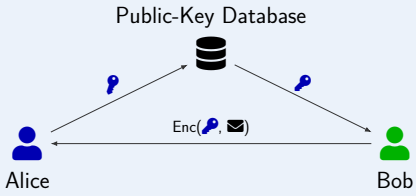
Registration-Based Encryption

Scalable and From Standard Lattice Assumptions

King's College London · Aalto University · Karlsruher Institut für Technologie · KASTEL

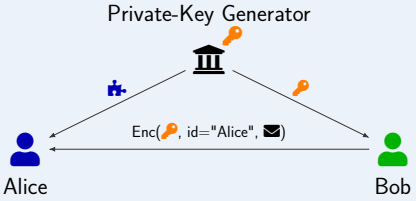
Primitives to Send an Encrypted Email

PKE



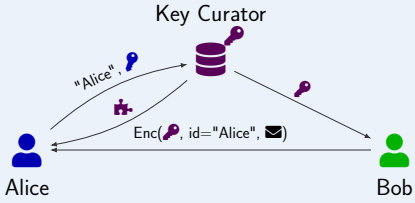
+ Lightweight
– Key management
– Recipient exposed on key retrieval

IBE



+ Key management
– Trusted PKG

RBE



+ Key management
+ No trusted authority
– Update frequency

Primitives: LE, BLE & RBE

Laconic Encryption

- $(\mathsf{pp}, \mathsf{st}, \mathsf{reg}) \leftarrow \mathsf{Setup}(1^n)$: generates initial state 🔑 and registry 🗄
- $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(\mathsf{pp})$: generates a key-pair with pk 🔑
- $\mathsf{ctxt} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{st}, \mathsf{id}, \mathsf{msg})$: encrypts msg for identity id
- $\mathsf{msg} \leftarrow \mathsf{Dec}(\mathsf{sk}, \mathsf{wit}, \mathsf{ctxt})$
- $\mathsf{st}' \leftarrow \mathsf{Upd}^{\mathsf{reg}}(\mathsf{pp}, \mathsf{st}, \mathsf{id}, \mathsf{pk})$: updates state 🔑 and registry 🗄 upon registration of a new identity id according to its pk 🔑
- $\mathsf{wit} \leftarrow \mathsf{WGen}^{\mathsf{reg}}(\mathsf{pp}, \mathsf{st}, \mathsf{id})$: generates 🧩 called witness

RBE — *LE + Efficiency Requirement:* Each identity's witness wit 🧩 updates at most $O(\log N)$ times; N defines the maximum number of registered identites.

Batched LE — *LE* with reg containing B sub-registries + *Efficiency Requirement:* only *small parts* of |ctxt| allowed to scale linearly with B

Transforming LE to RBE

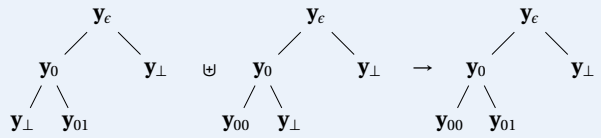
Encryption against B LE instances ⇒ |ctxt| scales **linearly** with B

Take B LE instances for a max. number of users $N = 2^B$. On registry of a new user, add it to an empty LE. Iteratively merge two LEs if there are two LE instances with the same number of users, leaving one empty.
⇒ Max. number of merges (witness updates) per identity $\leq \log N$

BLE to RBE — Sub-registries of BLE take over behaviour of multiple LE instances and their merging behaviour.

DKLLMR Construction [1]

Vector Commitment: SIS-based Merkle-Tree with $\mathsf{H}(\mathbf{x}) = \mathbf{A}_0 \| \mathbf{A}_1 \cdot \mathbf{G}^{-1}(\mathbf{x})$, public keys $\mathbf{y}_{00}, \mathbf{y}_{01}$, terminator symbol $\mathbf{y}_\perp$ and merge operation $\uplus$



PKE: Dual Regev Encryption w.r.t. $(\mathbf{A}_{\mathsf{id}}, \mathbf{v})$

$\mathsf{wit}_{\mathsf{id}} = -\mathbf{G}^{-1}\left(\left(\mathbf{y}_{\mathsf{id}_{[0:i]}\|b}\right)_{b\in\{0,1\}}\right)_{i\in[\ell]}$
$\Rightarrow \mathbf{A}_{\mathsf{id}} \cdot \mathsf{wit}_{\mathsf{id}} = \mathbf{v}$

$$\mathbf{A}_{\mathsf{id}} = \begin{pmatrix} \mathbf{A}_0\|\mathbf{A}_1 & & \\ \mathbf{i}_{\mathsf{id}_1} \otimes \mathbf{G} & \ddots & \\ & \ddots & \mathbf{A}_0\|\mathbf{A}_1 \\ & & \mathbf{i}_{\mathsf{id}_\ell} \otimes \mathbf{G} \quad \mathbf{B} \end{pmatrix}, \quad \mathbf{v} = \begin{pmatrix} \mathbf{y}_\epsilon \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{pmatrix},$$

where $\|$ concatenates matrices horizontally and $\mathbf{i}_j$ is the j-th unit vector, i.e.

$$\mathsf{ctxt} = (\mathbf{c}^T, d), \quad \mathbf{c}^T \approx \mathbf{r}^T \mathbf{A}_{\mathsf{id}} \bmod q, \quad d \approx \mathbf{r}^T \mathbf{v} + \left\lfloor \frac{q}{p} \right\rfloor \cdot \mu \bmod q.$$

Our Optimisations

1. BLE reuses $\mathbf{c}^T$ to share randomness along path to id across all B trees
2. Approximate $\mathbf{G}$ with optimal base $b \approx q^{1/3}$ to reduce |columns| of $\mathbf{A}$
3. Replace statistical arguments in game hops by computational ones
4. Lossy compression of ciphertext

Comparison to Prior Work in Practice

Open-source implementation in Rust supporting hardware-accelerated NTT and discrete Gaussian sampler with high throughput.

	Time (ms)				Size (MB)	
	Upd	Enc	WGen	Dec	ctxt	st
[1]	5 · 1565	150 · 2.95	5 · 0.061	5 · 6.27	150 · 49	8
Our RBE	33.24	15.56	3.38	10.43	7.0	0.34

Fig. 1: Benchmarks on laptop for $N = 2^{30}$, $|ID| \geq 2^{256}$, and $n = 1{,}000$. LE of [1] scaled to RBE.

Conclusion

- Reduction of ciphertext size from 7.2 **G**B [1] to 7.0 **M**B
- Highly scalable: $N = 2^{128}$ users results in |ctxt| = 7.1 MB
- Use-cases: Low frequency, high bandwidth (e.g. emails in companies)
- Further improvements will require different vector commitment

References & Further Details

[1] N. Döttling, D. Kolonelos, R. W. F. Lai, C. Lin, G. Malavolta, and A. Rahimi, *Efficient laconic cryptography from learning with errors*, in EUROCRYPT 2023. DOI: 10.1007/978-3-031-30620-4_14

- Paper: https://ia.cr/2026/717
- Implementation: jnsiemer/scalable_rbe_prototype

Michael Klooß, Russell W. F. Lai, Jan Niklas Siemer & Monisha Swarnakar
Karlsruhe Institute of Technology & KASTEL | Aalto University | King's College London