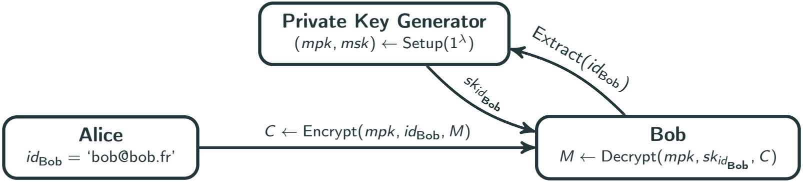
# Practical Implementation of Ring-SIS/LWE based Signature and IBE

Pauline Bert, Pierre-Alain Fouque, Adeline Roux-Langlois, and Mohamed Sabt

PQCrypto 2018, April 11

Univ Rennes, CNRS, IRISA

# Identity Based Encryption




## Advantages

- We no longer need certificates, PKI...
- We can add extra information to the identity.

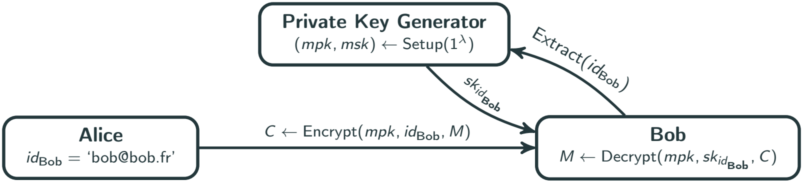
## Some Post-Quantum IBEs

2008 First lattice based IBE, by Gentry, Peikert, and Vaikuntanathan (ROM)

2010 First lattice based IBE in the standard model by Cash, Hofheinz, Kiltz, and Peikert following by work of Agrawal, Boneh, and Boyen,

2017 First code based IBE, by Gaborit, Hauteville, Phan and Tillich (ROM).

# Identity Based Encryption




## Contributions

- We propose an IBE scheme by mixing the Ring version of the IBE scheme à la ABB with the efficient trapdoor of Micciancio and Peikert,
- We also take a look at the underlying signature scheme,
- We implement these schemes in plain C++.

⟶ Both scheme have efficiency comparable to the DLP[1] IBE, and the Falcon NIST submission, with different assumptions (Ring-LWE/SIS vs NTRU).

[1] Ducas, Lyubashevsky, and Prest (2014). "Efficient Identity-Based Encryption over NTRU Lattices". In: *ASIACRYPT*.

# Outline

Hard Lattice Problems and Standard Model IBE framework

Ring Identity Based Encryption Scheme

Underlying Signature Scheme

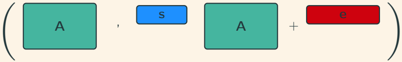
Conclusion

# Hard Lattice Problems and Standard Model IBE framework

# LWE[2]/SIS[3] problems

## Learning With Errors

Given




where

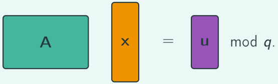
- $\mathbf{A} \hookleftarrow U(\mathbb{Z}_q^{n\times m})$,
- $\mathbf{s} \in \mathbb{Z}_q^n$,
- $\mathbf{e} \hookleftarrow D_{\mathbb{Z}^m,\alpha q}$.

The search problem is to find $\mathbf{s}$.

The decision problem is to distinguish $(\mathbf{A}, \mathbf{s}^T\mathbf{A} + \mathbf{e}^T)$ from $(\mathbf{A}, \mathbf{b}^T) \hookleftarrow U\left(\mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m\right)$.

## Short Integer Solution

Given an uniformly random matrix $\mathbf{A} \hookleftarrow U(\mathbb{Z}_q^{n\times m})$, find a non trivial short vector $\mathbf{x} \in \mathbb{Z}^m$ such that $\|\mathbf{x}\| \leq \beta$ and:




$\longrightarrow$ LWE/SIS are hard:

Regev/Ajtai gave reductions from worst-case problems on lattices to the average-case LWE/SIS problems.

---

[2] Regev (2005). "On lattices, learning with errors, random linear codes, and cryptography". In: *STOC*.

[3] Ajtai (1996). "Generating Hard Instances of Lattice Problems". In: *STOC*.

# Full trapdoor for LWE and SIS

A full trapdoor for the LWE and SIS problems is a short basis $\mathbf{T_A}$ of the lattice

$$\Lambda_q^\perp(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m \text{ such that } \mathbf{Ax} = \mathbf{0} \mod q\}.$$

- Given $\mathbf{A}$, it's hard to find such basis,
- we can generate $\mathbf{A}$ together with $\mathbf{T_A}$, thanks to algorithm $\mathrm{TrapGen}(n, m, q)$,
- we can use $\mathbf{T_A}$ to solve the SIS problem,
  - for the matrix $\mathbf{A}$,
  - for a matrix of the form $(\mathbf{A} \,|\, \mathbf{A}') \in \mathbb{Z}_q^{n\times(m+m')}$,

  i.e find a short non zero $\mathbf{x} \in \mathbb{Z}^{m+m'}$ such that $(\mathbf{A} \,|\, \mathbf{A}')\mathbf{x} = \mathbf{u} \mod q$.

# Public Key Encryption of Dual-Regev[4]

In this scheme, users can share a public matrix $\mathbf{A} \hookleftarrow U(\mathbb{Z}_q^{n \times m})$.

**Alice** | **Bob**

Bob: $sk = \mathbf{x} \hookleftarrow D_{\mathbb{Z}^m,\zeta}$

Bob → Alice: $pk = (\mathbf{A}, \mathbf{u} = \mathbf{A}\mathbf{x})$

Alice: $\mathbf{s} \hookleftarrow U(\mathbb{Z}_q^n)$, $\mathbf{e} \hookleftarrow D_{\mathbb{Z}^m,\tau}$

Alice → Bob: $\mathbf{c}_0^T = \mathbf{s}^T\mathbf{A} + \mathbf{e}^T$

Alice: $M \in \{0,1\}$, $e' \hookleftarrow D_{\mathbb{Z},\tau}$

Alice → Bob: $c_1 = \mathbf{s}^T\mathbf{u} + e' + M.\lfloor q/2 \rfloor$

Bob: $c_1 - \mathbf{c}_0^T\mathbf{x} = \underbrace{e' - \mathbf{e}^T\mathbf{x}}_{\text{small}} + M.\lfloor q/2 \rfloor$

$\longrightarrow$ IND-CPA secure based on the hardness of LWE.

[4] Gentry, Peikert, and Vaikuntanathan (2008). "Trapdoors for hard lattices and new cryptographic constructions". In: *STOC*.

**PKG**

$\mathbf{u} \hookleftarrow U(\mathbb{Z}_q^n)$

$(\mathbf{A}, \mathbf{T_A}) \leftarrow \mathrm{TrapGen}(n, m, q)$

$mpk = (\mathbf{A}, \mathbf{u}, \cdots)$ and $msk = \mathbf{T_A}$

$\mathbf{x}_{\mathrm{Bob}}$ such that

$\mathbf{A}_{\mathrm{Bob}}\mathbf{x}_{\mathrm{Bob}} = \mathbf{u} \mod q$

$\mathbf{x}_{\mathrm{Bob}}$

**Alice**

$\mathbf{s} \hookleftarrow U(\mathbb{Z}_q^n)$,

$\mathbf{e} \hookleftarrow D_{\mathbb{Z}^{m+m'},\tau}$

$M \in \{0,1\}$,

$e' \hookleftarrow D_{\mathbb{Z},\tau}$

$\mathbf{A}_{\mathrm{Bob}} = (\mathbf{A} \mid F(id_{\mathrm{Bob}})) \in \mathbb{Z}_q^{n\times(m+m')}$

where $F$ depends on the construction

$\mathbf{c}_0^T = \mathbf{s}^T \mathbf{A}_{\mathrm{Bob}} + \mathbf{e}^T$ →

$c_1 = \mathbf{s}^T \mathbf{u} + e' + M.\lfloor q/2 \rfloor$ →

**Bob**

$sk = \mathbf{x}_{\mathrm{Bob}}$

$pk = (\mathbf{A}_{\mathrm{Bob}}, \mathbf{u})$

$$c_1 - \mathbf{c}_0^T \mathbf{x}_{\mathrm{Bob}} = \underbrace{e' - \mathbf{e}^T \mathbf{x}_{\mathrm{Bob}}}_{small} + M.\lfloor q/2 \rfloor$$

[5] Cash et al. (2010). "Bonsai Trees, or How to Delegate a Lattice Basis". In: *EUROCRYPT*; Agrawal, Boneh, and Boyen (2010). "Efficient Lattice (H)IBE in the Standard Model". In: *EUROCRYPT*.

# Ring Identity Based Encryption Scheme

# From random lattice to ideal lattice

Consider the rings $R = \mathbb{Z}[x]/(x^n + 1)$ or $R_q = R/qR$, with $n$ a power of 2.

If we have $s$, $a \in R_q$, $s = s_0 + s_1 x + \cdots + s_{n-1}x^{n-1}$,

$$s \cdot a = \begin{pmatrix} s_0 & s_1 & \cdots & s_{n-1} \end{pmatrix} \begin{pmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ -a_{n-1} & a_0 & \cdots & a_{n-2} \\ & & \ddots & \\ -a_1 & -a_2 & \cdots & a_0 \end{pmatrix}$$

$\longrightarrow$ Smaller storage, faster operations.

**LWE:** Given $(\mathbf{A}, \mathbf{s}^T\mathbf{A} + \mathbf{e}^T \mod q)$, find $\mathbf{s} \in \mathbb{Z}_q^n$.
**SIS:** Given $\mathbf{A}$, find a short vector $\mathbf{x} \in \mathbb{Z}^m$ such that $\mathbf{A}\mathbf{x} = \mathbf{u} \mod q$.

**Ring-LWE:** Given $\mathbf{a} \in R_q^{m/n}$ and $(s \cdot a_1 + e_1, \cdots, s \cdot a_{m/n} + e_{m/n})$, find $s \in R_q$.
**Ring-SIS:** Given $\mathbf{a} \in R_q^{m/n}$, find $\mathbf{x} \in R^{m/n}$ such that $\mathbf{a}^T\mathbf{x} = u \mod q$.

# Ring Gadget Trapdoor of [MP12]

The trapdoor construction consists in an almost uniformly random vector of polynomials $\mathbf{a} = (a_1, \cdots, a_m) \in R_q^m$,

$$\mathbf{a} = \left(\mathbf{a}'^T \,\middle|\, h\mathbf{g} - \mathbf{a}'^T \mathbf{T}\right)^T.$$

where:

- $\mathbf{a}' \hookleftarrow U(R_q^{m-k})$,
- $\mathbf{g} = (1, 2, 4, \cdots, 2^{k-1}) \in R_q^k$ with $k = \lceil \log_2 q \rceil$ is the 'gadget vector',
- $h \in R_q$ is an invertible polynomial, called the tag,
- $\mathbf{T} \hookleftarrow D_{R^{(m-k)\times k},\sigma}$ is the trapdoor composed of Gaussian polynomials.

**FRD map [ABB10]**

A function $H : \{0,1\}^n \to R_q$ is an encoding with *Full-Rank Differences* if:

- for all $id$, $H(id)$ is invertible,
- for all $id \neq id'$, $H(id) - H(id') \in R_q$ is invertible.

**PKG**

$$u \hookleftarrow U(R_q),\ \mathbf{a}' \hookleftarrow U(R_q^{m-k})$$
$$\mathbf{T} \hookleftarrow D_{R^{(m-k)\times k},\sigma},$$
$$\mathbf{a} = ({\mathbf{a}'}^T \mid -{\mathbf{a}'}^T \mathbf{T})^T$$
$$mpk = (\mathbf{a}, u) \text{ and } msk = \mathbf{T}$$

$\mathbf{x}_{\text{Bob}}$ such that $\mathbf{a}_{\text{Bob}}^T \mathbf{x}_{\text{Bob}} = u \mod q$

PKG → Bob: $\mathbf{x}_{\text{Bob}}$

**Alice**

$$s \hookleftarrow U(R_q),$$
$$\mathbf{e}_0 \hookleftarrow D_{R^{m-k},\tau},$$
$$\mathbf{e}_1 \hookleftarrow D_{R^k,\gamma}$$

$$M \in R_2,\ e' \hookleftarrow D_{R,\tau}$$

$$\mathbf{a}_{\text{Bob}} = \mathbf{a} + (0 \mid H(id_{\text{Bob}})\mathbf{g})^T$$
$$= \left({\mathbf{a}'}^T \;\middle|\; H(id_{\text{Bob}})\mathbf{g} - {\mathbf{a}'}^T \mathbf{T}\right)^T$$

Alice → Bob: $\mathbf{c}_0^T = \mathbf{a}_{\text{Bob}} s + (\mathbf{e}_0^T \mid \mathbf{e}_1^T)^T$

Alice → Bob: $c_1 = u \cdot s + e' + M.\lfloor q/2 \rfloor$

**Bob**

$$sk = \mathbf{x}_{\text{Bob}}$$
$$pk = (\mathbf{a}_{\text{Bob}}, u)$$

$$c_1 - \mathbf{c}_0^T \mathbf{x}_{\text{Bob}} = \underbrace{e' - (\mathbf{e}_0^T \mid \mathbf{e}_1^T)^T \mathbf{x}_{\text{Bob}}}_{small} + M.\lfloor q/2 \rfloor$$

# Implementation Choices

- Plain C++ implementation using the NFLlib library[6],
- Preimage sampling à la MP12, recently improved by Micciancio and Genise[7],
- By setting $m - k = 2$, and $\mathbf{a}' = (1, a)$ we get

$$\mathbf{a} = (1, a \mid h \cdot g_1 - (a \cdot t_{2,1} + t_{1,1}), \cdots, h \cdot g_k - (a \cdot t_{2,k} + t_{1,k}))$$

$\longrightarrow$ Hardness of Ring-LWE with Gaussian secret of parameter $\sigma$,

[6] Aguilar Melchor et al. (2016). "NFLlib: NTT-Based Fast Lattice Library". In: *CT-RSA*.
[7] Genise and Micciancio (2018). "Faster Gaussian Sampling for Trapdoor Lattices with Arbitrary Modulus". In: *EUROCRYT*.

# Parameter Choices

We need to ensure:

- the hardness of two Ring-LWE instances, of parameter $q$, $n$ and:
  - Gaussian parameter $\sigma$, corresponding to the public key,
  - Gaussian parameter $\tau$, corresponding to the encryption part,
- the correctness of the scheme:

$$\|e' - (\mathbf{e}_0^T \mid \mathbf{e}_1^T)^T \mathbf{x}\| < q/4,$$

$\longrightarrow$ Estimation of the hardness of these LWE instances using the LWE estimator of Albrecht et al.[8].

$\longrightarrow$ Example, for $\lambda = 80$, we get $\log_2 q = 51$, $n = 1024$, and $\sigma, \tau \approx 5$.

[8] Albrecht, Player, and Scott (2015). "On the concrete hardness of Learning with Errors". In: *J. Mathematical Cryptology*.

# Experimental Results (IBE)

| Scheme | $(\lambda, n)$ | Setup (ms) | Extract (ms) | Encrypt (KB/s) | Decrypt (KB/s) |
|---|---|---|---|---|---|
| BF-128[9] | $(128, -)$ | – | 0.55 | 4.10 | 6.19 |
| DLP-14[10] | $(80, 512)$ | 4034 | 3.8 | 587 | 1405 |
| This paper [11] | $(80, 1024)$ | 1.67 | 4.02 | 230 | 1042 |

[9] Fouotsa (2013). "Calcul des couplages et arithmétique des courbes elliptiques pour la cryptographie". PhD thesis.

[10] McCarthy, Smyth, and O'Sullivan (2017). "A Practical Implementation of Identity-Based Encryption Over NTRU Lattices". In: *IMACC*.

[11] Timings obtained on a Intel i7-5600 2.6 GHz CPU.

# Underlying Signature Scheme

# Underlying Signature

$\mathrm{KeyGen}(1^\lambda) \to (vk, sk)$

1. Choose random $\mathbf{a}' \hookleftarrow U(R_q^{m-k})$,
2. Sample $\mathbf{T} \hookleftarrow D_{R^{(m-k)\times k},\sigma}$,
3. Compute $\mathbf{a} = (\mathbf{a}'^T \mid -\mathbf{a}'^T\mathbf{T})^T$,
4. Output $mpk = \mathbf{a} \in R_q^m$ and $msk = \mathbf{T} \in R^{(m-k)\times k}$.

We can compute $\mathbf{a}_M$ as $\mathbf{a}_M = \mathbf{a}^T + (0 \mid H(M)\mathbf{g})^T = (\mathbf{a}'^T \mid H(M)\mathbf{g} - \mathbf{a}'^T\mathbf{T})^T$.

$\mathrm{Sign}(vk = \mathbf{a}, sk = \mathbf{T}, M) \to \nu$

1. Sample $\mathbf{x} \leftarrow \mathrm{Extract}((\mathbf{a}, 0), \mathbf{T}, M)$, satisfying $\mathbf{a}_M^T\mathbf{x} = 0 \in R_q$,
2. Output $\nu = \mathbf{x} \in R_q^m$.

$\mathrm{Verify}(vk = \mathbf{a}, \nu = \mathbf{x}, M) \to \{\textbf{accept}, \textbf{reject}\}$

1. Accept iff $\mathbf{a}_M^T\mathbf{x} = 0 \mod q$ and $\|\mathbf{x}\| \leq t\zeta\sqrt{mn}$.

# Experimental Results (Signature)

Timings obtained on a Intel i7-5600 2.6 GHz CPU.

| Scheme | $(\lambda, n)$ | KeyGen (ms) | Sign (op/s) | Verify (op/s) |
|---|---|---|---|---|
| Falcon[12] | (195, 768) | 53.48 | 202 | 2685 |
| This paper | (170, 1024) | 0.96 | 540 | 21276 |

$\longrightarrow$ run on the same computer but not fair comparison: not as pessimistic with the choice parameters, naive implementation of the function $H$...

[12] Fouque et al. (2018). *Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU.* . NIST Submission.

# Conclusion

# Conclusion

Get the source code of this work from

`https://github.com/lbibe/code`

# Conclusion

Get the source code of this work from

`https://github.com/lbibe/code`

Future works:

1. Both IBE/Signature schemes achieve selective security

$\longrightarrow$ adaptive secure variants

2. IND-CCA1 variant of the IBE scheme
3. Module variants

$\longrightarrow$ more versatile choice of parameters

# Conclusion

Get the source code of this work from

`https://github.com/lbibe/code`

Future works:

1. Both IBE/Signature schemes achieve selective security

$\longrightarrow$ adaptive secure variants

2. IND-CCA1 variant of the IBE scheme
3. Module variants

$\longrightarrow$ more versatile choice of parameters

Thank You!