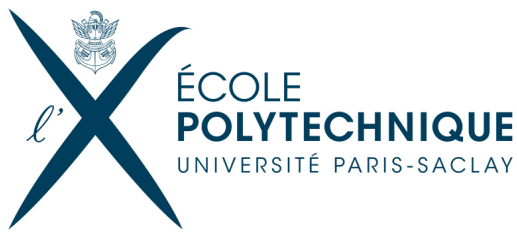
Attaque de Sidelnikov-Shestakov appliquée au cryptosystème de Chor-Rivest

INF 581 - Enseignement d'Approfondissement
D. Augot

Sylvain Colin & Gaspard Férey

Département d'Informatique
Ecole Polytechnique, France

27 Mars 2014



Cryptosystème de McEliece utilisant les codes de Reed-Solomon

Clef privée

- Une matrice $G = G(\alpha_1, \ \dots \ , \alpha_n, z_1, \ \dots \ z_n) = (z_i \alpha_i^j)_{i=1\dots n, j=0\dots k-1}$
- Une matrice inversible H de taille $k \times k$ dans $\mathbb{F}_q$.

Clef publique

- La représentation de $\mathbb{F}_q$.
- La matrice $M = H \cdot G$.
- L'entier $t = \lfloor \frac{n-k}{2} \rfloor$.

Messages originaux : vecteurs de $b \in \mathbb{F}_q^k$.
Message chiffré : $b \cdot M + e$ avec e de poids de Hamming inférieur à t et $b \cdot M = (z_i f_b(\alpha_i))_{1 \leq i \leq n}$ (f_b de degré au plus $k-1$).
Déchiffrement :

- On calcule $b \cdot H$ par un algorithme de déchiffrement de code GRS.
- On calcule b par multiplication par H^{-1}

Attaque de Sidelnikov et Shestakov

- Basée sur l'équivalence entre codes GRS :
  $\exists H', (\alpha'_3, \ \dots \ \alpha'_k, \alpha'_{k+2}, \ \dots \ , \alpha'_n)$ et $(z'_1, \ \dots \ z'_k, z'_{k+2}, \ \dots \ z'_n)$ tels que

$$H \cdot G = H' \cdot G(0, 1, \alpha'_3, \ \dots \ , \infty, \alpha'_{k+2}, \ \dots \ , \alpha'_n, z'_1, \ \dots \ , 1, z'_{k+2}, \ \dots \ z'_n)$$

- On calcule la forme échelon de M :

$$E(M) = \left(\begin{array}{c|c} I_k & (b_{i,j})_{1 \leq i \leq k, k+1 \leq j \leq n} \end{array} \right)$$

- On remarque que :

$$f_{b_i}(X) = c_{b_i} \cdot \prod_{1 \leq j \leq k, i \neq j} (X - \alpha_j)$$

avec $c_{b_i} = b_{i,k+1}$.

Attaque de Sidelnikov et Shestakov

Calcul des α_i :

- $\forall k+2 \leq j \leq n, \alpha_j = \frac{b_{2,j} \cdot c_{b_1}}{b_{2,j} \cdot c_{b_1} - b_{1,j} \cdot c_{b_2}}$
- $\forall 3 \leq i \leq k, \alpha_i = \alpha_{k+2} - \frac{b_{i,k+2}}{b_{1,k+2}} \cdot \frac{c_{b_1}}{c_{b_2}} \cdot (\alpha_{k+2} - 1)$
- On calcule un ensemble de α'_i équivalent et tous finis en trouvant un élément α différent de tous les α_i et en appliquant la transformation birationnelle $\phi : x \mapsto \frac{1}{x-\alpha}$

Calcul des z_i :

- On note $L_i(X) = \prod_{1 \leq j \leq k, i \neq j} (X - \alpha_j) = \frac{1}{c_{b_i}} \cdot f_{b_i}(X)$
- $\forall 1 \leq i \leq k, z_i = \frac{L_i(\alpha_{k+1})}{b_{i,k+1} \cdot L_i(\alpha_i)}$
- $\forall k+2 \leq j \leq n, z_j = \frac{b_{1,j}}{b_{1,k+1}} \cdot \frac{L_1(\alpha_{k+1})}{L_1(\alpha_j)}$

Calcul de H : $H = M_k \cdot G_k^{-1}$

Le cryptosystème de Chor-Rivest

Clef privée:

- $t \in \mathbb{F}_q$ dont le polynôme minimal est de degré h.
- g générateur $\mathbb{F}_q^*$.
- $0 \leq d < q$.
- π permutation de $\{0, ..., p-1\}$.

Clef publique:

$$c_i := d + \log_g(t + \alpha_{\pi(i)}) \mod q - 1$$

Message $m = [m_0 ... m_{p-1}]$ avec $\sum_i m_i = h$. Message chiffré:

$$E(M) := \sum_{i=0}^{p-1} m_i c_i \mod q - 1$$

On déchiffre en calculant

$$g^{E(M)-hd} = \prod_i \left(t + \alpha_{\pi(i)}\right)^{m_i}$$

Lien avec Reed-Solomon

Theorem

Pour $2 \leq k \leq p-2$, *supposons qu'il existe* k *polynômes* $(Q_i)_{1 \leq i \leq k}$ *de* $\mathbb{F}_p[X]$ *de degré inférieur à* $k-1$ *linéairement indépendants. Supposons connues les évaluations de ces polynômes en les* $\alpha_{\pi(j)}$, $m_{i,j} := Q_i(\alpha_{\pi(j)})$. *Alors la permutation* π *peut être retrouvée en temps polynomial en utilisant une attaque de Sidelnikov-Shestakov sur la matrice* $M = (m_{i,j})_{i,j} \in \mathcal{M}_{k,p}(\mathbb{F}_p)$.

Attaque de Vaudenay

Theorem

Quelque soit r divisant h, il existe un générateur g_{p^r} du groupe multiplicatif $\mathbb{F}_{p^r}^*$ (où F_{p^r} sous-corps de $\mathbb{F}_q$) et $Q \in \mathbb{F}_{p^r}[X]$ de degré h/r admettant $-t$ pour racine et tel que pour tout j, $Q(\alpha_{\pi(j)}) = g_{p^r}^{c_j}$.

Proof.

On a $g_{p^r} = g^{\frac{q-1}{p^r-1}}$ et

$$Q(X) = g_{p^r}^d \prod_{i=0}^{h/r-1} \left(X + t^{p^{ri}}\right)$$

Attaque de Vaudenay

Theorem

Si $r > \sqrt{h}$, et g_{p^r} connu, il existe une attaque du cryptosystème de Chor-Rivest en temps polynomial.

Proof.

Les r coordonnées de $g_{p^r}^{c_j}$ sont des polynômes de degré $h/r > r$ en les $\alpha_{\pi(j)}$. On utilise une attaque de Sidelnikov-Shestakov sur la matrice de ces coordonnées. □

Utilisation des puissances de g_{p^r}

Soit r diviseur de h et $(e_i)_{1\leq i\leq r}$ une base de $\mathbb{F}_{p^r}$. On note

- $U_w := \{u \in [0, p^r - 1] | w_p(u) \leq w\}$
- $h[i]$ la $i$ème coordonnée de $h \in \mathbb{F}_{p^r}$ dans la base (e_i).
- On définit $M^{(w)} \in \mathcal{M}_{r\cdot|U_w|,p}$

$$M^{(w)} := \left(g_{p^r}^{uc_j}[i]\right)_{(i,u)\in[1,r]\times U_w, 1\leq j\leq p}$$

- $u_w := \text{rank}\left(M^{(w)}\right)$

On a

$$u_w \leq r \cdot |U_w| = O\left(\frac{w^{r+1}}{r!}\right)$$

Theorem

Si $u_w = wh/r + 1 \leq p - 2$, Sidelnikov Shestakov fournit une attaque en temps polynomial.

Postulat

Postulate

Pour tout $r > 2$,

$$u_w = \min\left(\binom{w+r}{r}, w\frac{h}{r} + 1, p\right).$$

Vérifié sur

r	w	h/r
2	[1,17]	{1,2}
3	[1,17]	[1,30]
4	[1,17]	[1,30]
5	[1,17]	[1,30]

Condition sur r

On suppose

- $u_w = \min\left(\binom{w+r}{r}, w\frac{h}{r} + 1, p\right)$ (pour tout w)
- $h \sim p/\log p$
- h a de petits diviseurs
- Il existe w tel que

$$\begin{aligned} wh/r + 1 &\leq p - 2 \\ wh/r + 1 &\leq u_w \end{aligned}$$

On obtient

$$r \sim \frac{\log p}{\log\log p}$$

Algorithme

Input : Description de $\mathbb{F}_{p^h}$ et la clef publique: $(c_j)_{1 \leq j \leq p}$

- Calculer le plus petit diviseur r de h qui permette une attaque.
- Calculer l'ensemble U_w.
- Choisir une base $(e_i)_{1 \leq i \leq r}$ de $\mathbb{F}_{p^r}$ et générer la matrice projetant les éléments de $\mathbb{F}_q$ dans cette base.
- **Pour** tout générateur g_{p^r} possible de $\mathbb{F}_{p^r}$ **faire**
  - Générer M à partir de $wh/r + 1$ lignes indépendantes à partir des lignes de $M^{(w)}$
  - **Si** on peut trouver une ligne de $M^{(w)}$ indépendantes de celles de M
    Alors Passer au générateur suivant.
    Sinon Sortir de la boucle, retenir M et g_{p^r}.
- Effectuer une attaque de Sidelnikov Shestakov attack sur M pour générer toute les permutations possibles (π_i).
- **Pour chaque** permutation π **faire**
  - Déchiffrer en utilisant une attaque connaissant g_{p^r} et π.

Complexité en temps

$$(\text{Recherche exhaustive}) \times (\text{"Early abort"}) = O\left(p^{\frac{\log p}{\log\log p}+C}\right)$$

Conclusion

Notre algorithme

- a une bien meilleure complexité que celle de Vaudenay
- marche dès que $\Omega(p)$ coefficients $\alpha_{\pi(j)}$ sont connus
  - mais $O(p)$ permutations possibles générées

Questions ?