$(PK, SK) \leftarrow KeyGen(1^n)$

$$\xrightarrow{ct \leftarrow Enc_{PK}(x)}$$

$$\xleftarrow{ct^* = Eval_{PK}(f, ct)}$$

$Dec_{SK}(ct^*) = f(x)$

Bit-by-bit encryption

correctness:

security: $(PK, Enc_{PK}(0)) \approx (PK, Enc_{PK}(1))$

Efficiency: $|ct^*| \ll |f|$

Approach:

- Represent f as a circuit with NAND gates

- Given $Enc_{pk}(x)$, $Enc_{pk}(y)$

  derive $Enc_{pk}(x \text{ NAND } y)$

$$\parallel$$

$$1 - x \cdot y$$

Idea:

secret key $t \in \mathbb{Z}_q^m$

Encryption of x: $C \in \mathbb{Z}_q^{m \times m}$

s.t. $tC = x \cdot t$

eigenvector eigenvalue

Given: C_x, C_y

$$t(C_x + C_y) = (x+y) \cdot t$$

$$t \cdot (C_x \cdot C_y) = x \cdot t \cdot C_y = x \cdot y \cdot t$$

$$t \cdot I = 1 \cdot t$$

so $C_{NAND} = I - C_x \cdot C_y$

satisfies $t \cdot C_{NAND} = 1 - x \cdot y$

Problem: not secure! Eigenvectors are easy to find.

Proposed solution: add errors

$$t \cdot C = x \cdot t + e$$ small error.

How to implement? Why secure?

Detour: the "gadget matrix"

Recall that SIS says that

given $\boxed{A \in \mathbb{Z}_q^{n \times m}}$ and $\boxed{v} \in \mathbb{Z}_q^n$

hard to find $\boxed{r} \in \mathbb{Z}_q^m$ s.t. r "small" and

$$A \cdot r = v$$

Best for a special "gadget matrix"

$G \in \mathbb{Z}_q^{n \times m}$ this is easy.

<u>Claim:</u> $\forall\, m > n \lfloor \lg q \rfloor$ $\exists\, G \in \mathbb{Z}_q^{n \times m}$

and a poly(m) function $G^{-1} : \mathbb{Z}_q^n \to \{0,1\}^m$

s.t. $\forall\, v \in \mathbb{Z}_q^n$ $G \cdot G^{-1}(v) = v$

<u>pf:</u> Let $g = [1, 2, 4, \dots, 2^{\lfloor \lg q \rfloor}]$

$$G = \begin{array}{c} n \end{array} \left[\begin{array}{ccccc|c} -g- & & & & & \\ & -g- & & & & \\ & & -g- & & & 0 \\ & & & \ddots & & \\ & & & & -g- & \end{array} \right]$$

$$\underbrace{\qquad\qquad\qquad\qquad\qquad\qquad}_{n \cdot \lfloor \lg q \rfloor}$$

Given $v \in \mathbb{Z}_q$ let $g^{-1}(u) \in \{0,1\}^{\lceil \log q \rceil}$

be the bit decomposition of u:

$g^{-1}(u) = b_1, \dots b_{\lceil \log q \rceil}$ s.t. $\sum b_i \cdot 2^i = v$.

$$G^{-1}\left(\begin{bmatrix} v_1 \\ \vdots \\ v_n \end{bmatrix} \right) = \begin{bmatrix} g^{-1}(v_1) \\ \vdots \\ g^{-1}(v_n) \\ 0 \end{bmatrix}$$

■

For $V \in \mathbb{Z}_q^{n \times \ell}$ $V = \begin{bmatrix} | & & | \\ v_1 & \cdots & v_\ell \\ | & & | \end{bmatrix}$

def $G^{-1}(V) = \begin{bmatrix} | & & | \\ G^{-1}(v_1) & \cdots & G^{-1}(v_\ell) \\ | & & | \end{bmatrix}$

so that $G \cdot G^{-1}(V) = V$.

FHE Scheme:

Let $m > n \cdot \log q$

KeyGen(1^n):

$\bar{A} \leftarrow \mathbb{Z}_q^{n \times m}$

$s \leftarrow \mathbb{Z}_q^n$

$e \leftarrow \chi^m$

$b := s \cdot \bar{A} + e$

PK: $A = \begin{bmatrix} \bar{A} \\ b \end{bmatrix} \in \mathbb{Z}_q^{(n+1) \times m}$

SK: $t = [-s, 1] \in \mathbb{Z}_q^{n+1}$

$$\boxed{\begin{aligned} t \cdot A &= -s\bar{A} + b = e \\ &\approx 0 \end{aligned}}$$

$\mathrm{Enc}_{pk}(x)$: $\quad R \leftarrow \{0,1\}^{m \times m}$

$$C = AR + x \cdot G$$

$$\boxed{\begin{aligned} t \cdot C &= \underbrace{e \cdot R}_{e^*} + x \cdot t \cdot G \\ &\approx x \cdot t \cdot G \end{aligned}}$$

Note:

Let $\hat{t} = t \cdot G$, $\hat{C} = G^{-1}(C)$ then $\hat{t} \cdot \hat{C} \approx x \cdot \hat{t}$

$\mathrm{Dec}_{sk}(C)$: $\quad \mathrm{round}\left(\underbrace{t \cdot C \cdot G^{-1}\begin{pmatrix} 0 \\ \lfloor q/2 \rfloor \end{pmatrix}}\right)$

$$= x \cdot t \cdot G \cdot G^{-1}\begin{pmatrix} 0 \\ \lfloor q/2 \rfloor \end{pmatrix}$$

$$+ \quad e^* \cdot G^{-1}\begin{pmatrix} 0 \\ q/2 \end{pmatrix}$$

$$\approx x \cdot (-s, 1)\begin{pmatrix} 0 \\ \vdots \\ \lfloor q/2 \rfloor \end{pmatrix} \approx x \cdot \lfloor q/2 \rfloor$$

Given: C_x, C_y s.t.

$$t \cdot C_x = x \cdot t \cdot G + e_x$$

$$t \cdot C_y = y \cdot t \cdot G + e_y$$

$C_{add} = C_x + C_y$:

$$t \cdot C_{add} = (x+y) \cdot t \cdot G + (e_x + e_y)$$

$C_{mult} = C_x \cdot G^{-1}(C_y)$:

$$t \cdot C_{mult} = (x \cdot t \cdot G + e_x) \cdot G^{-1}(C_y)$$

$$= x(y t G + e_y) + e_x \cdot G^{-1}(C_y)$$

$$= x \cdot y \cdot t G + \underbrace{x \cdot e_y + e_x \cdot G^{-1}(C_y)}_{e_{mult}}$$

$$\approx (xy) \cdot t G$$

$C_{NAND} = G - C_x \cdot G^{-1}(C_y)$:

$$t \cdot C_{NAND} = t G - x y t G - e_{mult} \approx (1 - xy) \cdot t \cdot G$$

Error analysis:

- Assume χ is B-bounded
- A ciphertext has β-error if $tC = xtG + e$

$\|e\|_\infty \leq \beta$.

then:

- Fresh encryptions have $\beta = m \cdot B$ error.
- If C_x has β_x error C_y has β_y error

$C_{NAND} = C_x \cdot G^{-1}(C_y)$ has

$\beta_{NAND} = \beta_y + m \cdot \beta_x$ error

$= (m+1)\beta_{max}$

- If we evaluate a circuit of depth d then final ctext has

$\beta_{final} = (m+1)^d m \cdot B$

- Can decrypt as long as $m \cdot \beta_{final} < q/4$

$\Rightarrow \quad q > 4 \cdot (m+1)^{d+1} \cdot m \cdot B$

Efficiency scales with $\log q \approx d$.

"leveled FHE".

Security:

$$\left(A = \begin{bmatrix} \bar{A} \\ b = s\bar{A} + e \end{bmatrix}, \; C = AR + x \cdot G \right)$$

$$\approx \left(A \leftarrow \mathbb{Z}_q^{(n+1) \times m}, \; C = AR + x \cdot G \right)$$

by LWE security

$$\approx \left(A \leftarrow \mathbb{Z}_q^{(n+1) \times m}, \; C \leftarrow \mathbb{Z}_q^{(n+1) \times m} \right)$$

statistically indistinguishable by LHL.

Problems with leveled FHE:

- Need to know depth d a-priori.

- Efficiency (PK, sk, ct sizes, cost of each gate evaluation) scales with d.

Fix using bootstrapping:

- Use leveled FHE for some fixed $d \geq$ depth of FHE decryption $+1$.

- Give out $C_{sk} \leftarrow Enc_{pk}(sk)$

— For any ciphertext C at $\leq$ level d error

$$C_{new} = \mathrm{Eval}(\mathrm{Dec}_{(\cdot)}(C), C_{sk})$$

has error level $< d$. Can do

1 op.