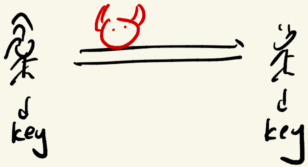
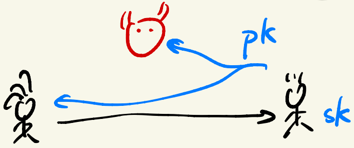
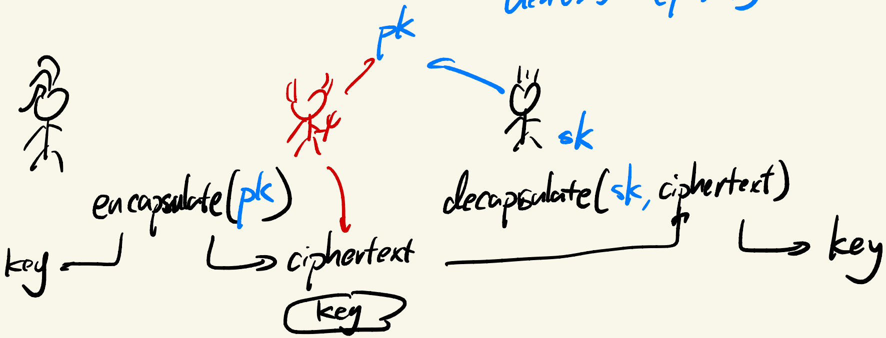
Lec 9. Public-key cryptography

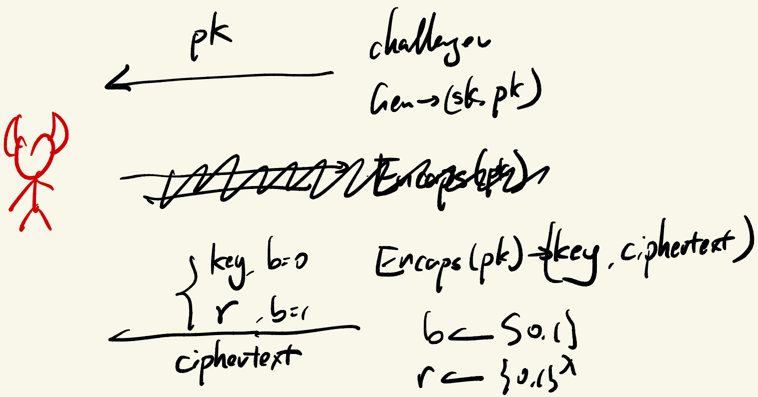
	Def	Security Def
Key-agreement Key-exchange		weak: 😈 can not guess key strong: 😈 can not distinguish key from a random string |||
key encapsulation	≡ 2-msg key exchange ≡ Public-key encryption for random messages	CPA-secure CCA-secure
Public-key encryption		indistinguishability in the presence of eavesdropper ‖ CPA-security = multi-msg CPA-security CCA-security

Key encapsulation

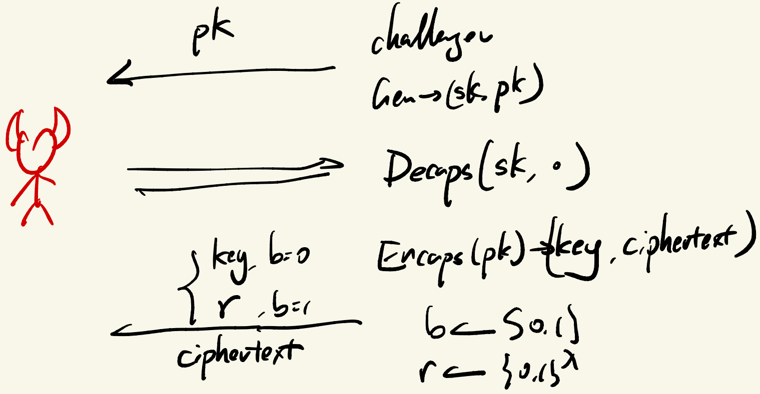
$\mathrm{Gen}(1^\lambda) \to (pk, sk)$



CPA security of key encapsulation



CCA security



Key-encapsulation + Private-Key Encryption ⟹ Public-key Encryption

(Gen, Encaps, Decaps) (~~Gen~~, Enc, Dec)

Assume Gen sample a random key

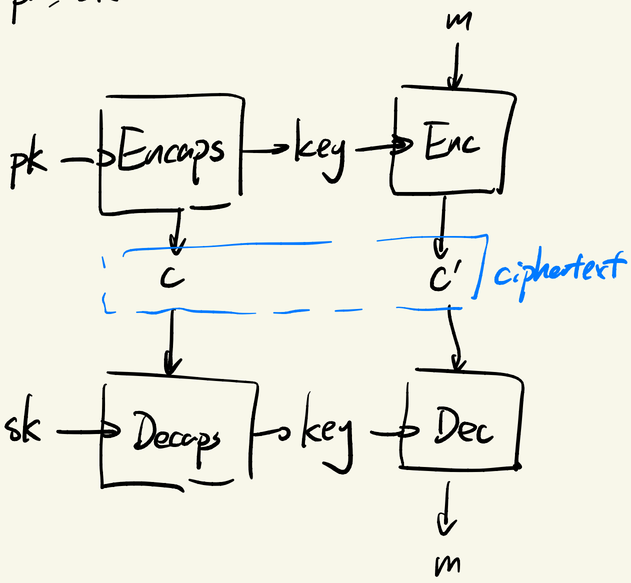
CPA-secure + CPA-secure ⟹ CPA-secure

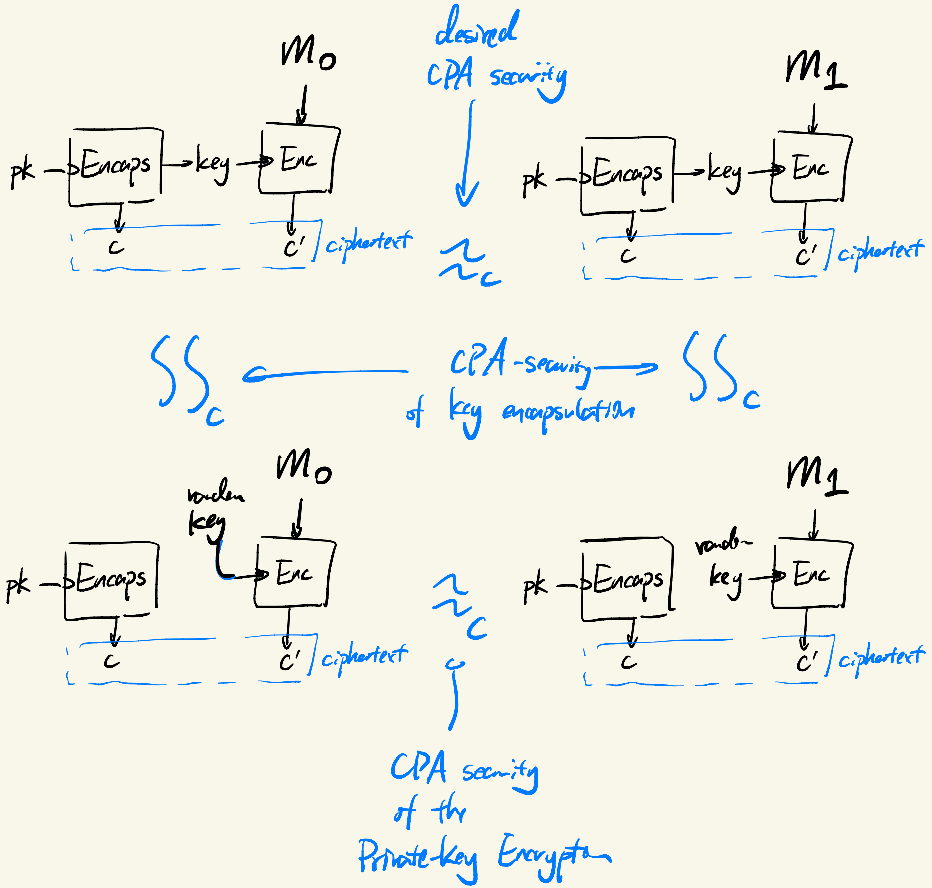
CCA-secure + CCA-secure ⟹ CCA-secure

Gen: $\mathrm{Gen}(1^{\lambda}) \to pk, sk$

Enc(pk, m)

Dec(sk, (c,c'))





m_0
desired
CPA security
m_1
pk
Encaps
key
Enc
c
c'
ciphertext
$\approx_c$
$\approx_c$
CPA-security
of key encapsulation
$\approx_c$
m_0
random key
m_1
random key
$\approx_c$
CPA security
of the
Private-key Encryption

Constructions of Key-Encapsulation / Public-Key encryption

① DDH-based $pk = (G, g, g^x)$.

$sk = x$

$$\mathrm{Encaps}(pk) \to (\underbrace{g^y}_{\text{ciphertext}}, \underbrace{g^{xy}}_{\text{key}})$$

not CCA-secure. $A(pk, g^y)$ query decapsulation of $(g^{y+1}) \to g^{x(y+1)}$

② RSA-based $pk = (N=pq, e)$

$sk = d$ s.t. $de \equiv 1 \mod \phi(N)$

$$\mathrm{Encaps}(pk) \to (\underbrace{\mathrm{hash}(r)}_{\text{key}}, \underbrace{r^e}_{\text{ciphertext}})$$

CCA. secure

③ RSA-based same.

$$\mathrm{Encaps}(pk) \to (\underbrace{\mathrm{lsb}(r^i) \text{ for } i=0,1,\cdots,\lambda-1}_{\text{key}}, \underbrace{r^{e^\lambda}}_{\text{ciphertext}})$$

④ "RSA-based" same.

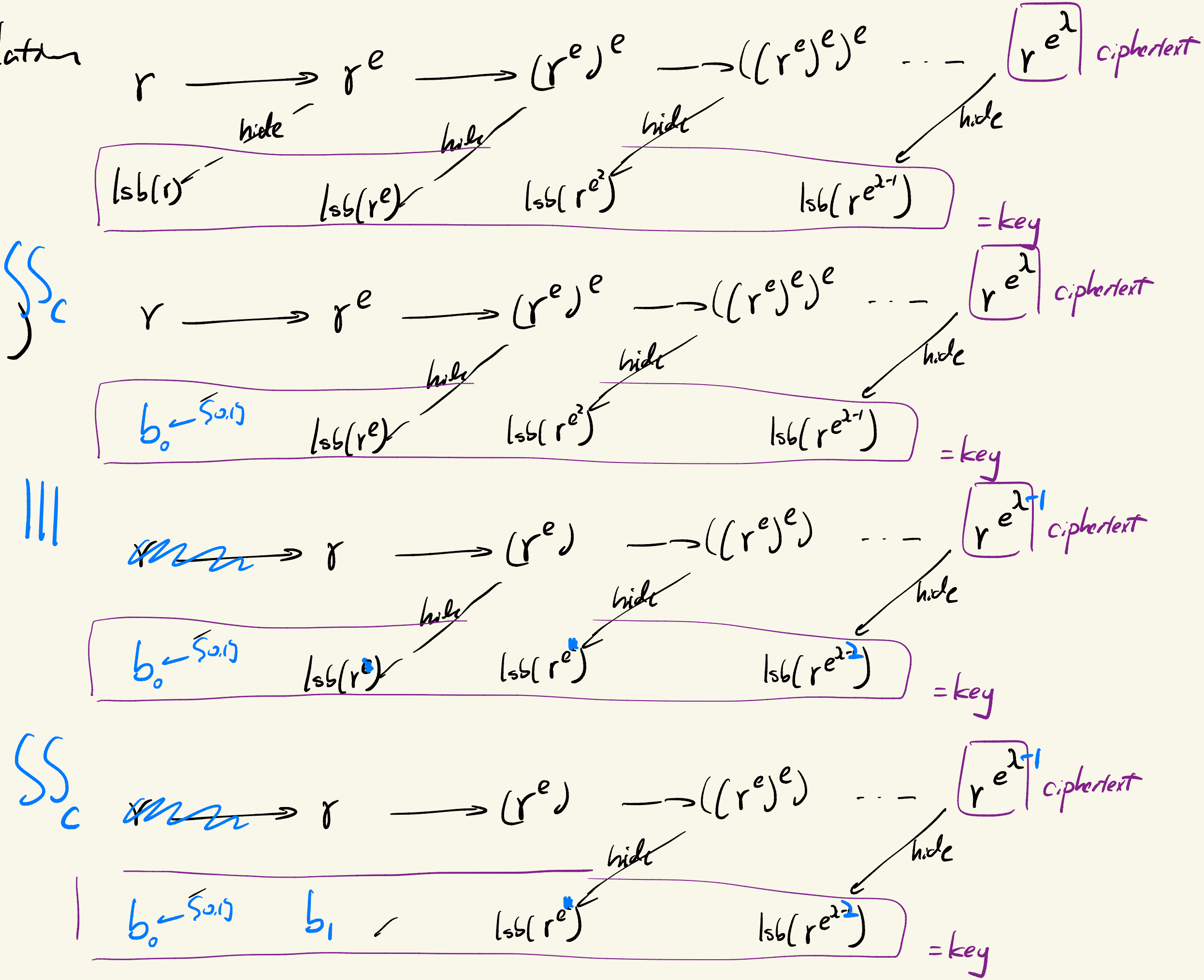
$$\mathrm{Encaps}(pk) \to (\text{last } \lambda\text{-bit of } r, r^e)$$

No proof.

Encapsulation

$(r^e, \mathrm{lsb}(r))$

$\approx_c (r^e, b)$



Quadratic Residue Assumption

$$r \xleftarrow{\$} QR_N \not\approx_c r \xleftarrow{\$} \mathbb{Z}_N^*$$

$$\approx_c \qquad\qquad \approx_c$$

$$r \xleftarrow{\$} QR_N \cup QNR_N$$

$N = pq$

$$\begin{array}{ccccc} \mathbb{Z}_N^* & \cong & \mathbb{Z}_p^* & \times & \mathbb{Z}_q^* \\ \rotatebox{90}{$\subseteq$} & & \rotatebox{90}{$\subseteq$} & & \rotatebox{90}{$\subseteq$} \\ QR_N & \cong & QR_p & \times & QR_q \end{array}$$

Goldwasser-Micali Encryption

$pk = (N = pq,\ a \in QNR_N) \quad sk = (p, q)$

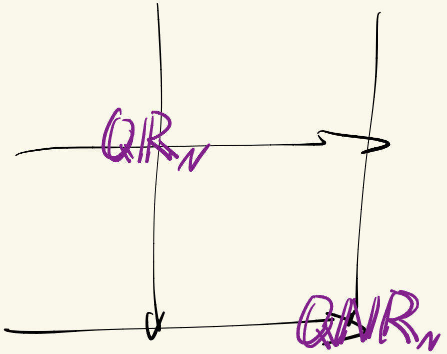
$$Enc(pk, m \in \{0,1\}) = r^2 \cdot a^m$$

$a \notin QR_p$

$$\mathbb{Z}_p^* = QR_p \cup aQR_p$$

$$\begin{array}{c} \mathbb{Z}_q^* \\ \parallel \\ QR_q \\ \cup \\ b\,QR_q \end{array}$$

$b \notin QR_q$



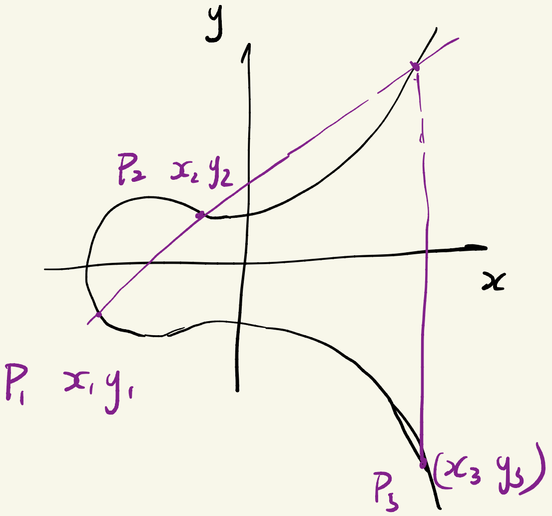
Elliptic Curve

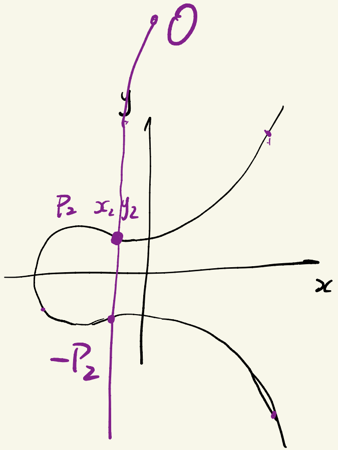
$E = \{(x, y) \in Q^2 \mid y^2 = x^3 + ax + b\}$

$y^2 = x^3 + ax + b$

$Z_{p1} \simeq Z_p^* \simeq EC$

RSA, DDH





$P_1 \boxplus P_2 = P_3$

$$\frac{\left(\frac{(y_1-y_2)x + (x_1y_2 - x_2y_1)}{x_1 - x_2}\right)^2 - (x^3 + ax + b) = 0}{(x - x_1)(x - x_2)}$$

$E/\mathbb{F}_p = \{(x,y) \in \mathbb{F}_p^2 \mid y^2 = x^3 + ax + b\} \cup \{0\}$ $\qquad |E| = 1 + p \pm O(\sqrt{p})$

$y^2 = x^3 + ax + b \qquad a, b \in \mathbb{F}_p$

$E(\mathbb{F}_{p^k}) = \{(x,y) \in \mathbb{F}_{p^k} \times \mathbb{F}_{p^k} \mid y^2 = x^3 + ax + b\} \cup \{0\}$

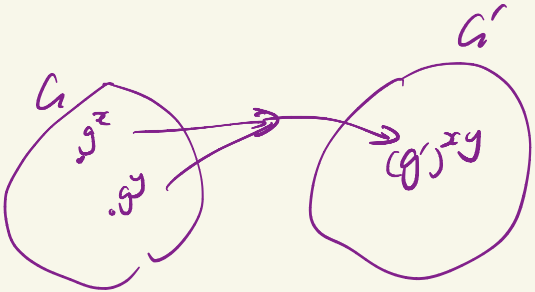
Let $q \mid |E|$ for some big prime q

$\Rightarrow \exists\, g \in E$ order$(g) = q$ $\quad g^q = 0$

Consider $\langle g \rangle = \{0, g, g^2, \dots, g^{q-1}\} \cong \mathbb{Z}_q$

Pairing:

$$\begin{tikzcd}
& \mathbb{Z}_q \arrow[dl, dash, "\simeq"'] \arrow[dr, dash, "\simeq"] & \\
G \subset E & & G_T \subset E' \\
g & & g_T
\end{tikzcd}$$



- $\mathrm{Gen}(1^\lambda) \to (q, G, g, G_T, g_T)$

- $\exists$ poly-time function e

$$e(g^x, g^y) \Rightarrow g_T^{\,xy}$$

- Decisional Bilinear Diffie-Hellman (DBDH)

$$(g^x, g^y, g^z, g^{xyz}) \approx_c (g^x, g^y, g^z, g^w)$$

Generic Group Model

- Oracle

$\mathbb{Z}_q$ $(2^\lambda > q)$

Rand permutation $\{0,1\}^\lambda \to \{0,1\}^\lambda$

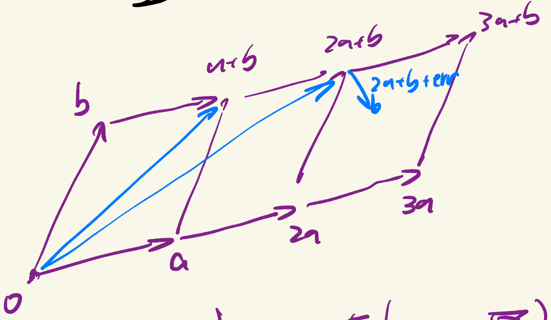
```
Size()   return $q$
Zero()   return $P(0)$
Rand()   $r \leftarrow \mathbb{Z}_q$
         return $P(r)$
GroupOperation$(a,b)$
    return $P(P^{-1}(a) + P^{-1}(b) \bmod q)$
```

Lattice-based Cryptography

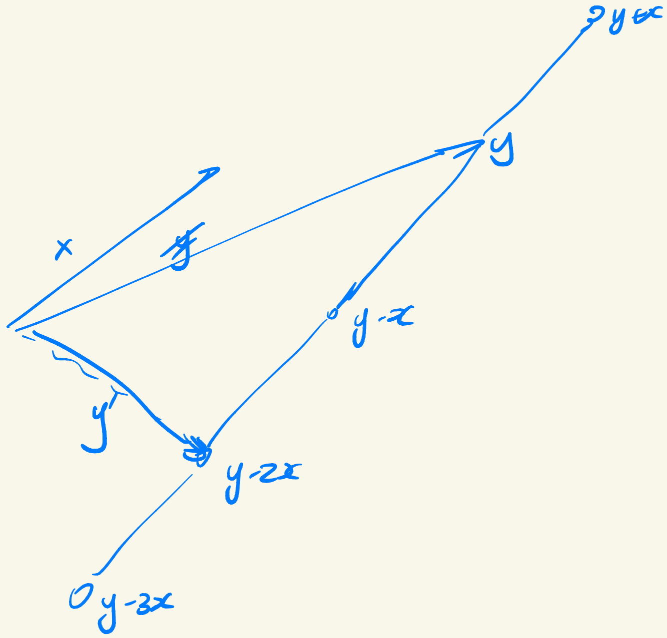
Lattice: a discrete subgroup of $\mathbb{R}^n$

a subgroup of $\mathbb{Z}^n$

e.g. $\mathbb{Z}^n$



$$\langle \bar{a}, \bar{b} \rangle = \{ i\bar{a} + j\bar{b} \mid i, j \in \mathbb{Z} \}$$



$$\langle x, y \rangle = \langle x, y \pm kx \rangle$$

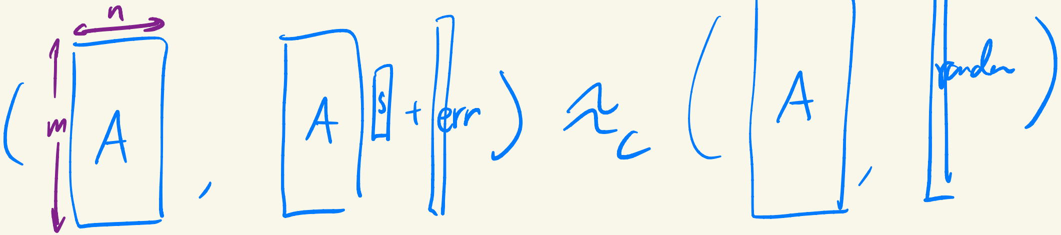
Learning with Error (LWE)

Assumption depends on error distribution

$\vec{a}_1, \quad \vec{a}_1^T \cdot \vec{s} + err \mod Q$

$\vec{a}_2, \quad \vec{a}_2^T \cdot \vec{s} + err \mod Q$

$\vec{a}_3, \quad \vec{a}_3^T \cdot \vec{s} + err \mod Q$



LWE-based PRG:

$$(A, \underbrace{s}_{n \times \lg Q}, \underbrace{err}_{\ll m \cdot \lg Q}) \to (A, \underbrace{As + err}_{m \cdot \lg Q})$$

LWE-based encryption

LWE-based Private-Key encryption

Gen$(1^\lambda) \to \underbrace{n, Q, s}_{k}$

Enc(k, m)

sample $\vec{a} \in \mathbb{Z}_Q^n$, err ← errdistribution

$\underline{a}, \underline{a^T s + err + m \cdot \frac{Q}{2}}$ mod Q

Dec(k, c)

$$\left\lceil \frac{c - a^T s \mod Q}{Q/2} \right\rfloor$$

Public-Key Encryption

$\bar{a}_i, \bar{a}_i^T s + err$

Gen$(1^\lambda) \to \underbrace{m, n, Q, A, As + err}_{pk}, \quad s$

sample small $(r_1 \cdots r_m) = r$

$\sum r_i \bar{a}_i, \quad \sum r_i (a_i^T s + err_i)$

$\underline{r^T A}, \quad \underline{r^T (As + err) + err + m \frac{Q}{2}}$

$(rA) \cdot s + \underbrace{(r^T \cdot err)}_{\text{bounded}} + err$