CSCI 1510

- SWHE from LWE (Continued)
- Bootstrapping SWHE to FHE
- Digital Signatures
- Hash-and-Sign Paradigm
- RSA-based Signatures
- Random Oracle Model

FHE Constructions

Step 1: Somewhat Homomorphic Encryption (SWHE)

- over Integers
- from LWE (GSW)

Step 2: Bootstrapping

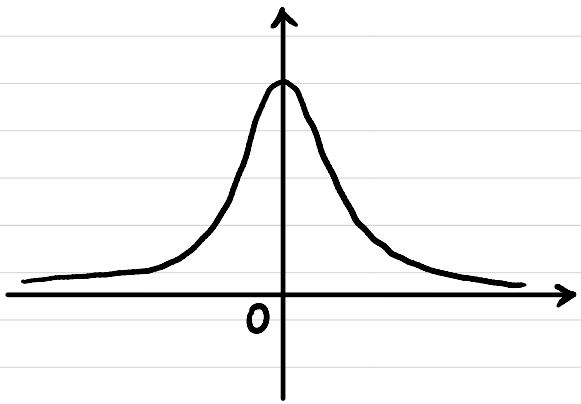
Post-Quantum Assumption: Learning With Errors (LWE)

n: security parameter

$q \sim 2^{n^\epsilon}$

$m = \Omega(n \log q)$

χ: distribution over $\mathbb{Z}_q$
(concentrated on "small integers")

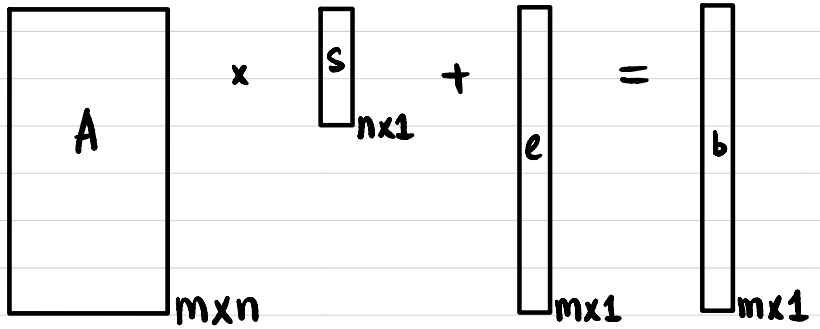


$$\Pr[|e| > \alpha \cdot q \mid e \leftarrow \chi] \leq \text{negl}(n)$$

$\alpha \ll 1$

Def We say the decisional $\text{LWE}_{n,m,q,\chi}$ problem is (quantum) hard if $\forall$ (quantum) PPT $\mathcal{A}$, $\exists$ negligible function $\epsilon(\cdot)$ s.t.

$$\left| \Pr\left[\begin{matrix} A \xleftarrow{\$} \mathbb{Z}_q^{m \times n} \\ s \xleftarrow{\$} \mathbb{Z}_q^n \\ e \leftarrow \chi^m \end{matrix} : \mathcal{A}(A, [As + e \bmod q]) = 1 \right] - \Pr\left[\begin{matrix} A \xleftarrow{\$} \mathbb{Z}_q^{m \times n} \\ b' \xleftarrow{\$} \mathbb{Z}_q^m \end{matrix} : \mathcal{A}(A, b') = 1 \right] \right| \leq \epsilon(n).$$



SWHE from LWE (GSW)

Attempt 1 (secret-key)

$$sk = t_{nx1} \quad \begin{bmatrix} s \\ 1 \end{bmatrix}_{nx1}$$

$Enc_{sk}(\mu)$: $\mu \in \{0,1\}$

How? Sample $C_0 \in \mathbb{Z}_q^{n \times n}$ s.t. $C_0 \cdot \vec{t} = \text{small}$

$$\begin{bmatrix} C_0 \end{bmatrix}_{nxn} \times \begin{bmatrix} t \end{bmatrix}_{nx1} = \begin{bmatrix} e \end{bmatrix}_{nx1}$$

$$C = C_0 + \mu \cdot I$$

(C: $n \times n$; I: identity matrix)

$Dec_{sk}(c)$: $C \cdot \vec{t} = (C_0 + \mu \cdot I) \cdot \vec{t} = \vec{e} + \mu \cdot \vec{t}$

CPA Security?

SWHE from LWE (GSW)

Attempt 1 (secret-key)

Without Error: $c \cdot \vec{t} = \mu \cdot \vec{t}$

With Error: $c \cdot \vec{t} = \mu \cdot \vec{t} + \vec{e}$

Homomorphism:

$c_1 \cdot \vec{t} = \mu_1 \cdot \vec{t}$

$c_2 \cdot \vec{t} = \mu_2 \cdot \vec{t}$

Homomorphism:

$c_1 \cdot \vec{t} = \mu_1 \cdot \vec{t} + \vec{e}_1$

$c_2 \cdot \vec{t} = \mu_2 \cdot \vec{t} + \vec{e}_2$

Additive Homomorphism?

$$c = c_1 + c_2$$

$$c \cdot \vec{t} = (c_1 + c_2) \cdot \vec{t} = (\mu_1 + \mu_2) \cdot \vec{t}$$

Additive Homomorphism?

$$c = c_1 + c_2$$

$$c \cdot \vec{t} = (c_1 + c_2) \cdot \vec{t} = (\mu_1 + \mu_2) \cdot \vec{t} + (\vec{e}_1 + \vec{e}_2)$$

Multiplicative Homomorphism?

$$c = c_1 \cdot c_2$$

$$\begin{aligned} c \cdot \vec{t} &= (c_1 \cdot c_2) \cdot \vec{t} \\ &= c_1 \cdot (c_2 \cdot \vec{t}) \\ &= c_1 \cdot \mu_2 \cdot \vec{t} \\ &= \mu_2 \cdot (c_1 \cdot \vec{t}) \\ &= \mu_2 \cdot \mu_1 \cdot \vec{t} \end{aligned}$$

Multiplicative Homomorphism?

$$c = c_1 \cdot c_2$$

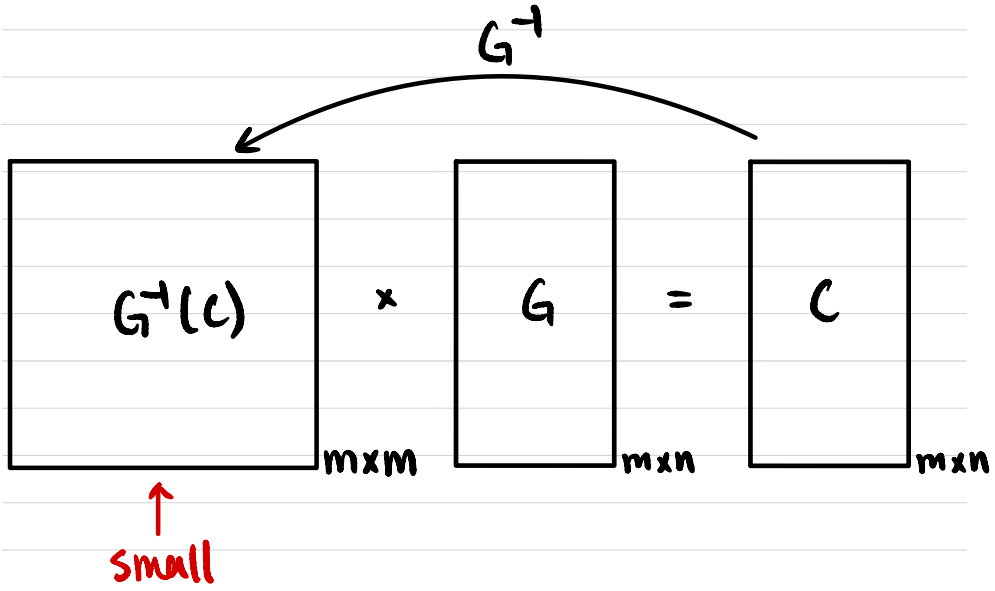
$$\begin{aligned} c \cdot \vec{t} &= (c_1 \cdot c_2) \cdot \vec{t} \\ &= c_1 \cdot (c_2 \cdot \vec{t}) \\ &= c_1 \cdot (\mu_2 \cdot \vec{t} + \vec{e}_2) \\ &= \mu_2 \cdot c_1 \cdot \vec{t} + c_1 \cdot \vec{e}_2 \\ &= \mu_2 \cdot (\mu_1 \cdot \vec{t} + \vec{e}_1) + c_1 \cdot \vec{e}_2 \\ &= \mu_2 \cdot \mu_1 \cdot \vec{t} + \mu_2 \cdot \vec{e}_1 + c_1 \cdot \vec{e}_2 \end{aligned}$$

SWHE from LWE (GSW)

Attempt 2 (secret-key)

Flattering Gadget:

Gadget matrix $G \in \mathbb{Z}_q^{m \times n}$

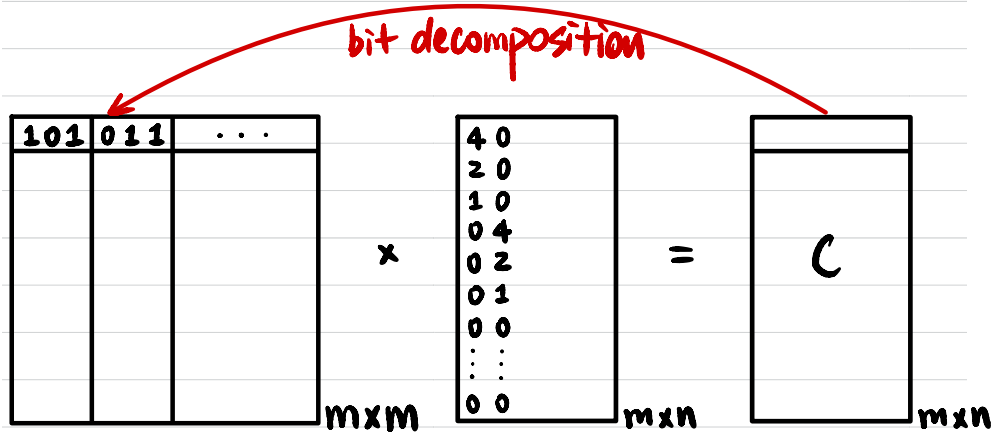


Inverse transformation

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

$\forall C \in \mathbb{Z}_q^{m \times n}$, $G^{-1}(C) = \text{small}$

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



$m = n \cdot \lceil \log q \rceil$

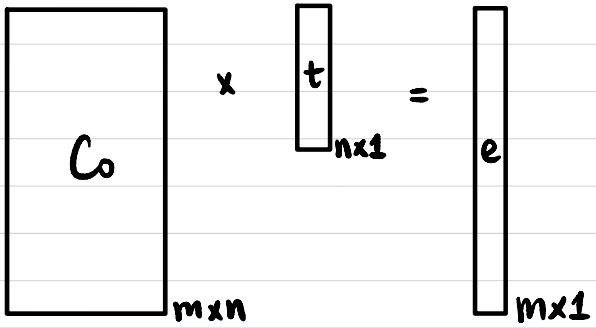
SWHE from LWE (GSW)

Attempt 2 (secret-key)

$$sk = t_{n\times 1} \begin{bmatrix} s \\ 1 \end{bmatrix}_{n\times 1}$$

$Enc_{sk}(\mu)$: $\mu \in \{0,1\}$

Sample $C_0 \in \mathbb{Z}_q^{m\times n}$ s.t. $C_0 \cdot \vec{t} = \text{small}$



$$C = C_0 + \mu \cdot G$$

↑ gadget matrix

$Dec_{sk}(C)$:

$$C \cdot \vec{t} = (C_0 + \mu \cdot G) \cdot \vec{t}$$
$$= \vec{e} + \mu \cdot (G \cdot \vec{t})$$

CPA Security?

Homomorphism:

$$C_1 \cdot \vec{t} = \mu_1 \cdot (G \cdot \vec{t}) + \vec{e}_1$$
$$C_2 \cdot \vec{t} = \mu_2 \cdot (G \cdot \vec{t}) + \vec{e}_2$$

Additive Homomorphism?

$$C = C_1 + C_2 \Rightarrow C \cdot \vec{t} = (\mu_1 + \mu_2) \cdot (G \cdot \vec{t}) + (\vec{e}_1 + \vec{e}_2)$$

Multiplicative Homomorphism?

$$C = G^{-1}(C_1) \cdot C_2$$

$$C \cdot \vec{t} = G^{-1}(C_1) \cdot C_2 \cdot \vec{t}$$
$$= G^{-1}(C_1) \cdot (\mu_2 \cdot (G \cdot \vec{t}) + \vec{e}_2)$$
$$= \mu_2 \cdot G^{-1}(C_1) \cdot G \cdot \vec{t} + G^{-1}(C_1) \cdot \vec{e}_2$$
$$= \mu_2 \cdot C_1 \cdot \vec{t} + G^{-1}(C_1) \cdot \vec{e}_2$$
$$= \mu_2 \cdot (\mu_1 \cdot (G \cdot \vec{t}) + \vec{e}_1) + G^{-1}(C_1) \cdot \vec{e}_2$$
$$= \mu_2 \cdot \mu_1 \cdot (G \cdot \vec{t}) + \mu_2 \cdot \vec{e}_1 + G^{-1}(C_1) \cdot \vec{e}_2$$

How homomorphic is it?

$$\#\text{MULT} = \log_m q$$

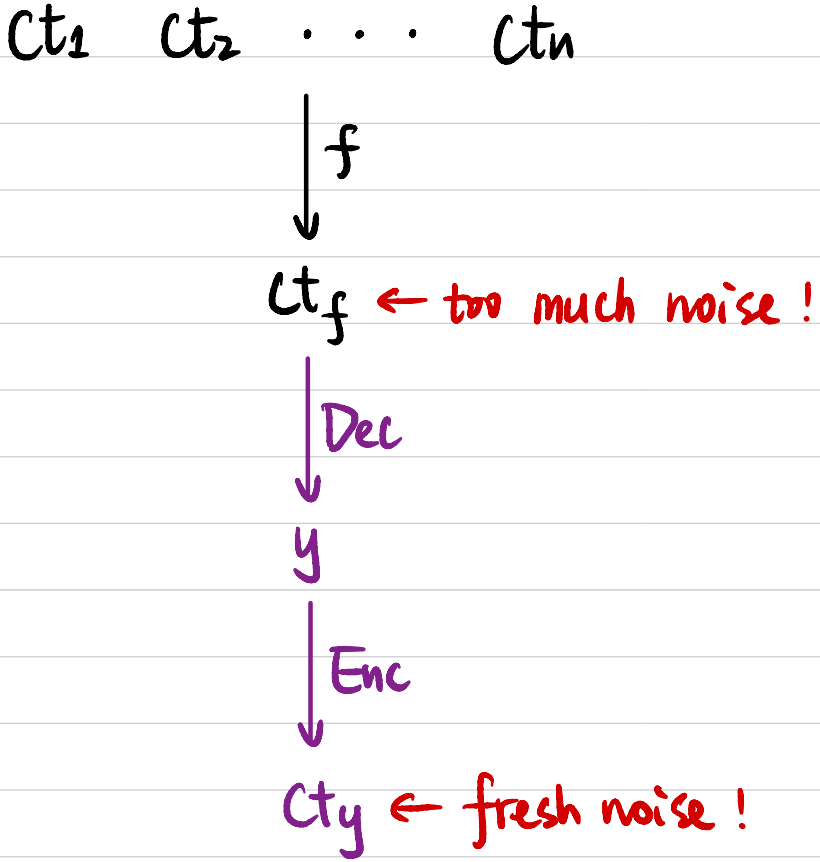
FHE Constructions

Step 1: Somewhat Homomorphic Encryption (SWHE)

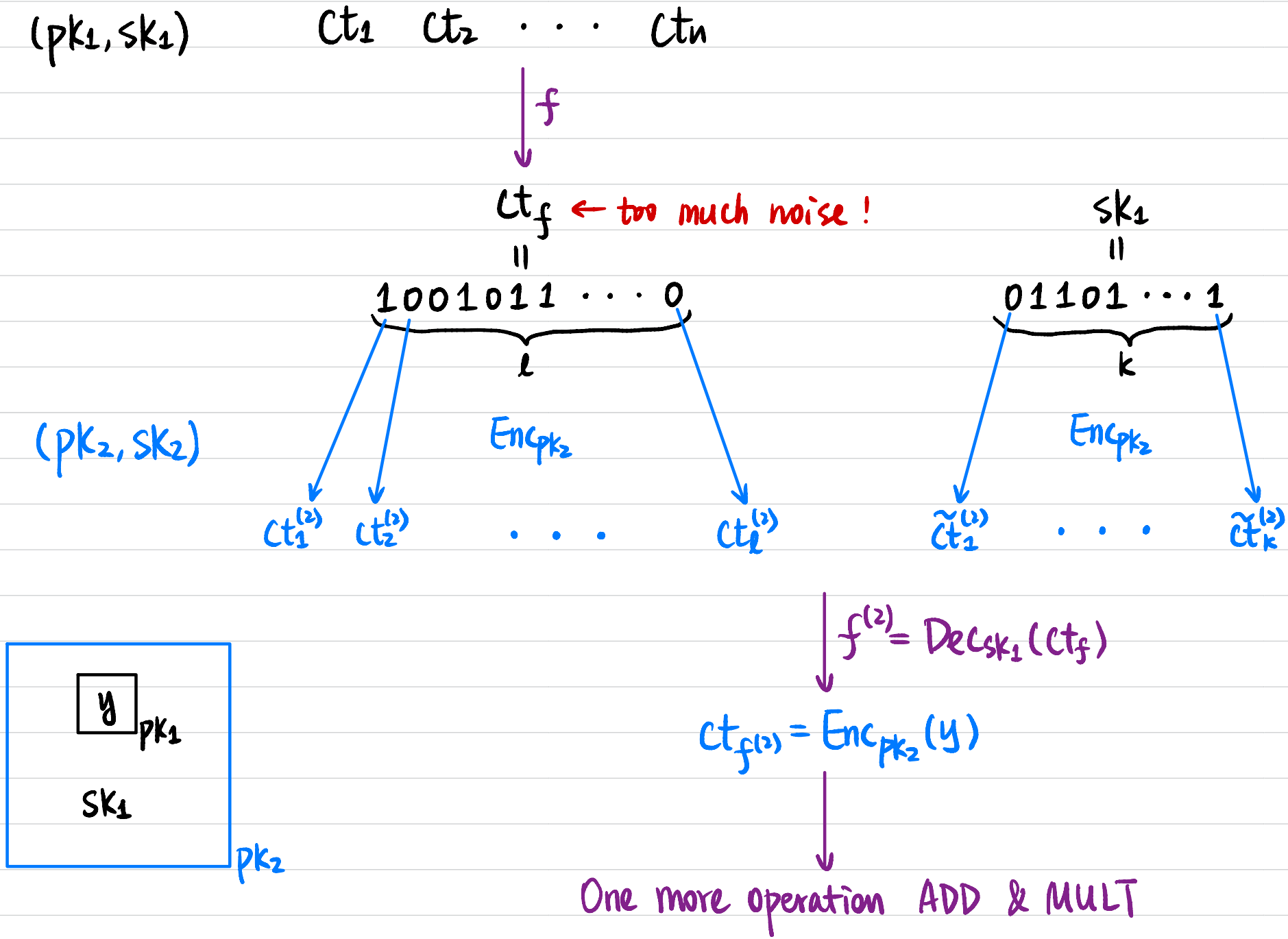
- over Integers
- from LWE (GSW)

Step 2: Bootstrapping

Step 2: Bootstrapping



Leveled FHE


(pk_1, sk_1)
Ct_1 Ct_2 $\cdots$ Ct_n
f
Ct_f ← too much noise !
=
1001011 $\cdots$ 0
ℓ
sk_1
=
01101 $\cdots$ 1
k
(pk_2, sk_2)
Enc_{pk_2}
Enc_{pk_2}
$Ct_1^{(2)}$ $Ct_2^{(2)}$ $\cdots$ $Ct_\ell^{(2)}$
$\tilde{Ct}_1^{(2)}$ $\cdots$ $\tilde{Ct}_k^{(2)}$
$f^{(2)} = Dec_{sk_1}(Ct_f)$
$Ct_{f^{(2)}} = Enc_{pk_2}(y)$
One more operation ADD & MULT
y
pk_1
sk_1
pk_2

Step 2: Bootstrapping

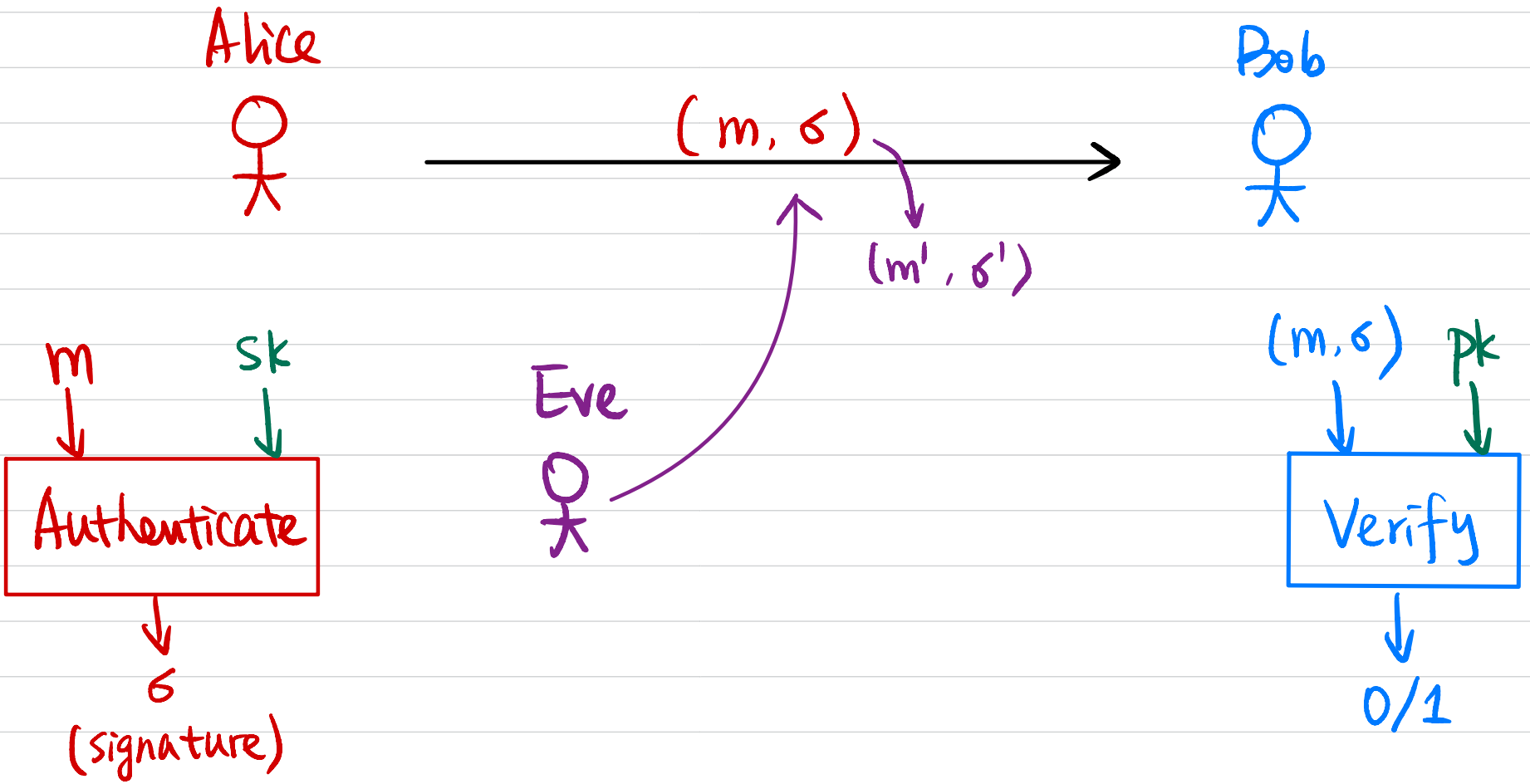
Leveled FHE: pk_1, pk_2, pk_3, . . . , pk_n

$Enc_{pk_2}(sk_1)$ $Enc_{pk_3}(sk_2)$ $Enc_{pk_n}(sk_{n-1})$

FHE: pk. $Enc_{pk}(sk)$

"circular secure" assumption

Digital Signature



Digital Signature

- **Syntax:**

  A digital signature scheme is defined by PPT algorithms (Gen, Sign, Vrfy):

  $(pk, sk) \leftarrow Gen(1^n)$

  $\sigma \leftarrow Sign_{sk}(m) \quad m \in M$

  $0/1 := Vrfy_{pk}(m, \sigma)$

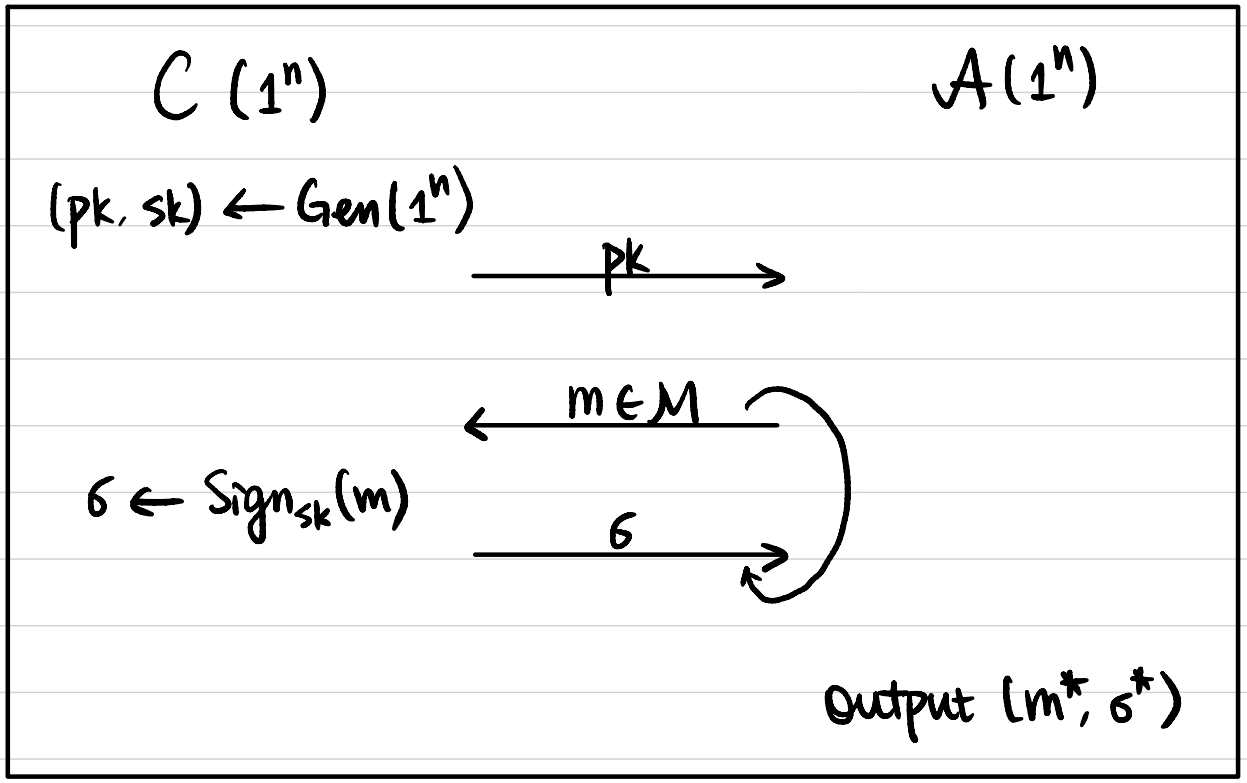
- **Correctness:** $\forall n$, $\forall (pk, sk)$ output by $Gen(1^n)$, $\forall m \in M$

  $$Vrfy_{pk}(m, Sign_{sk}(m)) = 1$$

- **Security?**

Digital Signature

Def A digital signature scheme $\Pi = (Gen, Sign, Vrfy)$ is **secure** if $\forall$ PPT $\mathcal{A}$,
$\exists$ negligible function $\varepsilon(\cdot)$ s.t. $Pr[SigForge_{\mathcal{A},\Pi} = 1] \leq \varepsilon(n)$.



$Q := \{m \mid m \text{ queried by } \mathcal{A}\}$

$SigForge_{\mathcal{A},\Pi} = 1$ ($\mathcal{A}$ succeeds) if

① $m^* \notin Q$, and

② $Vrfy_{pk}(m^*, \sigma^*) = 1$.

Hash-and-Sign Paradigm

Recall: Hash-and-MAC

Secure MAC for fixed-length messages
+
CRHF for arbitrary-length inputs
$\Rightarrow$ Secure MAC for arbitrary-length messages

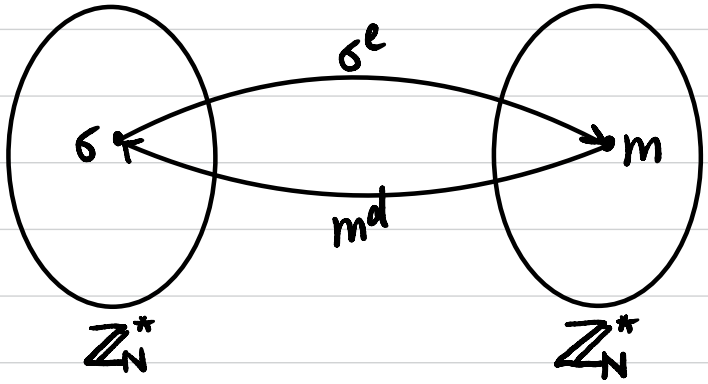
Hash-and-Sign

Secure Signature for fixed-length messages
+
CRHF for arbitrary-length inputs
$\Rightarrow$ Secure Signature for arbitrary-length messages

RSA-based Signatures

Plain RSA Signature:

- $Gen(1^n)$:
  $(N, e, d) \leftarrow GenRSA(1^n)$
  $pk := (N, e)$
  $sk := (N, d)$



- $Sign_{sk}(m)$: $m \in \mathbb{Z}_N^*$
  $\sigma := m^d \mod N$

- $Vrfy_{pk}(m, \sigma)$: $m \stackrel{?}{=} \sigma^e \mod N$

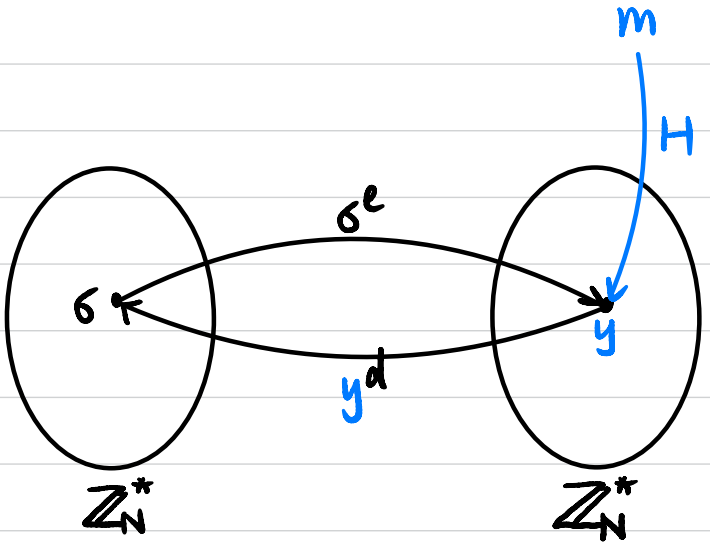
Is it secure? No!

No-message attack
Correlated message attack
Arbitrary message attack

RSA-based Signatures

RSA-FDH (Full Domain Hash) Signature:

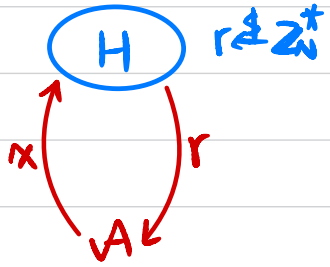
- $Gen(1^n)$:
  $(N, e, d) \leftarrow GenRSA(1^n)$
  $pk := (N, e)$
  $sk := (N, d)$
  Specify a hash function $H: \{0,1\}^* \to \mathbb{Z}_N^*$



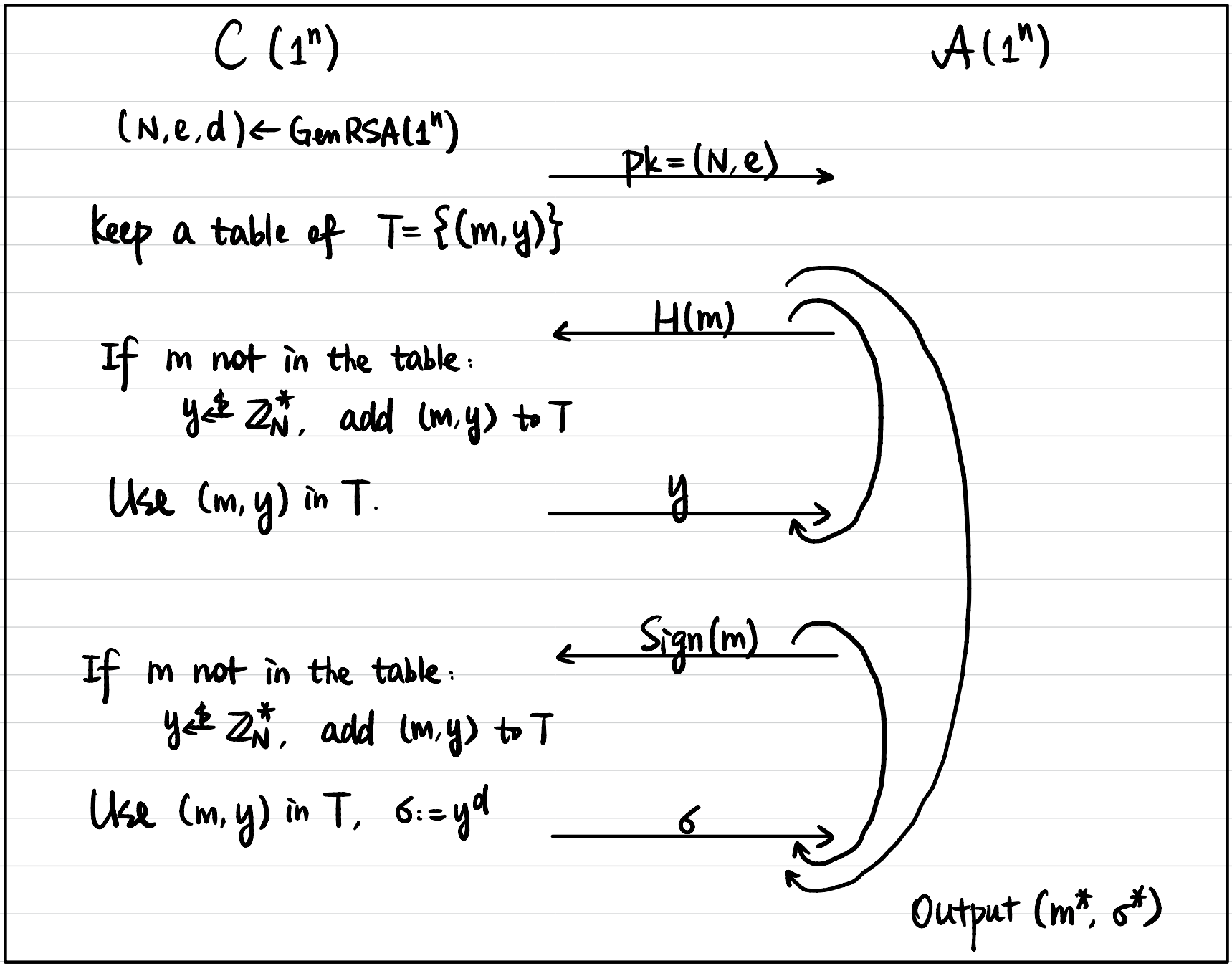
- $Sign_{sk}(m)$: $m \in \{0,1\}^*$
  $\sigma := H(m)^d \bmod N$

- $Vrfy_{pk}(m, \sigma)$: $H(m) \stackrel{?}{=} \sigma^e \bmod N$

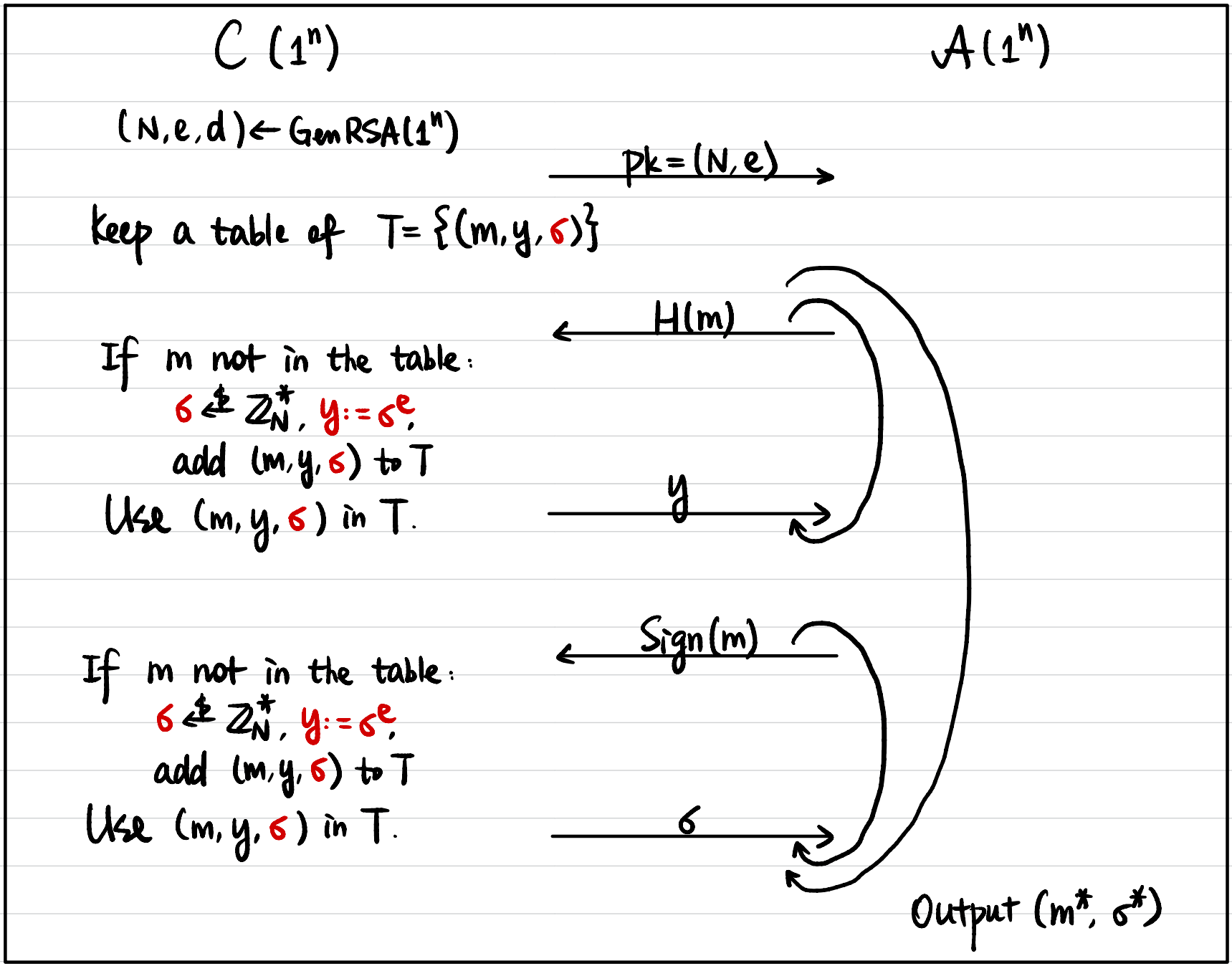
Thm If the RSA problem is hard relative to GenRSA and H is modeled as a random oracle, then this signature scheme is secure.



H₀: RSA-FDH Signature

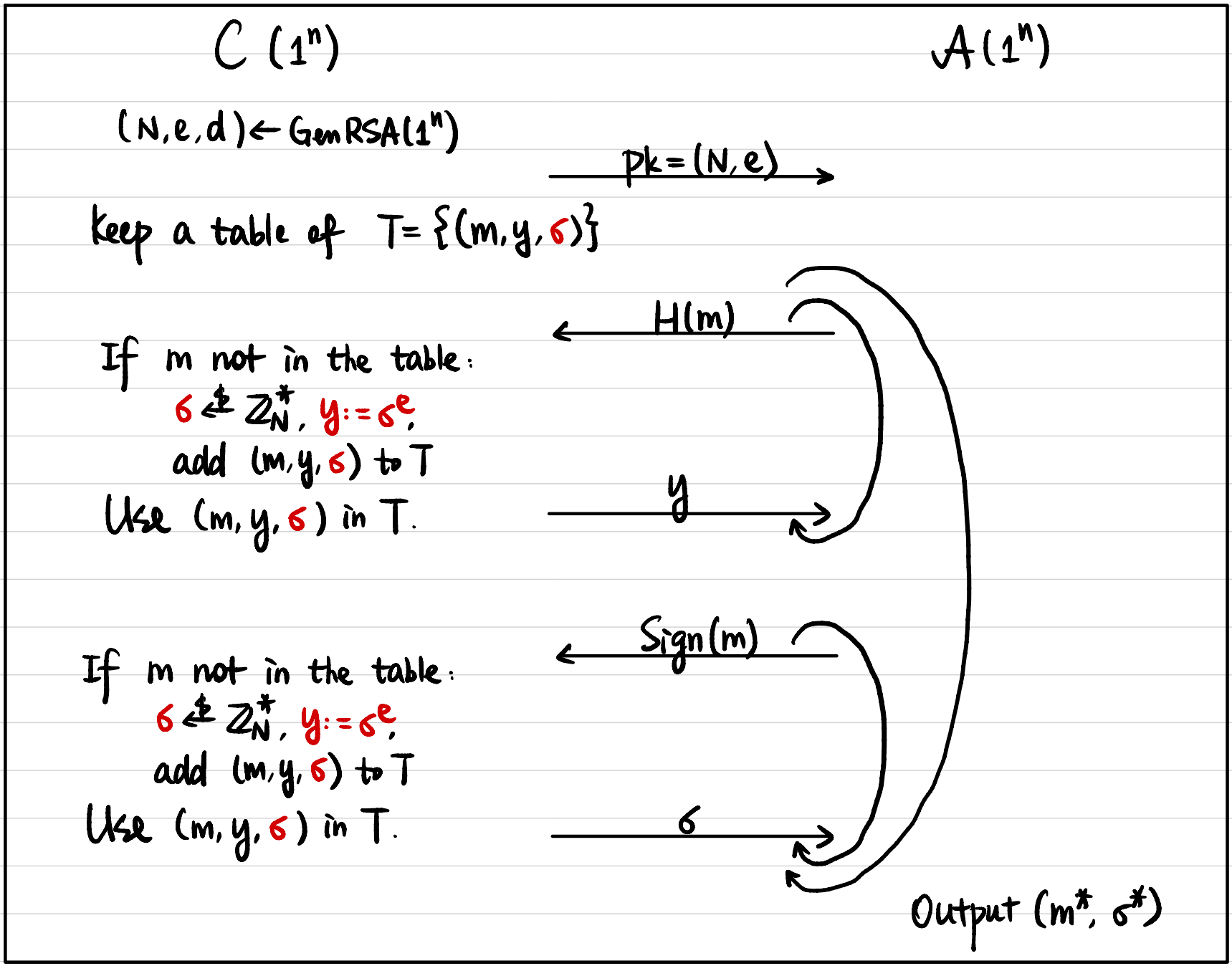


H_1: Sample $\sigma \xleftarrow{\$} \mathbb{Z}_N^*$, set $H(m) := \sigma^e$



H_1 has the same distribution as H_0.

$\mathcal{H}_2$: $m^* \in \{m \mid H(m) \text{ has been queried by } \mathcal{A}\}$



There is a negligible difference in $\mathcal{A}$'s success probability between $\mathcal{H}_1$ & $\mathcal{H}_2$.

$C(1^n)$ RSA $B(1^n)$ H_2 $A(1^n)$

$pk = (N, e, y^*)$ →

$pk = (N, e)$ →

$i \xleftarrow{\$} \{1, 2, \cdots, q\}$

Keep a table of $T = \{(m, y, \sigma)\}$

If i-th $H(\cdot)$ query:
$y := y^*$, add $(m, y^*, ?)$ to T

If m not in the table:
$\sigma \xleftarrow{\$} \mathbb{Z}_N^*$, $y := \sigma^e$,
add (m, y, σ) to T

Use (m, y, σ) in T.

← $H(m)$

y →

If m not in the table:
$\sigma \xleftarrow{\$} \mathbb{Z}_N^*$, $y := \sigma^e$,
add (m, y, σ) to T

Use (m, y, σ) in T.

← $Sign(m)$

σ →

← Output (m^*, σ^*)

Output σ^*