Groupe d'étude en théorie analytique des nombres

Jean-Louis Nicolas

Cryptographie et factorisation

Groupe d'étude en théorie analytique des nombres, tome 2 (1985-1986), exp. n° 10, p. 1-6
<http://www.numdam.org/item?id=TAN_1985-1986__2__A5_0>

Groupe d'étude en
THÉORIE ANALYTIQUE DES NOMBRES
3e année, 1985/86, n° 10, 6 p.

20 janvier 1986

CRYPTOGRAPHIE ET FACTORISATION

par Jean-Louis NICOLAS (*)

1. Introduction.

De tout temps le souci de communiquer secrètement, notamment à des fins militaires, a conduit à utiliser des langages chiffrés. Une des méthodes anciennes consiste à permuter les lettres de l'alphabet, par exemple a —> b , b —> c , etc. Ainsi CRYPTOGRAPHIE devient DSZQUPHSBQIJF. Mais la distribution des lettres dans les langues usuelles permet de percer rapidement ce genre de méthodes. Aussi décide-t-on de donner à chaque lettre une valeur numérique (d'où le langage : chiffrement, etc.) et de transformer les nombres ainsi obtenus.

Jusqu'en 1970, la cryptographie avait peu à voir avec les mathématiques. Ensuite la découverte du système R. S. A. (cf. § 3), puis du logarithme discret (cf. § 6), et plus récemment l'utilisation des courbes elliptiques, ont considérablement rapproché les spécialistes de cryptographie et les mathématiciens, particulièrement les arithméticiens.

Cet exposé a pour but de présenter quelques méthodes de chiffrement, et leurs liens avec la théorie des nombres. Pour une étude plus approfondie, on consultera avec intérêt les livres de KONHEIM [Kon] et de MEYER et MATYAS [Mey].

2. Le système D. E. S. (Data Encryption Standard) (cf. [Kon], p. 240, [Mey], p. 141).

Ce système, qui date de 1977, est basé sur l'idée (pas forcément juste) que plus on mélange les données, moins on a de chance de retrouver le message initial à partir du message chiffré.

C'est un système à clé secrète. La clé est un mot de 64 bits (c'est-à-dire un nombre binaire d'au plus 64 chiffres) dont 8 bits sont des bits de contrôle de parité. Il y a donc $2^{56} = 7{,}2\ 10^{16}$ clés possibles.

Chiffrement. - Le message est lui aussi un mot de 64 bits. En 16 étapes, on le transforme successivement en 16 autres mots de 64 bits. Dans chacune de ces étapes, le résultat est une fonction booléenne tout à fait explicite du mot d'entrée et de la clé.

Déchiffrement. - Le processus est symétrique ; c'est-à-dire qu'avec la même clé, si l'on introduit le message chiffré, on ressort le message initial.

Même s'il est un peu long, le chiffrement est facile à réaliser en microprogrammation. La machine comporte 64 fils d'entrée pour la clé, 64 fils d'entrée pour le

(*) Jean-Louis NICOLAS, Département de Mathématiques, Université de Limoges, 123 avenue Albert Thomas, 87060 LIMOGES CÉDEX.

message, et 64 fils de sortie pour le message chiffré.

Pour décrypter le message, la seule solution consiste à essayer les 2^{56} clés possibles, ce qui dépasse les capacités des ordinateurs actuels. Cependant, on ne peut exclure la possibilité d'existence d'un décryptage plus rapide.

Enfin les systèmes à clé secrète nécessitent la transmission de la clé entre les correspondants. Cela peut être réalisé en pratique par les systèmes ci-dessous.

3. Le système R. S. A. (RIVEST, SHAMIR, ADLEMAN) (cf. [Riv]).

Ce système est basé sur le fait, qu'actuellement, il est relativement facile de construire de grands nombres premiers (jusqu'à 200 chiffres) mais qu'il est pratiquement impossible de factoriser un nombre de 100 chiffres. Le record actuel est la factorisation d'un nombre de 81 chiffres (cf. [Bri], [Pom]).

Le chef de réseau choisit deux nombres premiers p et q d'une cinquantaine de chiffres. Il calcule $n = pq$, $\varphi(n) = (p - 1)(q - 1)$, et choisit un nombre d, premier avec $\varphi(n)$. Il calcule e tel que $ed \equiv 1 \mod \varphi(n)$. Il publie n et e dans un annuaire et garde secrets tous les autres nombres.

Un correspondant veut envoyer un message au chef de réseau. Il le met sous la forme de nombres M vérifiant $1 \leqslant M < n$ et envoie le message sous la forme $C = M^e \mod n$. Tout le monde peut connaitre C, mais seul le chef de réseau peut reconstruire M par la formule $M \equiv C^d \mod n$. Le calcul de d par un ennemi est équivalent à la factorisation de n et donc pratiquement impossible.

4. La fonction Ψ de De Bruijn.

Soit $P(n)$ le plus grand facteur premier de n. On définit pour x réel > 1 et pour $y \leqslant x$

$$\Psi(x , y) = \sum_{\substack{n \leqslant x \\ P(n) \leqslant y}} 1 .$$

Cette fonction étudiée d'abord par N. G. De BRUIJN (cf. [De B]) a fait l'objet d'un article récent de A. HILDEBRAND et G. TENENBAUM (cf. [Hil]) où l'on trouvera d'autres références.

On pose $u = (\log x)/(\log y)$. On a alors, pour $u \longrightarrow +\infty$ et $y \geqslant (\log x)^{1+\varepsilon}$,

$$\Psi(x , y) = x\, u^{-u(1+o(1))} .$$

On peut dire également que la probabilité qu'un nombre entier voisin de x ait tous ses diviseurs premiers $\leqslant y$ est environ u^{-u}.

Si l'on pose

$$L = L(x) = \exp(\sqrt{\log x \, \operatorname{loglog} x}) ,$$

l'estimation précédente de Ψ donne, lorsque α et v sont des constantes

$$\Psi(x^{\alpha}, L^{v}) = x^{\alpha} L^{-(\alpha/2v)+o(1)} .$$

Les algorithmes de factorisation récents utilisent tous la famille de nombres dont les facteurs premiers sont relativement petits, et leur temps d'exécution pour un nombre n s'évalue par la formule ci-dessus, et est de la forme $L_{(n)}^{c+o(1)}$ (cf. [Pom])

5. Le logarithme discret.

C'est un résultat bien connu que le groupe multiplicatif des éléments inversibles d'un corps fini est un groupe cyclique. Nous nous limiterons ici au cas des corps $\mathbb{Z}/p\mathbb{Z}$, et p sera un nombre premier assez grand (une centaine de chiffres).

La recherche d'un générateur g de $(\mathbb{Z}/p\mathbb{Z})^{*}$ n'est pas facile en théorie, mais en pratique on essaye les premières valeurs possibles, et l'algorithme aboutit rapidement au résultat.

Tout nombre a, $1 \leqslant a \leqslant p-1$, s'écrit donc

$$g^{x} \equiv a \mod p .$$

On appelle x le logarithme discret de a en base g, car on a la propriété $\log ab \equiv \log a + \log b \mod (p-1)$.

Le calcul du logarithme discret est, pour le moment, de difficulté comparable à la factorisation des entiers naturels. Esquissons ci-dessous deux méthodes.

A. - La méthode des pas de géants - pas de bébés (ou méthode du vernier).

On veut calculer x tel que $g^{x} = a$. On pose $u = [\sqrt{p}] + 1$, on calcule $E_u = \{1, g, \ldots, g^{u-1}\}$, et on l'ordonne.

Ensuite, pour $0 \leqslant k \leqslant u$, on teste si $a\, g^{-ku} \in E_u$. Si oui, on en déduit x.

Cet évènement se produit certainement puisque tout $x \leqslant p-1$ s'écrit

$$x = ku + t \quad \text{avec} \quad t \leqslant u-1 \quad \text{et} \quad k \leqslant \sqrt{p} .$$

La vitesse d'exécution de cet algorithme est $O(\sqrt{p} \log p)$.

B. - Une méthode plus rapide. (Cf. [Odl]).

1ere étape : Précalcul.

On calcule g^{t}, $1 \leqslant t \leqslant T$. On sélectionne les nombres $t_1, t_2, \ldots, t_k$ tels que $g^{t_i} \mod p$ n'ait que des facteurs premiers $\leqslant y$. On peut penser raisonnablement que

$$\frac{k}{T} \neq \frac{1}{p} \Psi(p, y) .$$

Supposons y choisi tel que $\pi(y) \leqslant k$. Le système d'équations, où $r = \pi(y)$

$$g^{t_i} = p_1^{\alpha_{i,1}} \ldots p_k^{\alpha_{i,k}} \quad 1 \leqslant i \leqslant k$$

s'écrit, si l'on pose $p_j = g^{x_j}$,

$$t_i = \sum_{j=1}^{r} \alpha_{i,j} x_j \quad 1 \leqslant i \leqslant k ,$$

permet en général de calculer les x_j , au prix de $O(r^3)$ opérations.

Si l'on choisit $y = L^{(1/2)+o(1)}$, $T = L^{(3/2)+o(1)}$, cette étape s'achève en $L^{(3/2)+o(1)}$ opérations, avec $L = L(p)$. (Cf. § 4).

2e étape : Calcul du logarithme discret de a .

On calcule $g^t a$, pour $1 \leqslant t \leqslant T'$, jusqu'à ce que $g^t a$ ait tous ses facteurs premiers $\leqslant y$. On peut estiner que la probabilité que $g^t a$ ait tous ses facteurs premiers $\leqslant y$ est voisine de $1/p\,\Psi(p , y) = L^{-(1+o(1))}$; donc, en choisissant $T' = L^{1+o(1)}$, on a de bonnes chances de trouver t tel que

$$g^t a = p_1^{\alpha_1} \ldots p_k^{\alpha_k} .$$

Si a s'écrit g^x , on calcule x par

$$t + x = \alpha_1 x_1 + \ldots + \alpha_k x_k .$$

Nous avons ainsi décrit un algorithme probabiliste de calcul du logarithme discret dont la vitesse d'exécution est euristiquement de $L^{(3/2)+o(1)}$. Par quelques amélioration, on peut arriver à $L^{1+o(1)}$.

6. Méthodes cryptographiques basées sur le logarithme discret.

(A) Échange de clé. - Cette méthode peut en particulier servir à échanger la clé du système D. E. S. entre les correspondants. On suppose p premier fixé et g un générateur connu de $(\mathbb{Z}/p\mathbb{Z})^*$. A choisit un nombre a secret, $1 < a < p - 1$. Il calcule g^a mod p qu'il publie dans un annuaire. B fait de même avec g^b .

A peut donc calculer $(g^b)^a$ en lisant g^b dans l'annuaire. B peut calculer $(g^a)^b = (g^b)^a$. Ils ont donc une clé secrète g^{ab} entre eux.

(B) Transmission de message. - Le message M est un nombre vérifiant $1 \leqslant M \leqslant p-1$, que l'on assimile à un élément de $(\mathbb{Z}/p\mathbb{Z})^*$. A choisit a comme précédemment, en s'assurant que $(a , p - 1) = 1$. Il calcule a' tel que $aa' \equiv 1 \mod (p - 1)$. B fait de même avec b et b' . Pour envoyer le message M , A envoie d'abord M^a à B . B renvoie $(M^a)^b = M^{ab}$ à A . A renvoie à B le message $(M^{ab})^{a'} = M^b$, et finalement B calcule $M^{bb'} = M$.

Un ennemi peut facilement connaitre M^a , M^{ab} , et M^b . Pour connaitre a' , il lui faut connaitre $a = \log M^{ab} / \log M^b$.

7. Méthode du sac à dos.

Le problème, classique en optimisation, du sac à dos (knapsack) est le suivant : étant donné des coefficients $c_1 , \ldots , c_k$ et un nombre C , trouver $x_1 , \ldots , x_k$ à valeur 0 ou 1 tels que

$$\sum_{1 \leqslant i \leqslant k} c_i x_i = C .$$

Lorsque $c_i = 2^i$, le problème est facile, et plus généralement, lorsque les c_i vérifient $c_i > \sum_{j \leq i-1} c_j$, mais, dans le cas général, c'est un problème difficile.

La méthode de cryptographie à clé publique est la suivante. Le chef de réseau choisit des coefficients c_1 , ... , c_k vérifiant les inégalités ci-dessus, et un nombre $N > c_1 + \ldots + c_k$. Il choisit ensuite un nombre D , premier avec N et calcule D' tel que $DD' \equiv 1 \mod N$. Il calcule ensuite

$$a_i = D' c_i \mod N \quad 1 \leq i \leq k .$$

Il publie a_1 , ... , a_k .

Pour envoyer le message constitué des bits m_1 , ... , m_k , on envoie au chef de réseau la quantité $C = \sum_{i=1}^{k} a_i m_i$. Pour déchiffrer, le chef de réseau calcule $DC \mod N$, qui est, par ailleurs, égal à $\sum_{i=1}^{k} c_i m_i$. Il est donc facile de calculer les m_i .

Malheureusement, cette méthode a été cassée par l'algorithme 3L , qui permet de trouver un vecteur raisonnablement court dans un réseau.

8. Courbes elliptiques.

Il est connu que les points de la courbe

$$y^2 = x^3 + ax + b \quad (a , b \text{ fixés})$$

forment un groupe abélien, si on leur adjoint le point à l'infini dans la direction de l'axe des y , qui est le zéro, et trois points alignés ont une somme nulle. Ceci est encore valable lorsque x et y appartiennent à un corps fini $\mathbb{Z}/p\mathbb{Z}$. On obtient alors un groupe fini, dont l'ordre est compris entre $p + 1 - 2\sqrt{p}$ et $p + 1 + 2\sqrt{p}$, et qui est soit un groupe cyclique, soit le produit de deux groupes cycliques.

Lorsque ce groupe est cyclique, on peut appliquer les méthodes cryptographiques exposées au § 6. L'inconvénient est que la loi de groupe est un peu plus difficile à calculer par l'ordinateur. En revanche, le logarithme discret dans ce groupe est plus difficile à évaluer : la méthode A du § 5 peut aisément s'adapter, mais pas la méthode B.

RÉFÉRENCES

[Bri] BRILLHART (J.), LEHMER (D. H.), SELFRIDGE (J.-L.), TUCKERMAN (B.) and WAGSTAFF (S. S.). - Factorizations of $b^n \pm 1$. - Providence, American Mathematical Society, 1983 (Contemporary Mathematics, 22).

[De B] De BRUIJN (N. G.). - On the number of positive integers $\leq x$ and free of prime factors $> y$, II, Koninkl. nederl. Akad. Wetensch., Proc., Series A, t. 69, 1966, p. 239-247 ; Indag. Math., t. 28, 1966, p. 239-247.

[Hil] HILDEBRAND (A.) and TENENBAUM (G.). - On integers free of large prime factors, Trans. Amer. math. Soc., t. 296, 1986, p. 265-290.

[Kon] KONHEIM (A. G.). - Cryptography. A primer. - New York, J. Wiley and Sons, 1981.

[Mey] MEYER (C. H.) and MATYAS (S. M.). - Cryptography : A new dimension in computer data security. - New York, J. Wiley and Sons, 1982.

[Odl] ODLYZKO (A. M.). - Discrete logarithms in finite fields and their cryptographic significance, "Advances in cryptology", p. 224-314. - Berlin, Springer-Verlag (Lecture Notes in Computer Science, 209).

[Pom] POMERANCE (C.). - Analysis and comparison of some integer factoring algorithms, "Computational methods in number theory", part 1, p. 89-139. - Amsterdam, Mathematisch Centrum, 1982 (Mathematical Centre Tracts, 154).

[Riv] RIVEST (R. L.), SHAMIR (A.) and ADLEMAN (L.). - A method for obtaining digital signatures and public-key cryptosystems, Communic. Assoc. comput. Mach., t. 21, 1978, p. 120-126.