

# 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

16 Décembre 2013



# Cryptosystème de McEliece utilisant les codes de Reed-Solomon

Du texte

- un point

# Attaque de Sidelnikov et Shestakov

# Le cryptosystème de Chor-Rivest

Clef privée:

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

Clef publique:

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

Message  $m = [m_0 \dots m_{p-1}]$  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

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

## Theorem

*Pour  $2 \leq k \leq p - 2$ , supposons qu'il existe  $k$  polynômes  $(Q_i)_{0 \leq i \leq k-1}$  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  $\pi$  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  $\mathbb{F}_{p^r}$  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

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



# 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  $g_{p^r}^j$  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

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

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

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

On a

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



# 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

$$wh/r + 1 \leq p - 2$$

$$wh/r + 1 \leq u_w$$

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  $(\pi_i)$ .
- **Pour chaque** permutation  $\pi$  **faire**
  - ▶ Déchiffrer en utilisant une attaque "à  $g_{p^r}$  et  $\pi$  connus".

# 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