AFF3CT et cryptographie post-quantique

Andrea Lesavourey

27 novembre 2023

Inria

Outline

Cryptographie à clef publique

Cryptographie à clef publique

Bob génère un *clef publique* **pk** et une *clef secrète* **sk**.

pk permet à tout le monde de chiffrer ;
Bob est le seul qui peut déchiffrer en utilisant **sk**.

Inria

Cryptographie à clef publique

Bob génère un *clef publique* **pk** et une *clef secrète* **sk**.

pk permet à tout le monde de chiffrer ;
Bob est le seul qui peut déchiffrer en utilisant **sk**.

Sécurité fondée sur un problème mathématique *difficile* à résoudre.

Factorisation (RSA) ou Logarithme discret (courbes elliptiques).

Inria

Cryptographie post-quantique

Problème : Algorithmes de Shor
Attaques en temps **polynomial quantique**.

Besoin d'un cryptographie **post-quantique** :
calculs classiques ;
résistance aux attaques quantiques.

Réseaux euclidiens, **Codes correcteurs**,
Systèmes polynomiaux, Fonctions de hachages
Variétés algébriques (courbes elliptiques).

Inria

Appels à standardisation

Appel du NIST en 2016.

Fin (presque) du processus.

Schémas de chiffrement :
Réseaux : KYBER.

Signatures :
Réseaux : DILITHIUM, FALCON.
Fonctions de hachage : SPHINCS+.

Un round de plus :
Codes : BIKE, CLASSIC MCELIECE, HQC

Inria

Appels à standardisation

Appel du NIST en 2016.

Fin (presque) du processus.

Schémas de chiffrement :
Réseaux : KYBER.

Signatures :
Réseaux : DILITHIUM, FALCON.
Fonctions de hachage : SPHINCS+.

Un round de plus :
Codes : BIKE, CLASSIC MCELIECE, HQC

Inria

Notre objectif.

Incorporer les codes correcteurs utilisés en cryptographie post-quantique dans AFF3CT.

Evaluer l'intérêt d'un cadre comme AFF3CT pour la cryptographie post-quantique.

Notre objectif.

Incorporer les codes correcteurs utilisés en cryptographie post-quantique dans AFF3CT.

Evaluer l'intérêt d'un cadre comme AFF3CT pour la cryptographie post-quantique.

Mon travail :

1. Recherche bibliographique sur les codes utilisés en cryptographie.
2. Définition et évaluation des objectifs en terme de performance, ou de fonctionnalités.
3. Incorporation dans AFF3CT.

Inria

Notre objectif.

Incorporer les codes correcteurs utilisés en cryptographie post-quantique dans AFF3CT.

Evaluer l'intérêt d'un cadre comme AFF3CT pour la cryptographie post-quantique.

Mon travail :

1. Recherche bibliographique sur les codes utilisés en cryptographie.
2. Définition et évaluation des objectifs en terme de performance, ou de fonctionnalités.
3. Incorporation dans AFF3CT.

Inria

Outline

1. Cryptographie post-quantique

2. Codes correcteurs d'erreurs.

3. Codes en cryptographie post-quantique.

Codes correcteurs d'erreurs

Definitions

Un *code correcteur d'erreurs linéaire* $\mathcal{C}$ est un sous-espace vectoriel d'un certain $\mathbb{F}_q^n$ pour un certain q. Le paramètre n est appelé la *longueur* de $\mathcal{C}$ et sa *dimension* est $\dim_{\mathbb{F}_q} \mathcal{C}$.

Definition (Distance minimale)

Notons d une distance sur l'espace $\mathbb{F}_q^n$, typiquement le poids de Hamming. La *distance minimale* d'un code $\mathcal{C}$ sur $\mathbb{F}_q$ est

$$\lambda_1(\mathcal{C}) \stackrel{\text{def}}{=} \min_{c \in \mathcal{C} \setminus \{0\}} d(0, c) = \min_{x \neq y} d(x, y).$$

Inria

Représentation d'un code

Un code peut-être représenté par :

- ⋆ une matrice génératrice G : $\forall c \in \mathcal{C}, \exists \lambda \in \mathbb{F}_q^k \mid c = \lambda \cdot G$;
- ⋆ une matrice de parité H : $c \in \mathcal{C} \iff H \cdot c^\top = 0$.

Grandes familles de codes – Reed-Solomon

Definition

Soit $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q^n \mid \forall i \neq j, \alpha_i \neq \alpha_j$.

Le *code de Reed-Solomon* sur $\mathbb{F}_q$ associé à α est

$$\mathrm{RS}_k(\alpha) \stackrel{\text{def}}{=} \left\{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f(X) \in \mathbb{F}_q[X], \deg f(x) < k\right\}.$$

Grandes familles de codes – Reed-Solomon

Definition

Soit $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q^n \mid \forall i \neq j, \alpha_i \neq \alpha_j$.

Le *code de Reed-Solomon* sur $\mathbb{F}_q$ associé à α est

$$\mathrm{RS}_k(\alpha) \stackrel{\text{def}}{=} \left\{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f(X) \in \mathbb{F}_q[X], \deg f(x) < k\right\}.$$

Décodage efficace et optimal.

Codes de Reed-Solomon généralisés and alternants

Definition

Soient $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q^n, \alpha_i \neq \alpha_j$ et $\beta = (\beta_1, \ldots, \beta_n) \in (\mathbb{F}_q^\times)^n$.

Le *code de Reed-Solomon généralisé* sur $\mathbb{F}_q$ associé à α, β est

$$\mathrm{GRS}_k(\alpha, \beta) \stackrel{\text{def}}{=} \{(\beta_1 f(\alpha_1), \ldots, \beta_n f(\alpha_n)) \mid f(X) \in \mathbb{F}_q[X], \deg f(x) < k\}.$$

On peut ensuite regarder le *code alternant*

$$\mathcal{A}_r(\alpha, \beta) \stackrel{\text{def}}{=} \mathrm{GRS}_k(\alpha, \beta)^\perp \bigcap \mathbb{F}_{q'}^n,$$

où $\mathbb{F}_{q'}$ est un sous-corps de $\mathbb{F}_q$.

Codes de Reed-Solomon généralisés and alternants

Definition

Soient $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q^n, \alpha_i \neq \alpha_j$ et $\beta = (\beta_1, \ldots, \beta_n) \in (\mathbb{F}_q^\times)^n$.

Le *code de Reed-Solomon généralisé* sur $\mathbb{F}_q$ associé à α, β est

$$\mathrm{GRS}_k(\alpha, \beta) \stackrel{\text{def}}{=} \left\{(\beta_1 f(\alpha_1), \ldots, \beta_n f(\alpha_n)) \mid f(X) \in \mathbb{F}_q[X], \deg f(x) < k\right\}.$$

On peut ensuite regarder le *code alternant*

$$\mathcal{A}_r(\alpha, \beta) \stackrel{\text{def}}{=} \mathrm{GRS}_k(\alpha, \beta)^\perp \bigcap \mathbb{F}_{q'}^n,$$

où $\mathbb{F}_{q'}$ est un sous-corps de $\mathbb{F}_q$.

Héritent des bonnes propriétés de décodage des codes de Reed-Solomon.

Inria

Codes cycliques et BCH

Definition

Considérons le cycle $\sigma : (c_0, \ldots, c_{n-1}) \mapsto (c_{n-1}, c_0, \ldots, c_{n-2})$.

Un code $\mathcal{C}$ est dit *cyclique* si $\sigma \cdot \mathcal{C} = \mathcal{C}$.

Exemple.

Un code de Reed-Solomon porté par un vecteur de la forme $(1, \alpha, \ldots, \alpha^{n-1})$ est cyclique.

Codes cycliques et BCH

Definition

Considérons le cycle $\sigma : (c_0, \ldots, c_{n-1}) \mapsto (c_{n-1}, c_0, \ldots, c_{n-2})$.

Un code $\mathcal{C}$ est dit *cyclique* si $\sigma \cdot \mathcal{C} = \mathcal{C}$.

Exemple.

Un code de Reed-Solomon porté par un vecteur de la forme $(1, \alpha, \ldots, \alpha^{n-1})$ est cyclique.

Diviseurs $g(X)$ de $X^n - 1$

Code cycliques | **Idéaux de** $\mathbb{F}_q[X]/(X^n - 1)$ | $c = g\lambda$

$c = (c_0, \ldots, c_{n-1})$ | $c(X) = \sum_{i=0}^{n-1} c_i X^i$ | Classes cyclotomiques I de $\mathbb{Z}/n\mathbb{Z}$

Action de σ | Mult. par X | $g(X) = \prod_{i \in I} X - \zeta_n^i$

Inria

Code cyclique avec générateur g tel que $\zeta_n^c, \ldots, \zeta_n^{c+s-1}$ sont des racines de g.

Algorithme de décodage efficace.

Codes Low Rank Parity Check (LRPC)

Fixons une base $(b_1, \ldots, b_n)$ de $\mathbb{F}_{q^m}/\mathbb{F}_q$.

Pour $x \in \mathbb{F}_{q^m}^n$, $x_i = \sum_{j=1}^{m} x_{i,j} b_j$. On définit $M(x) \stackrel{\text{def}}{=} [x_{i,j}]_{1 \leqslant i,j \leqslant n}$.

On définit alors la métrique rang : $w_r(x) \stackrel{\text{def}}{=} \operatorname{rang}(M(x))$.

Definition

Soit $H = (h_{i,j})_{1 \leqslant i,j, \leqslant n} \in M_{n-k,n}(\mathbb{F}_q)$ une matrice de rang plein telle que $\langle h_{i,j} \rangle_{\mathbb{F}_q}$ soit de petite dimension d. Alors le code $\mathcal{C}$ qui a H comme matrice de parité est un *code LRPC* de poids dual d

Problèmes difficiles sur les codes

On se donne toujours un code $\mathcal{C} \subseteq \mathbb{F}_q^n$.

Problèmes calculatoires :

Bounded Decoding Problem (BDD) :
Etant donnés $r > 0$ et $y \in \mathbb{F}_q^n$, calculer $c \in \mathcal{C}$ tel que $d(c, y) < r$.

Décodage par syndrome :
Etant donnés $r > 0$ et $s \in \mathbb{F}_q^n$ un syndrome, calculer $x \in \mathcal{C}$ tel que $H \cdot x^\top = s$ et $w_H(x) < r$.

Problèmes décisionnels :

Distinguer un syndrome :
Etant donnés $r > 0$ et (H, s), décider si $s = H \cdot x^\top$ avec $w_H(x) < r$ ou s est uniforme.

Inria

Outline

1. Cryptographie post-quantique

2. Codes correcteurs d'erreurs.

3. Codes en cryptographie post-quantique.

Classic McEliece

Utilise des codes de *Goppa* binaires :

On fixe $\alpha \in \mathbb{F}_{q^m}^n$ et $g(X) \in \mathbb{F}_{q^m}[X]$ tel que $\forall i \in [\![1, n]\!], g(\alpha_i) \neq 0$.

$$c \in \mathbb{F}_q^n \in \mathcal{C}(g, \alpha) \iff \sum_{i=1}^{n} \frac{c_i}{X - \alpha_i} \equiv 0 \mod g(X).$$

Code alternant donné par la matrice de parité

$$\begin{bmatrix} 1/g(\alpha_1) & \dots & 1/g(\alpha_n) \\ \alpha_1/g(\alpha_1) & \dots & \alpha_n/g(\alpha_n) \\ \vdots & & \vdots \\ \alpha_1^{d-1}/g(\alpha_1) & \dots & \alpha_n^{d-1}/g(\alpha_n) \end{bmatrix}$$

En écrivant tout sur $\mathbb{F}_2$ et en regardant la forme systématique, on obtient :

$$\left[\begin{array}{ccc|c} 1 & & & \\ & \ddots & & T \\ & & 1 & \end{array}\right]$$

Classic McEliece

Clef privée : (g, α).

Clef publique : T.

Chiffrement : $m \in \mathbb{F}_2^n \mapsto [\mathrm{Id} \mid T] \cdot m^\top$.

Déchifremment : $c \mapsto \texttt{decoder}(\mathcal{C}, c)$.

Inria

Classic McEliece

Clef privée : (g, α).

Clef publique : T.

Chiffrement : $m \in \mathbb{F}_2^n \mapsto [\mathrm{Id} \mid T] \cdot m^\top$.

Déchifremment : $c \mapsto \texttt{decoder}(\mathcal{C}, c)$.

Le décodage se ramène à celui d'un code de Reed-Solomon.

Hamming Quasi-Cyclic (HQC)

Sécurité fondée sur des codes quasi-cycliques : $\approx$ blocs de codes cycliques.

Utilise des codes de Reed-Solomon et Reed-Muller dans les calculs.

BIKE : Bit Flipping Key Encapsulation

Codes Quasi-Cyclic Moderate Density Parity Check (QS-MDPR) : Chiffrement de McEliece avec des codes quasi-cycliques.

Clef privée : $(h_0, h_1) \in H_w$.

Clef publique : $h = h_1/h_0$.

Chiffrement : $(m_0, m_1) \mapsto m_0 + m_1 h$.

Déchifremment : $c \mapsto \texttt{decoder}(ch_0, h_0, h_1)$.

L'algorithme de décodage `decoder` est ici le Black-Gray-Flip (BGF).

BIKE : Bit Flipping Key Encapsulation

Codes Quasi-Cyclic Moderate Density Parity Check (QS-MDPR) : Chiffrement de McEliece avec des codes quasi-cycliques.

Clef privée : $(h_0, h_1) \in H_w$.

Clef publique : $h = h_1/h_0$.

Chiffrement : $(m_0, m_1) \mapsto m_0 + m_1 h$.

Déchifremment : $c \mapsto \texttt{decoder}(ch_0, h_0, h_1)$.

L'algorithme de décodage `decoder` est ici le Black-Gray-Flip (BGF).

La sécurité est fondée sur :

Clef : distinguer h d'un élément aléatoire.

Message : distinguer $(m_0 + m_1 h, h)$ d'un couple aléatoire.

Inria