CRT-based Fully Homomorphic Encryption over the Integers with Shorter Public key

1 Preliminaries

Notation. We use a $\leftarrow A$ to denote choosing an element a from a set A randomly. When $\mathcal{D}$ is a distribution, we use a $\leftarrow \mathcal{D}$ to denote choosing an element a according to the distribution $\mathcal{D}$. We use $\mathbb{Z}_p := \mathbb{Z} \cap \left(\frac{-p}{2}, \frac{p}{2}\right]$ and x mod p denotes a number in $\mathbb{Z} \cap \left(\frac{-p}{2}, \frac{p}{2}\right]$ and $\langle x \rangle_p$ is x mod p in $\mathbb{Z} \cap [0, p)$ which is equivalent to x modulos p. We use notation $(a_i)^k$ for a vector $(a_1, .., a_k)$.

For pairwise coprime integers $p_1, ..., p_k$, we define $\mathrm{CRT}_{(p_1,..,p_k)}(m_1, .., m_k)$ as a number in $\mathbb{Z} \cap \left(\frac{-p}{2}, \frac{p}{2}\right]$ which is equivalent to $m_i$modulos p_i for all $i \in 1, ..., k$ where $x_0 = \prod_{i=1}^{k} p_i$. This is,

$$\mathrm{CRT}_{(p_1,..,p_k)}(m_1, .., m_k) = \sum_{i=1}^{k} m_i \; \hat{p_i} \; (\hat{p_i}^{-1} \mod p_i) \bmod x_0$$

where $\hat{p_i} = \frac{x_0}{p_i} = \frac{\prod_{j=0}^{k} p_j}{p_i}$

2 CRT-based Fully Homomorphic Encryption

The message space is $\prod_{i=1}^{k} \mathbb{Z}_{Q_i}$. If $Q_1, .., Q_k$ are pairwise coprime integers, the message space can be considered $\mathbb{Z}_Q$ where $Q = \prod_{i=1}^{k} Q_i$.

2.1 Parameters

We give some descriptions about the parameters.

λ : the security parameter

ρ : the bit length of the error

η : the bit length of the secret primes

γ : the bit length of a ciphertext

τ : the number of encryptions of zero in public key

k : the number of distinct secret primes

l_Q : the bit length of Q_i for $i = 1, ...k$

Roughly speaking, k determines the size of the message space. The parameter l_Q can bean integer from 2 to $\eta/8$ depending on the multiplicative depth of the scheme.

- $\gamma = \eta^2\omega(\log(\lambda))$ to resist Cohn and Heninger's attack [1] and the attack using Lagarias algorithm [2] on the approximate GCD problem
- $\eta =\widetilde{\Omega}\ (\lambda^2+\rho.\lambda)$), to resist the factoring attack using the elliptic curve method [3] and to permit enough multiplicative depth.
- $\rho =\widetilde{\mathcal{O}}\ (\lambda)$, to be secure against Chen-Nguyen's attack [4] and Howgrave-Graham's attack [5].
- $\tau = \gamma+\omega(\log(\lambda))$, in order to use left-over hash lemma in the security proof.

We choose $\gamma =\widetilde{\mathcal{O}}\ (\lambda^5)$, $\eta =\widetilde{\mathcal{O}}\ (\lambda^2)$, $\rho = 2\lambda$, $\tau = \gamma + \lambda$ which is similar to the DGHV's convenient parameter setting [6].

2.2 Construction

KeyGen$(\lambda, \rho, \eta, \gamma, \tau, l_Q, k)$: Choosen η-bit distinct primes $p_1, ..., p_k$ and $q_0 \leftarrow \mathbb{Z} \cap [0, \frac{2^\gamma}{\prod_{i=1}^k p_i})$ and set x_0. Choosen l_Q-bit integers $Q_1, ..., Q_k$ with $gcd(Q_i, x_0)$=1 for $i = 1, ..., k$. Output the public key p_k as follows:

$$pk = \Big(x_0, \{Q_i\}_{i=0}^k, X := \{x_j = CRT_{(q_0,p_1,..,p_k)}(e_{j0}, e_{j1}Q_1, .., e_{jk}Q_k)\}_{j=0}^{\tau}, \\ Y := \{y_l = CRT_{(q_0,p_1,..,p_k)}(e'_{l0}, e'_{l1}Q_1 + \delta_{l1}, .., e'_{lk}Q_k + \delta_{lk})\}_{j=0}^k\Big)$$

where $e_{j0}, e'_{l0} \leftarrow \mathbb{Z} \cap [0, q_0)$, $e_{ji} \leftarrow \mathbb{Z} \cap (-2^\rho, 2^\rho)$, $e'_{li} \leftarrow \mathbb{Z} \cap (-2^\rho, 2^\rho)$ for $i, l \in [1, k], j \in [1, \tau]$ and δ_{ij} in Kronecker delta. Output the secret key $sk = (p_1, ..., p_k)$.

Enc$(pk,\boldsymbol{m})$: For any $\boldsymbol{m}$=$(m_1, ..., m_k)$ with $m_i \in \mathbb{Z}_{\mathbb{Q}}$, outputs $c = \sum_{i=1}^k m_iy_i+ \sum_{j\in S} x_j \bmod x_0$ where S is a random subset of $\{1, ..., \tau\}$.

Dec$(sk,\boldsymbol{c})$: Output $(m_1, ..., m_k)$=$((c \bmod p_1) \bmod Q_1, ..., (c \bmod p_k) \bmod Q_k)$.

Remark 1. There are $(\tau + k)$ integers of γ-bit and k integers of l_Q-bit in the public key. The public key size is $\widetilde{\mathcal{O}}\ ((\tau + k)\gamma + kl_Q)$= $\widetilde{\mathcal{O}}\ (\lambda^{10})$ under the parameters in the Section 2.1

3 Our CRT encrytion Public Key Compression Technique

3.1 Description

`KeyGen`. Generate a random distinct η-bit prime integers $p_1, .., p_k$ and $q_0 \leftarrow \mathbb{Z} \cap [0, \frac{2^\gamma}{\prod_{i=1}^k p_i})$ and let $p = \prod_{i=1}^k p_i$, $q = \prod_{i=1}^k Q_i$ and $x_0 = pq$. Initialize a pseudo-random number generator f with a random seed se. Use $f(\text{se})$ to generate a set of integers $\chi_i \in [0, 2^\gamma)$ for $1 \leq i \leq \tau$. For all $1 \leq i \leq \tau$ compute:

$$\mu_i = \langle \chi_i \rangle_p + \xi_i.p - r_i \cdot q$$

where $r_i \leftarrow \mathbb{Z} \cap (-2^{\rho+k\cdot\eta}/q, 2^{\rho+k\cdot\eta}/q)$ and $\xi \leftarrow \mathbb{Z} \cap [0, 2^{\lambda+k\cdot\eta}/p)$. For all $1 \leq i \leq \tau$ compute:

$$x_i = \chi_i - \mu_i \tag{1}$$

Let p_k =(se, $x_0, \{Q_i\}_{i=1}^k, \{\mu_i\}_{i=1}^\tau, \{y_i\}_{i=1}^k,$) and $sk = (p_1, .., p_k)$where x_0, Q_i, y_i are same values as in the section 2.2.

`Encrypt`(p_k, m): use $f(\text{se})$ to recover the integers χ_i and let $x_i = \chi_i - \mu_i$ for all $1 \leq i \leq \tau$. And do the encryption same as the **Enc**(p_k, m) in section 2.2.

The main difference with the original CRT-based encryption scheme instead of storing the large x_i's in the public key we store only store the much smaller μ_i's. The new public key for the somewhat homomorphic scheme has size $\widetilde{\mathcal{O}}\ (k\eta\tau + k\gamma + kl_Q) = \widetilde{\mathcal{O}}\ (\lambda^7)$ instead of $\widetilde{\mathcal{O}}\ (\lambda^{10})$.

References

[1] H. Cohn and N. Heninger. Approximate common divisors via lattices. *IACR Cryptology ePrint Archive,*. 2011:437, 2011.

[2] J. C. Lagarias. The computational complexity of simultaneous diophantine approximation problems. *SIAM J. Comput.,*. 14(1):196-209, 1985.

[3] J. Lenstra, H. W. Factoring integers with elliptic curves. *The Annals of Mathematics,* 126(3):pp. 649-673, 1987.

[4] Y. Chen and P. Nguyen. Faster algorithms for approximate common divisors: Breaking fully-homomorphicencryption challenges over the integers. In D. Pointcheval and T. Johansson, editors, *Advances in Cryptology EUROCRYPT 2012, volume 7237 of Lecture Notes in Computer Science,* pages 502-519. Springer Berlin Heidelberg, 2012.

[5] N. Howgrave-Graham. Approximate integer common divisors. In *CaLC,* pages 51-66, 2001.

[6] M. v. Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. In H. Gilbert, editor, *Advances in Cryptology - EUROCRYPT 2010,* volume 6110 of *Lecture Notes in Computer Science*, pages 24-43. Springer Berlin - Heidelberg, 2010.