Rank-metric based cryptography and its implementations

Nicolas Aragon

Univ Rennes, CNRS, Inria, IRISA - EMSEC team

November 10, 2021

Journée thématique :
Algorithmes de chiffrement post-quantiques et sécurité matérielle

Post-quantum cryptography

The recent developments of **quantum computers** threaten currently used public-key cryptography.

Public-key cryptography needs difficult mathematical problems:

- Factorization
- Discrete logarithm
- **Decoding random codes**
- Searching for short vectors in lattice
- Solving multivariate systems
- ...

Post-quantum cryptography

The recent developments of **quantum computers** threaten currently used public-key cryptography.

Public-key cryptography needs difficult mathematical problems:

- Factorization
- Discrete logarithm
- **Decoding random codes**
- Searching for short vectors in lattice
- Solving multivariate systems
- ...

Post-quantum cryptography

The recent developments of **quantum computers** threaten currently used public-key cryptography.

Public-key cryptography needs difficult mathematical problems:

- Factorization
- Discrete logarithm
- **Decoding random codes**
- Searching for short vectors in lattice
- Solving multivariate systems
- ...

NIST standardization process

In 2017, the National Institute for Standards and Technology (NIST) started a standardization process to choose post-quantum encryption and digital signature schemes.

Code-based proposals

There were 19 code-base proposals:

- 17 encryption schemes,
- 2 digital signature schemes.

Code-based cryptography was the second most represented "family", behind lattice-based cryptography.

NIST standardization process

In 2017, the National Institute for Standards and Technology (NIST) started a standardization process to choose post-quantum encryption and digital signature schemes.

Code-based proposals

There were 19 code-base proposals:

- 17 encryption schemes,
- 2 digital signature schemes.

Code-based cryptography was the second most represented "family", behind lattice-based cryptography.

Error correcting codes

Definition (Linear code)

An $[n, k]$ linear code C over $\mathbb{F}_q$ is a vector subspace of $\mathbb{F}_q^n$. Elements of C are called codewords.

Definition (Generator matrix)

A generator matrix $G \in \mathbb{F}_q^{k \times n}$ of an $[n, k]_{\mathbb{F}_q}$ code C is a matrix such that its lines form a basis of the vector space C.

Definition (Parity-check matrix)

A parity-check matrix $H \in \mathbb{F}_q^{(n-k) \times n}$ of an $[n, k]_{\mathbb{F}_q}$ code C is a matrix such that:

$$x \in C \Leftrightarrow H.x^t = 0$$

Error correcting codes

Definition (Linear code)

An $[n, k]$ linear code C over $\mathbb{F}_q$ is a vector subspace of $\mathbb{F}_q^n$. Elements of C are called codewords.

Definition (Generator matrix)

A generator matrix $G \in \mathbb{F}_q^{k \times n}$ of an $[n, k]_{\mathbb{F}_q}$ code C is a matrix such that its lines form a basis of the vector space C.

Definition (Parity-check matrix)

A parity-check matrix $H \in \mathbb{F}_q^{(n-k) \times n}$ of an $[n, k]_{\mathbb{F}_q}$ code C is a matrix such that:

$$x \in C \Leftrightarrow H.x^t = 0$$

Error correcting codes

Definition (Linear code)

An $[n, k]$ linear code C over $\mathbb{F}_q$ is a vector subspace of $\mathbb{F}_q^n$. Elements of C are called codewords.

Definition (Generator matrix)

A generator matrix $G \in \mathbb{F}_q^{k \times n}$ of an $[n, k]_{\mathbb{F}_q}$ code C is a matrix such that its lines form a basis of the vector space C.

Definition (Parity-check matrix)

A parity-check matrix $H \in \mathbb{F}_q^{(n-k) \times n}$ of an $[n, k]_{\mathbb{F}_q}$ code C is a matrix such that:

$$x \in C \Leftrightarrow H.x^t = 0$$

Hamming metric

In the Hamming metric, we generally consider codes with coefficients in $\mathbb{F}_2$.

Definition (Hamming metric)

Let $\boldsymbol{x}$ a vector $(x_1, \ldots, x_n) \in \mathbb{F}_q^n$.
The Hamming weight of $\boldsymbol{x}$ is the number of non null coordinates of $\boldsymbol{x}$.
The Hamming distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ is the number of non null coordinates of $\boldsymbol{x} - \boldsymbol{y}$.

Syndrome decoding

Definition (Syndrome decoding)

Given $\boldsymbol{H}$ a random full-rank $(n-k) \times n$ matrix over $\mathbb{F}_q$, and a syndrome $\boldsymbol{s} \in \mathbb{F}_q^{n-k}$, the problem is to find $\boldsymbol{e}$ such that $\boldsymbol{He}^\intercal = \boldsymbol{s}$ and $\boldsymbol{e}$ has weight t. This problem has been proved NP-complete[a].

[a] Elwyn Berlekamp, Robert McEliece, and Henk Van Tilborg. "On the inherent intractability of certain coding problems (corresp.)". In: *IEEE Transactions on Information Theory* 24.3 (1978), pp. 384–386.

The McEliece cryptosystem

McEliece described the first code-based cryptosystem in 78[1].

Alice		Bob
$\boldsymbol{G} \xleftarrow{\$}$ generator matrix of a structured code		
$\boldsymbol{G}' \leftarrow$ scrambled version of $\boldsymbol{G}$	$\xrightarrow{\boldsymbol{G}'}$	
		$\boldsymbol{e} \xleftarrow{\$}$ error of weight r
	$\xleftarrow{\boldsymbol{c}}$	$\boldsymbol{c} \leftarrow \boldsymbol{m}\boldsymbol{G}' + \boldsymbol{e}$
$\boldsymbol{m} \leftarrow \text{Decode}(\boldsymbol{G}, \boldsymbol{c})$		

[1] Robert J McEliece. "A public-key cryptosystem based on algebraic coding theory". In: *Coding Thv* 4244 (1978), pp. 114–116.

Rank metric

In the rank metric, we consider codes with coefficients in $\mathbb{F}_{q^m}$.
Let $(\beta_1, \ldots, \beta_m)$ a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. To a vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ We can associate a matrix $\boldsymbol{M_x}$

$$\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_{q^m}^n \leftrightarrow \boldsymbol{M_x} = \begin{pmatrix} x_{11} & \ldots & x_{n1} \\ \vdots & \ddots & \vdots \\ x_{1m} & \ldots & x_{nm} \end{pmatrix} \in \mathbb{F}_q^{m \times n}$$

Rank weight and distance

$|\boldsymbol{x}|_R = \text{Rank}(\boldsymbol{M_x})$ and $d(\boldsymbol{x}, \boldsymbol{y}) = \text{Rank}(\boldsymbol{M_x} - \boldsymbol{M_y})$

Rank metric

In the rank metric, we consider codes with coefficients in $\mathbb{F}_{q^m}$.
Let $(\beta_1, \ldots, \beta_m)$ a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. To a vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ We can associate a matrix $\boldsymbol{M_x}$

$$\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}_{q^m}^n \leftrightarrow \boldsymbol{M_x} = \begin{pmatrix} x_{11} & \ldots & x_{n1} \\ \vdots & \ddots & \vdots \\ x_{1m} & \ldots & x_{nm} \end{pmatrix} \in \mathbb{F}_q^{m \times n}$$

Rank weight and distance

$|\boldsymbol{x}|_R = \text{Rank}(\boldsymbol{M_x})$ and $d(\boldsymbol{x}, \boldsymbol{y}) = \text{Rank}(\boldsymbol{M_x} - \boldsymbol{M_y})$

Ideal codes

To reduce the size of the parameters, some rank-metric based schemes are based on ideal codes.

Ideal matrix

Let $\mathcal{R} = \mathbb{F}_{q^m}[X]/\langle P \rangle$ where P is a polynomial of degree n with coefficients in $\mathbb{F}_q$. Let $\boldsymbol{y} = (y_0, \ldots, y_{n-1}) \in \mathcal{R}$. We can associate the polynomial $\boldsymbol{y}$ with the following ideal matrix:

$$\mathcal{M}(\boldsymbol{y}) = \begin{pmatrix} \boldsymbol{y} \\ X\boldsymbol{y} \mod P \\ X^2\boldsymbol{y} \mod P \\ \vdots \\ X^{(n-1)}\boldsymbol{y} \mod P \end{pmatrix} \in \mathbb{F}_{q^m}^{n \times n}.$$

Encryption schemes

The GPT cryptosystem

McEliece with Gabidulin codes[a].

[a] Ernst M Gabidulin, AV Paramonov, and OV Tretjakov. "Ideals over a non-commutative ring and their application in cryptology". In: *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer. 1991, pp. 482–489.

NIST standardization process

- ROLLO: "McEliece" with LRPC codes.
- RQC: Better security reduction, larger parameters.

Digital signatures

Hash and sign

Most "natural" idea: invert a random syndrome using a structured code ⇒ Ranksign, with LRPC codes, but it was broken[a].

[a] Thomas Debris-Alazard and Jean-Pierre Tillich. "Two attacks on rank metric code-based schemes: RankSign and an IBE scheme". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2018, pp. 62–92.

Proof of knowledge

Two approaches:

- Authentication scheme and the Fiat-Shamir transform,
- Adaptation of the Lyubashevsky[a] approach ⇒ Durandal.

[a] Vadim Lyubashevsky. "Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures". In: *International Conference on the Theory and Application of Cryptology and Information Security*. Springer. 2009, pp. 598–616.

Mathematical objects

Elements of $\mathbb{F}_{q^m}$

- Addition
- Multiplication
- Inversion
- ...

Elements of $\mathbb{F}_{q^m}^n$

- Rank computation
- Addition, multiplication, inversion
- ...

More complex functions

- Operations on vector subspaces of $\mathbb{F}_{q^m}$
- Decoding LRPC and Gabidulin codes
- Sampling low weight vectors
- ...

Mathematical objects

Elements of $\mathbb{F}_{q^m}$

- Addition
- Multiplication
- Inversion
- ...

Elements of $\mathbb{F}_{q^m}^n$

- Rank computation
- Addition, multiplication, inversion
- ...

More complex functions

- Operations on vector subspaces of $\mathbb{F}_{q^m}$
- Decoding LRPC and Gabidulin codes
- Sampling low weight vectors
- ...

Mathematical objects

Elements of $\mathbb{F}_{q^m}$

- Addition
- Multiplication
- Inversion
- ...

Elements of $\mathbb{F}_{q^m}^n$

- Rank computation
- Addition, multiplication, inversion
- ...

More complex functions

- Operations on vector subspaces of $\mathbb{F}_{q^m}$
- Decoding LRPC and Gabidulin codes
- Sampling low weight vectors
- ...

Overview

General choices

The RBC library is written in C and focuses on performance without sacrificing usability.
It is released under the LGPL license and is available at:

`http://rbc-lib.org/`

Target users

- Users who want to use the cryptographic schemes implemented in the library,
- Users who want to implement new schemes.

Overview

General choices

The RBC library is written in C and focuses on performance without sacrificing usability.
It is released under the LGPL license and is available at:

`http://rbc-lib.org/`

Target users

- Users who want to use the cryptographic schemes implemented in the library,
- Users who want to implement new schemes.

Design choices

Implementation of $\mathbb{F}_{q^m}$

- Only $q = 2$.
- $\mathbb{F}_{q^m}$ is implemented as $\mathbb{F}_2[X]/\langle P \rangle$ where P is an irreducible polynomial of degree m over $\mathbb{F}_2$.
- Specific algorithms for each value of m.

Memory representation

An element of $\mathbb{F}_{q^m}$ is stored using $\lceil \frac{m}{64} \rceil$ integers, where each bit represents a coefficient of the polynomial.

Design choices

Implementation of $\mathbb{F}_{q^m}$

- Only $q = 2$.
- $\mathbb{F}_{q^m}$ is implemented as $\mathbb{F}_2[X]/\langle P \rangle$ where P is an irreducible polynomial of degree m over $\mathbb{F}_2$.
- Specific algorithms for each value of m.

Memory representation

An element of $\mathbb{F}_{q^m}$ is stored using $\lceil \frac{m}{64} \rceil$ integers, where each bit represents a coefficient of the polynomial.

Design choices

Preprocessing system

The library uses a preprocessing system that generates optimized code for each finite field:

- Improves performances,
- Avoids duplicate code,
- Adds complexity.

Build system

The library offers three possible build targets:

- c32, for x86 processors.
- c64, for x64 processors.
- avx, for x64 processors with CLMUL and AVX support.

Design choices

Preprocessing system

The library uses a preprocessing system that generates optimized code for each finite field:

- Improves performances,
- Avoids duplicate code,
- Adds complexity.

Build system

The library offers three possible build targets:

- `c32`, for x86 processors.
- `c64`, for x64 processors.
- `avx`, for x64 processors with CLMUL and AVX support.

RBC on microcontroller

Available implementations

- c32 code is compatible with ARM Cortex-M microcontrollers.
- ROLLO and RQC are available in the round 2 `mupq` project:

`https://github.com/mupq/mupq`

Limitations

- Generated files are split into multiple folders.
- OpenSSL used for symmetric cryptography.

RBC on microcontroller

Available implementations

- c32 code is compatible with ARM Cortex-M microcontrollers.
- ROLLO and RQC are available in the round 2 `mupq` project:

`https://github.com/mupq/mupq`

Limitations

- Generated files are split into multiple folders.
- OpenSSL used for symmetric cryptography.

Cortex-M4 performance

Scheme	Keygen	Encaps	Decaps
ROLLO-I	16 927 603	1 926 332	7 009 943
RQC	5 756 747	11 340 541	71 551 978
frodokem640shake	91 940 068	109 310 982	109 009 172
kyber512	653 616	883 740	981 642
sikep434	672 303 199	1 100 796 989	1 174 307 957

Figure 1: Performances of several KEM on ARM Cortex-M4 in cycles. These implementations are in plain C and target 128 bits security.

Implementation security

Constant time implementations

The goal of the library is to provide efficient and secure implementations, for as many platforms as possible.

Right now the library only focuses on constant time.

There is the need to develop side-channel attacks and countermeasures.[a]

[a] Agathe Cheriere, Lina Mortajine, Tania Richmond, and Nadia El Mrabet. "Side-Channel Attack on ROLLO Post-Quantum Cryptographic Scheme.". In: *IACR Cryptol. ePrint Arch.* 2021 (2021), p. 477.

Future work

Security

- Patch existing security issues,
- Further analyze side-channel leakages,
- Provide a better review of existing code.

Features

- Include additionnal cryptosystems,
- Explore algorithmic improvements,
- Include additionnal mathematical objects.

Contributions are welcomed !

https://rbc-lib.org

Questions ?