Code based cryptography

J.-P. Tillich

March 24, 2015

1. Introduction

Post-quantum cryptography: (public-key) cryptography resistant to a quantum computer. Neither based on the hardness of factoring nor on computing discrete logs.

Based on the hardness

- finding a short codeword in a lattice lattice-based cryptography
- decoding a linear code code-based cryptography
- solving an algebraic system multivariate cryptography
- hash based cryptography

Advantages/drawbacks of code-based cryptography

Advantages

- Post Quantum;
- Efficient encryption and decryption (compared to RSA, El Gamal): the original McEliece has encryption ≈ 5 times faster than RSA 1024, decryption ≈ 150 times faster than RSA 1024.

Drawbacks

- Huge size of the keys: the original proposal (McEliece 1978) has a 67ko key (more than 500 times RSA 1024 for a similar security).

The hard problem upon which code-based cryptography is based

Input:

- $\boldsymbol{G}$ a full rank $k \times n$ matrix over $\mathbb{F}_q$
- $\boldsymbol{y} \stackrel{\text{def}}{=} \boldsymbol{x}\boldsymbol{G} + \boldsymbol{e}$ where $\boldsymbol{e} \in \mathbb{F}_q^n$ is of Hamming weight $\leqslant t$

Output:

$\boldsymbol{x} \in \mathbb{F}_q^k$

Decoding problem

Code

$$\begin{aligned} \mathcal{C} &= \{\boldsymbol{x}\boldsymbol{G} : \boldsymbol{x} \in \mathbb{F}_q^k\} \\ &= \{\boldsymbol{y} : \boldsymbol{H}\boldsymbol{y}^T = 0\} \end{aligned}$$

$\boldsymbol{G}$ is a generator matrix for $\mathcal{C}$
$\boldsymbol{H}$ is a parity-check matrix for $\mathcal{C}$.
$\mathcal{C}$ is an $[n, k]_q$ code.

Problem : decoding t errors (or less) for $\mathcal{C}$.

The McEliece cryptosystem

- 1978 McEliece cryptosystem

- Secret Key : A generator matrix $\boldsymbol{G}$ of an $[n,k]_q$ code $\mathcal{C}$ having an efficient t–correcting algorithm;

- Public Key : $\boldsymbol{G}' := S\boldsymbol{G}P$, where $S \in \mathsf{GL}(k, \mathbb{F}_q)$ and P is an $n \times n$ permutation matrix;

- Encryption : $m \in \mathbb{F}_q^k \longmapsto y \stackrel{\text{def}}{=} m\boldsymbol{G}' + e$ with $|e| = t$.

- Decryption : $y \longmapsto yP^{-1} = mS\boldsymbol{G} + eP^{-1} \longmapsto mS \longmapsto m$.

Codes that have an efficient decoding algorithm

- codes with an algebraic structure
- Low Density Parity Check Codes
- Convolutional codes
- Polar codes

Reed-Solomon codes

These are codes defined over large alphabets $\mathbb{F}_q$. We choose n distinct elements $x_1, \ldots, x_n \in \mathbb{F}_q$.

Let e be the evaluation function:

$$\begin{array}{rccl} \boldsymbol{e}: & \mathbb{F}_q[X] & \to & \mathbb{F}_q^n \\ & f & \mapsto & \boldsymbol{e}(f) = (f(x_1), \ldots, f(x_n)) \end{array}$$

and

$$L = \{f \in \mathbb{F}_q[X] \mid \deg f < k\}.$$

The Reed-Solomon code of dimension k is given by

$$C \stackrel{\text{def}}{=} \boldsymbol{e}(L).$$

Reed-Solomon decoding by interpolation

Let $y = (y_1, \ldots, y_n)$ be the received word and c be the closest codeword with $c = \boldsymbol{e}(f(X))$ where $\deg f(X) < k$.

let I be the set of positions where there is an error:

$$I = \{i \in 1..n, \quad f(x_i) \neq y_i\},$$

and construct the polynomial $E(X) = \prod_{i \in I}(X - x_i)$. Then we have

$$E(x_i)y_i = E(x_i)f(x_i), \quad i \in 1..n. \tag{1}$$

Decoding (II)

let

$$X^t + \sum_{i=0}^{t-1} e_i X^i \stackrel{\text{def}}{=} E(X)$$

$$\sum_{i=0}^{t+k-1} a_i X^i \stackrel{\text{def}}{=} E(X) f(X)$$

- $2t + k$ unknowns and n affine equations :

$$E(x_i) y_i = E(x_i) f(x_i), \quad i \in 1..n. \tag{2}$$

- One can hope to correct in this way $\frac{n-k}{2} = \frac{d-1}{2}$ errors.

Generalised Reed-Solomon codes

Generalised Reed-Solomon code $\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y})$ associated to a pair $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{F}_q^n \times \mathbb{F}_q^n$ where $\boldsymbol{x}$ is an n-tuple of distinct elements of $\mathbb{F}_q$ and the entries y_i are arbitrary nonzero elements in $\mathbb{F}_q$.

$$\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\text{def}}{=} \Big\{ (y_1 p(x_1), \ldots, y_n p(x_n)) : p \in \mathbb{F}_q[X], \deg p < k \Big\}.$$

Alternant code associated to $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_{q^m}^n$ given by

$$\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y}) \cap \mathbb{F}_q^n = \Big\{ (y_i p(x_i))_i : \deg p < k, y_i p(x_i) \in \mathbb{F}_q \Big\}$$

Let $r \stackrel{\text{def}}{=} n - k$ then this is an $[n, \geqslant n - rm]_q$ code that can correct $r/2$ errors.

Goppa code special kind of alternant codes.

2. Finding S and P

$$\boldsymbol{G}' = S\boldsymbol{G}P$$

Code $\mathcal{C}$ with generator matrix $\boldsymbol{G}$ is equivalent to the code $\mathcal{C}'$ with generator matrix $\boldsymbol{G}'$: there exists a permutation π of the codeword positions such that

$$\mathcal{C}' = \pi\mathcal{C}$$

i.e.

$$\mathcal{C}' = \{(c_{\pi(i)})_i : (c_i)_i \in \mathcal{C}\}$$

[Dinh-Moore-Russell11]

- Study quantum Fourier sampling techniques.
- Show that the group $G = (\mathsf{GL}_k(\mathbb{F}_q) \times S_n) \wr \mathbb{Z}_2$ resists to Fourier sampling techniques.
- Applies to show that Fourier sampling techniques are inefficient for finding S and P in the McEliece system.
- Applies strictly speaking only when there is one secret code.
- This problem can be solved efficiently classically for most codes...

Sendrier splitting algorithm

Dual code

$$< \boldsymbol{x}, \boldsymbol{y} > \stackrel{\text{def}}{=} \sum_i x_i y_i$$

$$\mathcal{C}^{\perp} \stackrel{\text{def}}{=} \{\boldsymbol{x} :< \boldsymbol{x}, \boldsymbol{c} >= 0,\ \forall \boldsymbol{c} \in \mathcal{C}\}$$

Hull of a code $\mathcal{C}$ $\stackrel{\text{def}}{=}$ $\mathcal{C}^{\perp} \cap \mathcal{C}$ typically of constant dimension. If

$$\mathcal{D} = \pi(\mathcal{C})$$

then

$$\begin{aligned} \mathcal{D}^{\perp} &= \pi(\mathcal{C}^{\perp}) \\ \mathcal{D}^{\perp} \cap \mathcal{D} &= \pi(\mathcal{C}^{\perp} \cap \mathcal{C}) \end{aligned}$$

Sendrier splitting algorithm

$$\mathcal{C}_i = \{(c_j)_{j \neq i}, (c_i)_i \in \mathcal{C}\}$$

if

$$\mathcal{D} = \pi(\mathcal{C})$$

then there exists a permutation π' such that

$$\mathcal{D}_{\pi(i)} = \pi'(\mathcal{C}_i)$$

3. Decreasing the key size

Best known attack in 2^{128} operations

cryptosystem	public key size
RSA	3072
McEliece (binary Goppa)	1,448,000
Rainbow (multivariate crypto)	1,200,000
NTRU (lattice based)	6104

Idea

Choosing a structured generator matrix $\boldsymbol{G}$

$$\boldsymbol{H} = \left(\begin{array}{c|c|c} & & \\ \hline \cdots & \boldsymbol{M}_i & \cdots \\ \hline & & \end{array}\right),$$

where the $\boldsymbol{M}_i$'s have the form

$$\boldsymbol{M}_i = \begin{pmatrix} a_0 & a_1 & \cdots & a_{\ell-1} \\ a_{\ell-1} & a_0 & \cdots & a_{\ell-2} \\ \vdots & \ddots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_0 \end{pmatrix}$$

$\Rightarrow$ reduction by a factor of ℓ of the key size

Equivalent Version

Let C_ℓ be the ring of circulant $\ell \times \ell$ matrices over $\mathbb{F}_q$

$$\Phi : C_\ell \rightarrow \mathbb{F}_q[X]/(X^\ell - 1)$$

$$\begin{pmatrix} a_0 & a_1 & \cdots & a_{\ell-1} \\ a_{\ell-1} & a_0 & \cdots & a_{\ell-2} \\ \vdots & \ddots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_0 \end{pmatrix} \mapsto a_0 + a_1 X + \cdots + a_{\ell-1} X^{\ell-1}$$

Φ is a ring isomorphism.

Decoding quasi-cyclic codes

Input:

- $\boldsymbol{G}$ a quasi-cyclic full rank $k_0\ell \times n_0\ell$ matrix over $\mathbb{F}_q$ formed by circulant blocks of size ℓ
- $\boldsymbol{y} \stackrel{\text{def}}{=} \boldsymbol{x}\boldsymbol{G} + \boldsymbol{e}$ where $\boldsymbol{e} \in \mathbb{F}_q^{n_0\ell}$ is of Hamming weight $\leqslant t$

Output:

$\boldsymbol{x} \in \mathbb{F}_q^{k_0\ell}$

Decoding quasi-cyclic codes

Input:

- $\boldsymbol{G}$ a full rank $k_0 \times n_0$ matrix over $\mathbb{F}_q[X]/(X^\ell - 1)$
- $\boldsymbol{y} \stackrel{\text{def}}{=} \boldsymbol{x}\boldsymbol{G} + \boldsymbol{e}$ where $\boldsymbol{e} \in \left(\mathbb{F}_q[X]/(X^\ell - 1)\right)^{n_0}$ is of weight $|\boldsymbol{e}| \leqslant t$

$$|\boldsymbol{e}| = \sum_{i=1}^{n_0} |\boldsymbol{e}_i|_H$$

Output:

$\boldsymbol{x} \in \left(\mathbb{F}_q[X]/(X^\ell - 1)\right)^{k_0}$

Quasi-cyclic Goppa codes

- Dramatic reduction in the key size [BergerCayrelGaboritOtmani09], [MisoczkiBarreto09] ≈ 10000 bits

Quasi-cyclic Goppa codes

- Dramatic reduction in the key size [BergerCayrelGaboritOtmani09], [MisoczkiBarreto09] ≈ 10000 bits
- Dramatic improvement in the attacks... [FaugèreOtmaniPerretTillich10], [GauthierLeander10], [FaugèreOtmaniPerretPortzamparcTillich14]

Idea

$\boldsymbol{G} = \left(C_0 \dots C_{n_0\ell-1}\right)$ is a $k_0\ell \times n_0\ell$ generator matrix in block circulant form of an alternant code $\mathcal{C}$,

$$\begin{aligned} C'_i &= \sum_{j=0}^{\ell-1} C_{i'\ell+j} \\ \boldsymbol{G}' &= \left(C'_0 \dots C_{n_0-1}\right) \end{aligned}$$

then $\boldsymbol{G}'$ generates an alternant code $\mathcal{C}_0$ of dimension $\leqslant k_0$.

$$\begin{array}{rcl} \text{alternant description of } \mathcal{C}_0 & : & \mathcal{C}_0 = \mathbf{GRS}_{r_0}(\boldsymbol{x}_0, \boldsymbol{y}_0) \cap \mathbb{F}_q^{n_0} \\ & \Downarrow & \\ \text{alternant description of } \mathcal{C} & : & \mathcal{C} = \mathbf{GRS}_{r}(\boldsymbol{x}, \boldsymbol{y}) \cap \mathbb{F}_q^{n_0\ell} \end{array}$$

4. Algebraic attack on McEliece based on Goppa codes [FaugèreOtmaniPerretTillich10]

$$\begin{aligned} \mathcal{C} &= \mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y}) \cap \mathbb{F}_q^n \\ (\boldsymbol{x}, \boldsymbol{y}) &\in \mathbb{F}_{q^m}^n \end{aligned}$$

There exists $\boldsymbol{y}' \in \mathbb{F}_{q^m}^n$ depending on $(\boldsymbol{x}, \boldsymbol{y})$ such that

$$\boldsymbol{H} = \begin{pmatrix} y'_1 & y'_2 & \cdots & y'_n \\ y'_1 x_1 & y'_2 x_2 & \cdots & y'_n x_n \\ \vdots & \vdots & \ddots & \vdots \\ y'_1 x_1^{r-1} & y'_2 x_2^{r-1} & \cdots & y'_n x_n^{r-1} \end{pmatrix}$$

is a parity-check matrix for $\mathcal{C}$ where $r \stackrel{\text{def}}{=} n - k$

Solve the system

Problem 1. *Let* $\boldsymbol{G} = (g_{ij})$ *be a generator matrix of* $\mathcal{C}$, $(\boldsymbol{x}, \boldsymbol{y}')$ *is given by a/the solution* X, Y *to the algebraic system*

$$\sum_{j=1}^{n} g_{ij} Y_j X_j^s = 0$$

for $i \in \{1, \ldots, k\}$, $s \in \{0, \ldots, r-1\}$.

5. Other families of codes

An $[n, k]$ code with a parity-check matrix with rows of weight $\leqslant w$ can decode $t \approx n/w$ errors.

MDPC code $\mathcal{C}$: code with a parity-check matrix with rows of weight $w = O(n^{1/2})$: corrects $t = \theta(n^{1/2})$ errors.

secret key : parity-check matrix of $\mathcal{C}$ with rows of weight $w = O(n^{1/2})$

public key : any random basis of $\mathcal{C} \rightarrow$ random generator matrix of $\mathcal{C}$.

quasi-cyclic MDPC codes

public key size for 128 bits of security

cryptosystem	public key size
RSA	3072
McEliece (binary Goppa)	1,448,000
Rainbow (multivariate crypto)	1,200,000
NTRU (lattice based)	6104
McEliece (QMDPC)	9857

The best decoding algorithms have all the same complexity as the very simple Prange algorithm in the binary case

...when $t = o(n)$

Prange algorithm : take k linear equations solve them and hope that they were not noisy.

The exponent of the best decoding algorithms is the same as the simple Prange algorithm when $t = o(n)$ in the binary case

Problem 2. *Input:*

- $\boldsymbol{G}$ *a full rank* $k \times n$ *matrix over* $\mathbb{F}_2$
- $\boldsymbol{y} \stackrel{def}{=} \boldsymbol{x}\boldsymbol{G} + \boldsymbol{e}$ *where* $\boldsymbol{e} \in \mathbb{F}_2^n$ *is of Hamming weight* $\leqslant t$

Output:
$\boldsymbol{x} \in \mathbb{F}_2^k$

If $t = o(n)$ and the ratio $R = \frac{k}{n}$ is fixed, then the best decoding algorithms have complexity $O\left(e^{-t\ln(1-R)(1+o(1))}\right)$...

QMDPC-McEliece=NTRU-like system

Problem 3. *Let* $h_1(X), h_2(X), s(X)$ *in* $\mathbb{F}_2[X]/(X^\ell - 1)$. *Find* $f(X) \in \mathbb{F}_2[X]/(X^\ell - 1)$ *of Hamming weight* $|f(X)| \stackrel{def}{=} \sum_i f_i = t$ *which is such that*

$$h_1(X) + f(X)h_2(X) = s(X) \mod (X^\ell - 1)$$

For $t = o(\ell)$ all known algorithms have complexity $O\left(2^{t(1+o(1))}\right)$.