Gentry-Sahai-Waters [GSW '13]

Gadget matrix $G \in \mathbb{Z}_q^{n \times m}$ $\quad m \geq n \log q$

there is a poly-time $G^{-1} : \mathbb{Z}_q^n \to \{0,1\}^m$

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

— extend to $G^{-1} : \mathbb{Z}_q^{n \times m} \to \{0,1\}^{m \times m}$

$G \cdot G^{-1}(M) = M$

Toy (insecure) scheme

$sk = t \in \mathbb{Z}_q^n$

$Enc_t(x) = C : \quad C \in \mathbb{Z}_q^{n \times n}$ s.t. $tC = x \cdot t$

(eigenvector, eigenvalue)

$C_1 = Enc_t(x), \quad C_2 = Enc_t(y)$

$t \cdot C_1 = x \cdot t \qquad t \cdot C_2 = y \cdot t$

- Add: $C_1 + C_2$

$t \cdot (C_1 + C_2) = x \cdot t + y \cdot t = (x+y) \cdot t$

- Mult: $C_1 \cdot C_2$

$t \cdot C_1 \cdot C_2 = (x \cdot t \cdot C_2) = x \cdot y \cdot t$

- $I = Enc_t(1)$

$\Rightarrow \quad I - C_1 \cdot C_2 = Enc_t(1 - x_1 \cdot x_2)$

Unfortunately insecure

$C : \quad t \cdot C = x \cdot t \Rightarrow (C - Ix)t = 0$ (solve for t)

→ add errors to get security

$$Enc_t(x) = \left[C \in \mathbb{Z}_q^{n \times m} \;:\; t \cdot C = x \cdot t \cdot G + e \right] \leftarrow$$

(small error)

C_1: $t \cdot C_1 = x \cdot t \cdot G + e_1$, C_2: $t \cdot C_2 = y \cdot t \cdot G + e_2$

- Add: $t(C_1 + C_2) = (x \cdot t \cdot G + e_1) + (y \cdot t \cdot G + e_2)$

$$= (x+y) \cdot t \cdot G + (e_1 + e_2)$$

- Mult: $t \cdot C_1 \cdot G^{-1}(C_2) = (x \cdot t \cdot G + e_1) \cdot G^{-1}(C_2)$

$$= x \cdot t \cdot C_2 + e_1 \cdot G^{-1}(C_2)$$

$$= x(y \cdot t \cdot G + e_2) + e_1 \cdot G^{-1}(C_2)$$

$$= x \cdot y \cdot t \cdot G + \underbrace{x \cdot e_2 + e_1 \cdot G^{-1}(C_2)}$$

if $\|e_1\|_\infty \le \beta_1$, $\|e_2\| \le \beta_2$

new error

$$x \cdot \beta_2 + \beta_1 \cdot m \leftarrow$$

- $G = Enc_t(1)$: $t \cdot G = 1 \cdot t \cdot G$

- $G - C_1 \cdot G^{-1}(C_2) = Enc_t(1 - x \cdot y)$

GSW Encryption

Key Gen: choose: $s \leftarrow \mathbb{Z}_q^{n-1}$

$\bar{A} \leftarrow \mathbb{Z}_q^{(n-1)\times m}$ $\quad m = n\lceil \log(q) \rceil + \lambda$

$e \leftarrow \chi^m$: $\|e\|_\infty \le \beta$

$$pk = A = \left[\begin{array}{c} \bar{A} \\ \hline b = s\bar{A} + e \end{array}\right] \in \mathbb{Z}_q^{n\times m}$$

$$sk = t = (-s, 1) \in \mathbb{Z}_q^n$$

$$t \cdot A = (-s, 1)\left[\begin{array}{c} \bar{A} \\ \hline b \end{array}\right] = -s\bar{A} + b = e$$

$Enc_{pk}(x)$: $R \leftarrow \{0,1\}^{m\times m}$

$$C = A \cdot R + x \cdot G$$

$$t \cdot C = t \cdot (A \cdot R + xG) = e \cdot R + x t G = x \cdot t \cdot G + e^*$$

$\rightarrow \|e^*\|_\infty \le m \cdot \beta$

- $Dec_{sk}(C)$: $t \cdot C = x \cdot t \cdot G + e^*$

$- \; t = (-s, 1)$ $\quad \begin{bmatrix} 1 & 2 & 4 & \cdots & \cdots & \lfloor q/2 \rfloor & \cdots \end{bmatrix}$

$x \cdot \frac{q}{2} + e$

$$- \; t \cdot C \cdot G^{-1}\begin{pmatrix} 0 \\ \vdots \\ \lfloor q/2 \rfloor \end{pmatrix} = (x \cdot t \cdot G + e^*) G^{-1}\begin{pmatrix} 0 \\ \vdots \\ 0 \\ q/2 \end{pmatrix}$$

$$= x \cdot \langle t, 0 \cdots, 0, \tfrac{q}{2} \rangle + \hat{e}$$

$$= x \cdot \lfloor q/2 \rfloor + \hat{e}$$

A, C, G : $n \times m$

security:

$$\left(\text{ph}=\ A=\begin{bmatrix}\bar{A}\\ s\bar{A}+e\end{bmatrix},\ C=AR+xG\right) \approx \left(A\leftarrow \mathbb{Z}_q^{n\times m},\ C\leftarrow \mathbb{Z}_q^{n\times m}\right)$$

LWE $\qquad$ SS LHL

$$\left(A\leftarrow\mathbb{Z}_q^{n\times m},\ C=AR+xG\right)$$

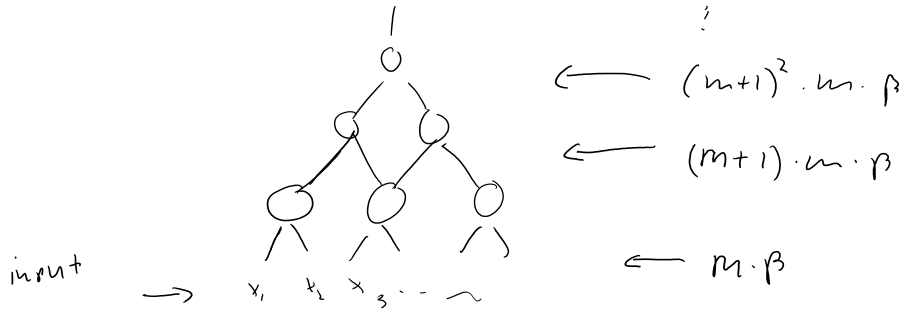
$$C: \quad \epsilon\cdot C = x\cdot\epsilon\cdot G + e$$

$$(-s,1)\cdot\begin{bmatrix}\vdots\\ a_i\\ \vdots\\ b_i\end{bmatrix} + x\cdot\epsilon G + e$$

$$-\langle s, a_i\rangle + b_i = [x\cdot\epsilon\cdot G]_i + e_i$$

$$b_i = \langle s, a_i\rangle + x\cdot 2^l\cdot s_j + e_i$$

$$(a_i, b_i) = \text{Enc}_s(x\cdot 2^l\cdot s_j)$$



depth d $\Rightarrow$ error $(m+1)^{d+1}\cdot\beta$

$\Rightarrow$ can decrypt $q\cdot(m+1)^{d+1}\cdot\beta \le q$

$\Rightarrow$ take-away: can evaluate any poly depth d circuit

	cycle	ladder
Problems:		
– need to decide max depth a-priori (leveled FHE)	✓	✗
– pk, sk, ct sizes, cost of each op grow d.	✓	$1/2$
– exp modulus q to error β ratio, (worse assumption)	$1/2$ quasi-poly	$1/2$

Bootstrapping:

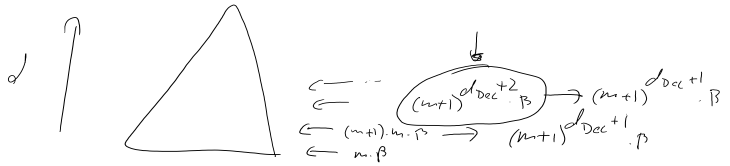
1. Assume $\boxed{ct_{sk} = Enc_{pk}(sk)}$ // error $m\beta$

$ct_x = Enc_{pk}(x)$ // error β^*

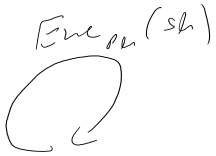
← may not be secure, no proof from LWE

$Eval\left(\underline{Dec(\cdot, ct_x)}, ct_{sk}\right)$

$= Enc_{pk}(x)$ // $\boxed{(m+1)^{d_{Dec}+1} \cdot \beta}$



$d_{Dec} \approx O(\log \lambda)$



Security?

---

2. $(pk_1, sk_1), \ldots, (pk_d, sk_d)$

give out:
$Enc_{pk_1}(sk_2)$, $Enc_{pk_2}(sk_3) \ldots$

security follows trivially