Efficient Lattice HIBE in the Standard Model with Shorter Public Parameters

Kunwar Singh[1], C. Pandu Rangan[2], and A.K. Banerjee[3]

[1] Computer Science and Engineering Department
NIT Trichy, Tiruchirappalli, India
kunwar@nitt.edu

[2] Computer Science and Engineering Department
IIT, Madras
rangan@cse.iitm.ac.in

[3] Mathematics Department
NIT Trichy, Tiruchirappalli, India
banerjee@nitt.edu

Abstract. The concept of identity-based cryptosystem was introduced by Adi Shamir in 1984. In this new paradigm users' public key can be any string which uniquely identifies the user. The task of Public Key Generator (PKG) in IBE is to authenticate identity of the entity, generate the private key corresponding to the identity of the entity and finally transmit the private key securely to the entity. In large network PKG has a burdensome job. So the notion of Hierarchical IBE (HIBE) was introduced in [11,12] to distribute the workload by delegating the capability of private key generation and identity authentication to lower-level PKGs. In Eurocrypt 2010 Agrawal et al [1] presented an efficient lattice based secure HIBE scheme in the standard model in weaker security notion i.e. selective-ID. Based on [1], Singh et al [18] constructed adaptive-ID secure HIBE with short public parameters and still the public parameters is very large (total $l'' \times h+2$ matrices). In this paper, we have reduced the size of the public parameters from $l'' \times h+2$ matrices to $l''+2$ matrices using Chatterjee and Sarkar's [8] and blocking technique [7], where h is the number of levels in HIBE.

Keywords: Lattice, Hierarchical Identity Base Encryption (HIBE), Learning With Error (LWE).

1 Introduction

The concept of identity-based cryptosystem was introduced by Adi Shamir in 1984 [16]. In this new paradigm, users' public key can be any string which uniquely identifies the user. For example, users' identifier information such as email, phone number and IP address can be public key. As a result, it significantly reduces cost and complexity of establishing public key infrastructure (PKI). Although Shamir constructed an identity-based signature scheme using RSA function but was not able to construct an identity-based encryption scheme and this remained open problem until 2001, when this open problem was independently solved by Boneh-Franklin [5] and Cocks [9].

The task of Public Key Generator (PKG) in IBE is to authenticate identity of the entity, generate the private key corresponding to identity of the entity and finally transmit the private key securely to the entity. In large network PKG has a burdensome job.

Linawati et al. (Eds.): ICT-EurAsia 2014, LNCS 8407, pp. 542–553, 2014.

So the notion of Hierarchical IBE (HIBE) was introduced in [11,12] to distribute the workload by delegating the capability of private key generation and identity authentication to lower-level PKGs. However, lower level PKGs do not have their own public parameters. Only root PKG has some set of public parameters.

In 1994, Peter Shor in his seminal paper showed that prime factorization and discrete logarithm problem can be solved in polynomial time on a quantum computer. In other words, once quantum computer comes into reality all of the public-key algorithms used to protect the Internet [20] will be broken. It facilitated research on new cryptosystems that remain secure in the advent of quantum computers. Till now there is no polynomial time quantum algorithm for lattice based problems. Ajtai's seminal result on the average case / worst case equivalence sparked great interest in lattice based cryptography. Informally, it means breaking the lattice based cryptosystem in the average case is as hard as solving some lattice based hard problems in the worst case. So it gives strong hardness guarantee for the lattice hard problems. Recently Regev [15] defined the learning with errors (LWE) problem and proved that, it also enjoys similar average case / worst case equivalence hardness properties.

Related Work. Recently Cash et al [6] and Peikert [14] have constructed secure HIBE in the standard model using basis delegation technique. Their construction considers an identity as a bit string and then assign a matrix coresponding to each bit. Agarwal et al [1] constructed an efficient lattice based secure HIBE scheme in the standard model in weaker security notion i.e. selective-ID. They have considered identities as one block rather than bit-by-bit. Singh et al [18] appllied Waters's [19] idea to convert Agrawal et al [1] selective-ID secure lattice HIBE to adaptive-ID secure HIBE then they have reduced the public parameters by using Chatterjee and Sarkar's [7] blocking technique. Blocking technique is to divide an l'-bit identity into l'' blocks of l'/l'' so that size of the vector $\overrightarrow{V}$ can be reduced from l' elements of G to l'' elements of G. Still the public parameters is very large (total $l'' \times h + 2$ matrices).

Our Contributions. In this paper first, we apply Waters's [19] idea to convert Agrawal et al [1] selective-ID secure lattice HIBE to adaptive-ID secure HIBE. With this technique, for an h-level HIBE has public parameters as $A_{1,1},\ldots,A_{1,l'},A_{2,1},\ldots,A_{2,l'},\ldots,A_{h,1},\ldots,A_{h,l'}$ and A_0, B. Here the public parameters is very large (total $l \times h + 2$ matrices). Similar to Chatterjee and Sarkar [8] we have used same public parameters $A_1,\ldots,A_{l'}$ for all levels. This way public parametrs is reduced from $l' \times h + 2$ matrices to $l' + 2$ matrices. Further we reduce the public parameters $(l' + 2)$ matrices to $l'' + 2$ matrices by using Chatterjee and Sarkar's [7] blocking technique. Size of the public parameter in Singh et al [18] scheme is $l'' \times h + 2$ matrices. In our present scheme we have reduced the public parameters to $l'' + 2$ matrices.

2 Preliminaries

2.1 Notation

We denote $[j] = \{0, 1, ..., j\}$. We assume vectors to be in column form and are written using bold letters, e.g. **x**. Matrices are written as bold capital letters, e.g. **X**. The norm $||.||$ here is the standard Euclidean norm in R^n.

Gram Schmidt Orthogonalization: $\widetilde{S} := \{\widetilde{s_1}, ..., \widetilde{s_k}\} \subset R^m$ denotes the Gram-Schmidt orthogonalization of the set of linearly independently vectors $S = \{s_1, ..., s_k\} \subset R^m$. It is defined as follows: $\widetilde{s_1} = s_1$ and $\widetilde{s_i}$ is the component of s_i orthogonal to span$(s_1, ..., s_i)$ where $2 \le i \le k$. Since $\widetilde{s_i}$ is the component of s_i so $\|\widetilde{s_i}\| \le \|s_i\|$ for all i.

2.2 Hierarchical IBE

Here definition and security model of HIBE are similar to [11,12,1]. User at depth l is defined by its tuple of ids : $(id/id_l) = (id_1, ..., id_l)$. The user's ancestors are the root PKG and the prefix of id tuples (users/lower level PKGs).

HIBE consists of four algorithms.

Setup(d, λ): On input a security parameter d(maximum depth of hierarchy tree) and λ, this algorithm outputs the public parameters and master key of root PKG.

Derive(PP,$(id/id_l), SK_{(id/id_l)}$): On input public parameters PP, an identity $(id/id_l) = (id_1, ..., id_l)$ at depth l and the private key $SK_{(id/id_{l-1})}$ corresponding to parent identity $(id/id_{l-1}) = (id_1, ..., id_{l-1})$ at depth $l-1 \ge 0$, this algorithm outputs private key for the identity (id/id_l) at depth l.

If $l = 1$ then $SK_{(id/id_0)}$ is defined to be master key of root PKG.

The private key corresponding to an identity $(id/id_l) = (id_1, ..., id_l)$ at depth l can be generated by PKG or any ancestor (prefix) of an identity (id/id_l).

Encrypt(PP,(id/id_l),M): On input public parameters PP, an identity (id/id_l), and a message M, this algorithm outputs ciphertext C.

Decrypt(PP,$SK_{(id/id_l)}$,C): On input public parameters PP, a private key $SK_{(id/id_l)}$, and a ciphertext C, this algorithm outputs message M.

2.3 Adaptive-ID (Full) Security Model of HIBE

We define adaptive-ID security model using a game that the challenge ciphertext is indistinguisable from a random element in the ciphertext space. The game proceeds as follows.

Setup: The challenger runs Setup$(1^\lambda, 1^d)$ and gives the PP to adversary and keeps MK to itself.

Phase 1: The adversary issues a query for a private key for identity $(id/id_k) = (id_1, ..., id_k)$, $k \le d$. Adversary can repeat this multiple times for different identities adaptivly.

Challenge: The adversary submits identity id* and message M. Identity id* and prefix of id* should not be one of the identity query in phase 1. The challenger choose a random bit $r \in \{0,1\}$ and a random string C with the size of the valid ciphertext. If $r = 0$ it assigns the challenge ciphertext $C^* := Encrypt(PP, id^*, M)$. If $r = 1$ it assigns the challenge ciphertext $C^* := C$. It sends C^* to the adversary as challenge.

Phase 2: Phase 1 is repeated with the restriction that the adversary can not query for id^* and prefix of id^*.

Guess: Finally, the adversary outputs a guess $r' \in \{0,1\}$ and wins if $r = r'$.

We refer an adversary $\mathscr{A}$ as an IND-ID-CPA adversary. Advantage of an adversary $\mathscr{A}$ in attacking an IBE scheme ξ is defined as

$$Adv_{d,\xi,A}(\lambda) = |Pr[r = r'] - 1/2|$$

Definition 1. *HIBE scheme ξ with depth d is adaptive-ID, indistinguishable from random if $Adv_{d,\xi,A}(\lambda)$ is a negligible function for all IND-ID-CPA PPT adversaries $\mathscr{A}$.*

2.4 Integer Lattices ([10])

A lattice L is defined as the set of all integer combinations

$$L(b_1, ..., b_n) = \left\{ \sum_{i=1}^{n} x_i b_i : x_i \in Z \text{ for } 1 \leq i \leq n \right\}$$

of n linearly independent vectors $\{b_1, ..., b_n\} \in R^n$. The set of vectors $\{b_1, ..., b_n\}$ is called a lattice basis.

Definition 2. *For q prime, $A \in Z_q^{n \times m}$ and $u \in Z_q^n$, define:*

$$\Lambda_q(A) := \{e \in Z^m \ \ s.t. \ \ \exists s \in Z_q^n \text{ where } A^T s = e \ (mod \ q)\}$$
$$\Lambda_q^{\perp}(A) := \{e \in Z^m \ \ s.t. \ \ Ae = 0 \ (mod \ q)\}$$
$$\Lambda_q^{u}(A) := \{e \in Z^m \ \ s.t. \ \ Ae = u \ (mod \ q)\}$$

Theorem 1. *([2]) Let q be prime and $m := \lceil 6n\log q \rceil$.*

There is PPT algorithm TrapGen(q,n) that outputs a pair ($A \in Z_q^{n\times m}, T \in Z^{n\times m}$) such that statistically distance between matrix A and a uniform matrix in $Z_q^{n\times m}$ is negligible and T is a basis for $\Lambda_q^{\perp}(A)$ satisfying

$$\|\widetilde{T}\| \leq O(\sqrt{n\log q}) \ \ and \ \ \|T\| \leq O(n\log q)$$

with overwhelming probability in n.

2.5 The LWE Hardness Assumption ([15,1])

The LWE (learning with error) hardness assumption is defined by Regev [15].

Definition 3. ***LWE:*** *Consider a prime q, a positive integer n, and a Gaussian distribution χ^m over Z_q^m. Given $(A, As + x)$ where matrix $A \in Z_q^{m\times n}$ is uniformly random and $x \in \chi^m$.*

LWE hard problem is to find s with non-negligible probability.

Definition 4. ***Decision LWE:*** *Consider a prime q, a positive integer n, and a Gaussian distribution χ^m over Z_q^m. The input is a pair (A, v) from an unspecified challenge oracle O, where $A \in Z_q^{m\times n}$ is chosen uniformly. An unspecified challenge oracle O is either a noisy pseudo-random sampler O_s or a truly random sampler $O_\$$. It is based on how v is chosen.*

1. *When v is chosen to be $As+e$ for a uniformly chosen $s \in Z_q^n$ and a vector $e \in \chi^m$, an unspecified challenge oracle O is a noisy pseudo-random sampler O_s.*
2. *When v is chosen uniformly from Z_q^m, an unspecified challenge oracle O is a truly random sampler $O_\$$.*

Goal of the adversary is to distinguish between the above two cases with non-negligible probability.

Or we say that an algorithm A decides the (Z_q, n, χ)-LWE problem if $|Pr[A^{O_s} = 1] - Pr[A^{O_\$} = 1]|$ is non-negligible for a random $s \in Z_q^n$.

Above decision LWE is also hard even if s is chosen from the Gaussian distribution rather than the uniform distribution [3,13].

2.6 Inhomogeneous Small Integer Solution (ISIS) Assumption

Definition 5. *Given an integer q, a matrix $A \in Z_q^{n\times m}$, a syndrome $u \in Z_q^n$ and real β, find a* short *integer vector $x \in Z_q^m$ such that $Ax = u \bmod q$ and $x \leq \beta$.*

3 Sampling Algorithms

Let A, B be matrices in $Z_q^{n\times m}$ and R be a matix in $\{-1,1\}^{m\times m}$. Let matrix $F = (AR + B) \in Z_q^{n\times 2m}$ and suppose we have to sample short vectors in $\Lambda_q^u(F)$ for some u in Z_q^n. This can be done either a SampleLeft or SampleRight algorithm.

SampleLeft Algorithm (A, M_1, T_A, u, σ)**([1]).** On input matrix $A \in Z_q^{n\times m}$ of rank n , a matrix M_1 in $Z_q^{n\times m_1}$, a "short" basis T_A of $\Lambda_q^\perp(A)$, a vector $u \in Z_q^n$ and a Gaussian parameter $\sigma > \|\widetilde{T_A}\|\omega(\sqrt{(\log(m+m_1))})$, this algorithm returns a vector $e \in Z^{m+m_1}$ sampled from a distribution which is statistically close to $D_{\Lambda_q^u(F_1),\sigma}$, where $F_1 = A|M_1$.

SampleRight Algorithm $(A, B, R, T_B, u, \sigma)$**([1]).** On input a rank n matrix A in $Z_q^{n\times m}$, $B \in Z_q^{n\times m}$ where B is rank n, a matrix R in $Z_q^{k\times m}$, let $s_R := \|R\|$, a basis T_B of $\Lambda_q^\perp(B)$ and a vector $u \in Z_q^n$ and a Gaussian parameter $\sigma > \|\widetilde{T_B}\| s_R \omega(\sqrt{\log(m)})$, this algorithm returns a vector $e \in Z^{m+k}$ sampled from a distribution which is statistically close to $D_{\Lambda_q^u(F_2),\sigma}$ where $F_2 = (A|AR+B)$.

4 Adaptively Secure HIBE Scheme in Standard Model

Our new scheme is a variant of Agarwal et al HIBE [1], but with short public parameters. In our scheme, identity id/id_l is represented as $id/id_l = (id_1, ..., id_l) = ((b_{1,1}||... ||b_{1,l''}), ..., (b_{l,1}||...||b_{l,l''}))$ where id_i is l' bit string and $b_{i,j}$ is $l'/l'' = \beta$ bit string. In Agrawal et al [1] selective-ID secure lattice HIBE, encryption matrix

$$F_{id/id_l} = (A_0|A_1 + H(id_1)B|\ldots|A_l + H(id_l)B) \in Z_q^{n\times(l+1)m}$$

We apply Waters's [19] idea to convert Agrawal et al [1] selective-ID secure lattice HIBE to adaptive-ID secure HIBE. With this technique, for an l-level HIBE has public parameters as $A_{1,1},\ldots,A_{1,l},A_{2,1},\ldots,A_{2,l},\ldots,A_{l,l}$ and A_0,B matrices. Now encryption matrix becomes

$$F_{id/id_l} = \left(A_0|\sum_{i=1}^{l} A_{1,i}b_{1,i}+B|\ldots|\sum_{i=1}^{l} A_{l,i}b_{l,i}+B\right)$$

Here the public parameters is very large (total $l \times l+2$ matrices). Similar to Chatterjee and Sarkar [8] we have used same public parameters $A_1,\ldots,A_l$ for all levels. This way public parametrs is reduced from $l \times l+2$ matrices to $l+2$ matrices. Further we reduce the public parameters $(l+2)$ matrices to $(l''+2)$ matrices by using Chatterjee and Sarkar's [7] blocking technique. Finally encryption matrix in our scheme is

$$F_{id/id_l} = \left(A_0|\sum_{i=1}^{l''} A_i b_{1,i}+B|\ldots|\sum_{i=1}^{l''} A_i b_{l,i}+B\right) \tag{1}$$

4.1 The HIBE Construction

Now we describe our adaptive secure HIBE scheme as follows.

Setup (d,λ)**:** On input a security parameter λ and a maximum hierarchy depth d, this algorithm set the parameters $q,n,m,\overline{\sigma},\overline{\alpha}$ as specified in the end of this section. Next we do the following.

1. Use algorithm TrapGen(q,n) to generate a matrix $A_0 \in Z_q^{n\times m}$ and a short basis T_{A_0} for $\Lambda_q^{\perp}(A_0)$ such that $\|\widetilde{T_{A_0}}\| \le O(\sqrt{n\log q})$.
2. Select $l''+1$ uniformly random $n \times m$ matrices $A_1,A_2,\ldots,A_{l''}$ and $B \in Z_q^{n\times m}$.
3. Select a uniformly random n - vector $u \in Z_q^n$.
4. Output the public parameters and master key, PP = $A_1,A_2,\ldots,A_{l''}$ and $B,A_0 \in Z_q^{n\times m}$, MK = $T_{A_0} \in Z_q^{m\times m}$.

Derive $(PP,(id/id_l),SK_{(id/id_{(l-1)})})$**:** On input public parameters PP, a private key SK$_{(id/id_{l-1})}$ corresponding to an identity (id/id_{l-1}) at depth $l-1$ the algorithm outputs a private key for the identity (id/id_l) at depth l.

From equation (1),

$$F_{id/id_l} = \left(A_0|\sum_{i=1}^{l''} A_i b_{1,i}+B|\ldots|\sum_{i=1}^{l''} A_i b_{l,i}+B\right) \tag{2}$$

Or $F_{id/id_l} = \left(F_{id/id_{l-1}}|\sum_{i=1}^{l''} A_i b_{l,i}+B\right)$.

Given short basis $SK_{(id/id_{(l-1)})}$ for $\Lambda_q^{\perp}(F_{id/id_{l-1}})$ and F_{id/id_l} as defined in (1), we can construct short basis $SK_{(id/id_l)}$ for $\Lambda_q^{\perp}(F_{id/id_l})$ by invoking

$$S \longleftarrow \text{SampleLeft}(F_{id/id_{l-1}},\sum_{i=1}^{l''} A_i b_{l,i}+B,SK_{(id/id_{(l-1)})},0,\sigma_l)$$

and output $SK_{(id/id_l)} \longleftarrow S$.

The private key corresponding to an identity $(id/id_l) = (id_1, \ldots, id_l)$ at depth l can be generated by PKG or any ancestor (prefix) of an identity (id/id_l) by repeatedly calling SampleLeft algorithm.

Encrypt (PP, Id, b)**:** On input public parameters PP, an identity (id/id_l) of depth l and a message $b \in \{0,1\}$,do the following:

1. Build encryption matrix

$$F_{id/id_l} = \left(A_0 | \sum_{i=1}^{l''} (A_i b_{1,i} + B) || \ldots || \sum_{i=1}^{l''} (A_i b_{l,i} + B) \right) \in Z_q^{n \times (l''+1)m}$$

.

2. Choose a uniformly random vector $s \xleftarrow{R} Z_q^n$.
3. Choose l'' uniformly random matrices $R_j \xleftarrow{R} \{-1,1\}^{m \times m}$ for $j = 1, \ldots, l''$. Define R_{id}[1] $= \sum_{i=1}^{l''} b_i R_i || \ldots || \sum_{i=1}^{l''} b_i R_i \in Z^{m \times l l'' m}$
4. Choose noise vector $x \xleftarrow{\overline{\Psi}_{\alpha_l}} Z_q$, $y \xleftarrow{\overline{\Psi}^m_{\alpha_l}} Z_q^m$ and $z \longleftarrow R_{id}^T y \in Z_q^{lm}$,
5. Output the ciphertext,

$$CT = \left(C_0 = u_0^T s + x + b \lfloor \frac{q}{2} \rfloor, C_1 = F_{id}^T s + \begin{bmatrix} y \\ z \end{bmatrix} \right) \in Z_q \times Z_q^{(l+1)m}$$

Decrypt $(PP, \mathbf{SK}_{(id/id_l)}, CT)$**:** On input public parameters PP, a private key SK_{id/id_l}, and a ciphertext CT = (C_0, C_1), do the following.

1. Set $\tau_l = \sigma_l \sqrt{m(l+1)} w(\sqrt{log(lm)})$. Then $\tau_l \geq \|\widetilde{SK}\| w(\sqrt{log(lm)})$.
2. $e_{id} \longleftarrow SamplePre(F_{id/id_l}, SK_{(Id/id_l)}, u, \tau_l)$
   Then $F_{id} e_{id} = u$ and $\|e_{id}\| \leq \tau_l \sqrt{m(l+1)}$
3. Compute $C_0 - e_{id}^T C_1 \quad \in Z_q$.
4. Compare w and $\lfloor \frac{q}{2} \rfloor$ treating them as integers in Z. If they are close, i.e., if $|w - \lfloor \frac{q}{2} \rfloor| < \frac{q}{4}$ in Z, output 1 otherwise output 0.

During Decryption:
$w_0 = C_0 - e_{id}^T C_1 = b \lfloor \frac{q}{2} \rfloor + x - e_{id}^T \left[\begin{smallmatrix} y \\ z \end{smallmatrix}\right]$.

Parameters and Correctness: We have during decryption, $w = C_0 - e_{id}^T c_1 = b \lfloor \frac{q}{2} \rfloor + x - e_{id}^T \left[\begin{smallmatrix} y \\ z \end{smallmatrix}\right]$. $x - e_{id}^T \left[\begin{smallmatrix} y \\ z \end{smallmatrix}\right]$ is called error term and for correctness it has to be less than $q/4$.

Lemma 1. *Norm of the error is less than* $[q 2^\beta l'' l^2 \sigma_l m \alpha_l \omega(\sqrt{\log m}) + O(2^\beta l'' l^2 \sigma_l m^{3/2})]$.

Proof: Lemma is essentially same as lemma 32 of [1] except now R_{id} is uniformly random matrix in $\{-2^\beta l'', 2^\beta l''\}^{m \times lm}$. So now $|R_{id}|$ will be equal to $2^\beta l'' R_{id}$. Hence error term will have extra factor $2^\beta l''$.

[1] In security proof, R_{id} is used to answer adversary's secret key query and also for valid challenge ciphertext, error vector has to be $\left[\begin{smallmatrix} y \\ R_{id}^T y \end{smallmatrix}\right]$.

For the scheme to work correctly, it is required that:

- the error is less than $q/4$ i.e. $\alpha_l < [2^\beta l'' l^2 \sigma_l m\omega(\sqrt{\log m})]^{-1}$ and $q = \Omega(2^\beta l'' l^2 \sigma_l m^{3/2})$
- that TrapGen can operate (i.e $m > 6n\log q$)
- That σ_l is sufficiently large for SimpleLeft and SimpleRight (i.e. $\sigma_l > \|\widetilde{T_B}\| s_R \omega(\sqrt{\log m})\ ) = 2^\beta l'' \sqrt{l} m\omega(\sqrt{\log m})$
- that Regev's reduction applies (i.e. $(q2^\beta)^l > 2Q$), where Q is the number of identity queries from the adversary)

To satisfy these requirements we set the parameters $(q, m, \sigma_l, \alpha_l)$ as follows, taking n to be the security parameter:

$$m = 6n^{1+\delta}, \qquad \sigma_l = l'' \sqrt{l} m\omega(\sqrt{\log n})$$

$$q = max((2Q/2^\beta)^{1/l}, (2^\beta l'')^2 l^{2.5} m^{2.5} \omega(\sqrt{\log n})), \alpha_l = [(2^\beta l'')^2 l^{2.5} m^2 \omega(\sqrt{\log m})]^{-1} \tag{3}$$

From above requirements, we need $q = (2^\beta l'')^2 l^{2.5} m^{2.5} \omega(\sqrt{\log n})$.

4.2 Security Proof

Our proof of theorem will require an abort-resistant hash function defined as follows.

Abort-Resistant Hash Functions

Definition 6. *Let $H = \{\hbar : X \longrightarrow Y\}$ be family of hash functions from X to Y where $0 \in Y$. For a set of $Q+1$ inputs $\overline{x} = (x_0, x_1, ..., x_Q) \in X^{Q+1}$, non-abort probability of $\overline{x}$ is defined as*

$$\alpha(\overline{x}) = Pr[\hbar[x_0] = 0 \ \wedge \ \hbar[x_1] \neq 0 \ \wedge ... \wedge \ \hbar[x_Q] \neq 0]$$

where range of the probability is the random selection of $\hbar$ in H.

H is $(Q, \alpha_{min}, \alpha_{max})$ abort-resistance if $\forall \overline{x} = (x_0, x_1, ..., x_Q) \in X^{Q+1}$ with $x_0 \notin \{x_1, ..., x_Q\}$ we have $\alpha(\overline{x}) \in [\alpha_{min}, \alpha_{max}]$. we use the following abort-resistant hash family very similar to [1].

For a prime number q let $(Z_q^{l''})^* = Z_q^{l''}$-$\{0^l\}$ and the family is defined as

$$H : \{\hbar : ((Z_{2\beta}^{l''})^* | ... | (Z_{2\beta}^{l''})^*) \longrightarrow (Z_q | ... | Z_q)\}$$

$$\hbar(id) = \hbar(id_1 | ... | id_l) = \left(1 + \sum_{i=1}^{l''} h_i b_{1,i}\right) | \dots | \left(1 + \sum_{i=1}^{l''} h_i b_{l,i}\right) \tag{4}$$

where h_i and $b_{k,i}$ are defined in section 4.1.

Lemma 2. *For prime number q and $0 < Q < q$. Then the hash family H defined in (3) is $(Q, \frac{1}{q^l}(1 - \frac{Q}{q^l}), \frac{1}{q^l})$ abort-resistant.*

Proof: The proof is similar to [1]. Consider a set of $\overline{id}$ of $Q+1$ inputs $id^0,...,id^Q$ in $(Z_q^{ll''})^*$ where $id^0 \notin \{id^1,...,id^Q\}$ and $id^i = \{id_1,...,id_l\}$. Since number of functions in $H = q^{l''}(2^\beta)^{l''l}$ and for $i = 0,...,Q+1$ let S_i be function $\hbar$ in H such that $\hbar(id^i) = 0$. Hence number of such functions $= |S_i| = \frac{q^{l''}(2^\beta)^{l''l}}{q^l}$. and $\frac{|S_0 \wedge S_j| \leq q^{l''}(2^\beta)^{l''l}}{q^{2l}}$ for every $j > 0$. Number of functions in H such that $\hbar(id^0) = (0|...|0)$ but $\hbar(id^i) \neq 0$ for $i = 1,...,Q$. $= |S|$ and

$$|S| = |S_0 - (S_1 \vee ...S_Q)| \geq |S_0| - \sum_{i=1}^{Q} |S_0 \wedge S_i|$$

$$\geq \frac{q^{l''}(2^\beta)^{l''l}}{q^l} - Q\frac{q^{l''}(2^\beta)^{l''l}}{q^{2l}}$$

Therefore the no-abort probability of identities is atleast equal to $\frac{\frac{q^{l''}(2^\beta)^{l''l}}{q^l} - \frac{Qq^{l''}(2^\beta)^{l''l}}{q^{2l}}}{q^{l''}(2^\beta)^{l''l}} = \frac{1}{q^l}(1 - \frac{Q}{q^{2l}})$ Since $|S| \leq |S_0|$, so the no-abort probability is atmost $\frac{|S_0|}{q^{l''}(2^\beta)^{l''l}} = \frac{1}{q^l}$.

Theorem 2. *Our HIBE scheme is IND-ID-CPA secure provided that the $(Z_q, n, \bar{\psi}_{\alpha_d})$-LWE assumptions hold.*

Proof. Here proof is similar to [1,17]. We show that if there exist a PPT adversary A that breaks our *HIBE* scheme with non-negligible probability then there exists a PPT challenger B that answers whether an unspecified challenge oracle O is either a noisy pseudo-random sampler O_s or a truly random sampler $O_\$$ by simulating views of adversary A.

Setup: Challenger B generates uniformly random matrix A_0 in $Z_q^{n\times m}$ as follows. Challenger B obtains $m+1$ LWE samples i.e. $(u_i, v_i) \in Z_q^n \times Z_q$ $(0 \leq i \leq m+1$ from an unspecified challenge oracle, which get parsed as matrix $A_0 = (u_1, \ldots, u_m)$. Matrix B is generated by using algorithm TrapGen, which returns random matrix B in $Z_q^{n\times m}$ and a Trapdoor T_B for $\Lambda_q^\perp(B)$. Challenger also chooses l'' uniformly random matrices $R_i \in [-1, l]^{m\times m}, i \in [1, l'']$ and l'' random scalars $h_i \in Z_q, i \in [1, l'']$. Next it constructs the matrices A_i as

$$A_i \longleftarrow A_0 R_i + h_i B$$

By lemma 3, the statistical distance between distribution of A_i's and the uniform distribution is negligible.

Phase 1

$$F_{id/id_l} = \left(A_0 | \sum_{i=1}^{l''} A_i b_{1,i} + B||...||\sum_{i=1}^{l''} A_i b_{l,i} + B\right) \quad (5)$$

Substituting the value of matrices A_i from equation (4)

$$F_{id/id_l} = \left(A_0|A_0\left(\sum_{i=1}^{l''} R_i b_{1,i}\right) + B\left(1+\sum_{i=1}^{l''} h_i b_{1,i}\right)||...||A_0\left(\sum_{i=1}^{l''} R_i b_{l,i}\right) + B\left(1+\sum_{i=1}^{l''} h_i b_{l,i}\right)\right)$$

Or $F_{id} = (A_0|A_0R_{id} + Bh_{id})$ where $R_{id} = \sum_{i=1}^{l''} R_i b_{1,i}||...||\sum_{i=1}^{l''} R_i b_{l,i}$ and $B_{id} = B\hbar_{id} = B(1+\sum_{i=1}^{l''} h_i b_{1,i})||...||(1+\sum_{i=1}^{l''} h_i b_{l,i})$.

If $\hbar_{id}$ is not equal to zero then challenger responds the private key query of $id = (id^1, id^2, ..., id^l)$ by running

$$SK_{id} \longleftarrow \text{SampleRight}(A_0, B_{id}, R_{id}, T_B, 0, \sigma_l)$$

and sending SK_{id} to A. $\hbar_{id}$ is equal to zero will be part of abort resistant hash function.

Challenge: Adversary declares target identity $id^* = (id_1, id_2, ..., id_l)$ and bit message $b^* \in \{0,1\}$. Simulator B creates challenge ciphertext for declared target identity as follows:

1. Set
$$v^* = \begin{pmatrix} v_1 \\ \vdots \\ v_m \end{pmatrix} \in Z_q^m$$
where $v_1, \ldots, v_m$ be entries from LWE instance.
2. Blind the bit message by letting
$$C_0^* = v_0 + b^* \lfloor \frac{q}{2} \rceil \in Z_q$$
3. Challenger also chooses l'' uniformly random matrices $R_i^* \in [-1, l]^{m\times m}, i \in [1, l'']$. Let
$$R_{id^*} = (R_1^*|...|R_{l''}^*)$$
and set
$$C_1^* = \begin{pmatrix} v^* \\ (R_{id^*})^T v^* \end{pmatrix} \in Z_q^{m+l''m}$$
4. Randomly choose a bit $r \leftarrow \{0,1\}$. If $r = 0$, send ciphertext $CT^* = (C_0^*, C_1^*)$ to the adversary. If $r = 1$ choose a random $(C_0, C_1) \in Z_q \times Z_q^{m+l''m}$ and send (C_0, C_1) to the adversary.

Phase 2: Simulator repeats the same method used in Phase 1 with the restriction that the adversary can not query for id^* and prefix of id^*.

Artificial Abort: This artificial abort technique was introduced by Waters [19]. Chatterjee and Sarkar [7] presented a detailed exposition on artificial abort. Since probability of abort depends on the set of private key queries so it is possible that an adversary's success probability and simulator' abort probability are not independent. The purpose of the artificial abort step is to ensure that simulator aborts with almost same probability irrespective of any set of queries made by the adversary. This step increases the run time of the simulator.

We obtain the lower bound λ for the probability that challenger B does not abort. Let ab be the event that challenger B aborts and Σ' is the set of queries made by the adversary. Waters [19] has proved that probability challenger B does not abort is very close to λ for all adversarial queries.

$$|Pr[\overline{ab}|Y \in \Sigma'] - \lambda| \leq \frac{\varepsilon}{2}$$

From lemma 3 no-abort probability of identities is atleast equal to $\frac{1}{q^l}(1 - \frac{Q}{q^{2l}})$. With $Q \leq q^l/2$ no-abort probability of identities will be atleast equal to $\frac{1}{q^l}$.

Simulator requires an additional $\chi = O(\varepsilon^{-2} ln(\varepsilon^{-1})\lambda ln(\lambda^{-1}))$ time for artificial abort stage. Bellare and Ristenport [4] showed that artificial step can be avoided. They have provided following security reduction formula without artificial abort step.

$$Adv^{dbdh}(B) \geq \frac{\gamma_{min}}{2} Adv^{IND-CPA}_{Waters} + (\gamma_{min} - \gamma_{max})$$

From the above expression it is clear that Bellare and Ristenport's [4] proof will work when $\gamma_{min} - \gamma_{max}$ is negligible. But in our case $\gamma_{min} - \gamma_{max}$ is $-\frac{Q}{q^2}$ and it can be made negligible with large q which will affect the performance of the scheme. So we have used Waters [19] artificial abort.

When the LWE oracle is pseudorandom then $F_{id^*} = (A_0|A_0\overline{R}_{id^*})$ since $h_{id^*} = 0$ and

$$v^* = A_0^T s + y$$

for some uniform noise vector $y \in Z_q^m$ distributed as $\bar{\Psi}_\alpha^m$. Therefore

$$C_1^* = \begin{pmatrix} A_0^T s + y \\ (A_0 R_{id^*})^T s + (R_{id^*})^T y \end{pmatrix} = (F_{id^*})^T s + \begin{pmatrix} y \\ (R_{id^*})^T y \end{pmatrix}$$

Above C_1^* is a valid C_1 part of challenge ciphertext. Again $C_0^* = u_0^T + x + b^* \lfloor \frac{q}{2} \rceil$ is also a valid C_0 part of challenge ciphertext. Therefore (C_0^*, C_1^*) is valid challenge ciphertext.

When LWE oracle is random oracle, v_0 is uniform in Z_q and v^* is uniform in Z_q^m. Therefore challenge ciphertext is always uniform in $Z_q \times Z_q^{l''m}$. Finally adversary A terminates with correct output, adversary B answers that an unspecified challenge oracle O is a noisy pseudo-random sampler O_s else an unspecified challenge oracle O is a truly random sampler $O_\$$ and terminates the simulation.

So probabilistic algorithm B solves the $(Z_q, n, \overline{\Psi}_\alpha)$-LWE problem in about the time $= t_1 + O(\varepsilon^{-2} ln(\varepsilon^{-1})\lambda ln(\lambda^{-1}))$ and with $\varepsilon' \geq \varepsilon/4q^l$

5 Conclusion

We have shown that by converting selective-ID HIBE to adaptive-ID HIBE security degradation is exponential in number of levels. The open problem is to construct adaptive-ID HIBE secure scheme without exponentialial degradation.

References

1. Agrawal, S., Boneh, D., Boyen, X.: Efficient Lattice (H)IBE in the Standard Model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 553–572. Springer, Heidelberg (2010)
2. Alwen, J., Peikert, C.: Generating Shorter Bases for Hard Random Lattices. In: International Symposium on Theoretical Aspects of Computer Science, STACS 2009, pp. 75–86. IBFI Schloss Dagstuhl (2009)
3. Applebaum, B., Cash, D., Peikert, C., Sahai, A.: Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 595–618. Springer, Heidelberg (2009)
4. Bellare, M., Ristenpart, T.: Simulation without the Artificial Abort: Simplified Proof and Improved Concrete Security for Waters' IBE Scheme. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 407–424. Springer, Heidelberg (2009)
5. Boneh, D., Franklin, M.: Identity Based Encryption From the Weil Pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)
6. Cash, D., Hofheinz, D., Kiltz, E.: How to Delegate a Lattice Basis. In: IACR Cryptology ePrint Archive (2009)
7. Chatterjee, S., Sarkar, P.: Trading Time for Space: Towards an Efficient IBE Scheme with Short(er) Public Parameters in the Standard Model. In: Won, D.H., Kim, S. (eds.) ICISC 2005. LNCS, vol. 3935, pp. 424–440. Springer, Heidelberg (2006)
8. Chatterjee, S., Sarkar, P.: HIBE With Short(er) Public Parameters Without Random Oracle. In: Lai, X., Chen, K. (eds.) ASIACRYPT 2006. LNCS, vol. 4284, pp. 145–160. Springer, Heidelberg (2006)
9. Cocks, C.: An Identity Based Encryption Scheme Based on Quadratic Residues. In: Honary, B. (ed.) Cryptography and Coding 2001. LNCS, vol. 2260, pp. 360–363. Springer, Heidelberg (2001)
10. Micciancio, D., Goldwasser, S.: Complexity of Lattice Problems: A Cryptographic Perspective, vol. 671. Kluwer Academic Publishers (2002)
11. Gentry, C., Silverberg, A.: Hierarchical ID-Based Cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548–566. Springer, Heidelberg (2002)
12. Horwitz, J., Lynn, B.: Toward Hierarchical Identity-Based Encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 466–481. Springer, Heidelberg (2002)
13. Lindner, R., Peikert, C.: Better key sizes (and attacks) for LWE-based encryption. In: Kiayias, A. (ed.) CT-RSA 2011. LNCS, vol. 6558, pp. 319–339. Springer, Heidelberg (2011)
14. Peikert, C.: Bonsai trees (or, arboriculture in lattice-based cryptography). In: Cryptology ePrint Archive, Report 2009/359 (2009)
15. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: STOC, pp. 84–93. ACM (2005)
16. Shamir, A.: How to share a secret. Commun. ACM 22(11), 612–613 (1979)
17. Singh, K., Pandu Rangan, C., Banerjee, A.K.: Lattice based identity based proxy re-encryption scheme. Journal of Internet Services and Information Security (JISIS) 3(3/4), 38–51 (2013)
18. Singh, K., Pandurangan, C., Banerjee, A.K.: Adaptively secure efficient lattice (H)IBE in standard model with short public parameters. In: Bogdanov, A., Sanadhya, S. (eds.) SPACE 2012. LNCS, vol. 7644, pp. 153–172. Springer, Heidelberg (2012)
19. Waters, B.: Efficient identity-based encryption without random oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (2005)
20. You, I., Hori, Y., Sakurai, K.: Enhancing svo logic for mobile ipv6 security protocols. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA) 2(3), 26–52 (2011)