Multikey Homomorphic Encryption from NTRU

Li Chen

lichen.xd at gmail.com

Xidian University

January 12, 2014

Outline

References

[1] Damien Stehlé and Ron Steinfeld. Making ntru as secure as worst-case problems over ideal lattices. In *Advances in Cryptology–EUROCRYPT 2011*, pages 27–47. Springer, 2011.

[2] Adriana López-Alt, Eran Tromer, and Vinod Vaikuntanathan. On-the-fly multiparty computation on the cloud via multikey fully homomorphic encryption. In *Proceedings of the 44th symposium on Theory of Computing*, pages 1219–1234. ACM, 2012.

[3] Vadim Lyubashevsky and Daniele Micciancio. Generalized compact knapsacks are collision resistant. In *Automata, Languages and Programming*, pages 144–155. Springer, 2006.

Claim: Our slides are based on reference [1], [2]

1 Variant of NTRU Encryption

Notations & Public Parameters:

- security parameter κ.
- a prime $q = q(\kappa)$.
- a ring $R \triangleq \mathbb{Z}[x]/\langle x^n + 1\rangle$, and a ring $R_q \triangleq R/qR = \mathbb{Z}_q[x]/\langle x^n + 1\rangle$.
- B-bounded polynomial, $\forall f \in R$, if $\|f\|_\infty \leq B$.
- a B-bounded distribution χ over the ring R, i.e. $\Pr[\|f\|_\infty \leq B : f \leftarrow \chi] = 1 - \varepsilon$, where ε is negligible.

KenGen(1^κ) : sample bounded polynomials $f', g \leftarrow \chi$, and set $f = 2f' + 1$ (so that $f \equiv 1 \pmod 2$). Set $h = 2gf_{-1} \in R_q$, then public key pk=$\{h\}$, secrete key sk=$\{f\}$ (if f is not invertible over R_q, resample f').

Enc(m, pk) : Suppose the plaintext $m \in \{0, 1\}$, sample bounded polynomials $s, e \leftarrow \chi$. Output ciphertext $c \leftarrow hs + 2e + m \in R_q$.

Dec(c, sk) : Let $\mu = cf \in R_q$, output $m \leftarrow \mu \bmod 2$.

Correctness:

$$\begin{aligned} \mu &= cf = (hs + 2e + m)f = 2gf^{-1}sf + 2ef + mf \\ &= 2gs + 2ef + mf \end{aligned}$$

Since $\|2gs + 2ef + mf\|_\infty < \frac{q}{2}$, $\mu \bmod 2 = mf \bmod 2 = m$. Where $\|\mu\|_\infty$ means the L_∞ norm of $\mu \bmod x^n + 1$.

Lemma 1.1 ([3]) Let $n \in \mathbb{N}$, $\phi(x) = x^n + 1$, and $R \triangleq \mathbb{Z}[x]/\langle \phi(x) \rangle$. For arbitrary $s, t \in R$,

$$\|s \cdot t \pmod{\phi(x)}\| \leq \sqrt{n} \cdot \|s\| \cdot \|t\|$$
$$\|s \cdot t \pmod{\phi(x)}\|_\infty \leq n \cdot \|s\|_\infty \cdot \|t\|_\infty$$

Parameter Setting: Since g, s, f, e are all bounded by $2B + 1$, then $gs \pmod{()x^n + 1}$ and $fe \pmod{()x^n + 1}$ are bounded by $n(2B + 1)^2$, thus $2gs + 2ef + mf \pmod{()x^n + 1}$ is bounded by $4n(2B + 1)^2 + 2B + 1 < 40nB^2$.
So, for a fresh ciphertext to be decrypted correctly, we only to set $q > 80nB^2$.

2 Somewhat homomorphic from NTRU

Initiation: Let (h_1, f_1), (h_2, f_2) be two public-secret key pairs. Given two plaintext bits m_1, m_2, let $c_1 = \mathsf{Enc}(m_1, h_1)$, $c_2 = \mathsf{Enc}(m_2, h_2)$, i.e. $c_1 = h_1 s_1 + 2e_1 + m_1$, $c_2 = h_2 s_2 + 2e_2 + m_2$.

$$\begin{aligned} f_1 f_2 (c_1 + c_2) &= 2[gs(f_1 + f_2) + f_1 f_2 (e_1 + e_2)] + f_1 f_2 (m_1 + m_2) \\ f_1 f_2 (c_1 c_2) &= 2[2 g_1 s_1 g_2 s_2 + 2 g_1 s_1 f_2 e_2 + 2 g_2 s_2 f_1 e_1 \\ &\quad + g_1 s_1 f_2 m_2 + g_2 s_2 f_1 m_2 + e_1 f_1 f_2 m_2 + e_2 f_1 f_2 m_1] \\ &\quad + f_1 f_2 m_1 m_2 \end{aligned}$$

Thus by setting a proper q such that there is no wrap-around modulo q, then ,

$$m_1 + m_2 \leftarrow \mathsf{Dec}(c_1 + c_2, f_1 f_2))$$

$$m_1 m_2 \leftarrow \mathsf{Dec}(c_1 c_2, f_1 f_2)$$

Notation: Let $(h_i, f_i), i = 1, \cdots, N$ be N public-secret key pairs, $m_1, m_2, \cdots, m_N$ be N plaintext bits, $c_1, c_2, \cdots, c_N$ be the ciphertexts respectively encrypted using public key $h_1, h_2, \cdots, h_N$, i.e. $c_i = h_i s_i + 2e_i + m_i, i = 1, \cdots, N$.

To decrypt $c_1 + c_2 + \cdots + c_N$ and $c_1 c_2 \cdots c_N$, the minimum joint key need is $\prod_{i=1}^{N} f_i$.

Let $c' = c_1 c_2$ and $c'' = c_2 c_3$, to decrypt c' we need joint key $f_1 f_2$, to decrypt c'' we need joint key $f_2 f_3$. To decrypt $c' + c''$ we need joint key $f_1 f_2 f_3$, to decrypt $c' c''$ we need joint key $f_1 f_2^2 f_3$.

Let D be the degree of the evaluated circuit, N be the number of parties involved, The size of the joint key need to decrypt an evaluated ciphertext grows exponentially both with D and N. The exponential grow dependant on N can not be eliminate, but that dependant on D can be eliminate.

3 Optimized Scheme

KenGen(1^κ) : Sample bounded polynomials $f', g \leftarrow \chi$, and set $f = 2f' + 1$. Set

$$\mathsf{pk} \leftarrow h = 2gf_{-1} \in R_q, \quad \mathsf{sk} \leftarrow f$$

For all $\tau \in [0, \cdots, \lfloor \log q \rfloor]$, sample $s_\tau, e_\tau \leftarrow \chi$, and compute $\gamma_\tau = hs_\tau + 2e_\tau + 2^\tau f \in R_q$. Set

$$\mathsf{ek} = (\gamma_0, \cdots, \gamma \lfloor \log q \rfloor)$$

Enc(m, pk) : Suppose the plaintext $m \in \{0, 1\}$, sample bounded polynomials $s, e \leftarrow \chi$. Output ciphertext $c \leftarrow hs + 2e + m \in R_q$.

Dec$(c, \mathsf{sk}_1, \cdots, , \mathsf{sk}_N)$: $\mathsf{sk}_i = f_i$, compute $\mu = cf_1 \cdots f_N \in R_q$, output $m \leftarrow \mu \bmod 2$.

$\mathsf{Add}(c_1, \mathsf{K}_1, c_2, \mathsf{K}_2,)$: Output $c_{\mathsf{add}} \leftarrow c_1 + c_2$ and $\mathsf{K}_{\mathsf{add}} = \mathsf{K}_1 \bigcup \mathsf{K}_2$.

$\mathsf{Mult}(c_1, \mathsf{K}_1, c_2, \mathsf{K}_2,)$: Compute $\tilde{c}_0 = c_1 c_2 \in R_q$, and let $\mathsf{K}_1 \bigcap \mathsf{K}_2 = \{\mathsf{pk}_{i_1}, \mathsf{pk}_{i_2}, \cdots, \mathsf{pk}_{i_r}\}$.

- If $\mathsf{K}_1 \bigcap \mathsf{K}_2 = \varnothing$, $c_{\mathsf{mult}} \leftarrow \tilde{c}_0$.
- Otherwise, for $j \in [r]$ and $\tau \in [0, \lfloor \log q \rfloor]$, define $\tilde{c}_{j-1,\tau}$, such that

$$\tilde{c}_{j-1} = \sum_{\tau=0}^{\lfloor \log q \rfloor} \tilde{c}_{j-1,\tau} 2^{\tau}$$

is the binary representation of $\tilde{c}_{j-1}$. For each $\mathsf{pk}_{i_j} = f_{i_j}$, let $\mathsf{ek}_{i_j} = (\gamma_{i_j,0}, \cdots, \gamma_{i_j,\lfloor \log q \rfloor})$, where $\gamma_{i_j,\tau} = hs_\tau + 2e_\tau + 2^\tau f_{i_j}$. Then let

$$\tilde{c}_j = \sum_{\tau=0}^{\lfloor \log q \rfloor} \tilde{c}_{j-1,\tau} \gamma_{i_j,\tau}$$

Finally, output $c_{\mathsf{mult}} = \tilde{c}_r$, and $\mathsf{K}_{\mathsf{mult}} = \mathsf{K}_1 \bigcup \mathsf{K}_2$.

Correctness: Let $f_{\mathsf{K}_1} = \prod_{i \in \mathsf{K}_1} f_i$, $f_{\mathsf{K}_2} = \prod_{i \in \mathsf{K}_2} f_i$, and $F_0 = f_{\mathsf{K}_1} f_{\mathsf{K}_2}$, $F_j = F_{j-1} f_{i_j}^{-1}$, for $j = 1, 2, \cdots, r$. Obviously, $F_r = \prod_{i \in \mathsf{K}_{\mathsf{mult}}} f_i$. Just need to show $F_r \tilde{c}_r = F_0 \tilde{c}_0 \pmod 2$.

$$\begin{aligned} F_j \tilde{c}_j &= f_{i_j} F_j \cdot (f_{i_j})^{-2} \cdot (f_{i_j} \tilde{c}_j) \\ &= F_{j-1} \cdot (f_{i_j})^{-2} \cdot (f_{i_j} \tilde{c}_j) \\ &= F_{j-1} \cdot (f_{i_j})^{-2} \cdot \left(\sum_{\tau=0}^{\lfloor \log q \rfloor} \tilde{c}_{j-1,\tau} (\gamma_{i_j,\tau} f_{i_j}) \right) \end{aligned}$$

Since $\gamma_{i_j,\tau} f_{i_j} = 2(g_{i_j} s_{i_j,\tau} + f_{i_j} e_{i_j,\tau}) + 2^\tau f_{i_j}^2 = 2E_\tau + 2^\tau f_{i_j}^2$, we have,

$$\begin{aligned} F_j \tilde{c}_j &= F_{j-1} \cdot (f_{i_j})^{-2} \cdot \left(\sum_{\tau=0}^{\lfloor \log q \rfloor} \tilde{c}_{j-1,\tau} (2E_\tau + 2^\tau f_{i_j}^2) \right) \\ &= 2\left(F_{j-1} \cdot (f_{i_j})^{-2} \sum_{\tau=0}^{\lfloor \log q \rfloor} \tilde{c}_{j-1,\tau} E_\tau \right) + F_{j-1} \sum_{\tau=0}^{\lfloor \log q \rfloor} \tilde{c}_{j-1,\tau} 2^\tau \\ &= 2\left(F_{j-1} \cdot (f_{i_j})^{-2} \sum_{\tau=0}^{\lfloor \log q \rfloor} \tilde{c}_{j-1,\tau} E_\tau \right) + F_{j-1} \tilde{c}_{j-1} \end{aligned}$$

4 More Optimize & Application Scenario

Internal Discussed...

Thanks! & Questions?