Lecture 8: Attribute-Based Encryption, Predicate Encryption and Functional Encryption

1 Definitions

Attribute-based encryption (ABE) and predicate encryption (PE) generalize identity-based encryption (IBE) in the following way.

- Setup produces MPK, MSK.
- Enc uses MPK to encrypt a message m relative to attributes $\boldsymbol{\mu} = (\mu_1, \dots, \mu_\ell) \in \{0,1\}^\ell$. *(In an IBE scheme, $\boldsymbol{\mu}$ is the identity.)*
- KeyGen uses MSK to generate a secret key SK_f for a given Boolean function $f : \{0,1\}^\ell \to \{0,1\}$. *(IBE is the same as ABE where f is restricted to be the point function $f_{ID'}(ID) = 1$ iff $ID = ID'$.)*
- Dec gets $\boldsymbol{\mu}$ (attributes are in the clear) and uses SK_f to decrypt a ciphertext C if $f(\boldsymbol{\mu}) = 1$ (true). If $f(\boldsymbol{\mu}) = 0$, Dec simply outputs $\perp$.

The difference between ABE and PE is in the type of security they achieve. In a nutshell, in ABE, the ciphertext is not required to hide the attribute vector $\boldsymbol{\mu}$, rather only the message m. PE requires attribute-hiding, which is formalized in one of two ways, weak attribute-hiding or (strong) attribute-hiding, which we will explain below. It turns out that PE with strong attribute-hiding is equivalent to functional encryption (FE) which we will show in later lectures gives us an indistinguishability obfuscation (IO) scheme.

2 The Key Lattice Equation

Let us abstract out the mathematics behind the GSW FHE scheme into a *key lattice equation* which will guide us through constructing the rest of the primitives in this lecture, in particular an attribute-based encryption (ABE) scheme and a predicate encryption scheme for the orthogonality predicate.

Recall the approximate eigenvector relation:

$$\mathbf{s}^T \mathbf{A}_i \approx \mu_i \mathbf{s}^T \mathbf{G}$$

and rewrite it as

$$\mathbf{s}^T(\mathbf{A}_i - \mu_i \mathbf{G}) \approx \mathbf{0} \tag{1}$$

Let $\mathbf{A}_f$ be the homomorphically evaluated ciphertext for a function f. We know that

$$\mathbf{s}^T \mathbf{A}_f \approx f(\boldsymbol{\mu}) \mathbf{s}^T \mathbf{G}$$

or

$$\mathbf{s}^T(\mathbf{A}_f - f(\boldsymbol{\mu})\mathbf{G}) \approx \mathbf{0} \tag{2}$$

We will generalize this to arbitrary matrices $\mathbf{A}_1, \dots, \mathbf{A}_\ell$ – not necessarily ones that share the same eigenvector.

First, we know that $\mathbf{A}_f$ is a function of $\mathbf{A}_1, \dots, \mathbf{A}_\ell$ and f (*but not* $\mu_1, \dots, \mu_\ell$). Henceforth, when we say $\mathbf{A}_f$, we will mean a matrix obtained by the GSW homomorphic evaluation procedure. (That is, homomorphic addition of two matrices is matrix addition; homomorphic multiplication is matrix multiplication after bit-decomposing the second matrix).

Second, and very crucially, we can show that for any sequence of matrices $\mathbf{A}_1, \ldots, \mathbf{A}_\ell$,

$$[\mathbf{A}_1 - \mu_1\mathbf{G} \| \ldots \| \mathbf{A}_\ell - \mu_\ell\mathbf{G}]\ \mathbf{H}_{f,\boldsymbol{\mu}} = \mathbf{A}_f - f(\boldsymbol{\mu})\mathbf{G}$$

where $\mathbf{H}_{f,\boldsymbol{\mu}}$ is a matrix with small coefficients. We call this the key lattice equation.

To see this for addition, notice that

$$[\mathbf{A}_1 - \mu_1\mathbf{G} \| \mathbf{A}_2 - \mu_2\mathbf{G}]\ \underbrace{\begin{bmatrix} \mathbf{I} \\ \mathbf{I} \end{bmatrix}}_{\mathbf{H}_{+,\mu_1,\mu_2}} = \mathbf{A}_1 + \mathbf{A}_2 - (\mu_1 + \mu_2)\mathbf{G} = \mathbf{A}_+ - (\mu_1 + \mu_2)\mathbf{G}$$

and for multipication,

$$[\mathbf{A}_1 - \mu_1\mathbf{G} \| \mathbf{A}_2 - \mu_2\mathbf{G}]\ \underbrace{\begin{bmatrix} \mathbf{G}^{-}(\mathbf{A}_2) \\ \mu_1\mathbf{I} \end{bmatrix}}_{\mathbf{H}_{\times,\mu_1,\mu_2}} = \mathbf{A}_1\mathbf{G}^{-}(\mathbf{A}_2) - \mu_1\mu_2\mathbf{G} = \mathbf{A}_\times - \mu_1\mu_2\mathbf{G}$$

By composition, we get that

$$[\mathbf{A}_1 - \mu_1\mathbf{G} \| \mathbf{A}_2 - \mu_2\mathbf{G} \| \ldots \| \mathbf{A}_\ell - \mu_\ell\mathbf{G}]\ \mathbf{H}_{f,\boldsymbol{\mu}} = \mathbf{A}_f - f(\boldsymbol{\mu})\mathbf{G}$$

where $\mathbf{H}_{f,\mu}$ is a matrix with small entries (roughly proportional to $m^{O(d)}$ where d is the circuit depth of f).

An Advanced Note: Given arbitrary matrices $\mathbf{A}_i$ and $\mathbf{A}_f$, there exists such a small matrix $\mathbf{H}$; but if $\mathbf{A}_f$ is arbitrary, it is hard to find.

Let's re-derive FHE from the key equation:

- The ciphertexts are the matrices $\mathbf{A}_i$ and we picked them such that

$$\mathbf{s}^T\mathbf{A} \approx \mu\mathbf{s}^T\mathbf{G}$$

- Homomorphic evaluation is computing $\mathbf{A}_f$ starting from $\mathbf{A}_1, \ldots, \mathbf{A}_\ell$.

- Correctness of homomorphic eval follows from the key equation: We know that

$$\mathbf{s}^T[\mathbf{A}_1 - \mu_1\mathbf{G} \| \ldots \| \mathbf{A}_\ell - \mu_\ell\mathbf{G}] \approx \mathbf{0}$$

  by the equation above that characterizes ciphertexts. Therefore, by the key equation,

$$\mathbf{s}^T[\mathbf{A}_f - f(\boldsymbol{\mu})\mathbf{G}] = \mathbf{s}^T[\mathbf{A}_1 - \mu_1\mathbf{G} \| \ldots \| \mathbf{A}_\ell - \mu_\ell\mathbf{G}]\ \mathbf{H}_{f,\boldsymbol{\mu}} \approx \mathbf{0}$$

  as well meaning that $\mathbf{A}_f$ is an encryption of $f(\boldsymbol{\mu})$. Note that no one needs to know or compute the matrix $\mathbf{H}$; it only appears in the analysis.

3 Predicate Encryption for the Orthogonality Predicate

We show the construction of a predicate encryption scheme (Agrawal, Freeman and Vaikuntanathan; Asiacrypt 2011) supporting the *orthogonality predicate* (alternatively, linear functions with equality test) from the LWE assumption. Such a function is defined by a vector $y \in \mathbb{Z}_q^L$, takes as input $x \in \mathbb{Z}_q^L$ and outputs 1 if

$$\langle y, x \rangle = 0 \bmod q$$

The construction proceeds as follows:

- $\mathsf{Setup}(1^\lambda)$: generate $L \log q + 1$ matrices $\mathbf{A}_0, \mathbf{A}_1, \dots, \mathbf{A}_{L \log q} \in \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{v} \in \mathbb{Z}_q^n$, where $n = n(\lambda)$, $m = m(\lambda)$ and $q = q(\lambda)$ are LWE parameters. Here, $\mathbf{A}_0$ is a uniformly random matrix generated together with its trapdoor $\mathbf{T}_0$, and all other matrices are generated uniformly at random without trapdoors.

  The master public and secret key are

$$\mathsf{mpk} = (\mathbf{A}_0, \dots, \mathbf{A}_{L \log q}, \mathbf{v}) \qquad \text{and} \qquad \mathsf{msk} = (\mathsf{mpk}, \mathbf{T}_0)$$

- $\mathsf{KeyGen}(\mathsf{msk}, y)$ where $y \in \mathbb{Z}_q^L$: Let $y' = \mathbf{G}^{-1}(y) \in \{0,1\}^{L \log q}$ denote the bit decomposition of y. Use the trapdoor $\mathbf{T}_0$ to generate a vector $\mathbf{r}$ such that

$$\left[\mathbf{A}_0 \,\middle\|\, \sum_{i \in [L \log q]} y'_i \mathbf{A}_i \right] \cdot \mathbf{r} = \mathbf{v} \bmod q$$

  The function-specific private key is $\mathsf{sk}_y := (y, \mathbf{r})$. (Recall that using the Gaussian sampling algorithm and the trapdoor for the matrix above, one can generate such a vector whose marginal distribution is Gaussian with a fixed parameter $\sigma = \sigma(\lambda)$.)

- $\mathsf{Enc}(\mathsf{mpk}, x)$ where $x \in \mathbb{Z}_q^L$ and $m \in \{0,1\}$: Let $x' = \mathbf{G}^T x \in \mathbb{Z}_q^{L \log q}$ be the powers-of-two encoding of x. The ciphertext is

$$\mathbf{c}^T := [\mathbf{c}_0^T \mid \mathbf{c}_1^T \mid \dots \mid \mathbf{c}_{L \log q}^T \mid c'] \in \mathbb{Z}_q^{mL \log q + 1}$$

  where

$$\begin{aligned} \mathbf{c}_0^T &= \mathbf{s}^T \mathbf{A}_0 + \mathbf{e}_0^T \\ \mathbf{c}_i^T &= \mathbf{s}^T (\mathbf{A}_i + x'_i \mathbf{G}) + \mathbf{e}_i \ \ (\text{for } 1 < i \le L \log q) \\ c' &= \mathbf{s}^T \mathbf{v} + e' + m \lceil q/2 \rceil \end{aligned}$$

- $\mathsf{Dec}(\mathsf{sk}_y, \mathbf{c})$: Compute

$$(\mathbf{c}_y)^T := \mathbf{s}^T \left[\mathbf{A}_0 \,\middle\|\, \sum_i y'_i \mathbf{A}_i + y'_i x'_i \mathbf{G} \right] + [\mathbf{e}_0^T \parallel \sum_i y'_i \mathbf{e}_i^T]$$

  and then

$$c' - \mathbf{c}_y^T \cdot \mathbf{r} \,,$$

  round the result to the nearest multiple of $q/2$ and output the resulting bit.

4 Attribute-based Encryption for all Predicates

Here is an ABE scheme (called the BGG+ scheme) using the key equation. It's best to view this as a generalization of the Agrawal-Boneh-Boyen IBE scheme.

- KeyGen outputs matrices $\mathbf{A}, \mathbf{A}_1, \ldots, \mathbf{A}_\ell$ and a vector $\mathbf{v}$ and these form the MPK. The MSK is the trapdoor for $\mathbf{A}$.
- Enc computes
$$\mathbf{s}^T[\mathbf{A}||\mathbf{A}_1 - \mu_1\mathbf{G}||\ldots||\mathbf{A}_\ell - \mu_\ell\mathbf{G}]$$
(plus error, of course, and we will consider that understood.) Finally, the message is encrypted as $\mathbf{s}^T\mathbf{v} + e + m\lfloor q/2 \rfloor$.
- Let's see how Dec might work. You (and in fact anyone) can compute
$$\mathbf{s}^T[\mathbf{A}||\mathbf{A}_1 - \mu_1\mathbf{G}||\ldots||\mathbf{A}_\ell - \mu_\ell\mathbf{G}]\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_{f,\boldsymbol{\mu}} \end{bmatrix} = \mathbf{s}^T[\mathbf{A}||\mathbf{A}_f - f(\boldsymbol{\mu})\mathbf{G}]$$
using the key equation.

  If you had a short $\mathbf{r}$ that maps $[\mathbf{A}||\mathbf{A}_f - \mathbf{G}]$ to $\mathbf{v}$, that is
$$[\mathbf{A}||\mathbf{A}_f - \mathbf{G}]\ \mathbf{r} = \mathbf{v}$$
you can decrypt and find m. (Can you fill in the blanks?)

Two notes:

- The security definition mirrors IBE exactly, and the security proof of this scheme mirrors that of the ABB IBE scheme that we did in the last lecture. I will leave it to you as an exercise. The reference is the work of Boneh et al., Eurocrypt 2014.
- One might wonder if the attributes $\boldsymbol{\mu}$ need to be revealed. The answer is "NO", in fact one can construct an attribute-hiding ABE scheme (also called a predicate encryption scheme). There are two flavors of security of such a scheme, the weaker one can be realized using LWE (Gorbunov, Vaikuntanathan and Wee, Crypto 2015) and the stronger one implies indistinguishability obfuscation, a *very* powerful cryptographic primitive which we don't know how to construct from LWE yet. More in the next lecture.