Cryptology

Scribe: Fabrice Mouhartem

M2IF

Chapter 1

Identity Based Encryption from Learning With Errors

In the following we will use this two tools which existence is not proved here. The first tool description can be found in [1] and the second tool can be retrieved in [2].

Tool 1 There exists a probabilistic polynomial time algorithm `GenBasis` that takes n, m, q with m bigger than a $\Omega(n \log n)$ as inputs, and returns a matrix T (called the *trapdoor* of A) of size $m \times n$ in $\mathbb{Z}_q^{m \times n}$ such that:

- $\Delta(A, \text{Unif}) \leq 2^{-n}$
- The maximum norm of a row of T is quasilinear:

$$\|T\| = \max_i \|t_i\| = \mathcal{O}\left(n \log n\right)$$

- the t_i's form a basis of the lattice:

$$\Lambda^{\perp}(A) = \left\{x \in \mathbb{Z}^m : x^T A = 0 \mod q\right\}$$

Remark. For this algorithm to be polynomial time, n and m are written in unary in order to make the input exponentially bigger.

Tool 2 Let L be a n-dimensional lattice with basis (b_i), a vector c in $\mathbb{Z}^n$ and s greater than $\max \|b_i\| \cdot \Omega\left(\sqrt{\log n}\right)$. There exists a probabilistic pseudo-random polynomial time algorithm `GPVSample` that samples from $D_{L,s,c}$:

$$D_{L,s,c}(b) \sim \exp\left(-\pi \frac{\|b-c\|^2}{s^2}\right)$$

For such an s we have the following properties:

$$\max_{b \in L} D_{L,s,c}(b) \leq 2^{-n}$$

and

$$\Pr_{b \in D_{L,s,c}}\left(\|b-c\| \geq s\sqrt{n}\right) \leq 2^{-n}$$

1.1 The Gentry-Peikert-Vaikuntanathan Identity Based Encryption scheme

In this section we will present the Gentry Peikert Vaikuntanathan Identity Based Encryption [2]. It is a cryptosystem which security is based on the Learning With Errors problem.

Setup We sample A, T using **tool 1**. Let $s = \mathcal{O}\left(n \log n \cdot \sqrt{n}\right)$. We define the master public key and the master secret key with:

$$MPK : A, MSK : T$$

Key extract(MSK, id**)** We first take $u = H(id)$, where H is a hash function from a binary word to $\mathbb{Z}_q^n$ modeled as a random oracle.
Then we use linear algebra to find $r_0 \in \mathbb{Z}_q^m$ such that

$$r_0^T A = u^T \mod q$$

Finally we use **tool 2** to sample r from the distribution:

$$r_0 + \underbrace{D_{\Lambda^\perp(A),s,-r_0}}_{r_1}$$

Thus:

$$r^T A = r_0^T A + r_1^T A = r_0^T A + 0 = u^T \mod q$$

And we have (tail bound) with probability at least $1 - 2^n$:

$$\|r\| = \|r_0 + r_1\| = \|r_1 - (-r_0)\| \leq \sqrt{n}s$$

r is id's secret key sk_{id}.

Enc($MPK, id, M \in \{0,1\}$**)** Let $u := H(id)$. We sample s from $U(\mathbb{Z}_q^n)$ and (e_1, e_2) from $D_{\mathbb{Z}^m,\alpha q} \times D_{\mathbb{Z},\alpha q}$ (gaussian), and we send:

$$\begin{aligned} c_1 &= As + e_1 \\ c_2 &= \langle u, s \rangle + e_2 + \lfloor q/2 \rfloor M \end{aligned}$$

Dec($sk_{id}, (c_1, c_2)$**)** We compute

$$c_2 - r^T c_1 = e_2 + \lfloor q/2 \rfloor M - r^T e_1 \mod q$$

if the result is greater than $q/4$, output 1, else output 0.

Correctness Let us compute the decryption error:

$$\begin{aligned} |e_2 - r^T e_1| &\le |e_2| + \|r\| \cdot \|e_1\| \\ &\le \alpha q\sqrt{n} + \alpha q\sqrt{m}\sqrt{n} \cdot \|r\| \\ &\text{(with probability } \ge 1 - c2^n \text{ for some constant } c \\ &\le \alpha q\sqrt{n}[1 + \sqrt{m}\sqrt{mn}\log q\sqrt{\log n} \end{aligned}$$

We have to set α such that

$$\alpha q\sqrt{n}[1 + m\log q\sqrt{\log q}] \le \frac{q}{8}$$
$$\alpha\sqrt{n}[1 + m\log q\sqrt{\log q}] \le \frac{1}{8}$$

Remark. The parameter α is some $\frac{1}{\mathrm{poly}(m,n,\log q)}$

Security Indistinguishability under Chosen Plaintext Attack (**IND-CPA**) with selective-id security in the Random Oracle Model (**ROM**).

Security Game We have an attacker $\mathcal{A}$ and a challenger $\mathcal{C}$, as we are in the selective-id assumption, $\mathcal{A}$ starts by choosing and identity id^* to attack and send it to $\mathcal{C}$. Then $\mathcal{C}$ selects master public and secret keys from `Setup` and send the master public key (**MPK**) to $\mathcal{A}$.

Algorithm $\mathcal{A}$ starts by making extraction queries for id different from id^* and also Random Oracle queries to $\mathcal{C}$.

Without loss of generality if id is different from id^*. We assume that $H(id)$ is always queried just before **Extract**(id). $H(id^*)$ is queried before the challenge phase.

As we are encrypting a bit, there is no need to produce two different messages M_0, M_1 as they will be 0 and 1. This is why $\mathcal{C}$ decides to encrypt a bit b with identity secret key id^* and send the resulting ciphertext c^* to $\mathcal{A}$.

Finally $\mathcal{A}$ returns a bit b', and wins if b' is b.

From this initial game we derive the following games:

Game 0 Real game

Game 1 *Change Setup to Setup':*
We sample: the matrix A uniform and the vector u^* uniform (u^* is $H(id^*)$)

Change Extract to Extract':
if $Extract'(id)$ has been queried before **then** give same answer.
else sample r from $D_{\mathbb{Z}^m,s,0}$
set $u := r^T A$.
return that sk_{id} is r and $H(id)$ is u.

Enc stays the same as in Game 0.
Using **tool 1** we have that:

$$\Delta(A \text{ in game } 0, A \text{ in Game } 1) \le 2^{-n}$$

For all fresh Random Oracle queries, the reply is uniform. Distribution of sk_{id} (conditioned on A, $H(id) = u$) are

$$\begin{aligned} &u \leftarrow U(\mathbb{Z}_q^m) \\ &r \leftarrow r_0^T + D_{\Lambda^\perp(A),s,-r_0)} \end{aligned}$$

Question: Does the following experiment

$$\begin{aligned} &r \leftarrow D_{\mathbb{Z}^m,s,0} \\ &u = r^T A \end{aligned}$$

gives the same distribution for (r, u)? (the proof is somewhat non-trivial, left as exercise).

Game 2 *Enc changed to Enc′ in challenge phase:*

$$Enc'(MPK, id^*, M) : Unif + (0, 0, \ldots, 0, \lfloor q/2 \rfloor M) \in \mathbb{Z}_q^{m+1}$$

As depicted as follow:

$$\begin{array}{|c|c} & 0 \\ \text{Unif} & \in \mathbb{Z}_q^{m+1} \\ & \lfloor q/2 \rfloor M \end{array}$$

Any adversary that can distinguish between Games 1 and 2 may be able to break Learning With Errors (**LWE**).
LWE input : (B, b).
Answer $b = Bs+c$ if in Game 1 or $b =$ unif if in Game 2: the matrix B is built as u concatenated at the right of u. **Enc**: $b + (0, 0, \ldots, 0, \lfloor q/2 \rfloor M)$

1.2 The Agrawal-Boneh-Boyen IBE

The Agrawal Boneh Boyen Identity Based Encryption [3] scheme is based on the Learning With Errors problem. An interesting property of this scheme is that it is provably secure in the standard model, therefore we have no need of a random oracle assumption.

Encoding of identities as matrices in $\mathbb{Z}_q^{n\times n}$: we want a map H from ID to $\mathbb{Z}_q^{n\times n}$ such that

$$\begin{cases} H(id) - H(id') \\ \forall id \neq id' \end{cases}$$

is full rank/invertible

Let $\Phi(x)$ be a degree n irreducible polynomial modulo q (as the density is greater than $1/n$ and irreducibility can be tested in polynomial time, then the Φ-generation is polynomial-time).

$$\begin{aligned} ID &= \mathbb{Z}_q[x]_{/(\Phi(x))} \\ |ID| &= q^n \end{aligned}$$

$$H(id) = [id, x \cdot id \bmod \Phi, \ldots, x^{n-1} \cdot id \bmod \Phi]$$

The k-th column coefficients of $H(id)$ consists in the $x^k \cdot id(x) \mod \Phi$:

$$\left[\begin{array}{ccccccc} | & | & | & | & & | & \\ | & | & | & | & & | & \\ id & x \cdot id & x^2 \cdot id & x^3 \cdot id & \cdots & x^{n-1} \cdot id & \mathrm{mod}\Phi \end{array}\right]$$

Remark. The map:

$$\begin{aligned} ID &\to H(ID) \\ id &\mapsto H(id) \end{aligned}$$

is a ring homomorphism: $H(id + id') = H(id) + H(id')$ and $H(id \cdot id') = H(id) \cdot H(id')$

The set ID is a field: $H(id) - H(id') = 0 \implies id = id'$, then we can embed $\mathbb{F}_{q^n}$ into $\mathbb{Z}_q^{n \times n}$

Setup We generate A_0, T_0 using **tool 1**.
And we sample matrices A_1, B uniform in $\mathbb{Z}_q^{m \times n}$.
The vector u is sampled uniform in $\mathbb{Z}_q^n$.
We return: $MSK = T_0, MPK = (A_0, A_1, B, u)$

Extract(MPK, id**)** Define A_{id}:

$$A_{id} = \begin{pmatrix} A_0 \\ A_1 + B \cdot H(id) \end{pmatrix} \in \mathbb{Z}_q^{3m \times n}$$

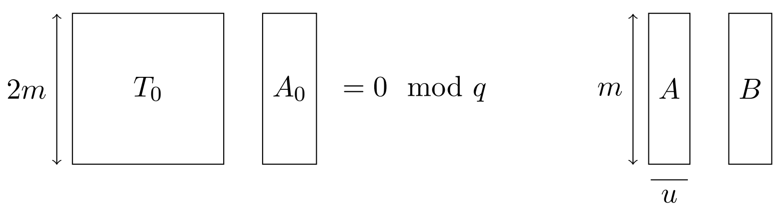
$$\begin{pmatrix} T_0 & 0 \\ T' & I \end{pmatrix} \begin{pmatrix} A_0 \\ A_i + B \cdot H(id) \end{pmatrix} = 0$$

and

$$T_{id} = \begin{pmatrix} T_0 \\ T' \end{pmatrix}$$

How to find T' such that $T_{id} \cdot A_{id} = 0$?
To find t'_i such that $t_i'^T \cdot A_0 = - \ldots$, proceed as in GPV's **Extract**:

$$\|t'_i\| \le \mathcal{O}\left(\sqrt{m} n \log q \sqrt{\log n}\right)$$



The matrix T_{id} is small (largest row smaller than $\mathcal{O}\left(\sqrt{m} n \log q \sqrt{\log n}\right)$). It is made of linearly independent rows. Further, we have $T_{id} A_{id} = 0$. We then have $3m$ linearly independent short vectors in $\Lambda^\perp(A_{id})$. It is sufficient to efficiently sample from $D_{\Lambda^\perp(A_{id}), s, c}$ for all c and s greater

than an $\Omega(m^{1.5} n \log q \sqrt{\log n})$. (like in GPV's extract)
Using gaussian tail bound, we sample r such that:

$$r^T A_{id} = u^T \mod q$$
$$\|r\| \leq \mathcal{O}\left(\sqrt{m}s\right)$$

Enc($MPK, id, M \in \{0,1\}$**)** Recover:

$$A_{id} = \begin{matrix} A_0 \\ A_1 + BH(id) \end{matrix}$$

Sample:

$$s \leftarrow U(\mathbb{Z}_q^n)$$
$$e_1 \in D_{\mathbb{Z}^{2m},\alpha q}$$
$$R \leftarrow \{-1,1\}^{m \times 2m}$$
$$\text{set } e_2 := R \cdot e_1$$
$$e_2 \leftarrow D_{\mathbb{Z},\alpha q}$$

Send $c = (c_1, c_2, c_3)$ with the following components:

$$c_1 = A_0 \cdot s + e_1 + 0$$
$$c_2 = A_1 + B \cdot H(id) + e_2 + 0$$
$$c_3 = 0 + e_3 + \lfloor q/2 \rfloor M$$

Dec($sk_{id}, (c_1, c_2, c_3)$**)** Compute:

$$c_3 - r^T \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \lfloor q/2 \rfloor M + \text{(decryption noise)}$$

Correctness Get a bound on decryption noise. Set α such that | decryption noise| is smaller than $q/8$ with probability greater than $1 - k2^{-n}$ for some constant k.

Security IND-CPA selective id in the standard model.

Game 0 real game

Game 1 change **Setup** and **Extract**:
Change setup to setup':
The Challenger gets id^* from Adversary.
He samples:

$$R^* \leftarrow \{-1,1\}^{m \times 2m} \text{ (to be used for challenge)}$$
$$T_B, B \leftarrow \textbf{tool 1}$$
$$A_0 \leftarrow \mathbb{Z}_q^{m \times n}$$
$$u \leftarrow \mathbb{Z}_q^n$$

Set $A_1 = R^* \cdot A_0 - BH(id^*)$

We want that A_0, A, B, u should "look" as in Game 0.
For A_0, B and u it is ok as they are chosen uniformly in Game 0. For A_1 with small statistical distance to uniform by the Leftover Hash Lemma (**LHL**). Therefore:

$$\Delta(MPK^{Game0}, MPK^{Game1}) \leq 2^{-\Omega(n)}$$

Change Extract to Extract'($id \neq id'$):

$$A_{id} = \begin{pmatrix} A_0 \\ A_1 + B \cdot H(id) \end{pmatrix} = \begin{pmatrix} A_0 \\ R^* A_0 + B(H(id) - H(id^*)) \end{pmatrix}$$

How do I find a T_{id} for A_{id} using T_B?

$$T_B \cdot B = 0 \implies T_B \cdot B \cdot (H(id) - H(id^*)) = 0$$

$$\begin{pmatrix} I & T' \\ 0 & T_B \end{pmatrix} \begin{pmatrix} A_0 \\ B \end{pmatrix} \begin{pmatrix} \\ \square \end{pmatrix} = 0$$

The vectors t_i' are small, constructed as the T' in real Extract.
Possible as $\square = H(id) - H(id^*)$ is full rank.

$$\underbrace{\begin{pmatrix} I & T' \\ 0 & T_B \end{pmatrix} \underbrace{\begin{pmatrix} I & \\ -R^* & I \end{pmatrix} \begin{pmatrix} A_B \\ R^* A_0 + BH(id - id^*) \end{pmatrix}}_{\begin{pmatrix} A_0 \\ BH(id - id^*) \end{pmatrix}}}_{T_{id}} = 0$$

Rest of Extract' is identical to Extract

Game 2 Enc' in challenge phase : $Unif + (0, 0, \ldots, \lfloor q/2 \rfloor M) \in \mathbb{Z}_q^{3m+1}$

Exercise : show that if an adversary can distinguish Games 2 and 3 it may be used to break **LWE**. (how to extend **LWE**'s "b" into a cipertext?)

Note: the correctness proof is not trivial and is based of a strengthened Leftover Hash Lemma.

Bibliography

[1] Joël Alwen and Chris Peikert. "Generating shorter bases for hard random lattices." Theory of Computing Systems 48.3 (2011): 535-553.

[2] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. "Trapdoors for hard lattices and new cryptographic constructions." Proceedings of the fortieth annual ACM symposium on Theory of computing. ACM, 2008.

[3] Shweta Agrawal, Dan Boneh, and Xavier Boyen. "Efficient lattice (H) IBE in the standard model." Advances in Cryptology–EUROCRYPT 2010. Springer Berlin Heidelberg, 2010. 553-572.