1 DDH-IP Encryption

Let $GroupGen$ be a probabilistic polynomial time algorithm that takes as input a security parameter 1^λ, and outputs a triplet $(\mathbb{G}, p, g)$ where $\mathbb{G}$ is a group of order p that generated by $g \in \mathbb{G}$ where p is an λ-bit number. For any two tuples (g, g^a, g^b, g^{ab}) and (g, g^a, g^b, g^c), they are computationally indistinguishable, where $(\mathbb{G}, p, g) \leftarrow GroupGen(1^\lambda)$ and $a, b, c \in \mathbb{Z}_p$ are chosen independently and uniformly at random. A simple inner-product functional encryption scheme is described as $IP = (Setup, KeyDer, Encrypt, Decrypt)$ and each component is explained as follows. The scheme input is $\boldsymbol{x} = (x_1, x_2, \ldots, x_\ell)$ from an entity A, and another entity B with $\boldsymbol{y} = (y_1, y_2, \ldots, y_\ell)$ outputs $\boldsymbol{x}\cdot\boldsymbol{y}$ with DDH assumption based security.

- $Setup(1^\lambda, 1^\ell) \rightarrow (mpk, msk)$. A triplet $(\mathbb{G}, p, g)$ is sampled based on $GroupGen(1^\lambda)$. Set $\boldsymbol{s} = (s_1, s_2, \ldots, s_\ell) \leftarrow \mathbb{Z}_p^\ell$ and $\boldsymbol{h} = (h_1 = g^{s_1}, h_2 = g^{s_2}, \ldots, h_\ell = g^{s_\ell})$. The outputs are obtained as $msk = \boldsymbol{s}$ and $mpk = \boldsymbol{h}$.

- $Encrypt(mpk, \boldsymbol{x}) \rightarrow \boldsymbol{Ct}$. Choose a random $r \leftarrow \mathbb{Z}_p$ and compute $ct_0 = g^r$ then for each $i \in [\ell]$, $ct_i = h_i^r \cdot g^{x_i}$.Then ciphertext $\boldsymbol{Ct} = (ct_0, (ct_i)i \in [\eta])$.

- $KeyDer(msk, \boldsymbol{y}) \rightarrow sk_y$. Calculate $sk_y = \boldsymbol{y} \cdot msk$.

- $Decrypt(mpk, \boldsymbol{Ct}, sk_y, y)$. It returns the inner product $x \cdot y$ as logorithm in basis g of $\Pi_{i\in[\ell]} ct_i^{y_i} / ct_0^{sk_y}$.

Finally, the entity B calculates the inner product x and y with the privacy of x reserved according to Eq.(1).

$$g^{x\cdot y} = \Pi_{i\in[\ell]} ct_i^{y_i} / ct_0^{sk_y}. \tag{1}$$