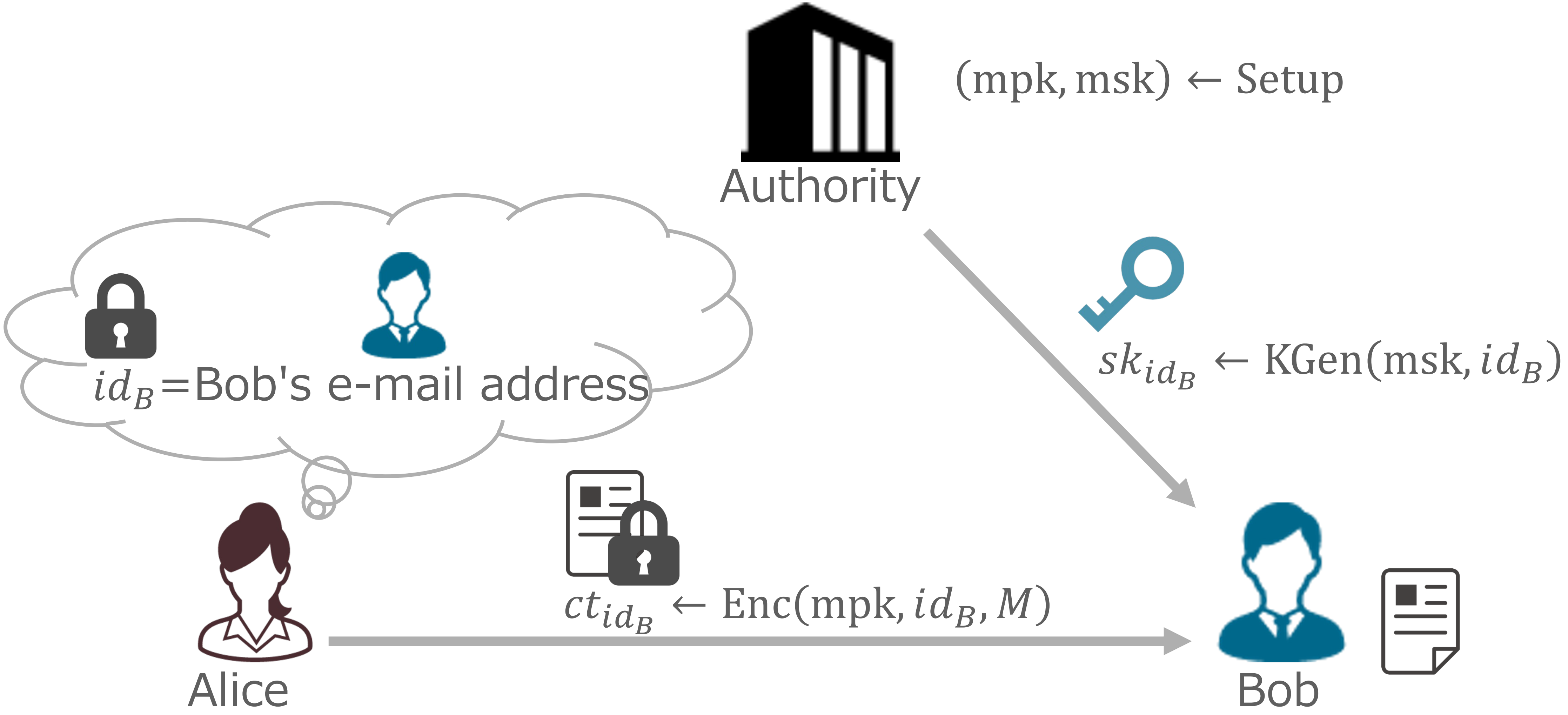
Efficient Identity-Based Encryption with Adaptive Tight Anonymity from RLWE

Toi Tomita and Junji Shikata
Yokohama National University (YNU), Japan

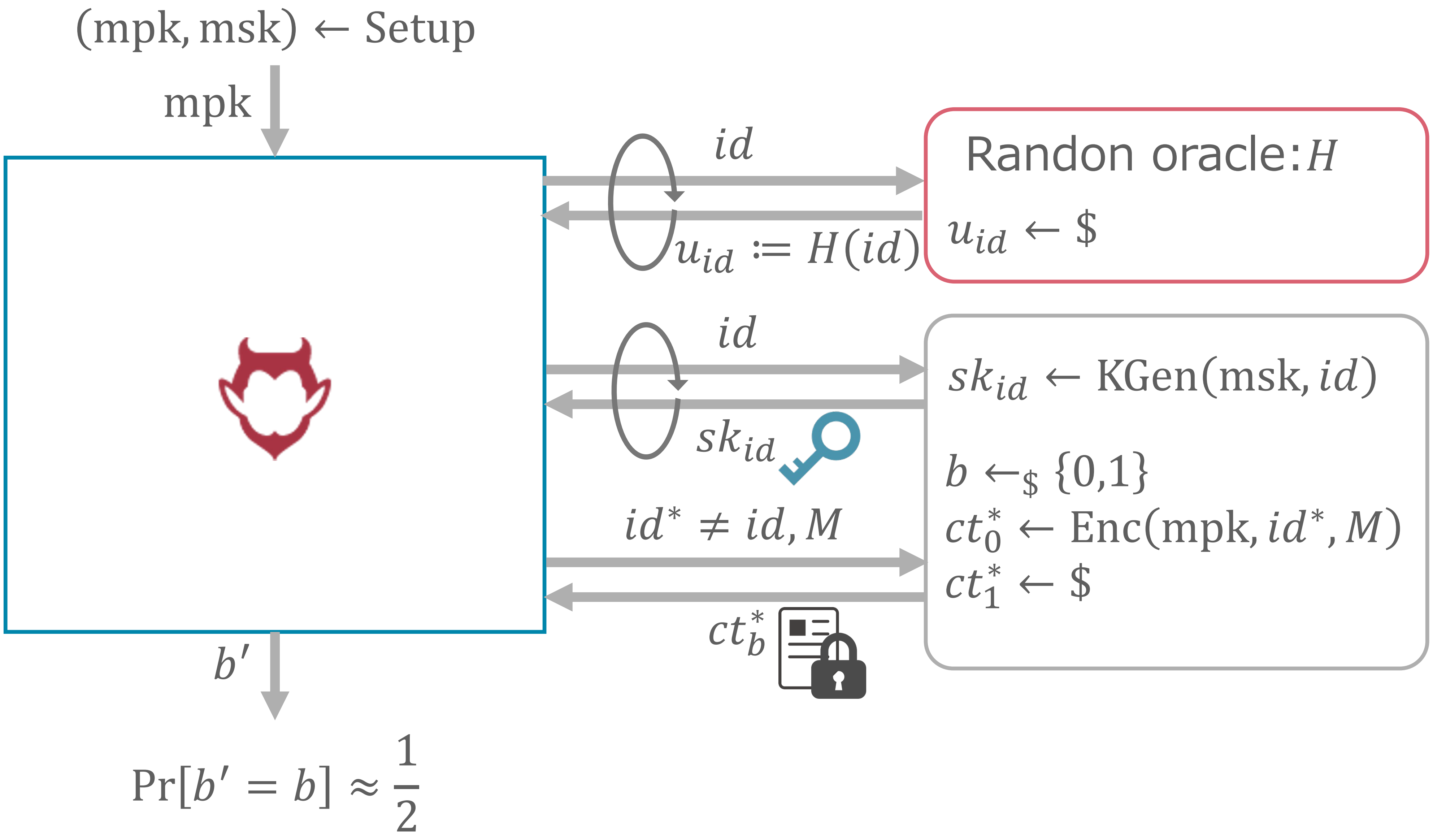
PQCrypto 2024

Background

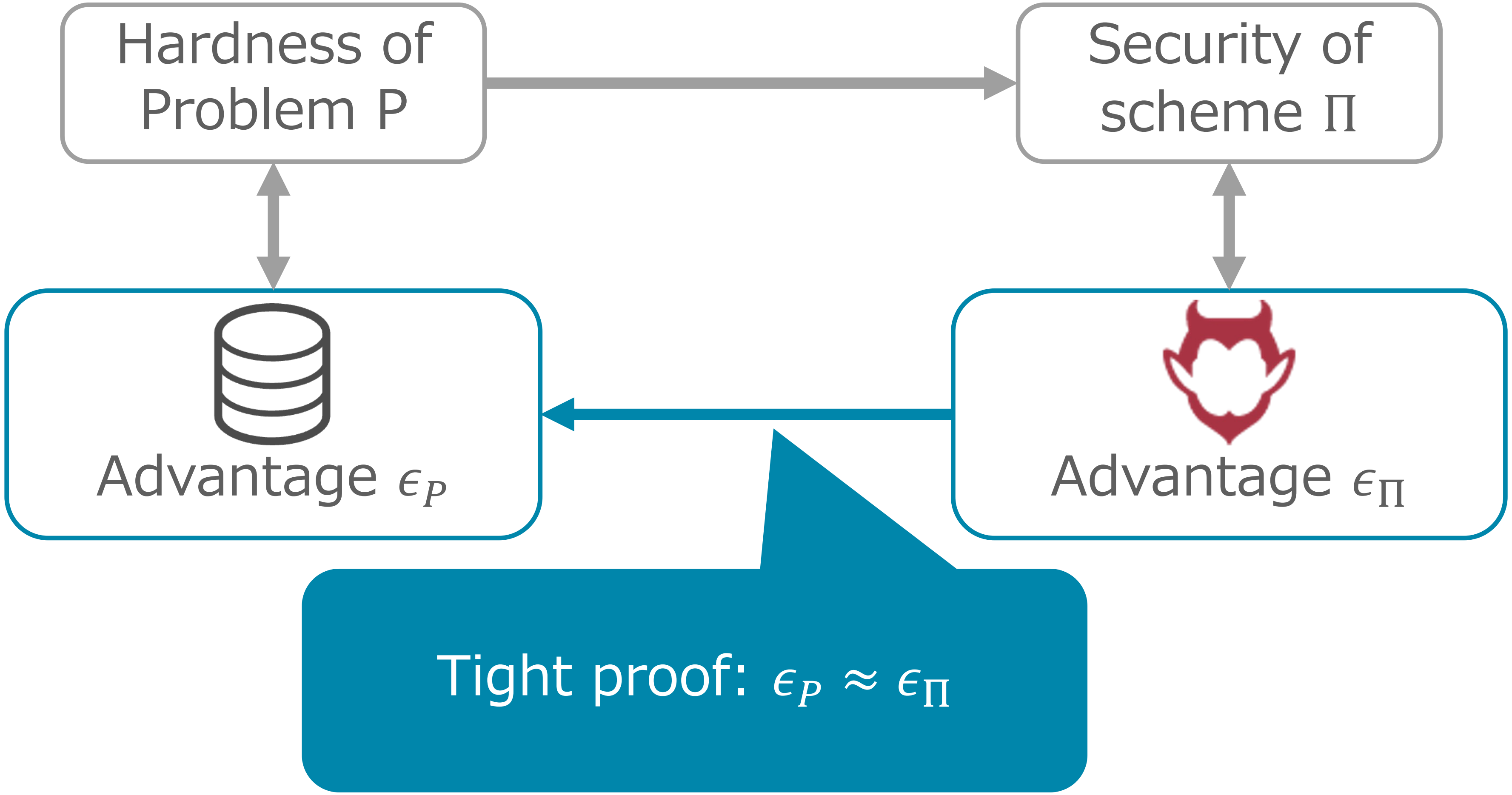
Identity-Based Encryption (IBE) [Sha84]



Security of IBE (in the Randon Oracle Model)



Reduction Cost



Previous Works (Lattice-Based IBE in the (Q)ROM)

Scheme	|mpk|	|sk|	|ct|	Assumption	Tight?	(Q)ROM?
[GPV08]	$O(n^2 \log^2 q)$	$O(n \log^2 q)$		LWE	No	ROM
[Zha12]	$O(n^2 \log^2 q)$	$O(n \log^2 q)$		LWE	No	QROM
[DLP14]	$O(n \log q)$	$O(n \log q)$		NTRU	No	ROM
[KYY18]	$O(n^2 \log^2 q)$	$O(n \log^2 q)$		LWE	Yes	QROM
	$O(n \log^2 q)$			RLWE		
[JHTW24]	$O(n \log^2 q)$	$O(n \log^2 q)$		RLWE	No	ROM

Can we construct an efficient and tightly secure IBE scheme?

Our Contribution

<table>
<tr><th>Scheme</th><th>$|\mathrm{mpk}|$</th><th>$|\mathrm{sk}|$</th><th>$|\mathrm{ct}|$</th><th>Assumption</th><th>Tight?</th><th>(Q)ROM?</th></tr>
<tr><td>[GPV08]</td><td>$O(n^2 \log^2 q)$</td><td colspan="2">$O(n \log^2 q)$</td><td>LWE</td><td>No</td><td>ROM</td></tr>
<tr><td>[Zha12]</td><td>$O(n^2 \log^2 q)$</td><td colspan="2">$O(n \log^2 q)$</td><td>LWE</td><td>No</td><td>QROM</td></tr>
<tr><td>[DLP14]</td><td>$O(n \log q)$</td><td colspan="2">$O(n \log q)$</td><td>NTRU</td><td>No</td><td>ROM</td></tr>
<tr><td rowspan="2">[KYY18]</td><td>$O(n^2 \log^2 q)$</td><td colspan="2" rowspan="2">$O(n \log^2 q)$</td><td>LWE</td><td rowspan="2">Yes</td><td rowspan="2">QROM</td></tr>
<tr><td>$O(n \log^2 q)$</td><td>RLWE</td></tr>
<tr><td>[JHTW24]</td><td>$O(n \log^2 q)$</td><td colspan="2">$O(n \log^2 q)$</td><td>RLWE</td><td>No</td><td>ROM</td></tr>
<tr><td>Ours</td><td>$O(n \log q)$</td><td colspan="2">$O(n \log q)$</td><td>RLWE</td><td>Yes</td><td>QROM</td></tr>
</table>

Contribution:
An efficient and tightly secure IBE scheme from RLWE

Our Approach

Our Approach

Scheme: GPV-IBE + **Approximate** trapdoor

Proof: [KYY18]'s proof + LWE with **hints**

Gentry-Peikert-Vaikuntanathan IBE [GPV08]

$\mathrm{Setup}(1^\lambda) \to (\mathrm{mpk}, \mathrm{msk})$

- $\mathrm{mpk} = \boldsymbol{A} \in \mathbb{Z}_q^{n\times m}, H: \{0,1\}^* \to \mathbb{Z}_q^n$
- $\mathrm{msk} = \tau_{\boldsymbol{A}}$: trapdoor for $\boldsymbol{A}$

$\mathrm{KGen}(\mathrm{msk}, id) \to sk_{id}$

- Short $\boldsymbol{z}_{id} \in \mathbb{Z}^m$ s.t.

$$\boldsymbol{A}\,\boldsymbol{z}_{id} = \boldsymbol{u}_{id} \left(:= H(id)\right)$$

$\mathrm{Enc}(\mathrm{mpk}, id, M) \to ct_{id}$

- $\boldsymbol{c}_0 \approx \boldsymbol{s}\,\boldsymbol{A}$
- $\boldsymbol{c}_1 \approx \boldsymbol{s}\,\boldsymbol{u}_{id} + M \cdot \frac{q}{2}$

$\boldsymbol{s} \leftarrow_\$ \mathbb{Z}_q^{1\times n}$

$\mathrm{Dec}(sk_{id}, ct) \to M$

$$M \cdot \frac{q}{2} \approx \boldsymbol{c}_1 - \boldsymbol{c}_0\,\boldsymbol{z}_{id}$$

[KYY18]'s Proof: Overview

Simulator samples $\boldsymbol{z}_{id}$ and programs $H(id) \coloneqq \boldsymbol{A}\boldsymbol{z}_{id}$ for ***all*** identities id.
→ Simulator can answer ***all*** secret key queries.
→ Simulator can generate the challenge ciphertext for ***all*** identities.

⬇

Simulator behaves identically for all identities.
→ Since the simulator never aborts, the ***security proof is tight.***

[KYY18]'s Proof: Simulation the Challenge Ciphertext

$$\boldsymbol{c}_0 = \boldsymbol{s}\boldsymbol{A} + \boldsymbol{e}$$

$$c_1 = \boldsymbol{c}_0\boldsymbol{z}_{id^*} + M \cdot \frac{q}{2} = \boldsymbol{s}\boldsymbol{A}\boldsymbol{z}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*} + M \cdot \frac{q}{2}$$

$$\approx \boldsymbol{s}\boldsymbol{u}_{id^*} + e' + M \cdot \frac{q}{2}$$

Distributions of e' and $\boldsymbol{e}\boldsymbol{z}_{id^*}$ are different.
→ Adjust by noise re-randomization of [KY16].

LWE

$$\boldsymbol{c}_0 \leftarrow \$$$

$$\boldsymbol{c}_1 = \boldsymbol{c}_0\boldsymbol{z}_{id^*} + M \cdot \frac{q}{2}$$

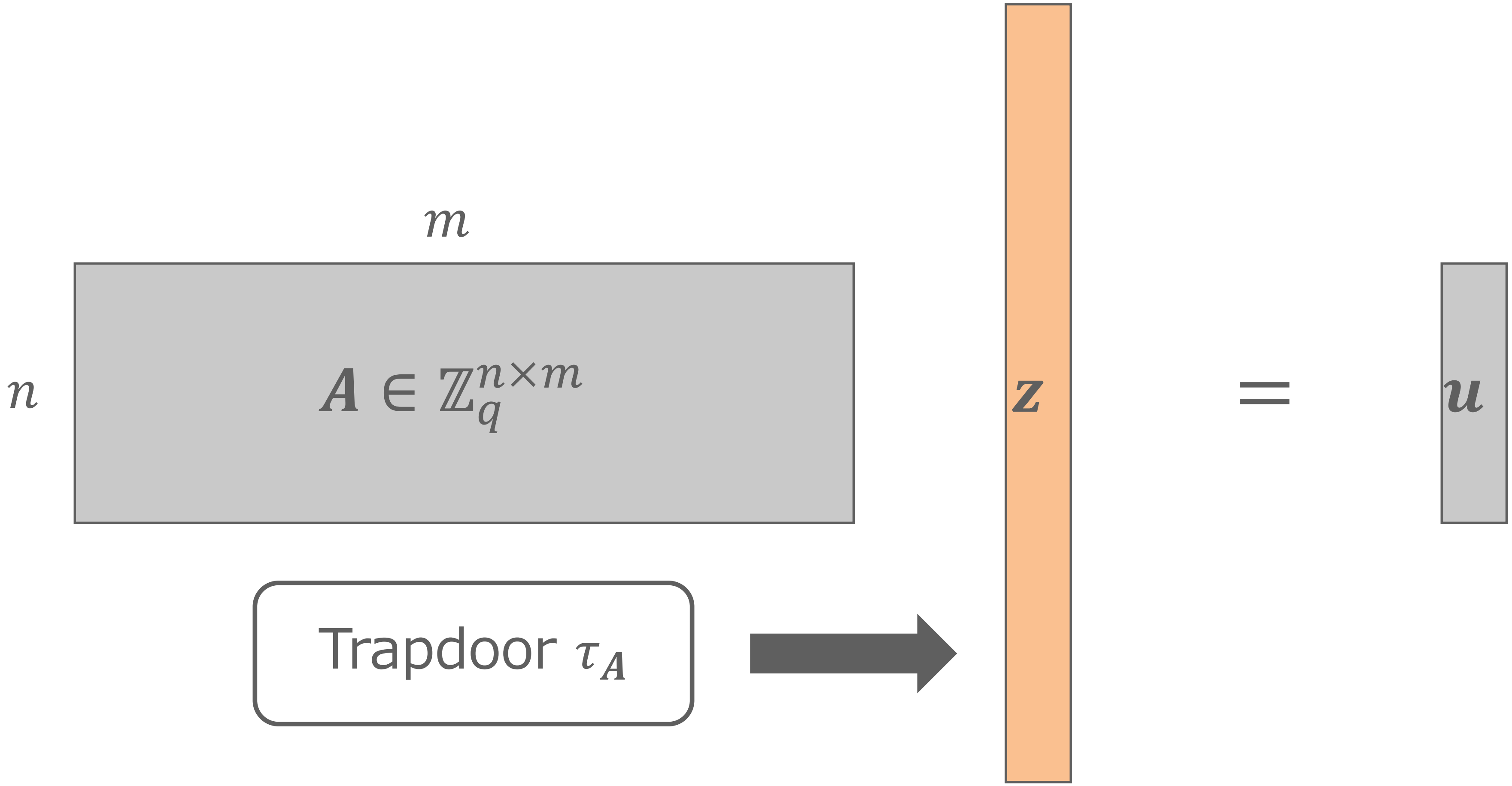
Regularity lemma using entropy of $\boldsymbol{z}_{id^*}$

$$\boldsymbol{c}_0 \leftarrow \$$$
$$c_1 \leftarrow \$$$

No information on M and id^*!

Source of Inefficiency: Trapdoor Sampling [GPV08,MP12]

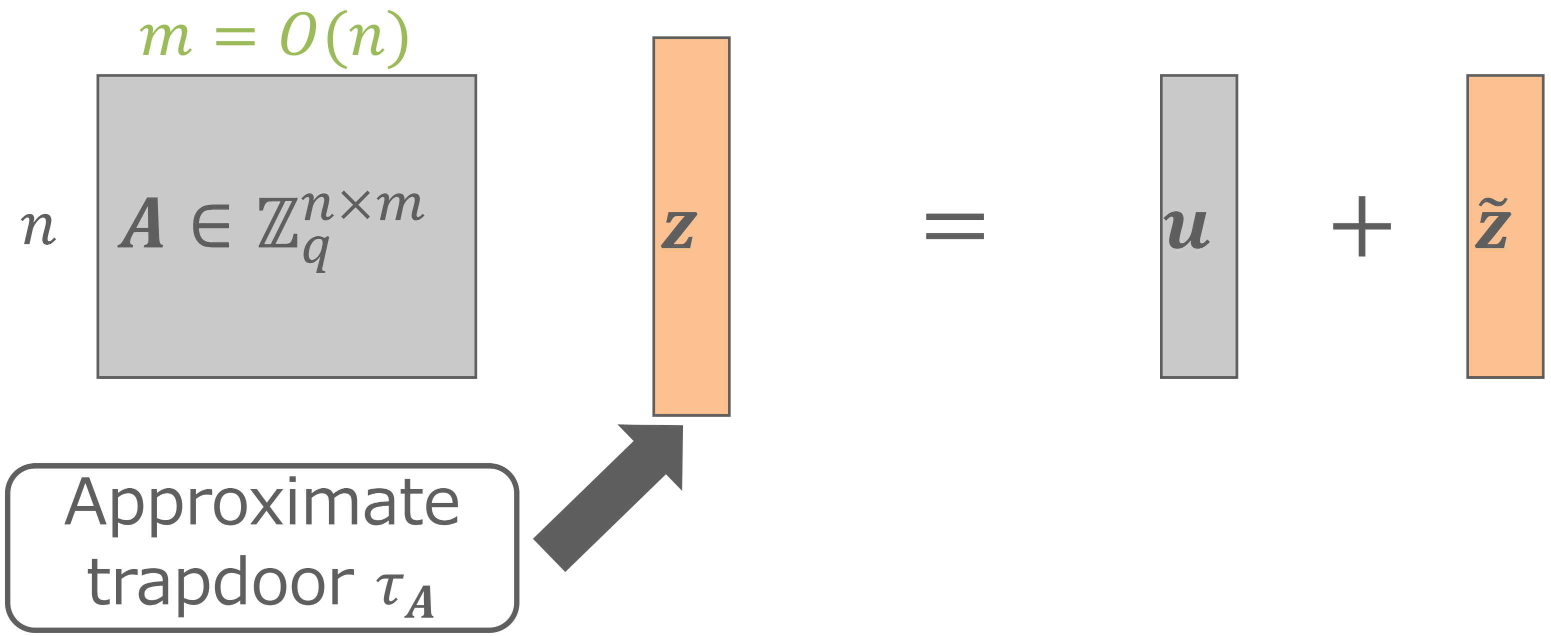
- We can efficiently find $\boldsymbol{z}$ by using the *trapdoor* $\tau_{\boldsymbol{A}}$ for $\boldsymbol{A}$
- But, to use the trapdoor sampling, it is necessary to set $m = O(n \log q)$
- $\rightarrow$ Large mpk, sk_{id}, and ct_{id} ☹



Approximate Trapdoor Sampling [CGM19,YJW23]

[YJW23]: **Approximate** Trapdoor τ_A

- We can efficiently find $\boldsymbol{z}$ even for smaller $\boldsymbol{m = O(n)}$ by using $\tau_{\boldsymbol{A}}$.
- $\rightarrow$ Smaller mpk, sk_{id}, and ct_{id} ☺



Our Scheme : [GPV08] + [YJW23]

$\mathsf{Setup}(1^\lambda) \to (\mathsf{mpk}, \mathsf{msk})$

- $\mathsf{mpk} = \boldsymbol{A}$, $H: \{0,1\}^* \to \mathbb{Z}_q^n$
- $\mathsf{msk} = \tau_A$: **Approximate** Trapdoor for $\boldsymbol{A}$

$\mathsf{KGen}(\mathsf{msk}, id) \to sk_{id}$

- Short $\boldsymbol{z}_{id} \in \mathbb{Z}^m$ s.t.

$$\boldsymbol{A}\,\boldsymbol{z}_{id} = \boldsymbol{u}_{id} + \tilde{\boldsymbol{z}}$$

$\mathsf{Enc}(\mathsf{mpk}, id, M) \to ct$

- $\boldsymbol{c}_0 \approx \boldsymbol{s}\,\boldsymbol{A}$
- $\boldsymbol{c}_1 \approx \boldsymbol{s}\,\boldsymbol{u}_{id} + M \cdot \frac{q}{2}$

Short $\boldsymbol{s} \leftarrow_\$ \mathbb{Z}^n$

$\mathsf{Dec}(sk_{id}, ct) \to M$

$$M \cdot \frac{q}{2} \approx \boldsymbol{c}_1 - \boldsymbol{c}_0\,\boldsymbol{z}_{id}$$

Smaller mpk, sk_{id}, and ct_{id} are obtained!

Attempts: Following [KYY18]

Simulator samples $(\boldsymbol{z}_{id}, \tilde{\boldsymbol{z}}_{id})$ and programs $H(id) \coloneqq \boldsymbol{A}\boldsymbol{z}_{id} - \tilde{\boldsymbol{z}}_{id}$ for ***all*** id.
→ Simulator can answer ***all*** secret key queries.
→ Can simulator simulate the challenge ciphertext?

$$\boldsymbol{c}_0 = \boldsymbol{s}\boldsymbol{A} + \boldsymbol{e}$$

$$\begin{aligned} c_1 &= \boldsymbol{c}_0\boldsymbol{z}_{id^*} + M \cdot \frac{q}{2} \\ &= \boldsymbol{s}\boldsymbol{A}\boldsymbol{z}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*} + M \cdot \frac{q}{2} \\ &= \boldsymbol{s}\boldsymbol{u}_{id^*} + \boldsymbol{s}\tilde{\boldsymbol{z}}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*} + M \cdot \frac{q}{2} \\ &\approx_? \boldsymbol{s}\boldsymbol{u}_{id^*} + e' + M \cdot \frac{q}{2} \end{aligned}$$

Unfortunately, this additional error term $\boldsymbol{s}\tilde{\boldsymbol{z}}_{id^*}$ cannot be adjusted by noise re-rand.
The noise re-rand. can adjust the error appearing ***before*** the evaluation of $\boldsymbol{A}\boldsymbol{z}_{id^*}$, but not the error appearing ***after*** the evaluation of $\boldsymbol{A}\boldsymbol{z}_{id^*}$.

Simulating the Challenge Ciphertext with Hints

Our idea: Simulate using $\boldsymbol{z}_{id^*}$ and $\boldsymbol{s}\tilde{\boldsymbol{z}}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*}$

$$\boldsymbol{c}_0 = \boldsymbol{s}\boldsymbol{A} + \boldsymbol{e}$$

$$c_1 = \boldsymbol{c}_0\boldsymbol{z}_{id^*} - (\boldsymbol{s}\tilde{\boldsymbol{z}}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*}) + e' + M \cdot \frac{q}{2}$$

$$= \boldsymbol{s}\boldsymbol{A}\boldsymbol{z}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*} - (\boldsymbol{s}\tilde{\boldsymbol{z}}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*}) + e' + M \cdot \frac{q}{2}$$

$$= \boldsymbol{s}\boldsymbol{u}_{id^*} + \boldsymbol{s}\tilde{\boldsymbol{z}} + \boldsymbol{e}\boldsymbol{z}_{id^*} - (\boldsymbol{s}\tilde{\boldsymbol{z}}_{id^*} + \boldsymbol{e}\boldsymbol{z}_{id^*}) + e' + M \cdot \frac{q}{2}$$

$$= \boldsymbol{s}\boldsymbol{u}_{id^*} + e' + M \cdot \frac{q}{2}$$

LWE with **Hints** [MKMS22,WLL24]

→ LWE is hard even given $\boldsymbol{s}\tilde{\boldsymbol{z}} + \boldsymbol{e}\boldsymbol{z}_{id^*}$

→ Hardness of LWE with ***many*** hints ≈ Hardness of LWE

Simulating the Challenge Ciphertext with Hints

$$\boldsymbol{c}_0 = \boldsymbol{s}A + \boldsymbol{e}$$

$$c_1 = \boldsymbol{c}_0 \mathbf{z}_{id^*} - (\boldsymbol{s}\tilde{\mathbf{z}} + \boldsymbol{e}\mathbf{z}_{id^*}) + e' + M \cdot \frac{q}{2}$$

LWE with **hints** $(\boldsymbol{s}\tilde{\mathbf{z}} + \boldsymbol{e}\mathbf{z}_{id^*})$

$$\boldsymbol{c}_0 \leftarrow \$$$

$$c_1 = \boldsymbol{c}_0 \mathbf{z}_{id^*} - (\boldsymbol{s}\tilde{\mathbf{z}} + \boldsymbol{e}\mathbf{z}_{id^*}) + e' + M \cdot \frac{q}{2}$$

(Gaussian) regularity

$$\boldsymbol{c}_0 \leftarrow \$$$
$$c_1 \leftarrow \$$$

No information on M and id^*!

Conclusion

Conclusion

Contribution: An efficient and tightly secure IBE scheme from RLWE

Approach:

- Scheme: GPV-IBE + Compact **approximate** trapdoor
- Proof: [KYY18]’s proof + LWE with **Hints**
  - -> Our proof is somewhat generic since it applies to any approximate trapdoor.

Future Works:

- Improving concrete parameters
- Extending the module-lattice setting

Thank you for listening!!

Appendixes

Security Proof

In the security proof,

- Simulator samples $\{(\boldsymbol{z}_i, \tilde{\boldsymbol{z}}_i)\}_i$ for ***all*** queries.
- Simulator receives the LWE instance $\left(\boldsymbol{A}, \boldsymbol{c}_0 = \begin{cases} \boldsymbol{sA} + \boldsymbol{e} \\ \leftarrow \$ \end{cases}, \{\boldsymbol{s z}_i + \boldsymbol{e}\tilde{\boldsymbol{z}}_i\}_i\right)$.
- For ***all*** id_i, simulator programs $H(id_i) \coloneqq \boldsymbol{A z}_i - \tilde{\boldsymbol{z}}_i$.
  - $\rightarrow$ Simulator can answer ***all*** secret key queries.
  - $\rightarrow$ Simulator can generate the challenge ciphertext for ***all*** id.

As with [KYY18], the ***security proof is tight.***