 2$ and $m \geq 5n\log q$. There exists a probabilistic polynomial-time algorithm **TrapGen** that outputs a pair $\boldsymbol{A} \in \mathbb{Z}_q^{n \times m}$, $\boldsymbol{T}_{\boldsymbol{A}} \in \mathbb{Z}_q^{n \times m}$ such that $\boldsymbol{A}$ is statistically close to uniform and $\boldsymbol{T}_{\boldsymbol{A}}$ is a basis for $\boldsymbol{\Lambda}^{\perp}(\boldsymbol{A})$ with length $L = ||\widetilde{\boldsymbol{T}_{\boldsymbol{A}}}|| \leq m \cdot \omega(\sqrt{\log m})$ with all but $n^{-\omega(1)}$ probability.

3.3 Discrete Gaussians

Definition 4: Let $m \in \mathbb{Z}_{>0}$ be a positive integer and $\boldsymbol{\Lambda} \subset \mathbb{R}^m$ an m-dimensional lattices. For any vector $\boldsymbol{c} \subset \mathbb{R}^m$ and any positive parameter $\sigma \in \boldsymbol{R}_{>0}$, we define:

$\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x}) = \exp\left(-\pi\left[\left(||\boldsymbol{x} - \boldsymbol{c}||^2\right)/\sigma^2\right]\right)$: a Gaussian-shaped function on $\mathbb{R}^m$ with centre $\boldsymbol{c}$ and parameter σ.

$\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{\Lambda}) = \sum_{\boldsymbol{x} \in \boldsymbol{\Lambda}} \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x})$: the (always converging) discrete integral of $\rho_{\sigma,\ \boldsymbol{c}}$ over the lattice $\boldsymbol{\Lambda}$.

$\mathcal{D}_{\boldsymbol{\Lambda},\sigma,\boldsymbol{c}}$: the discrete Gaussian distribution over $\boldsymbol{\Lambda}$ with centre $\boldsymbol{c}$ and parameter σ

$$\forall \boldsymbol{y} \in \boldsymbol{\Lambda}, \quad \mathcal{D}_{\boldsymbol{\Lambda},\sigma,\boldsymbol{c}}(\boldsymbol{\Lambda}) = \frac{\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{y})}{\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{\Lambda})}$$

For notational convenience, $\rho_{\sigma,0}$ and $\mathcal{D}_{\boldsymbol{\Lambda},\sigma,0}$ are abbreviated as ρ_{σ} and $\mathcal{D}_{\boldsymbol{\Lambda},\sigma}$.

3.3.1 Sampling discrete Gaussians over lattices

Gentry *et al.* [18] constructed the following algorithm which is given a basis $\boldsymbol{B}$ for the m-dimensional lattice $\boldsymbol{\Lambda}$ with $\sigma \geq ||\tilde{\boldsymbol{B}}|| \cdot \omega(\sqrt{\log m})$. It outputs a sampling from the discrete Gaussian $\mathcal{D}_{\boldsymbol{\Lambda},\sigma,\boldsymbol{c}}$.

SampleGaussian(Λ, B, σ, c) [18]: On input lattice $\boldsymbol{\Lambda}$, a basis $\boldsymbol{B}$ for $\boldsymbol{\Lambda}$, a positive Gaussian parameter σ, and a centre vector $\boldsymbol{c} \in \mathbb{R}^m$, it outputs a fresh random vector $\boldsymbol{x} \in \boldsymbol{\Lambda}$ drawn from a distribution statistically close to $\mathcal{D}_{\boldsymbol{\Lambda},\sigma,\boldsymbol{c}}$.

3.3.2 Preimage sampling algorithm

We will use the following algorithm from [18]. *Algorithm SamplePre*(A, T_A, u, σ): Let $q \geq 2$, $m \geq 2n\log q$, on input a matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n \times m}$ with 'short' trapdoor basis $\boldsymbol{T}_{\boldsymbol{A}}$ for $\Lambda_q^{\perp}(\boldsymbol{A})$, a target image $\boldsymbol{u} \in \mathbb{Z}_q^n$ and a Gaussian parameter $\sigma \geq ||\widetilde{\boldsymbol{T}_{\boldsymbol{A}}}|| \cdot \omega(\sqrt{\log m})$, outputs a sample $\boldsymbol{e} \in \mathbb{Z}^{\mathbf{m}}$ from a distribution that is within negligible statistical distance of $\mathcal{D}_{\boldsymbol{\Lambda}_q^{u}(\boldsymbol{A}),\sigma}$.

3.4 Hardness assumption

Security of our constructions reduces to the LWE problem, which a classic hard problem on lattices is extensively studied and used defined by Regev [14].

Definition 5: Let a prime q, a positive integer n and a distribution χ over $\mathbb{Z}_q$, all public. An $(\mathbb{Z}_q, n, \chi)$-LWE problem instance consists of access to an unspecified challenge oracle $\mathcal{O}$, being, either, a noisy pseudo-random sampler $\mathcal{O}_s$ carrying some constant random secret key $\boldsymbol{s} \in \mathbb{Z}_q^n$ or, a truly random sampler $\mathcal{O}_{\$}$, whose behaviours are respectively as follows:

$\mathcal{O}_s$: outputs samples of the form $(\boldsymbol{u}_i, v_i) = (\boldsymbol{u}_i, \boldsymbol{u}_i^{\mathrm{T}}\mathrm{s} + x_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, where $\boldsymbol{s} \in \mathbb{Z}_q^n$, is a uniformly distributed persistent value that is invariant across invocations, $x_i \in \mathbb{Z}_q$ is a freshly ephemeral additive noise component with distribution χ, and $\boldsymbol{u}_i$ is uniform in $\mathbb{Z}_q^n$.

$\mathcal{O}_{\$}$: outputs truly uniform random samples $(\boldsymbol{u}_i, v_i)$ from $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

The $(\mathbb{Z}_q, n, \chi)$-LWE problem allows a number of queries to be made to the challenge oracle $\mathcal{O}$. We say that an algorithm $\mathcal{A}$ decides the $(\mathbb{Z}_q, n, \chi)$-LWE problem if

$$\mathrm{LWE} - \mathrm{adv}[\mathcal{A}] := \left| \Pr\left[\mathcal{A}^{\mathcal{O}_s} = 1\right] - \Pr\left[\mathcal{A}^{\mathcal{O}_{\$}} = 1\right] \right|$$

is non-negligible for a random $\boldsymbol{s} \in \mathbb{Z}_q^n$.

Regev [14] shows that the LWE problem is as hard as the worst-case SIVP and GapSVP for certain noise distributions χ

under a quantum reduction [23]. Recall that the symbol $\lfloor x \rceil$ denotes the closest integer to x for $x \in \mathbb{R}$.

Definition 6: For an $\alpha \in (0, 1)$ and a prime q, let $\bar{\Psi}_\alpha$ denote the distribution over $\mathbb{Z}_q$ of the random variable $\lfloor q\boldsymbol{X} \rceil$ mod q, where $\boldsymbol{X}$ is a normal random variable with mean 0 and standard deviation $\alpha/\sqrt{2\pi}$.

Theorem 1 [23]: If there exists an efficient, possibly quantum, algorithm for deciding the $\left(\mathbb{Z}_q, n, \bar{\Psi}_\alpha\right)$-LWE problem for $q > 2\sqrt{n}/\alpha$ then there exists an efficient quantum algorithm for approximating the SIVP and GapSVP problems, to within $\tilde{\mathcal{O}}(n/\alpha)$ factors in the l_2 norm, in the worst case.

We need the following lemma to ensure decryption works correctly. The proof is implicited in [18].

Lemma 1: Let $\boldsymbol{e}$ be some vector in $\mathbb{Z}^m$ and let $\boldsymbol{y} \leftarrow \bar{\Psi}_\alpha^m$. Then the quantity $|\boldsymbol{e}^{\mathrm{T}}\boldsymbol{y}|$ treated as an integer in $[0, q-1]$ satisfies

$$|\boldsymbol{e}^{\mathrm{T}}\boldsymbol{y}| \leq ||\boldsymbol{e}||q\alpha\omega(\sqrt{\log m}) + ||\boldsymbol{e}||\sqrt{m}/2 \tag{1}$$

with all but negligible probability in m.

As a special case, Lemma 1 shows that if $\boldsymbol{y} \leftarrow \bar{\Psi}_\alpha^m$ is treated as an interger in $[0, q-1]$ then $|\boldsymbol{y}| < q\alpha\omega(\sqrt{\log m}) + 1/2$ with all but negligible probability in m.

3.5 Basis delegation without dimension increase

In this subsection, we introduce the short lattice basis delegation algorithm that keeps that lattice dimension does not increase proposed in [16]. Here we present the basis delegation mechanism.

Distributions on low norm matrices. A matrix $\boldsymbol{R}$ in $\mathbb{Z}^{m\times m}$ is $\mathbb{Z}_q$-invertible if $\boldsymbol{R}$ mod q is invertible as a matrix in $\mathbb{Z}_q^{m\times m}$. This construction makes use of $\mathbb{Z}_q$-invertible matrices $\boldsymbol{R}$ in $\mathbb{Z}^{m\times m}$ where all the columns of $\boldsymbol{R}$ are low norm.

Definition 7: Define $\sigma_{\boldsymbol{R}} = \tilde{L}_{\mathrm{TG}}\omega(\sqrt{\log m}) = \sqrt{n\log q}\cdot\omega(\sqrt{\log m})$. Let $\mathcal{D}_{m\times m}$ denote the distribution on matrices in $\mathbb{Z}^{m\times m}$ defined as $\left(\mathcal{D}_{\mathbb{Z}^m,\sigma_{\boldsymbol{R}}}\right)^m$ conditioned on the resulting matrix being $\mathbb{Z}_q$-invertible.

Algorithm SampleR(1^m). This algorithm samples matrices in $\mathbb{Z}^{m\times m}$ from a distribution that is statistically close to $\mathcal{D}_{m\times m}$.

1. Let $\boldsymbol{T}$ be the canonical basis of the lattice $\mathbb{Z}^m$.
2. For $i = 1, \ldots, m$, let $\boldsymbol{r}_i \leftarrow \mathit{SampleGaussian}\left(\mathbb{Z}^m, \boldsymbol{T}, \sigma_{\boldsymbol{R}}, 0\right)$
3. If $\boldsymbol{R}$ is $\mathbb{Z}_q$-invertible, output $\boldsymbol{R}$; otherwise repeat step 2.

Agrawal show that step 2 will need to be repeated fewer than two times in expectation for prime q in the full version of the algorithm. We now describe basis delegation algorithm which does not increase the dimensions of the underlying matrices.

Algorithm RandBasis(S, σ). On input a basis S of an m-dimensional lattice $\Lambda_q^\perp(\boldsymbol{A})$ and a Gaussian parameter $\sigma \geq ||\tilde{S}|| \cdot \omega(\sqrt{\log n})$, outputs a new basis S' of $\boldsymbol{\Lambda}_q^\perp(\boldsymbol{A})$ such that

- with overwhelming probability $|||\tilde{S}'|| \leq \sigma\sqrt{m}$ and
- up to a statistical distance, the distribution of S' does not depend on S. That is, the random variable *RandBasis*(S, σ) is statistically close to *RandBasis*($\boldsymbol{T}$, σ) for any other basis $\boldsymbol{T}$ of $\boldsymbol{\Lambda}_q^\perp(\boldsymbol{A})$ satisfying $|||\tilde{\boldsymbol{T}}|| \leq \sigma/\omega(\sqrt{\log n})$.

Algorithm BasisDel(A, R, T_A, σ). The algorithm inputs a rank n matrix $\boldsymbol{A}$ in $\mathbb{Z}_q^{n\times m}$, a $\mathbb{Z}_q$-invertible matrix $\boldsymbol{R}$ in $\mathbb{Z}_q^{n\times m}$ sampled from $\mathcal{D}_{m\times m}$ (or a product of such), a basis $\boldsymbol{T}_{\boldsymbol{A}}$ of $\boldsymbol{\Lambda}_q^\perp(\boldsymbol{A})$ and a parameter $\sigma \in \mathbb{R}_{>0}$. Let $\boldsymbol{B} := \boldsymbol{A}\boldsymbol{R}^{-1}$ in $\mathbb{Z}_q^{n\times m}$, the algorithm outputs a basis $\boldsymbol{T}_{\boldsymbol{B}}$ of $\boldsymbol{\Lambda}_q^\perp(\boldsymbol{B})$. The σ satisfies $\sigma \geq ||\widetilde{\boldsymbol{T}_{\boldsymbol{A}}}|| \cdot \sigma_{\boldsymbol{R}}\sqrt{m}\omega(\log^{3/2} m)$.

$\boldsymbol{T}_{\boldsymbol{B}}$ is distributed statistically close to the distribution *RandBasis*(T, σ) where $\boldsymbol{T}$ is an arbitrary basis of $\boldsymbol{\Lambda}_q^\perp\left(\boldsymbol{A}\boldsymbol{R}^{-1}\right)$ satisfying $||\tilde{\boldsymbol{T}}|| < \sigma/\omega(\sqrt{(\log m)})$. If $\boldsymbol{R}$ is a product of l matrices sampled from $\mathcal{D}_{m\times m}$ then bound on σ degrades to $\sigma \geq ||\widetilde{\boldsymbol{T}_{\boldsymbol{A}}}|| \cdot \left(\sigma_{\boldsymbol{R}}\sqrt{m}\omega(\log^{1/2} m)\right)^l \cdot \omega(\log m)$.

Algorithm SampleRwithBasis(A). Let $a_1, \ldots, a_m \in \mathbb{Z}_q^n$ be the m columns of the matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$.

1. Run *TrapGen*(q, n) to generate a random matrix $\boldsymbol{B} \in \mathbb{Z}_q^{n\times m}$ with rank n and a basis $\boldsymbol{T}_{\boldsymbol{B}}$ of such that $||\widetilde{\boldsymbol{T}_{\boldsymbol{B}}}|| \leq \tilde{L}_{\mathrm{TG}} = \sigma_{\boldsymbol{R}}/\omega(\sqrt{\log m})$.
2. For $i = 1, \ldots, m$, do:
   (1) Sample $\boldsymbol{r}_i \in \mathbb{Z}^m$ from *SamplePre*(B, T_B, a_i, σ_R) then $\boldsymbol{B}\boldsymbol{r}_i = a_i$ mod q and $\boldsymbol{r}_i$ is sampled from a distribution statistically close to $\mathcal{D}_{\boldsymbol{\Lambda}_q^{a_i}(\boldsymbol{B}),\sigma_{\boldsymbol{R}}}$.
   (2) Repeat step (1) until $\boldsymbol{r}_i$ is $\mathbb{Z}_q$ linearly independent of $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{i-1}$.
3. Let $\boldsymbol{R} \in \mathbb{Z}^{m\times m}$ be the matrix whose columns are $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m$. Then $\boldsymbol{R}$ has rank m over $\mathbb{Z}_q$. Output $\boldsymbol{R}$ and $\boldsymbol{T}_{\boldsymbol{B}}$. The generated basis $\boldsymbol{T}_{\boldsymbol{B}}$ satisfies $||\widetilde{\boldsymbol{T}_{\boldsymbol{B}}}|| \leq \sigma_{\boldsymbol{R}}/\omega(\sqrt{\log m})$ with overwhelming probability.

4 Our construction

In this section, we first give the concrete construction of the lattice-based t-HABE scheme in the standard model. Then we introduce a lemma to ensure the correctness for decryption.

The concrete constructions are as follows:

Setup(1^λ, 1^d, 1^f): On input a security parameter λ, attribute set size d and the maximum depth of attribute f, do:

1. Use algorithm *TrapGen*(1^λ) to select $2d$ uniformly random $n \times m$ matrices $\boldsymbol{A}_i^b \in \mathbb{Z}_q^{n\times m}$ (for all $i \in [d]$, $b \in \{0, 1\}$) together with a full rank set vectors $\boldsymbol{T}_i^b \subseteq \boldsymbol{\Lambda}_q^\perp(\boldsymbol{A}_i^b)$ such that $||\widetilde{\boldsymbol{T}_i^b}|| \leq m \cdot \omega(\sqrt{\log m})$.
2. Select a uniformly random vector $\boldsymbol{u} \in \mathbb{Z}_q^n$, $2df$ random matrices $\boldsymbol{R}_{i,j}^b \in \mathcal{D}_{m\times m}$ and $2df$ random matrices $\boldsymbol{V}_{i,j}^b \in \mathcal{D}_{m\times m}$ (for $1 \leq i \leq d$, $1 \leq j \leq f$).
3. Output the public parameters and master key

$$\mathbf{PP} = \left(\left\{\boldsymbol{A}_i^b\right\}_{i\in[d],b\in\{0,1\}}, \left\{\boldsymbol{R}_{i,j}^b\right\}_{i\in[d],j\in[f],b\in\{0,1\}}, \left\{\boldsymbol{V}_{i,j}^b\right\}_{i\in[d],j\in[f],b\in\{0,1\}}, \boldsymbol{u}\right)$$

$$\mathbf{MK} = \left(\left\{\boldsymbol{T}_i^b\right\}_{i\in[d],b\in\{0,1\}}\right)$$

Extract(PP, $\mathrm{SK}_{\mathrm{Att}_1}$, Att_2, k): On input public parameter PP, a private key $\mathrm{SK}_{\mathrm{Att}_1}$ which is associated with Att_1, the attribute set $\mathrm{Att}_2 = \{\mathrm{att}_1, \ldots, \mathrm{att}_h\}$ and threshold k. For attribute att_i, we denote 'high' level attribute j as $\mathrm{att}_{j/l} = (\mathrm{att}_{j,0}, \mathrm{att}_{j,1}, \ldots, \mathrm{att}_{j,l}) \in \{0, 1\}^{l+1}$ belong to attribute set Att_1, and the 'low'

doi: 10.1049/iet-ifs.2013.0111

level attribute j denote as $\text{att}_{j/c} = (\text{att}_{j,0}, \text{att}_{j,1}, \ldots, \text{att}_{j,l}, \ldots \text{att}_{j,c}) \in \{0,1\}^{c+1}$ belong to attribute set Att_2, where $c < f$.

Define $\boldsymbol{S}^1_{j|0} = \boldsymbol{T}^1_j$, $\boldsymbol{F}^1_{j|0} = \boldsymbol{A}^1_j$, $\boldsymbol{S}^0_{j|0} = \boldsymbol{T}^0_A$, $\boldsymbol{F}^0_{j|0} = \boldsymbol{A}^0_j$, $\boldsymbol{F}^1_{j|l} = \boldsymbol{A}^1_j \left(\boldsymbol{R}^{\text{att}_{j,1}}_{j,1}\right)^{-1} \left(\boldsymbol{R}^{\text{att}_{j,2}}_{j,2}\right)^{-1} \cdots \left(\boldsymbol{R}^{\text{att}_{j,l}}_{j,l}\right)^{-1}$, $\boldsymbol{F}^0_{j|l} = \boldsymbol{A}^0_j \left(\boldsymbol{V}^{\text{att}_{j,1}}_{j,1}\right)^{-1} \left(\boldsymbol{V}^{\text{att}_{j,2}}_{j,2}\right)^{-1} \cdots \left(\boldsymbol{V}^{\text{att}_{j,l}}_{j,l}\right)^{-1}$. Compute $\boldsymbol{R}_j = \left(\boldsymbol{R}^{\text{att}_{j,l+1}}_{j,l+1}\right) \cdots \left(\boldsymbol{R}^{\text{att}_{j,c}}_{j,c}\right)$, $\boldsymbol{F}^1_{j|c} = \boldsymbol{F}^1_{j|l} \boldsymbol{R}^{-1}_j$, $\boldsymbol{V}_j = \left(\boldsymbol{V}^{\text{att}_{j,l+1}}_{j,l+1}\right) \cdots \left(\boldsymbol{V}^{\text{att}_{j,c}}_{j,c}\right)$, $\boldsymbol{F}^0_{j|c} = \boldsymbol{F}^0_{j|l} \boldsymbol{V}^{-1}_j$.

Run $\boldsymbol{S}^1_{j|c} \leftarrow \textit{BasisDel}\left(\boldsymbol{F}^1_{j|l}, \boldsymbol{R}_j, \boldsymbol{S}^1_{j|l}, \sigma_l\right)$ to obtain a short random basis for $\boldsymbol{\Lambda}^{\perp}_q\left(\boldsymbol{F}^1_j\right)$. Run $\boldsymbol{S}^0_{j|c} \leftarrow \textit{BasisDel}\left(\boldsymbol{F}^0_{j|l}, \boldsymbol{V}_j, \boldsymbol{S}^0_{j|l}, \sigma_l\right)$ to obtain a short random basis for $\boldsymbol{\Lambda}^{\perp}_q\left(\boldsymbol{F}^0_j\right)$. Construct l shares of $\boldsymbol{u} = (u_1, \ldots, u_n) \in \mathbb{Z}^n_q$ using a Shamir secret-sharing scheme applied to each coordinate of $\boldsymbol{u}$ independently. Namely for each $k \in [n]$, choose a uniformly construct the jth share vector

$$\boldsymbol{u} = \left(\hat{u}_{j,1}, \ldots, \hat{u}_{j,n}\right) = \left(p_1(j), p_2(j), \ldots, p_n(j)\right) \in \mathbb{Z}^n_q$$

Looking ahead (to decryption), note that for all $J \subset [d]$ such that $|J| \geq k$, we can compute fractional Lagrangian coefficients L_j such that $\boldsymbol{u} = \sum_{j \in J} L_j \cdot \hat{\boldsymbol{u}}_j (\text{mod}\, q)$. That is , we interpret L_j as a fraction of integers, which we can also evaluate (mod$q$). Using trapdoor MK and the algorithm *SamplePre* from Section 3.3.2, find $\boldsymbol{e}_j$ such that $\boldsymbol{F}^{\text{att}_{j,0}}_j \boldsymbol{e}_j = \hat{\boldsymbol{u}}_i$. Output the secret key for attribute set with hierarchy as

$$\text{SK}_{\text{Att}_2} = \left(\text{Att}_2, \{e_1, \ldots, e_l\}, \left\{\mathbf{S}^b_{j|c_j}\right\}_{j \in [d], b \in \{0,1\}}\right)$$

Encrypt(PP, Att', b): On input public parameter PP, an attitude set Att', and a message bit $b \in \{0, 1\}$ do,

1. For all $i \in [d]$, compute $\boldsymbol{F}^1_i = \boldsymbol{A}^1_i \left(\boldsymbol{R}^{\text{att}'_{i,1}}_{i,1}\right)^{-1} \left(\boldsymbol{R}^{\text{att}'_{i,2}}_{i,2}\right)^{-1} \cdots \left(\boldsymbol{R}^{\text{att}'_{i,c'_j}}_{i,c'_j}\right)^{-1}$, $\boldsymbol{F}^0_i = \boldsymbol{A}^0_i \left(\boldsymbol{V}^{\text{att}'_{i,1}}_{i,1}\right)^{-1} \left(\boldsymbol{V}^{\text{att}'_{i,2}}_{i,2}\right)^{-1} \cdots \left(\boldsymbol{V}^{\text{att}'_{i,c'_j}}_{i,c'_j}\right)^{-1}$ and let $D = (d!)^2$.
2. Choose a uniformly random $\boldsymbol{s} \in \mathbb{Z}^n_q$, a noise term $x \leftarrow \chi_{\{\alpha, q\}}$ and $\boldsymbol{x}_i \leftarrow \chi_{\{\alpha,q\}^m}$. Set $c_0 \leftarrow \boldsymbol{u}^{\mathbf{T}} \boldsymbol{s} + Dx + b\left\lfloor \frac{q}{2} \right\rfloor \in \mathbb{Z}_q$, $\boldsymbol{c}_i \leftarrow \left(\boldsymbol{F}^{\text{Att}'_{i,0}}_i\right)^{\mathbf{T}} \boldsymbol{s} + Dx_i \in \mathbb{Z}^m_q$ for all $i \in [d]$. Output the ciphertext

$$\text{CT}_{\text{Att}'} = \left(c_0, \{c\}_{i\, i \in [d]}, \text{Att}'\right)$$

Decrypt(PP, $\text{CT}_{\text{Att}'}$, SK_{Att}): On input parameters PP, a private key SK_{Att} and a ciphertext $\text{CT}_{\text{Att}'}$.

1. Let $J \subset [d]$ denotes the set of matching attributes in Att and Att', that is, $i \in J$, $\text{att}_{i,0} = \text{Att}'_{i,0}$. For all $j \in J$, checks whether $c_j \leq c'_j$. If $c_j > c'_j$, we remove j'th attribute from set J. Finally, we check $|J|$ exceed the threshold k. If $|J| < k$, output $\perp$. Otherwise, if all attributes $c_j \neq c'_j$, we use the *Extract* algorithm to generate $\text{SK}_{j/c'}$ and construct the matrix $\boldsymbol{F}_j$. Then we can compute fractional Lagrangian coefficient L_j, so that

$$\sum_{j \in J} L_j \boldsymbol{F}_j \boldsymbol{e}_j = \boldsymbol{u} (\text{mod}\, q)$$

2. Compute $r = \boldsymbol{c}_0 - \sum_{j \in J} L_j \cdot \boldsymbol{e}^{\mathrm{T}}_j \boldsymbol{c}_j (\text{mod}\, q)$, view it as the interger $r \in [-\lfloor q/2 \rfloor, \lfloor q/2 \rfloor) \subset \mathbb{Z}$.
3. If $|r| < q/4$, output 0, else output 1.

Correctness

In order to decrypt the ciphertext correctly, we only need to consider the case $|J| \geq k$. Let L_j be the fractional Lagrangian coefficients as described above. Then

$$\begin{aligned} r &= \boldsymbol{c}_0 - \sum_{j \in J} L_j \boldsymbol{e}^{\mathrm{T}}_j \boldsymbol{c}_j (\text{mod}\, q) \\ &= \boldsymbol{u}^{\mathrm{T}} \boldsymbol{s} + Dx + b\left\lfloor \frac{q}{2} \right\rfloor - \sum_{j \in J} L_j \boldsymbol{e}^{\mathrm{T}}_j \left((\boldsymbol{F})^{\mathrm{T}}_j \boldsymbol{s} + Dx_j\right) (\text{mod}\, q) \\ &= b\left\lfloor \frac{q}{2} \right\rfloor + \left(\boldsymbol{u}^{\mathrm{T}} \boldsymbol{s} + \sum_{j \in J} \left(L_j \boldsymbol{F}_j \boldsymbol{e}_j\right)^{\mathbf{T}} \boldsymbol{s}\right) \\ &\quad + \left(Dx - \sum_{j \in J} D\left(L_j \boldsymbol{e}_j\right)^{\mathbf{T}} \boldsymbol{x}_j\right) (\text{mod}\, q) \\ &\simeq b\left\lfloor \frac{q}{2} \right\rfloor \end{aligned}$$

It is suffices to set the parameters so that with overwhelming probability

$$\left| D\boldsymbol{x} - \sum_{j \in J} D L_j \boldsymbol{e}^{\mathrm{T}}_j \boldsymbol{x}_j \right| \leq D|\boldsymbol{x}| + \sum_{j \in J} D^2 \left| \boldsymbol{e}^{\mathrm{T}}_j \boldsymbol{x}_j \right| < q/4$$

For the first inequality, we use the following lemma proposed in [19] to state that the number are integers bounded above by $D^2 \leq (d!)^2$.

Lemma 2: Let $D = (d!)^2$. Given $k \leq d$ number $I_1, \ldots, I_k \in [1, \ldots, d]$, define the Lagrangian coefficients $L_j = \prod_{i \neq j} \frac{-I_i}{\left(I_j - I_i\right)}$

Then, for every $1 \leq j \leq k$, DL_j is an interger, and $|DL_j| \leq D^2 \leq (d!)^4$.

5 Security and analysis

5.1 Security proof

In this subsection, we show our lattice-based t-HABE scheme can ensure the ciphertext securty by using the LWE problem. We proof our scheme is selective secure against chosen attribute set attacks and chosen plaintext attack by giving following theorem.

Theorem 2: Let $\mathcal{A}$ be a PPT adversary with advantage $\varepsilon > 0$ against the selective security (IND-sAtr-CPA) game for t-HABE scheme, then there exists a PPT algorithm $\mathcal{B}$ that decides the LWE problem with advantage $\varepsilon/(d+1)$.

Proof: The LWE problem is provided as a sampling oracle $\mathcal{O}$ which can be either truly random oracle $\mathcal{O}_{\$}$ or noisy pseudo-random $\mathcal{O}_s$ for some secret key $\boldsymbol{s} \in \mathbb{Z}^n_p$. The

simulation between $\mathcal{A}$ and $\mathcal{B}$ can be created in the following way:

Instance. $\mathcal{B}$ requests from $\mathcal{O}$ and receives $(lm+1)$ LWE samples that we denote as $(\boldsymbol{w}_1, v_1)$, $\{\{(\boldsymbol{w}_1^1, v_1^1), (\boldsymbol{w}_1^2, v_1^2), \ldots, (\boldsymbol{w}_1^m, v_1^m)\}, \ldots., \{(\boldsymbol{w}_d^1, v_d^1), (\boldsymbol{w}_d^2, v_d^2), \ldots, (\boldsymbol{w}_d^m, v_d^m)\}\} \in \left\{\mathbb{Z}_q^n \times \mathbb{Z}_q\right\}^{(dm+1)}$.
Targeting: $\mathcal{A}$ announces to $\mathcal{B}$ the attribute set it intends to attack, namely $\mathrm{Att}^* = \{\mathrm{att}_1^*, \ldots, \mathrm{att}_d^*\}$.
Setup: $\mathcal{B}$ constructs the systems public parameters PP as follows:

1. The d matrices $\boldsymbol{A}'^{\mathrm{att}_i^*}_i$, $i \in [d]$ are chosen from the LWE change challenge $\{(\boldsymbol{w}_i^1, \boldsymbol{w}_i^2, \ldots, \boldsymbol{w}_i^m)\}_{i\in[d]}$.

The d matrices $\boldsymbol{A}'^{\overline{\mathrm{att}_i^*}}_i$, $i \in [d]$, are chosen using *TrapGen* with a trapdoor $\boldsymbol{T}'^{\overline{\mathrm{att}_i^*}}_i$.

It then sample random matrices $\boldsymbol{R}_{i,1}^{\mathrm{att}_{i,1}^*}, \boldsymbol{R}_{i,2}^{\mathrm{att}_{i,2}^*} \ldots \boldsymbol{R}_{i,f_i}^{\mathrm{att}_{i,f_i}^*} \in \mathbb{Z}^{m\times m}$, $\boldsymbol{V}_{i,1}^{\mathrm{att}_{i,1}^*}, \boldsymbol{V}_{i,2}^{\mathrm{att}_{i,2}^*} \ldots \boldsymbol{V}_{i,f_i}^{\mathrm{att}_{i,f_i}^*} \in \mathbb{Z}^{m\times m}$ from the distribution $\mathcal{D}_{m\times m}$. Set $\boldsymbol{A}_i^{\mathrm{att}_{i,0}^*} = \boldsymbol{A}'^{\mathrm{att}_{i,0}^*}_i \boldsymbol{R}_{i,f_i}^{\mathrm{att}_{i,f_i}^*} \cdots \boldsymbol{R}_{i,2}^{\mathrm{att}_{i,2}^*} \boldsymbol{R}_{i,1}^{\mathrm{att}_{i,1}^*}$, $\boldsymbol{A}_i^{\overline{\mathrm{att}_{i,0}^*}} = \boldsymbol{A}'^{\mathrm{att}_{i,0}^*}_i \boldsymbol{V}_{i,f_i}^{\mathrm{att}_{i,f_i}^*} \ldots \boldsymbol{V}_{i,2}^{\mathrm{att}_{i,2}^*} \boldsymbol{V}_{i,1}^{\mathrm{att}_{i,1}^*}$.
Now consider the d matrices and sets $\boldsymbol{F}_i^{\mathrm{att}_{i,0}^*} = \boldsymbol{A}_i^{\mathrm{att}_{i,0}^*} \left(\boldsymbol{R}_{i,1}^{\mathrm{att}_{i,1}^*}\right)^{-1} \left(\boldsymbol{R}_{i,2}^{\mathrm{att}_{i,2}^*}\right)^{-1} \ldots \left(\boldsymbol{R}_{i,j}^{\mathrm{att}_{i,j}^*}\right)^{-1}$ for $j = 0, \ldots, (f_i - 1)$, set $\boldsymbol{F}_i^{\overline{\mathrm{att}_{i,0}^*}} = \boldsymbol{A}_i^{\overline{\mathrm{att}_{i,0}^*}} \left(\boldsymbol{V}_{i,1}^{\mathrm{att}_{i,1}^*}\right)^{-1} \left(\boldsymbol{V}_{i,2}^{\mathrm{att}_{i,2}^*}\right)^{-1} \ldots \left(\boldsymbol{V}_{i,j}^{\mathrm{att}_{i,j}^*}\right)^{-1}$ for $j = 0, \ldots, (f_i - 1)$.
Invoke SampleRwithBasis $\left(\boldsymbol{F}_i^{\mathrm{att}_{i,0}^*}\right)$ to generate a matrix $\boldsymbol{R} \sim \mathcal{D}_{m\times m}$ and short basis $\boldsymbol{T}_i^{\mathrm{att}_{i,0}^*}$ For $\boldsymbol{\Lambda}_q^{\perp}\left(\boldsymbol{B}_i^{\mathrm{att}_{i,0}^*} = \boldsymbol{F}_i^{\mathrm{att}_{i,0}^*}(\boldsymbol{R})^{-1}\right)$. Return $\boldsymbol{R}_{i,j}^{1-\mathrm{att}_{i,0}^*} \leftarrow \boldsymbol{R}$.
Invoke SampleR(1^m) to generate a matrix $\boldsymbol{V} \sim \mathcal{D}_{m\times m}$. Run $\mathit{BasisDel}\left(\boldsymbol{A}'^{\overline{\mathrm{att}_i^*}}_i, \boldsymbol{V}_{i,f_i}^{\mathrm{att}_{i,f_i}^*} \ldots \boldsymbol{V}_{i,j+2}^{\mathrm{att}_{i,j+2}^*} \boldsymbol{V}_{i,j+1}^{\mathrm{att}_{i,j+1}^*}, \boldsymbol{V}, \boldsymbol{T}'^{\overline{\mathrm{att}_{i,0}^*}}_i, \sigma\right)$ to generate $\boldsymbol{T}_i^{\overline{\mathrm{att}_{i,0}^*}}$. Return $\boldsymbol{V}_{i,j}^{1-\mathrm{att}_{i,0}^*} \leftarrow \boldsymbol{V}$.
The vector $\boldsymbol{u}$ is constructed from the LWE challenge $\boldsymbol{u} = \boldsymbol{w_1}$. The public parameters are returned to the adversary.
Queries. $\mathcal{B}$ answers each private-key extraction query for attribute set as follows: $\mathcal{B}$ answers a query on, for every attribute in the attribute sets $\mathrm{Att}^* = \{\mathrm{att}_1^*, \ldots, \mathrm{att}_h^*\}$, which of length as follow:

1. For attribute $i \in$ Att, let $j \in [r]$ be the shallowest level at which $\mathrm{att}_{i,j} \neq \mathrm{att}_{i,j}^*$. If $\mathrm{att}_{i,j} = \mathrm{att}_{i,j}^*$ for all $j \in [r]$, the simulation abort and failed.
2. By construction $\boldsymbol{B}_i^{\mathrm{att}_{i,0}^*} = \boldsymbol{A}_i^{\mathrm{att}_{i,0}^*} \cdot \left(\boldsymbol{R}_{i,1}^{\mathrm{att}_{i,1}^*}\right)^{-1} \ldots \left(\boldsymbol{R}_{i,j-1}^{\mathrm{att}_{i,j-1}^*}\right)^{-1} \left(\boldsymbol{R}_{i,j}^{\mathrm{att}_{i,j}}\right)^{-1}$, $\boldsymbol{B}_i^{\overline{\mathrm{att}_{i,0}^*}} = \boldsymbol{A}_i^{\overline{\mathrm{att}_{i,0}^*}} \cdot \left(\boldsymbol{V}_{i,1}^{\mathrm{att}_{i,1}^*}\right)^{-1} \ldots \left(\boldsymbol{V}_{i,j-1}^{\mathrm{att}_{i,j-1}^*}\right)^{-1} \left(\boldsymbol{V}_{i,j}^{\mathrm{att}_{i,j}}\right)^{-1}$.
3. Run $\mathit{BasisDel}\left(\boldsymbol{B}_i^{\mathrm{att}_{i,0}^*}, \boldsymbol{R}_{i,l}^{\mathrm{att}_{i,l}} \ldots \boldsymbol{R}_{i,j+2}^{\mathrm{att}_{i,j+2}} \boldsymbol{R}_{i,j+1}^{\mathrm{att}_{i,j+1}}, \boldsymbol{T}_i^{\mathrm{att}_{i,0}^*}, \sigma_j\right)$ to generate the short basis for attribute att_i. Run $\mathit{BasisDel}\left(\boldsymbol{B}_i^{\overline{\mathrm{att}_{i,0}^*}}, \boldsymbol{V}_{i,l}^{\mathrm{att}_{i,l}} \ldots \boldsymbol{V}_{i,j+2}^{\mathrm{att}_{i,j+2}} \boldsymbol{V}_{i,j+1}^{\mathrm{att}_{i,j+1}}, \boldsymbol{T}_i^{\overline{\mathrm{att}_{i,0}^*}}, \sigma_j\right)$ to generate the short basis for attribute $\overline{\mathrm{att}_i}$
4. Let $\mathrm{Att} \cap \mathrm{Att}^*: = I \subset [d]$ and let $|I| = t < k$. Then, note that $\mathcal{B}$ has trapdoors for the matrices corresponding to the set $\bar{I}$, where $|\bar{I}| = l - t$. W.l.o.g., we assume that the first t attribute of $\mathrm{att}_{i,0}$ are equal to $\mathrm{att}_{i,0}^*$.
5. Represent the share of u symbolically as $\hat{\boldsymbol{u}}_i = \boldsymbol{u} + \boldsymbol{a}_1 i + \boldsymbol{a}_2 i^2 + \cdots + \boldsymbol{a}_{k-1} i^{k-1}$ where $\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_{k-1}$ are vector variables of length n each.
6. For $i \in [t]$ s.t. $\mathrm{att}_{i,0}^* = \mathrm{att}_{i,0}$, pick $\boldsymbol{e}_i$ using $\mathit{SamplePre}\left(\boldsymbol{B}_j^{\mathrm{att}_{j,0}}, \boldsymbol{T}_j^{\mathrm{att}_{j,0}}, \hat{\boldsymbol{u}}_j, \sigma\right)$ s.t. $\hat{\boldsymbol{u}}_i = \boldsymbol{B}_i^{\mathrm{att}_{i,0}} \boldsymbol{e}_i$.
7. Since $t \leq k-1$, and there are $k-1$ variables $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{k-1}$ by choosing $k-1-t$ shares $\hat{\boldsymbol{u}}_{t+1}, \ldots, \hat{\boldsymbol{u}}_{k-1}$ randomly, the values for $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{k-1}$ are determined. This determines all d share $\hat{\boldsymbol{u}}_1, \ldots, \hat{\boldsymbol{u}}_d$.
8. To find $\boldsymbol{e}_j$ s.t. $\boldsymbol{B}_j^{\mathrm{att}_{j,0}} \boldsymbol{e}_j = \hat{\boldsymbol{u}}_j$ for $j = t+1, \ldots, d$, invoke $\mathit{SamplePre}\left(\boldsymbol{B}_j^{\mathrm{att}_{j,0}}, \boldsymbol{T}_j^{\mathrm{att}_{j,0}}, \hat{\boldsymbol{u}}_j, \sigma\right)$.
9. Return $(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_j)$.

Note that the distribution of the public parameters and keys in the real scheme is statistically indistinguishable from that in the simulation.
Challenge. $\mathcal{A}$ outputs a message bit $b^* \in \{0, 1\}$. $\mathcal{B}$ responds with a challenge ciphertext for attribute set Att*:

1. Let $\boldsymbol{c}_0 = Dv_1 + b\lfloor q/2 \rfloor$.
2. Let $\boldsymbol{c}_i = \left(Dv_i^1, Dv_i^2, \ldots, Dv_i^m\right)$ for $i \in [d]$.

Guess. The adversary $\mathcal{A}$ outputs a guess b'. The simulator $\mathcal{B}$ uses that guess to determine an answer on the LWE oracle, output 'genuin'. If $b' = b$, else output 'random'. □

5.2 Parameters

In order to ensure that decryption of the cryptosystem works with high probability and the security reductions are meaningful. The number of attributes are d and the level of attribute is f. Our parameters are set under the following constraints:

For the correctness to hold, we need to satisfy (1). Since $D = (d!)^2$ and $||\boldsymbol{e}_j|| \leq \tau_f \sqrt{m} = \sigma_f m \omega\left(\sqrt{\log m}\right)$, we have

$$
\begin{aligned}
&D|x| + \sum_{j\in J} D^2 \left|\boldsymbol{e}_j^{\mathrm{T}} \boldsymbol{x}_j\right| \\
&\quad \leq D \cdot q\alpha_f \omega\left(\sqrt{\log m}\right) + d \cdot D^2 q\alpha_f \sigma_f m \omega(\log m) \\
&\qquad + d \cdot D^2 \sigma_f m^{3/2} \omega\left(\sqrt{\log m}\right) \\
&\quad = (d!)^2 \cdot q\alpha_f \omega\left(\sqrt{\log m}\right) + d \cdot (d!)^4 q\alpha_f \sigma_l m \omega(\log m) \\
&\qquad + d \cdot (d!)^4 \sigma_f m^{3/2} \omega\left(\sqrt{\log m}\right) \\
&\quad \leq (d)^{2d} \cdot q\alpha_f \omega\left(\sqrt{\log m}\right) + (d)^{4d+1} q\alpha_f \sigma_f m \omega(\log m) \\
&\qquad + (d)^{4d+1} \sigma_f m^{3/2} \omega\left(\sqrt{\log m}\right)
\end{aligned}
$$

where we use the fact $(d!)^2 \leq (d)^{2d}$.

For the lattice *TrapGen* algorithm can operate, we need $m \geq 5n\log q$. To ensure the correctness, it set $\alpha_f < \left[(d)^{2d} q\omega\left(\sqrt{\log m}\right) + (d)^{4d+1} q\sigma_f m \omega(\log m)\right]^{-1}$, $q > (d)^{4d+1} \sigma_f m^{3/2} \omega\left(\sqrt{\log m}\right)$. Use properties of *RandBasis*$(\cdot, \sigma)$ the Gram-Schmidt norm of a short basis $\boldsymbol{T}_{i,f}$ at level f satisfies w.h.p. $\|\widetilde{\boldsymbol{T}_{i,f}}\| \leq \sigma_f \sqrt{m}$, the *BasisDel* used in *Extract* can operate $\sigma_f > \|\boldsymbol{T}_{f-1}\| \sigma_{\boldsymbol{R}} \sqrt{m} \omega\left(\log^{3/2} m\right)$ which follows from

doi: 10.1049/iet-ifs.2013.0111

$\sigma_f > \sigma_{f-1} m^{3/2} \omega(\log^2 m)$ and Regev's reduction applies $q > 2\sqrt{n}/\alpha_f$ for all f.

5.3 Scheme analysis

In this subsection, we compare lattice-based t-HABE scheme with existing schemes to indicate that our scheme have superior advantage over the existing ones. At first, Boneh and Franklin [24] used weil pairing to construct the IBE scheme in the random oracle model. Later, Gentry and Silverberg [25] proposed HIBE which is a generalisation of IBE whose ID is organisational hierarchy. Agrawal *et al.* [15] used lattice to construct HIBE scheme in the standard model. All the schemes mentioned above do not have the error-tolerance property. In order to solve this problem, FIBE(t-ABE) was constructed by Sahai and Waters [2] while lattice-based FIBE was created by Agrawal *et al.* [19] which can resist to quantum cryptanalysis. Soon after that, lattice-based ABE scheme was proposed by Boyen [20], but this scheme cannot achieve hierarchy attributes. Li *et al.* [8] constructed HABE scheme and proved its security in random oracle model, but this scheme is based on paring operation which could not resist quantum computing. In this paper, lattice-based t-HABE was proposed and its security was reduced to LWE problem in the standard model. As generalisation of ABE scheme, ours can achieve error-tolerance property, hierarchy property and quantum cryptanalysis resist property. We make the comparison to list the table below.

Scheme/ functionality	Model	Hierarchy	Error-tolerance	Quantum cry-resist
IBE [24]	R.O	no	no	no
HIBE [25]	R.O	yes	no	no
Lattice-based HIBE [15]	standard	yes	no	yes
FIBE/t-ABE [2]	standard	no	yes	no
Lattice-based FIBE [19]	standard	no	yes	yes
Lattice-based ABE [20]	standard	no	yes	yes
HABE [8]	standard	yes	yes	no
Ours	standard	yes	yes	yes

6 Conclusions

In order to decrease the number of attributes used to encrypt and decrypt the document, hierarchical attribute is introduced to ABE. In this paper, lattice-based threshold HABE scheme is proposed as a post-quantum cryptosystem. This scheme has proved to be selectively secure in the standard model from the hardness of the LWE problem. To our best knowledge, our scheme is the first realisation of HABE from lattices.

7 Acknowledgments

This research is supported by the Changjiang Scholars and Innovative Research Team in University under Grant no. IRT1078; the Key Program of NSFC-Guangdong Union Foundation under Grant no. U1135002; Major national S&T program under Grant no. 2011ZX03005-002; the Fundamental Research Funds for the Central Universities under Grant no. JY10000903001 and the National Natural Science Foundation of China (61303218, 61370078). The authors thank the sponsors for their support and the reviewers for helpful comments.

8 References

1 Goyal, V., Pandey, O., Sahai, A., Waters, B.: 'Attribute-based encryption for fine-grained access control of encrypted data'. Proc. 13th ACM Conf. on Computer and Communications Security, 2006, pp. 89–98
2 Sahai, A., Waters, B.: 'Fuzzy identity-based encryption'. Advances in Cryptology–EUROCRYPT 2005, 2005, pp. 557–557
3 Bethencourt, J., Sahai, A., Waters, B.: 'Ciphertext-policy attribute-based encryption'. IEEE Symp. on Security and Privacy, 2007 (SP'07), 2007, pp. 321–334
4 Cheung, L., Newport, C.: 'Provably secure ciphertext policy abe'. Proc. 14th ACM Conf. on Computer and Communications security, 2007, pp. 456–465
5 Ostrovsky, R., Sahai, A., Waters, B.: 'Attribute-based encryption with non-monotonic access structures'. Proc. 14th ACM Conf. on Computer and Communications Security, 2007, pp. 195–203
6 Katz, J., Sahai, A., Waters, B.: 'Predicate encryption supporting disjunctions, polynomial equations, and inner products'. EUROCRYPT, 2008, pp. 146–162
7 Waters, B.: 'Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization'. Public Key Cryptography (PKC 2011), 2011, pp. 53–70
8 Li, J., Wang, Q., Wang, C., Ren, K.: 'Enhancing attribute-based encryption with attribute hierarchy', *Mob. Netw. Appl.*, 2011, **16**, (5), pp. 553–561
9 Regev, O.: 'Lattice-based cryptography'. Advances in Cryptology-CRYPTO 2006, 2006, pp. 131–141
10 Ajtai, M.: 'Generating hard instances of lattice problems (extended abstract)'. STOC, 1996, pp. 99–108
11 Micciancio, D.: 'Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions'. Proc. 43rd Annual IEEE Symp. on Foundations of Computer Science, 2002, 2002, pp. 356–365
12 Ajtai, M., Dwork, C.: 'A public-key cryptosystem with worst-case/average-case equivalence'. Proc. 29th annual ACM Symp. on Theory of Computing, 1997, pp. 284–293
13 Regev, O.: 'New lattice-based cryptographic constructions', *J. ACM*, 2004, **51**, (6), pp. 899–942
14 Regev, O.: 'On lattices, learning with errors, random linear codes, and cryptography'. Proc. 37th Annual ACM Symp. on Theory of Computing, 2005, pp. 84–93
15 Agrawal, S., Boneh, D., Boyen, X.: 'Efficient lattice (h)ibe in the standard model'. EUROCRYPT, 2010, pp. 553–572
16 Agrawal, S., Boneh, D., Boyen, X.: 'Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical ibe'. CRYPTO, 2010, pp. 98–115
17 Cash, D., Hofheinz, D., Kiltz, E., Peikert, C.: 'Bonsai trees, or how to delegate a lattice basis'. Advances in Cryptology–EUROCRYPT 2010, 2010, pp. 523–52
18 Gentry, C., Peikert, C., Vaikuntanathan, V.: 'Trapdoors for hard lattices and new cryptographic constructions'. Proc. 40th Annual ACM Symp. on Theory of Computing, 2008, pp. 197–206
19 Agrawal, S., Boyen, X., Vaikuntanathan, V., Voulgaris, P., Wee, H.: 'Fuzzy identity based encryption from lattices'. IACR Cryptology ePrint Archive, 2011:414, 2011
20 Boyen, X.: 'Attribute-based functional encryption on lattices'. TCC, 2013, pp. 122–142
21 Ajtai, M.: 'Generating hard instances of the short basis problem'. ICALP, 1999, pp. 1–9
22 Alwen, J., Peikert, C.: 'Generating shorter bases for hard random lattices', *Theory Comput. Syst.*, 2011, **48**, (3), pp. 535–553
23 Peikert, C.: 'Public-key cryptosystems from the worst-case shortest vector problem: extended abstract'. STOC, 2009, pp. 333–342
24 Boneh, D., Franklin, M.: 'Identity-based encryption from the weil pairing'. Advances in CryptologyCRYPTO 2001, 2001, pp. 213–229
25 Gentry, C., Silverberg, A.: 'Hierarchical id-based cryptography'. Advances in Cryptology'ASIACRYPT 2002, 2002, pp. 548–566