Algorithmes arithmétiques pour la cryptologie — 26 octobre 2015, Paris

Avancées récentes sur le problème du logarithme discret

Razvan Barbulescu
CNRS et IMJ-PRG

Plan du cours

- Tailles des clés
- Sécurité des couplages

Niveau de sécurité

Définition

On dit qu'un cryptosystème offre la sécurité s si la meilleure attaque connue requière 2^s opérations élémentaires.

Utilisation

1. On peut comparer la vitesse des différents cryptosystèmes en les réglant à la même sécurité.
2. Si on utilise ensemble de la cryptographie symétrique et asymétrique on peut les régler au même niveau de sécurité.

La loi de Moore

À cause de l'évolution des ordinateurs (loi de Moore), le même niveau de sécurité est considérer suffisant à un moment donné mais trop faible quelques années plus tard. En 2015, les principaux niveaux de sécurité sont:

- 80 bits
- 128 bits
- 256 bits.

Taille des clés RSA (1/2)

Complexité

Le meilleur algorithme pour factoriser des clés RSA est le crible algébrique (NFS).

- sa complexité est $L_N(1/3, c)^{1+o(1)}$ avec $c = \sqrt[3]{64/9} \approx 1.923$; le terme $o(1)$ est problématique pour extrapoler;
- selon un travail de Lenstra et Verheul (Selecting cryptographic key sizes, 2001), le terme $o(1)$ est petit pour les tailles cryptographique et sa dérivée est négligeable, donc on peut extrapoler sur des petits intervalles.
- il est raisonnable d'utiliser le modèle de complexité $\kappa L_N(1/3, c)$ pour une constante κ à déterminer expérimentalement.

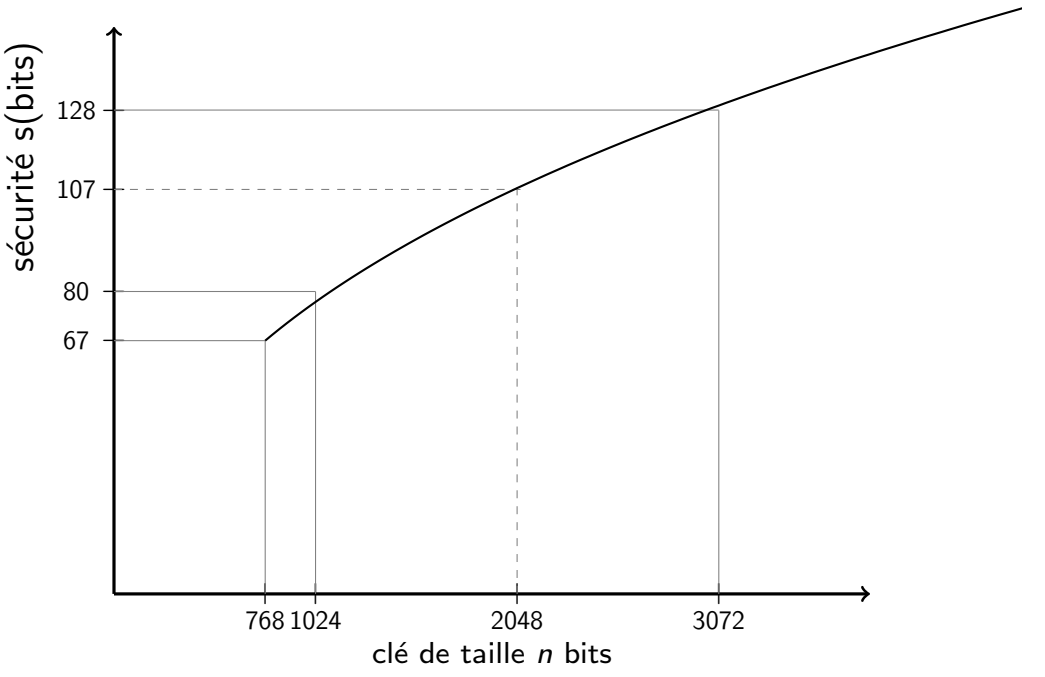
Records de factorisation

- RSA challenge, 768 bits;
- 2009, équipe commune à Nancy, Lausanne, Bonn, Tokyo, Amsterdam et Redmond;
- coût: 2000 années CPU sur des coeurs de 3GHz:

$$\log_2(3G \cdot 2000 \cdot 3.15e7) \approx 66.7.$$

RSA 768 offre une sécurité de 67 bits.

Taille des clés RSA (2/2)



Formule d'extrapolation

$$2^s = 2^{67} \frac{L_{2^n}(1/3, c)}{L_{2^{768}}(1/3, c)}$$

Recommandations gouvernementales

Mécanisme

- La NIST (National Institute of Standards and Technology) émet des spécifications, e.g. “Federal Information Processing Standards Publication 186-4” et ne valide que produits conformes à ses recommandations.
- L’ANSSI (Agence nationale de la sécurité des systèmes d’information) ne valide que les produits conformes avec le “RGS”, issue tous les 2 ans.
- ENISA (European Union Agency for Network and Information Security) émet également des recommandations.

Référentiel général de sécurité version 2.0 (2014): clés RSA

1. (RègleFact-1) La taille minimale du module est de 2048 bits, pour une utilisation ne devant pas dépasser l’année 2030. (L’application d’un paradigme fondamental de la cryptographie, qui consiste à dimensionner les systèmes non pas en se plaçant juste à la limite des capacités d’attaquants connus mais en s’imposant une marge de sécurité, milite pour l’emploi de modules d’au moins 2048 bits, même si aucun module de 1024 bits n’a été officiellement factorisé à ce jour. Par conséquent, nous considérons que l’emploi de modules de 1024 bits constitue une prise de risque incompatible avec des critères de sécurité raisonnables.)
2. (RègleFact-2) Pour une utilisation au-delà de 2030, la taille minimale du module est de 3072 bits.

Autres tailles de clé

DSA (algorithme de signature digitale, basé sur le log discret dans $\mathbb{F}_p$)

Même algorithme (NFS), avec les mêmes paramètres, et ainsi la même taille de clé (RegLogp-1 et 2).

ECDSA (Elliptic curves discrete logarithm)

- le meilleur algorithme en pratique est Pollard's rho de complexité $\kappa 2^{n/2}$;
- le record actuel sur courbes elliptiques à coefficients sur un corps premier correspond à $n = 113$ dans $\approx 2^{60}$ opérations, donc $\kappa \approx 1$.
- Le RGS de l'année 2014 recommande $n = 256$ pour un niveau de sécurité de 128 bits (RègleECp-2).

Taille des clés des couplages

Les couplages ne sont pas standardisés actuellement.

Requière la difficulté de deux problèmes

1. Log discret en $\mathrm{GF}(p^n)$;
2. Log discret sur courbes elliptiques.

Degré de plongement n

Le coût du chiffrement dépend de la courbe elliptique (donc $\log p$) mais aussi du degré de plongement n. Il faut donc équilibrer le niveau de sécurité.

Taille des clés des couplages

Les couplages ne sont pas standardisés actuellement.

Requière la difficulté de deux problèmes

1. Log discret en GF(p^n);
2. Log discret sur courbes elliptiques.

Degré de plongement n

Le coût du chiffrement dépend de la courbe elliptique (donc $\log p$) mais aussi du degré de plongement n. Il faut donc équilibrer le niveau de sécurité.

- pour 80 bits de sécurité on prend $\log p \approx 160$ et $\log(p^n) \approx 1024$, donc $n = 6$;
- pour 128 bits de sécurité on prend $\log p \approx 256$ et $\log(p^n) \approx 3072$, donc $n = 12$.

Special number field sieve (SNFS)

Definition

Pour chaque d, un entier N est d-SNFS s'il existe une base de numération $m \in \mathbb{N}$ telle que $N < m^{d+1}$ et les chiffres de N en base m sont bornées par une constante absolue C.

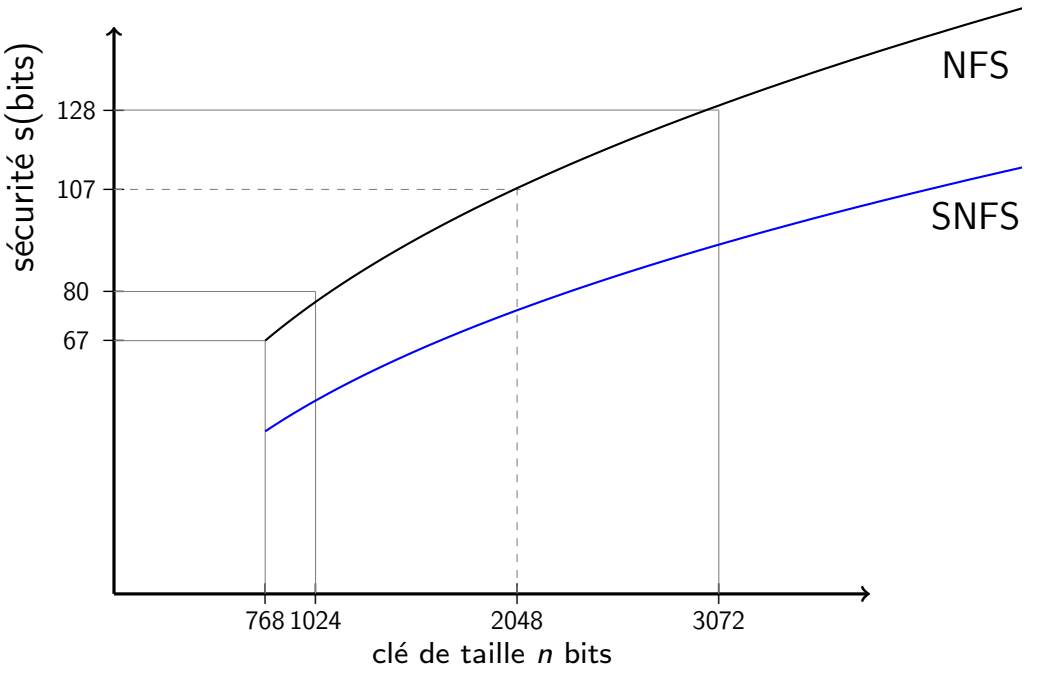
Example

- tous nombre N est 1-SNFS et $\lfloor \log N \rfloor$-SNFS, mais les nombres 6-SNFS ou 7-SNFS sont rares: C^7 au lieu de 2^x nombres de x bits.
- les nombres de type $2^n \pm 1$ sont d-SNFS pour tout d petit:

Conséquences

- Soit d le degré du polynôme f dans NFS, en pratique 6 ou 7. Les algorithmes NFS pour factorisation respectivement logarithmes discrets a une complexité $L_N(1/3, \sqrt[3]{\frac{32}{9}})$ quand on l'entrée (N ou p) est d-SNFS. À comparer avec la complexité de NFS de $L_N(1/3, \sqrt[3]{\frac{64}{9}})$.
- Aoki et al. (2007) ont factorisé un entier 6-SNFS de 1039 bits en 2^{43} opérations.

Difficulté de factoriser des modules SNFS



Formule d'extrapolation

$$2^s = 2^{43} \frac{L_{2^n}(1/3, c_{\mathrm{SNFS}})}{L_{2^{1024}}(1/3, c_{\mathrm{SNFS}})}$$

SNFS par erreur?

Test si un entier est d-SNFS

Require: integer N, degree d and absolute bound κ
Ensure: YES/NO an d-SNFS number, and polynomial f

1: **for** $(f_d, \ldots, f_0)$ in $[-\kappa, \kappa]^{d+1}$ **do**
2: Find the integer roots $\mathcal{R}$ of $\sum f_i x^i - N$
3: **if** $\mathcal{R} \neq \emptyset$ **then**
4: **return** YES and $\sum_{i=0}^{d} f_i x^i$
5: **end if**
6: **end for**
7: **return** NO

Complexité

Le calcul des racines entières d'un polynôme f à coefficients entiers se fait en temps polynomial. En effet, on trouve les racines de f modulo un petit nombre premier, disons 17, ensuite on relève les racines modulo 17^2, 17^3, etc (Hensel). Comme les racines sont bornées par $\sum_{i=0}^{d-1} |f_i|$, il suffit de s'arrêter après $\log(\|f\|_\infty)$ itérations.

Couplages rapides

Constructions générales

Étant donnés q et k on construit des courbes elliptiques E pour lesquelles on sait calculer un couplage de $E \times E$ vers $\mathbb{F}_{q^k}$.

- Pinch et Cocks (2001);
- Dupont, Enge et Morain (2005).

Constructions spéciales

Pour certains degrés de plongement k, on a des constructions spécifiques. L'avantage est que les couplages construits ainsi sont plus rapides.

- $\mathbb{F}_{2^{4n}}$ et $\mathbb{F}_{3^{6n}}$;
- $\mathbb{F}_{p^n}$ avec n petit, principalement

$$\boxed{n = 2, 3, 4, 6 \text{ et } 12.}$$

Caractéristique 2 et 3

Faiblesse

- L'algorithmes quasi-polynomiaux (B., Gaudry, Joux, Thomé 2013) et (Granger, Kleinjung, Zumbrägel 2014) ont montré la faiblesse asymptotique;
- Les records de Granger et al, Joux et Pierrot, Adj et al. (2013,2014,2015) ont montré la nécessité d'augmenter la taille des clés.

Interdiction

Le rapport de l'ENISA 2013 interdit tout cryptosystème basé sur la difficulté du calcul de logarithmes discrets en caractéristique 2 et 3.

Degré de plongement $k = 2$

Évolution de la difficulté de $\mathbb{F}_{p^2}$

D'après B., Gaudry, Guillevic et Morain (août 2014)

- La complexité asymptotique a utilisée est $L_{p^2}(1/3, c)$ avec c compris entre $\sqrt[3]{48/9}$ et $\sqrt[3]{64/9}$;
- Quand $Q = p^k$ a 180 chiffres décimaux (600 bits), le cas $k = 2$ est 260 fois plus faible que $k = 1$.

Raison de l'accélération: méthode de conjugaison

On arrive à construire des polynômes f et g tels que

- f a degré 4 et des coefficients bornés par $\log Q$, par ex. $x^4 + 1$;
- g est de la forme $vx^2 + ux + v$ avec $u, v \approx \sqrt{p} = Q^{1/4}$.

L'équivalent pour la factorisation ou log discret dans $\mathbb{F}_p$ serait $p^{1/3}$ pour $\log_2 Q \leq 3072$, donc $\mathbb{F}_{p^2}$ est plus rapide pour les tailles cryptographiques.

Degré de plongement $k = 3$

Évolution de la difficulté de $\mathbb{F}_{p^3}$

D'après B., Gaudry, Guillevic et Morain (octobre 2015)

- La complexité asymptotique a utilisée est $L_{p^2}(1/3, c)$ avec c compris entre $\sqrt[3]{48/9}$ et $\sqrt[3]{64/9}$;
- Quand $Q = p^k$ a 156 chiffres décimaux (512 bits), le cas $k = 3$ est un peu plus faible que $k = 1$.

Raison de l'accélération

- Conjugaison: on obtient des polynômes f et g de degré 6 et 3 où $\|f\|_\infty = O(1)$ et $\|g\|_\infty = Q^{1/6}$, alors que pour la factorisation et $\mathbb{F}_p$ on aurait $Q^{2/9}$.
- Automorphismes: l'application $x \mapsto \sigma(x) = 1 - 1/(x-1)$ est un automorphisme. Si on a trouvé une relation $a - bx$ on obtient gratuitement deux autres $a - \sigma(x)b$ et $a - \sigma^2(x)b$.

Degré de plongement $k = 4$

Évolution de la difficulté de $\mathbb{F}_{p^3}$

D'après B., Gaudry, Guillevic et Morain (octobre 2015) et Sarkar-Singh (2015)

- La complexité asymptotique a utilisée est $L_{p^2}(1/3, c)$ avec c compris entre $\sqrt[3]{48/9}$ et $\sqrt[3]{64/9}$;
- Quand $Q = p^k$ a 120 chiffres décimaux, le cas des corps $k = 4$ est approximativement aussi dur que celui de $k = 1$.

Accélération supplémentaire possible

D'après B. et Pierrot 2014 la complexité de NFS baisse quand on utilise $V > 2$ polynômes. Il suffit que (a, b) soit friable pour deux polynômes parmi les V, donc la probabilité de succès est multipliée par $V(V-1)/2$. Cela est intéressant même si le coût du crible est multiplié par $V/2$ et la taille de la base de facteurs est plus grande.

Degré de plongement $k = 6$

Évolution de la difficulté de $\mathbb{F}_{p^3}$

D'après Sarkar-Singh (2015)

- La complexité asymptotique à utiliser est $L_{p^2}(1/3, c)$ avec c compris entre $\sqrt[3]{48/9}$ et $\sqrt[3]{64/9}$;
- Afin de faire des records de calcul avec $Q = p^k$ de moins de 180 chiffres décimaux, il faut implanter le crible en dimension 3. Cela consiste à trouver les entiers (a, b, c) tels que $F(a, c, d)$ est B-friable quand F est un polynôme à trois variables. Les principales difficultés:
  - l'approximation de la valeur de $F(a, b, c)$ est plus dure à faire.
  - le coût des précalculs arithmétique n'est plus caché par les accès mémoire;

Accélération supplémentaire possible

- Si on utilise des polynômes f et g de même degré on peut implanter une version MNFS.
- Si on utilise a méthode de Sarkar et Singh, on a des valeurs plus petites à tester pour friabilité. La méthode menace autres degrés si k est composé.

Degré de plongement $k = 12$

BN repose sur $\mathbb{F}_{p^{12}}$ où p est 4-SNFS

Évolution de la difficulté de $\mathbb{F}_{p^3}$

- La complexité asymptotique à utiliser est $L_{p^2}(1/3, c)$ avec $c = \sqrt[3]{64/9}$ et non pas $\sqrt[3]{128/9}$ comme on savait précédemment (Joux et Pierrot 2013);
- Une analyse précise montre que les couplages de Baretto-Naehrig avec p^{12} de 1024 bits demandent de trouver des entiers friables de même taille que pour la factorisation de modules RSA de 1024 bits. L'évolution quand on extrapole à des tailles plus grandes pourrait être plus faible que celle de RSA.

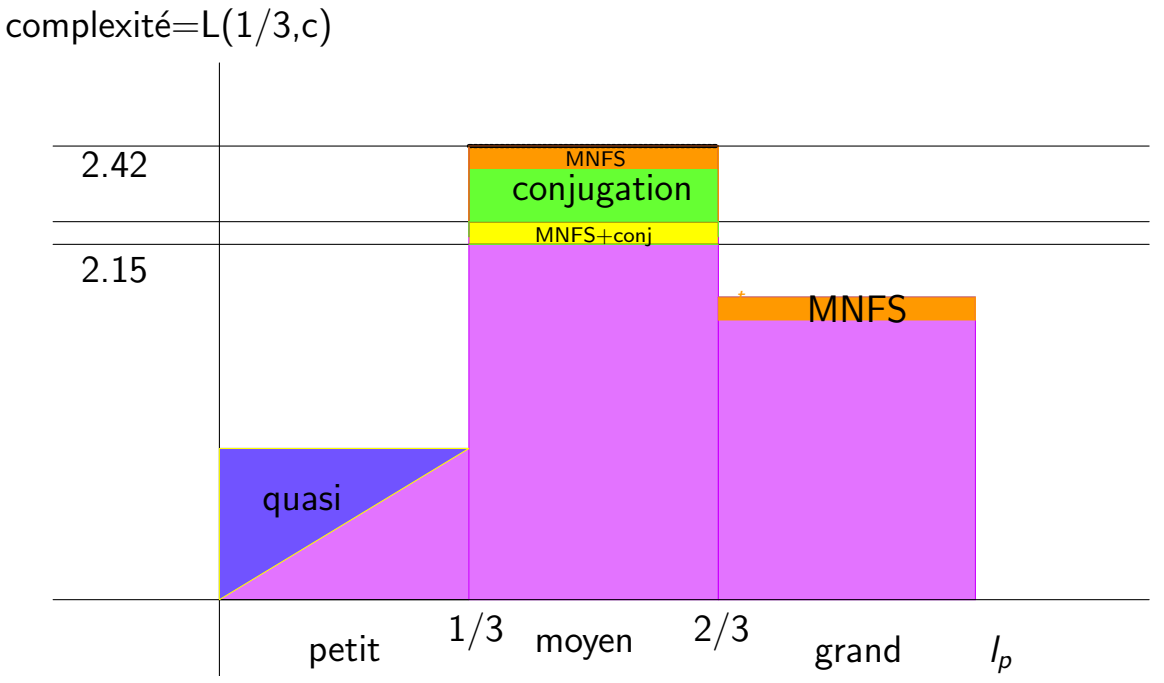
Raison de l'accélération: les tours d'extensions

- Joux et Pierrot ont donné une nouvelle méthode de sélection polynomiale.
- Tower number field sieve: On fixe un corps de nombres $\mathbb{Q}(\iota)$ de degré 12 où p est inerte. On choisit f et g comme si on voulait résoudre des logs discrets dans $\mathbb{F}_p$. Au lieu de chercher des paires a et b dans $\mathbb{Z}$, on les cherche dans $\mathbb{Q}(\iota)$. Cela permet de découpler p et n afin d'utiliser les propriétés de p.

Autre avancées récentes sur p^n

1. Élimination du coût des colonnes lourdes dans l'algèbre linéaire (Thomé || Joux et Pierrot 2015). .
2. MNFS pour la méthode de sélecton polynomiale par conjugaison La complexité du cas quand p est moyen déscend de 2.20 à 2.15 (Pierrot 2015).
3. Le calcul de logarithmes individuels est accéléré quand $n \neq 1$ (Guillevic 2015).
4. Record de calcul dans p^n quand p a 18 bits (Sarkar et Singh).

Complexité du log discret dans $\mathbb{F}_Q$ quand $p = L_Q(l_p)$



Conclusions

Taille des clés

- Les clés se calculent avec la notation

$$L_Q(\alpha, c) = \exp((c + o(1)) \log Q^{\alpha} (\log \log Q)^{1-\alpha}.$$

  qui est imprécise.
- Les records de calcul sont très importants et permettent de mettre à jour la sécurité estimée.
- Pour les couplages on équilibre la sécurité de RSA et celle de ECDSA.

Couplages

- On peut construire des couplages pour tout degré de plongement k mais les plus rapide sont quand $k = 2, 3, 4, 6$ et 12.
- Beaucoup des couplages rapides pourrait être affectés si les avancées récentes se confirment par des records de plus en plus grands.
- Le couplages non attaqués sont p^k avec $k > 6$ premier et p de forme générale (pas SNFS).