THÉORIE DES NOMBRES ET CRYPTOGRAPHIE

par

David Kohel & Igor E. Shparlinski

Résumé. — Des constructions en cryptographie moderne sont basées sur la théorie des nombres. Toutefois, les liens entre ces deux domaines sont plus profonds qu'il n'y paraît. Le développement de la cryptographie moderne a eu lieu en parallele avec des developpements et des questions centrales en théorie des nombres. Après des rappels de constructions en cryptographie à clef publique, à base de l'arithmétique modulaire, des corps finis, des réseaux et des courbes elliptiques, nous décrivons quelques unes de ces racines en théorie des nombres. La première concerne l'hypothèse de Riemann et les questions associées sur la distribution des nombres premiers et des nombres friables, et des distributions des diviseurs d'entiers. Puis on considère les origines de la cryptographie à base de courbes elliptiques, en commençant par le théorème de Hasse, les conjectures de Weil, et l'algorithme de Schoof. Finalement on se place dans le contexte du théorème de Mordell et de la conjecture de Birch et Swinnerton-Dyer. En conclusion on considère les perspectives d'avenir pour ces cryptosystèmes.

Abstract. — Modern cryptographic constructions are based on constructions from number theory, but many of the links go deeper than typically realized. The development of modern cryptography runs in parallel to developments and central questions in number theory. After recalling some of the constructions used in modern public key cryptography, based on modular arithmetic, finite fields, lattices and elliptic curves, we describe some of their number theoretic origins. The first concerns the Riemann hypothesis and associated questions of distributions of prime numbers and smooth numbers, and of distributions of divisors of integers. Next we consider the origins of elliptic curve cryptography, beginning from Hasse's theorem, the conjectures of Weil, and Schoof's algorithm. Finally we mention the context of Mordell's theorem and the conjectures of Birch and Swinnerton-Dyer. In conclusion we consider the future prospects of these cryptosystems.

1. Introduction

La théorie des nombres est un domaine des mathématiques qui depuis des siècles a été perçu comme une discipline pure sans aucune interaction avec le monde réel, mais qui a emergé au milieu du XXème siècle au coeur de la recherche en technologie de l'information. En particulier, la théorie des nombres joue un rôle important dans la théorie de l'information, la théorie des codes, et la cryptographie. Parmi les applications en cryptographie, certaines parmi les plus intéressantes et les plus novatrices résident en cryptographie à clef publique (ou asymétrique). Nous nous concentrons ici sur celles-ci, même s'il existe aussi des applications en cryptographie symétrique.

On rappelle qu'en cryptographie symétrique une seule clef secrète sert à la fois pour le chiffrement et le déchiffrement des messages. Par conséquent, les protagonistes doivent d'abord avoir accès à un canal sécurisé pour établir la clef secrète commune dont ils se serviront pour communiquer plus tard. Le canal sécurisé prend typiquement la forme d'une réunion face-à-face, mais peut être un courrier fiable qui transporte les clefs entre les protagonistes de la communication. Ces cryptosytèmes sont sûrs et efficaces, mais le besoin d'un canal sécurisé limite leur utilité dans de nombreuses situations pratiques.

Dans un cryptosystème à clef publique, la clef secrète est remplacée par une paire de clefs — une clef publique pour le chiffrement et une clef privée qui appartient au destinataire pour le déchiffrement. Dans une communication en cryptographie à clef publique, les participants utilisent un canal complètement public (non-sécurisé), soit pour établir une clef secrète, soit pour un envoyer un message. Ces cryptosystèmes sont typiquement plus lents que leurs analogues symétriques, mais se trouvent partout grâce à leur souplesse et leur adaptablilité à des tâches comme la signature numérique, le vote électronique, la preuve de connaissance, etc. Leur sécurité repose sur la difficulté presumée d'un problème computationnel en théorie des nombres – nous allons en décrire quelques uns. La communauté des cryptologues se subdivise en deux espèces. Les cryptographes, qui recherchent des algorithmes les plus efficaces possibles pour le chiffrement et le déchiffrement avec un niveau donné de securité, et les cryptanalystes, qui jouent un rôle important pour comprendre la sécurité de ces systèmes, en cherchant des algorithmes pour résoudre ou contourner les problèmes difficiles sur lesquels ils sont basés (on parle d'*attaques* contre le cryptosystème). Certaines des applications les plus excitantes de la théorie des nombres en cryptographie concernent les algorithmes pour résoudre ces problèmes difficiles.

2. Origines historiques

2.1. Une nouvelle direction en cryptographie. — En 1976, l'article original de Diffie et Hellman, *New directions in cryptography* [**19**] a introduit les fondements de la cryptographie à clef publique, et, en même temps, le premier schéma permettant à deux protagonistes de créer un secret commun en utilisant seulement un canal de communication ouvert. Leur idée simple et élégante s'appelle actuellement le protocole d'échange de clef de Diffie-Hellman.

On décrit ce protocole entre deux protagonistes A et B. D'abord, les interlocuteurs choisissent en grand nombre premier p. De nos jours, "grand" veut dire $p \approx 2^n$, pour n (le nombre de *bits* de p) entre 512 et 1024, ou pour les paranoïaques, on peut choisir un nombre premier de 2048 bits. Le premier est "public", dans le sens que l'on permet à n'importe qui de découvrir le choix de p. Puis les deux protagonistes utilisent p pour définir un corps fini $\mathbb{F}_p$ de p éléments (l'anneau $\mathbb{Z}/p\mathbb{Z}$ des résidus des entiers modulo p). Dans le groupe $\mathbb{F}_p^*$ des unités, ils choisissent un élément primitif, c'est-à-dire un générateur g tel que

$$\mathbb{F}_p^* = \{g^0, g^1, \dots, g^{p-2}\},$$

en utilisant le fait que ce groupe est cyclique.

Puis les interlocuteurs A et B choisissent des éléments "privés" k et ℓ dans $\mathbb{Z}/(p-1)\mathbb{Z} = \{0, 1, \dots, p-2\}$ et calculent g^k et g^ℓ dans $\mathbb{F}_p^*$, respectivement (en utilisant au plus $2n$ opérations dans $\mathbb{F}_p^*$ avec une stratégie de "multiplication-et-élévation-au-carré"). Après l'échange des valeurs g^k et g^ℓ sur le canal non-sécurisé, chaque protagoniste peut facilement calculer

$$\left(g^k\right)^\ell = \left(g^\ell\right)^k = g^{k\ell},$$

qui servira comme clef secrète commune pour un cryptosystème symétrique au choix.

Clairement, ce protocole ne fournit pas de protection absolue car un adversaire peut toujours trouver k, étant donnés g et $h = g^k$. L'entier k s'appelle le logarithme discret de h en base g, et on écrit $k = \log_g h$. Le problème de trouver k, étant donnés g et h dans $\mathbb{F}_p^*$, est le fameux *problème du logarithme discret*. Une approche naïve – mais déterministe – de ce problème est simplement de calculer la suite g^i pour $i = 0, 1, \dots$ jusqu'à l'obtention de l'identité $g^i = h$, rendant $i = k\,(= \log_g h)$. La connaissance de k laisse la clef publique sans sécurité. Néanmoins, pour p grand, cette algorithme est computationnellement infaisable. Ainsi, une valeur de p autour de 2^{120} fournit déjà une protection suffisante. Comme nous l'avons remarqué, des valeurs de p beaucoup plus grandes sont recommandées — cela parce qu'il y a des attaques plus efficaces sur le problème du logarithme discret, voir Section 4.2. Les algorithmes les plus efficaces appartiennent à la classe d'algorithmes dits de *calcul d'indices*, introduits par Odlyzko [**50**]. De nos jours, le représentant le plus efficace de cette classe est le *crible sur les corps des nombres* (voir [**14**] pour l'état de l'art actuel). Le

comportement de ces algorithmes est basé sur nos connaissances (démontrées) et des heuristiques (conjecturales) sur l'arithmétique des entiers.

On remarque aussi que la sécurité du schéma dépend énormément du plus grand nombre premier q divisant $p-1$, plutôt que de la taille de $p-1$.[1] Par conséquent, il est plus courant de choisir g parmi les éléments de $\mathbb{F}_p^*$ d'ordre un grand nombre premier q divisant $p-1$, au lieu d'un élément d'ordre $p-1$ parce que l'on gagne de la vitesse en calculant g^k et g^ℓ sans perdre en sécurité.

On termine cette section en remarquant que le protocole d'échange de clef de Diffie-Hellman n'est pas un cryptosystème — aucune information qui porte un message n'a été transmise entre A et B, et donc il doit être completé par un cryptosystème symétrique traditionnel.

2.2. In RSA We Trust ! [2]— Le cryptosystème de RSA, inventé par Rivest, Shamir et Adleman **[54]**, est le premier schéma permettant l'échange de message sans le besoin d'un cryptosystème symétrique (avec une clef secrète).

Le cryptosystème RSA est basé sur le *théorème d'Euler*, un des résultats les plus connus et fondamentaux de l'arithmétique, qui dit que pour des entiers a et N avec $\mathrm{pgcd}(a,N)=1$ on a

$$a^{\varphi(N)} \equiv 1 \bmod N. \tag{1}$$

où φ est l'indicatrice d'Euler. Dans le cas de RSA, N est le produit pq de deux nombres premiers distincts p et q, et $\varphi(N)=(p-1)(q-1)$.

Maintenant, si A veut recevoir un message privé, il a besoin d'établir un couple de clefs publique–privée comme suit :

– Choisir deux grands nombres premiers p et q et prendre $N=pq$.
– Choisir un exposant e avec $\gcd(e,\varphi(N))=1$.
– Calculer l'exposant de déchiffrement d tel que $ed \equiv 1 \bmod \varphi(N)$.

Puis A publie (N,e) comme sa clef publique, et garde d comme clef privée, en se débarrassant de p, q et $\varphi(N)$ de manière sûre. [3]

1. En effet la sécurité dépend aussi de la taille de $p-1$ ($\approx \log_2(p)$ bits), mais de manière *subexponentielle*, tandis que la sécurité est bornée de manière *exponentielle* par la taille de q (de $\log_2(q)$ bits). Le calibrage entre les tailles relatives de q et p de sorte que les bornes de sécurité dues à chacun soient équilibrées, est à l'origine du choix d'un nombre premier q de 160-bits couplé à un nombre premier p de 512 à 1024 bits dans la norme américaine, le *Digital Signature Algorithm*.

2. Ce titre fait reférence aux inscriptions figurant sur les monnaies américaines.

3. En pratique, les nombres premiers p et q sont retenus, et l'exposant de déchiffrement est gardé sous forme de deux paires, $(d_1, p-1)$ et $(d_2, q-1)$ tels que $d_1 = d \bmod (p-1)$ et $d_2 = d \bmod (q-1)$. L'exposant de déchiffrement d est donc bien défini modulo le diviseur propre $\mathrm{ppcm}(p-1,q-1)$ de $\varphi(N)$.

Pour envoyer un message m de manière sûre à A, en supposant que m est codé par un élément de $\mathbb{Z}/N\mathbb{Z}$, l'expéditeur B calcule et envoie le *texte chiffré* :

$$c \equiv m^e \bmod N.$$

Maintenant, pour déchiffrer, A calcule

$$c^d \equiv (m^e)^d \equiv m^{ed} \equiv m \bmod N.$$

Cette dernière égalité vient du théorème d'Euler (1) parce que $ed = 1 + k\varphi(N)$ pour un entier k.

On remarque que la sécurité de RSA repose sur la difficulté de trouver l'exposant de déchiffrement d, étant données les valeurs publiques e et N. L'attaque la plus évidente est d'essayer de calculer $\varphi(N)$. Certainement, si l'adversaire peut factoriser N et ainsi trouver les valeurs de p et q, alors la valeur de $\varphi(N)$, et ensuite de d, s'en déduit facilement. En effet, il est facile de voir que la tâche de trouver $\varphi(N)$ est équivalente en temps polynômial au problème de factorisation de N. Dans une direction, ceci est trivial, et dans l'autre direction, si $\varphi(N)$ est connu, on obtient deux relations :

$$p + q = N - \varphi(N) + 1 \text{ et } pq = N,$$

donc p et q sont retrouvés comme les racines de $x^2 - (p+q)x + pq$.

On remarque qu'une fois que la factorisation de N est connue, les chiffrements par la clef publique (e, N) sont facilement inversibles pour chaque valeur de e.

Il est naturel de se demander s'il existe des attaques contre des valeurs particulières de e et d ou même contre un message particulier. On pensait généralement que, si certaines précautions étaient suivies, il n'y aurait pas d'autre attaque contre RSA que l'attaque par factorisation. Cela a été démontré en 2002 par Cramer et Shoup **[13]**, et par Fujisaki, Okamoto, Pointcheval et Stern **[23]** : un cryptosystème RSA *correctement implémenté* n'admet que l'attaque par factorisation, voir Section 3.2 pour des dangers d'utilisation négligente de RSA.

2.3. Le cryptosystème d'ElGamal.— Comme nous l'avons remarqué, le schéma original de Diffie-Hellman sert seulement pour établir une clef secrète commune. Néanmoins, en 1985, Elgamal [4] a observé qu'une petite modification de ce schéma donne un cryptosystème **[20]**.

Le cryptosystème d'ElGamal **[20]** utilise un nombre premier p, avec grand diviseur premier q de $p-1$, et un générateur g du sous-groupe cyclique G d'order q de $\mathbb{F}_p^*$, tous publics. Des destinataires multiples peuvent établir des clefs publiques compatibles à partir d'un triplet (p, q, g) donné. En particulier,

4. Le cryptographe égyptien a choisi de publier sous le nom Taher Elgamal (au lieu d'El Gamal), mais le nom plus courant du cryptosystème s'écrit ElGamal.

le destinataire A choisit un élément k au hasard dans l'anneau résiduel $\mathbb{Z}/q\mathbb{Z}$ et calcule $h = g^k$ dans $\mathbb{F}_p^*$. Il publie la *clef publique* (p, q, g, h), et retient k pour sa clef privée.

Pour *chiffrer* un message m de $\mathbb{F}_p$, l'expéditeur B choisit un ℓ au hasard dans $\mathbb{Z}/q\mathbb{Z}$, calcule la paire

$$(r, s) = (g^\ell, mh^\ell) \in \mathbb{F}_p^* \times \mathbb{F}_p,$$

qu'il envoie à A. Pour *déchiffrer* ce texte chiffré, A calcule

$$r^k = g^{k\ell} = h^\ell \text{ et puis } s(r^k)^{-1} = sh^{-\ell} = m.$$

On remarque que le choix aléatoire de ℓ associe des textes chiffrés multiples à un message donné.

L'avantage majeur de ce cryptosystème sur RSA est qu'il se généralise facilement à un groupe abélien arbitraire. En particulier, il a donné naissance à la cryptographie à base de courbes elliptiques (voir Section 2.5).

2.4. L'ascension, la chute, et la résurrection de cryptosystèmes à sac à dos [5]**.** — L'idée de baser un cryptosystème sur le problème du *sac à dos* a été introduite par Merkle et Hellman [**45**] en 1978. D'abord, on rappelle ce problème :

Problème. — *Étant donnés n objets de tailles $a_1, \ldots, a_n$ et un "sac à dos" de capacité A, trouver une sélection $I \subseteq \{1, \ldots, n\}$ d'objets qui tiennent juste dans le sac à dos, c'est-à-dire*

$$A = \sum_{i \in I} a_i,$$

ou démontrer qu'aucun tel choix n'existe.

Ce problème se décrit de manière équivalente comme le problème de trouver un vecteur entier $(x_1, \ldots, x_n)$ in $\mathbb{Z}^n$, avec x_i dans $\{0, 1\}$, tel que

$$A = (a_1, \ldots, a_n) \cdot (x_1, \ldots, x_n) = \sum_{i=1}^{n} a_i x_i$$

En prenant $a_{n+1} = -A$ et $x_{n+1} = 1$, on peut exprimer cette dernière équation comme

$$(a_1, \ldots, a_{n+1}) \cdot (x_1, \ldots, x_n) = \sum_{i=1}^{n+1} a_i x_i = 0,$$

5. Le titre de cette section est une version plus optimiste du fameux article, *The rise and fall of knapsack cryptosystems*, d'Odlyzko [**51**], grâce à des développements récents.

et reconnaître le problème comme celui de trouver un vecteur court dans le complément orthogonal de $(a_1, \ldots, a_{n+1})$ dans $\mathbb{Z}^{n+1}$, engendré sur $\mathbb{R}$ par

$$\{v_1 = (-a_2, a_1, 0, \ldots, 0), \ldots, v_n = (0, \ldots, 0, -a_{n+1}, a_n)\}.$$

Ainsi on place le cryptosystème "sac à dos" dans le contexte plus général des cryptosystèmes à base de réseaux (voir ci-dessous). Tandis que le problème du sac à dos est NP-complet, dans des cas particuliers il est assez facile de le résoudre. Par exemple, c'est le cas pour le problème du sac à dos *super-croissant*, pour lequel on a

$$a_1 + \cdots + a_{i-1} < a_i \text{ pour } i = 2, \ldots, n.$$

Dans ce cas, étant donné A, tel que

$$A = \sum_{i \in I}^{n} a_i,$$

on peut récuperer $x_i \in \{0, 1\}$ qui déterminent I en utilisant un algorithme "glouton." Précisement, en commençant par $(x_1, \ldots, x_n) = 0$, pour le plus grand indice i tel que $a_i \leq A$, on incrémente x_i de 1 et décremente A de a_i. Puis on itère jusqu'à ce que $A = 0$, un $x_i > 1$, ou aucun tel i existe.

On peut, néanmoins, essayer de cacher la structure super-croissante en choisissant $p > a_n$, un entier $\lambda \not\equiv 0 \bmod p$ au hasard, et un élément π du groupe symétrique S_n, puis en publiant une permutation

$$(c_1, \ldots, c_n) = (b_{\pi(1)}, \ldots, b_{\pi(n)})$$

de résidus $b_i \equiv \lambda a_i \bmod p$, pour $i = 1, \ldots, n$. Ensuite, on chiffre un vecteur binaire $(y_1, \ldots, y_n) \in \{0, 1\}^n$ par

$$C = \sum_{i=1}^{n} c_i y_i.$$

La personne qui connaît p, λ et π peut calculer $A \equiv \lambda^{-1} C \bmod p$, trouver $(x_1, \ldots, x_n)$ pour le sac à dos super-croissant, et puis calculer

$$y_i = x_{\pi^{-1}(i)}, \text{ pour } i = 1, \ldots, n.$$

Cette idée est très simple, et le chiffrement et le déchiffrement sont tout les deux très rapides. Malheureusement, ce schéma et toutes les extensions semblables sont cassées par l'application du fameux algorithme LLL de Lenstra, Lenstra et Lovász **[37]**, qui est très efficace pour trouver un vecteur court dans un réseau. À la suite de plusieurs tentatives, sans réussite, de contruire un cryptosystème fiable autour de problèmes difficiles dans un réseau, cette direction a été abandonnée pendant un certain temps (voir Odlyzko **[51]**).

Les cryptosystèmes à sac à dos appartiennent à la classe des *cryptosystèmes à base de réseaux*, qui ont initialement tous partagés le même destin. On rappelle

qu'un réseau L de rang r dans un espace vectoriel $\mathbb{R}^n$ de dimension n est un sous-groupe discret de la forme

$$L = \{\mathbf{v} = \sum_{i=1}^{r} c_i \mathbf{v}_i \ : \ c_1, \ldots, c_r \in \mathbb{Z}\}$$

où la *base* $\{\mathbf{v}_1, \ldots, \mathbf{v}_r\}$ de L est un ensemble linéairement indépendant dans $\mathbb{R}^n$.

La première percée théorique, qui a relancé l'interêt pour les cryptosystèmes à base de réseaux, s'est passé en 1997, quand Ajtai et Dwork [**2**] et Goldreich, Goldwasser et Halevi [**25**] ont réouvert cette direction. Bien que ces cryptosystèmes et leurs variantes sont soit impraticables, soit susceptibles d'attaques (ou les deux) voir [**46, 49**], ils ont démontré la vitalité de l'idée d'utiliser un problème difficile de la géométrie des nombres dans un but cryptographique. De plus, en même temps, le cryptosystème NTRU très pratique a été inventé par Hoffstein, Pipher and Silverman [**32**]. Presque deux décennies d'attaques sur NTRU ont donné lieu à une série de modifications et de réglages du schéma original, mais il semble qu'il a survecu à toutes ces tentatives et fournit un cryptosystème efficace et sûr.

En général, on identifie les problèmes suivants, qui réapparaissent dans plusieurs scénarios d'intérêt cryptographique.

Problème de la base minimale : Étant donné un réseau $L \subseteq \mathbb{R}^n$, trouver une base telle que son plus long vecteur soit de longueur minimale.

Problème du vecteur le plus court : Étant donné un réseau $L \subseteq \mathbb{R}^n$, trouver un vecteur $\mathbf{u} \in L$ non-nul de longueur minimale.

Problème du vecteur le plus proche : Étant donnés un réseau $L \subseteq \mathbb{R}^n$ et un vecteur "cible" $\mathbf{z} \in \mathbb{R}^n$, trouver un vecteur $\mathbf{u} \in L$ le plus proche possible de $\mathbf{z}$.

Tous ces problèmes sont difficiles, et on pense qu'il n'existe aucun algorithme sous-exponentiel pour ces problèmes. Néanmoins, l'algorithme classique LLL de Lenstra, Lenstra et Lovász [**37**] et ses variantes réussissent à résoudre plusieurs exemples de ces problèmes, ce qui a engendré à son tour de nombreuses attaques cryptographiques (voir [**49**]).

Finalement, récemment un nouveau problème appelé *apprendre avec erreurs* en rapport avec les réseaux a été introduit par Regev [**53**]. Cela a ouvert une nouvelle direction prometteuse en cryptographie à base de réseaux, appellée *chiffrement homomorphique*, qui est en pleine croissance (voir [**11, 39, 40**] pour les développements les plus recents).

2.5. Cryptographie à base de courbes elliptiques. — Une brève inspection des protocoles de Diffie-Hellman et ElGamal (voir les sections 2.1 et 2.3) permet de voir qu'ils peuvent être adaptés à n'importe quel groupe abélien, et

non seulement à des sous-groupes de $\mathbb{F}_p^*$. Ainsi, en 1986, Koblitz [**35**] et Miller [**47**] ont proposé l'utilisation du groupe des points d'une courbe elliptique sur un corps fini. Pour mettre en oeuvre les protocoles de Diffie-Hellman et ElGamal, il est necessaire de déterminer l'ordre du groupe des points, [6] et en 1985, Schoof [**55**] a publié un algorithme, avec complexité polynômiale, pour déterminer cet ordre. Cela a permis Koblitz et Miller d'introduire la cryptographie à base des courbes elliptiques l'année suivante.

On rappelle qu'une courbe elliptique E sur un corps K (de caractéristique differente que 2 et 3) peut être donnée par une courbe plane

$$y^2 = x^3 + ax + b,$$

complétée avec un "point à l'infini" O. [7] L'ensemble de points admet une opération d'addition, définie par de fonctions rationnelles, donnant à E une structure de groupe. Sur le corps fini $K = \mathbb{F}_p$, l'ensemble des points rationnels

$$E(K) = \{O\} \cup \{(x, y) \in K^2 \ : \ y^2 = x^3 + ax + b\}$$

a cardinalité $p - t + 1$, où $|t| \leq 2\sqrt{p}$ par le théorème de Hasse [**29**] datant de 1936. C'est un cas spécial des conjectures plus générales sur les courbes énoncées par Artin [**3**] dans sa thèse de 1921 (publiée en 1924). Les conjectures importantes de Weil [**61**] en 1949 (finalement démontrées par Deligne [**15**] en 1974) ont donné une interprétation topologique et continue de ce problème discret de comptage de points.

L'interêt cryptographique vient du fait que l'ensemble des points rationnels $E(K)$ admet une structure de groupe avec identité O. Cette structure peut être dérivée du théorème de Bézout en géométrie algébrique — toute droite coupe E en trois points, comptés sur la clôture algébrique et avec multiplicités, et trois points alignés ont par définition somme O. La loi de groupe est algébrique, au sens que la somme de deux points rationnels $P = (x_1, y_1)$ and $Q = (x_2, y_2)$, s'exprime par des fonctions rationnelles n'impliquant que les coordonnées (x_1, y_1) et (x_2, y_2) des points et les coefficients (a, b) de la courbe.

Bien que l'arithmétique du groupe des courbes elliptiques soit plus compliquée, l'avantage de la cryptographie à base de courbes elliptiques est que la méthode de calcul d'indices n'est pas applicable au problème du logarithme discret sur les courbes elliptiques sur $\mathbb{F}_p$. En effet, la première attaque, trouvée

6. À proprement parler, seulement le protocole d'ElGamal a besoin de la connaissance de l'ordre du groupe. Néanmoins, dans le protocole de Diffie-Hellman, si l'ordre du groupe n n'est pas premier, et si n, avec sa factorisation, devient public, la sécurité est limitée par la taille du plus grand nombre premier divisant n. L'algorithme de Schoof assure qu'on peut choisir une courbe elliptique d'ordre premier.

7. Le point à l'infini est le point $O = (0 : 1 : 0)$ dans $\mathbb{P}^2(K)$ sur l'extension de la courbe du plan affine $\mathbb{A}^2$ au plan projectif $\mathbb{P}^2$.

par Menezes, Okamoto et Vanstone [**44**], était indirecte. Ils ont utilisé le *couplage de Weil* pour transférer le logarithme discret sur une courbe elliptique au groupe multiplicatif $\mathbb{F}_{p^n}^*$ d'une extension du corps $\mathbb{F}_p$. Pour une courbe aléatoire, néanmoins, le degré n de cette extension est de taille prohibitive pour poser un problème [**5**]. Néanmoins, pour des courbes sur certaines extensions de $\mathbb{F}_p$, il existe des attaques plus efficaces dues à Gaudry, Hess et Smart [**24**] et dans des travaux plus récents, à Diem [**17, 18**].

3. Des attaques et du mauvais usage

3.1. Préambule. — Tout cryptosystème à clef publique dépend d'un problème (computationnellement) difficile en théorie des nombres, comme la factorisation des entiers ou le logarithme discret. L'avancement algorithmique sur ces problèmes est surveillé attentivement et extrapolé à court et moyen termes, et les tailles des paramètres sont ajustées à chaque avancement majeur. Néanmoins, il y a beaucoup d'attaques contre des erreurs de mise en oeuvre ou un mauvais emploi de cryptosystèmes théoriquement fiables, ce qui permet de contourner le problème difficile sous-jacent. Ici nous donnons quelques exemples de ce genre d'attaques.

3.2. Accélérer RSA. — Puisque le temps de calcul du texte chiffré $c \equiv m^e \bmod N$ et le temps de déchiffrement $m \equiv c^d \bmod N$ dépendent linéairement de la taille (en bits) de e et d, il est tentant de choisir un des deux petit. Souvent le chiffrement et le déchiffrement ont lieu avec des ressources computationnelles très differentes (par exemple, la puce dans une carte de crédit contre l'ordinateur central d'une banque).

En 1990, Wiener [**62**] a démontré que pour $d < N^{1/4}$, on peut factoriser N de manière efficace en utilisant e et la connaissance que d est petit. Dix ans plus tard, Boneh et Durfee [**9**] ont démontré, sous certaines hypothèses heuristiques, que la même chose est vraie pour d jusqu'à $N^{0.292}$. Actuellement il est précisément recommandé que d soit supérieur à $N^{1/2}$, et conseillé d'éviter tout choix de petit exposant de déchiffrement.

Par contre, aucune attaque générale n'est connue contre un petit exposant de chiffrement, même contre $e = 3$. Néanmoins, il faut faire attention, comme Håstad [**30**] l'a indiqué, parce qu'il existe une attaque très simple et efficace quand le même message m est envoyé à plusieurs destinataires en utilisant des clef publiques RSA $(N_1, e), \ldots, (N_s, e)$ qui partagent un exposant e commun. (On suppose que les N_i sont premiers entre eux, sinon on peut les factoriser.) Par exemple, en supposant que $e = s = 3$, un adversaire reçoit trois textes chiffrés $c_i \equiv m^3 \bmod N_i$, $i = 1, 2, 3$. En utilisant le *théorème chinois*, il peut

calculer un entier c tel que

$$c \equiv m^3 \bmod N_1N_2N_3 \text{ avec } 0 \leq c < N_1N_2N_3.$$

Puisque $0 < m < N_i$ pour tout $i = 1, 2, 3$, on a aussi

$$0 \leq m^3 < N_1N_2N_3.$$

On conclut que $c = m^3$ et donc pour trouver m il suffit d'extraire la racine cubique de c sur les réels (en notant que le résultat m est un entier). Cette attaque très simple a été généralisée dans plusieurs directions par Håstad [**30**] et étendue par Coppersmith [**12**]. En effet l'attaque de Coppersmith [**12**], au lieu de travailler avec un message chiffré en utilisant plusieurs clefs RSA, s'applique à certains messages liés, chiffrés avec la même clef RSA.

3.3. Une utilisation naïve d'ElGamal. — Comme les cryptosystèmes à clef publique sont plus lents que leurs analogues symétriques, ils s'utilisent typiquement seulement pour l'échange d'une clef secrète (pour un cryptosystème symétrique). Un échange naïf de clef, sans prétraitement de la clef, implique que le texte clair serait un entier assez petit — autour de 80–128 bits pour une clef de taille minimale pour assurer une protection contre des attaques par *force brute*. Boneh, Joux and Nguyen [**10**] montrent que l'utilisation du cryptosystème d'ElGamal pour échanger une telle clef sans prétraitement donne lieu à des résultats désastreux. On donne ici une brève description d'une de leurs attaques.

On rappelle le cadre du cryptosystème d'ElGamal, décrit en section 2.3. Soit G un sous-groupe de $\mathbb{F}_p^*$ d'ordre premier q engendré par g. Un texte chiffré est de la forme $(r, s) = (g^\ell, mh^\ell)$, où $h \in G$. On suppose que $1 \leq m \leq B$, avec B beaucoup plus petit que p — typiquement entre 80 et 128 bits au lieu de 512 à 1024 bits pour p. On choisit aussi des bornes B_1 et B_2 qui sont des paramètres de l'algorithme, qui gérent un compromis entre la complexité de l'algorithme et la probabilité de réussite.

Étape 1 : Calculer $s^q = m^q h^{\ell q} = m^q$.
Étape 2 : Pour $c = 1, \ldots, \lceil B_1 \rceil$ calculer, trier et stocker c^q dans un tableau.
Étape 3 : Pour $d = 1, \ldots, \lceil B_2 \rceil$ calculer $s^q/d^q = (m/d)^q$, et déterminer si cette valeur est dans le tableau de l'étape 2, et sortir (c, d) tel que $m = cd$ en cas d'égalité $(m/d)^q = c^q$.

L'algorithme marche toujours avec $B_1 = 1$ et $B_2 = B$, où il se réduit à une recherche exhaustive par force brute, produisant $c = 1$ et $d = m$. Un choix plus intéressant est $B_1 = B_2 = B^{1/2+\varepsilon}$ pour un petit $\varepsilon > 0$. Pour analyser ce cas, on définit la fonction $H(x, y, z)$, qui est le nombre d'entiers $m \leq x$ tels qu'il existe un entier $d \mid m$ avec $y < d \leq z$. Ford [**22**] donne une suite d'améliorations remarquables des bornes précèdentes sur $H(x, y, z)$. Dans le

cas qui nous intéresse, une version simplifié de sa preuve de [**22**, Theorem 7], pour $0 < \alpha < \beta < 1$ fixés, implique la minoration

$$C(\alpha, \beta)\, x \le H(x, x^\alpha, x^\beta),$$

où $C(\alpha, \beta) > 0$ est une constante qui ne dépend que de α et β. Par conséquent, on voit que l'algorithme réussit pour une proportion positive de messages, puisqu'une suffisamment grande proportion d'entiers positifs aléatoires $m \le B$ admet une représentation $m = cd$ avec

$$B^{1/2-\varepsilon} \le c, d \le B^{1/2+\varepsilon}.$$

En particulier, en prenant $B = 2^{80}$ (comme ci-dessus pour une clef symétrique de niveau de sécurité minimal) on voit que l'attaque se termine en temps un peu supérieur à 2^{40} étapes.

4. La théorie des nombres en coulisses

4.1. Nombres premiers et nombres friables. — Puisque la théorie des nombres joue un rôle majeur dans la plupart des constructions en cryptographie à clef publique, il est naturel que les objets de base en théorie des nombres, les nombres premiers, soient d'intérêt primordial en cryptographie. Même si heuristiquement les propriétés des nombres premiers sont bien comprises (et la plupart du temps ceci est suffisant pour les applications), la plupart des questions théoriques majeures restent complétement ouvertes. L'exemple le plus célèbre d'une telle question ouverte est la célèbre *Hypothèse de Riemann*, qui a des implications directes sur la distribution des nombres premiers et où la dernière avancée majeure est dûe aux travaux indépendants de Korobov [**36**] and Vinogradov [**60**] il y a presque soixante ans. Trouver des grands nombres premiers d'un nombre de bits donné est une tâche très courante dans la construction de plusieurs cryptosystèmes. Comme nous n'avons pas de formules ni de constructions explicites pour les nombres premiers, il faut d'abord décrire un ensemble de taille raisonnable, qui contient certainement un nombre premier. Puis, tous les éléments de cet ensemble peuvent être vérifiés de manière consécutive ou dans un ordre aléatoire (ce qui est raisonnable si l'ensemble contient de façon certaine "beaucoup" de nombres premiers).

On pense généralement que si p_n est le n-ième nombre premier, alors

$$p_{n+1} - p_n \le c(\log p_n)^2$$

pour une constante absolue $c > 0$, qui est une version moins stricte de la fameuse conjecture de Cramér (voir [**26**]). Néanmoins, en ce moment, il n'y a pas d'approches faisables pour démontrer cette conjecture, et le meilleur résultat connu, dû à Baker, Harman et Pintz [**4**], n'affirme que l'inégalité plus faible $p_{n+1} - p_n \le cp_n^{0.525}$. D'un autre côté, on peut simplement choisir des

entiers aléatoires dans un intervalle de la forme $[2^k, 2^{k+1} - 1]$ et tester leur éventuelle primalité avec un test probabiliste de primalité (voir [**14**]). Avec une probabilité écrasante, ceci produit un nombre premier de k bits en un temps polynômial. Tao, Croot et Helfgott [**58**] ont étudié des algorithmes déterministes pour trouver des nombres premiers, mais ces algorithmes sont naturellement plus lents. Du côté positif, il a y environ une décennie, Agrawal, Kayal et Saxena [**1**] ont conclut la quête pour un algorithme déterministe en temps polynômial de preuve de primalité.

Un autre ensemble d'entiers qui a une grande importance en cryptographie est l'ensemble des nombres dit *y-friables*. Ce sont les entiers n tels que tous les diviseurs premiers $p \mid n$ satisfont $p \leq y$. En particulier, le design et l'analyse de plusieurs algorithmes cryptographiques sont basés sur notre capacité à contrôler la fonction $\psi(x, y)$ qui compte le nombre d'entiers $n \leq x$ qui sont y-friables. On renvoie aux articles de Granville [**27**] et de Hildebrand et Tenenbaum [**31**] pour des résultats précis et des conjectures sur $\psi(x, y)$. Ici, on énonce seulement, de manière très informelle, que pour une grande quantité de paramètres x et y, on a

$$\psi(x, y) \approx xu^{-u} \text{ où } u = \frac{\log x}{\log y}. \tag{2}$$

On ne donne pas de définition précise de $\approx$ dans cette équation, on entend juste donner une intuition du comportement de cette fonction.

4.2. Calcul d'indice. — On explique une version simplifiée de l'algorithme d'Odlyzko [**50**]. On suppose fixés un nombre premier p et des éléments g et h de $G = \langle g \rangle \subseteq \mathbb{F}_p^*$, avec G d'ordre q, et on note $\log_g h$ le logarithme discret de h en base g.

On fixe une borne y, que l'on optimisera plus tard, et on suppose tout d'abord que l'on connaît les logarithmes discrets

$$\log_g p_1, \ldots, \log_g p_s,$$

pour tous les nombres premiers $p_1, \ldots, p_s$ inférieurs à y. Sous cette hypothèse, que l'on justifiera par la suite, on poursuit la stratégie suivante.

Étape 1 : Choisir un entier aléatoire m et calculer

$$v \equiv hg^m \equiv g^{k+m} \bmod p,$$

en remarquant que $\log_g v = \log_g h + \log_g(g^m) = \log_g h + m$.

Étape 2 : En supposant que v se relève en un entier y-friable, on essaie de trouver une factorisation $v = p_1^{\alpha_1} \ldots p_s^{\alpha_s}$, par essais de division (force brute). En cas de réussite, alors

$$\log_g v = \alpha_1 \log_g p_1 + \ldots + \alpha_s \log_g p_s \bmod q.$$

et on donne comme résultat $\log_g h = \alpha_1 \log_g p_1 + \ldots + \alpha_s \log_g p_s - m \bmod q$. Si non, on retourne à l'*Étape 1.*

Maintenant, on voit que le coût de l'étape 1 est négligeable. Le coût de l'étape 2 est d'environ y opérations avec des nombres de taille inférieure à p, et soit on trouve la représentation, soit on décide qu'elle n'existe pas. Par conséquent, par (2), on trouve une espérance de u^u pour le nombre de repétitions de ces deux étapes, où

$$u = \log p / \log y,$$

sous l'hypothèse que v est un entier aléatoire de taille inférieure à p. Le coût total est donc d'environ yu^u. En prenant $y = \exp\left(\sqrt{\log p \log\log p}\right)$ on obtient un algorithme de complexité autour de $\exp\left(2\sqrt{\log p \log\log p}\right)$.

Il est certainement trop tôt pour annoncer le succès, parce qu'on a besoin de justifier l'hypothèse initiale que les logarithmes discrets $\log_g p_i$ sont connus. Pour le faire, on applique le même algorithme à l'ensemble entier de générateurs $\{p_1, \ldots, p_s\}$ au lieu d'un seul élément h. On s'arrête après avoir obtenu suffisamment de relations linéaires

$$\alpha_{1,i} \log_g p_1 + \ldots + \alpha_{s,i} \log_g p_s - m_i \equiv 0 \bmod q.$$

en s variables telles qu'il en existe s linéairement indépendantes, que l'on résout avec environ $s^3 \leq y^3$ opérations arithmétiques. Alors, notre hypothèse initiale est satisfaite.

En faisant attention à maintenir un équilibre entre la phase de collection de relations et la phase d'algébre linéaire, cela nous donne un algorithme ayant une complexité finale *sous-exponentielle*, d'environ

$$\exp\left(c\sqrt{(\log p)(\log\log p)}\right),$$

pour une constante c.

L'approche ci-dessus peut être, et a été, améliorée de plusieurs façons, et a été rigourousement analysée. En particulier, la phase de génération de relations doit être modifiée pour produire de petits éléments v, pour optimiser la probabilité de trouver des entiers friables. Aujourd'hui, le plus rapide de ces algorithmes de calcul d'indices est le *crible de corps de nombres* [**38**] (voir aussi [**14**]), qui atteint une complexité bornée par

$$\exp\left(c(\log p)^{1/3}(\log\log p)^{2/3}\right),$$

pour une constante c.

Jusqu'à récemment, la même construction en petite caractéristique a fourni une alternative possible au protocoles de Diffie-Hellman et ElGamal, en remplaçant $\mathbb{F}_p$ par le corps $\mathbb{F}_{\ell^n}$, où ℓ est un petit nombre premier et $n \approx \log_\ell(p)$. Un algorithme de calcul d'indices analogue s'applique, avec une complexité

similaire. Cette attaque remplace l'anneau des entiers $\mathbb{Z}$ par l'anneau des polynômes $\mathbb{F}_\ell[x]$ sur le corps de base $\mathbb{F}_\ell$, en suivant les grandes lignes ci-dessus, se généralise au *crible de corps de fonctions*. A partir de 2012, une nouvelle construction de Joux [**34**] a donné naissance à une amélioration spectaculaire du crible de corps de fonctions. Après une succession rapide d'améliorations et de records de calcul, cela a abouti à l'algorithme quasi-polynômial en temps de Barbulescu, Gaudry, Joux, and Thomé [**6**].

Cet exemple illustre le fait qu'un cryptosystème à clef publique, basé sur un problème computationnel perçu comme difficile, n'est pas plus sûr que la meilleure attaque sur le problème sous-jacent. Sa sécurité présumée peut donc changer compte-tenu d'un nouvel algorithme, ou, comme évoqué en Section 5, un nouveau modèle computationnel. Cela souligne l'importance de la recherche active sur les problèmes sous-jacents et sur les algorithmes pour les résoudre afin de mettre à jour notre connaissance des attaques éventuelles.

4.3. Courbes elliptiques. — L'analogie entre les groupes $\mathbb{F}_p^*$ et $E(\mathbb{F}_p)$ pour une courbe elliptique E sur $\mathbb{F}_p$ va plus loin. Le groupe multiplicatif $\mathbb{F}_p^*$ s'identifie aux points sur $\mathbb{F}_p$ de la courbe affine $\mathbb{G}_m$ d'équation $uv = 1$, i.e.

$$\mathbb{G}_m(\mathbb{F}_p) = \{(u, v) \in \mathbb{F}_p^2 \ : \ uv = 1\} \cong \mathbb{F}_p^*.$$

Une courbe elliptique est une cubique projective non singulière. Or dans la famille des cubiques $y^2 = f(x)$, il y a aussi des courbes singulières, comme

$$C : y^2 = x^3 + x^2.$$

Par restriction au lieu non singulier $C_{ns} = C\backslash\{(0, 0)\}$, la structure de groupe usuelle devient la *variété groupe* qui est isomorphe à $\mathbb{G}_m$. L'isomorphisme est donné par la formule explicite

$$(x, y) \longmapsto \left(\frac{y - x}{y + x}, \frac{y + x}{y - x}\right).$$

Pour comprendre la distinction entre $\mathbb{G}_m$ et la courbe elliptique E, notons que $\mathbb{G}_m$ est une courbe plane affine de genre 0, alors que E est une courbe projective complète (compacte) de genre 1. Par conséquent, comme nous l'avons vu, le groupe multiplicatif $\mathbb{G}_m(\mathbb{Q}) = \mathbb{Q}^*$ vient avec une infinité de générateurs (les nombres premiers dans $\mathbb{Z}_{>0} \subset \mathbb{Q}^*$). D'autre part, d'après le théorème de Mordell [**48**] de 1922, pour toute courbe elliptique, le groupe $E(\mathbb{Q})$ des points rationnels est de type fini. Ce groupe, connu de nos jours sous le nom de groupe de Mordell-Weil, est le produit du groupe fini $E(\mathbb{Q})_{\text{tors}}$ et d'un groupe abélien libre de rang r,

$$E(\mathbb{Q}) \cong \mathbb{Z}^r \times E(\mathbb{Q})_{\text{tors}}.$$

De plus, Mazur [**43**, Theorem (8)] a montré que $E(\mathbb{Q})_{\text{tors}}$ est isomorphe à l'un des items d'un catalogue de 15 groupes d'ordres inférieurs à 12. On ignore

actuellement si les rangs des courbes elliptiques sur $\mathbb{Q}$ sont bornés. Le plus grand rang connu, pour tout sous-groupe de points d'une courbe elliptique sur $\mathbb{Q}$, est 28. Jusqu'à présent, les tentatives d'attaques utilisant la structure du groupe de Mordell-Weil, par analogie avec le rôle des nombres premiers dans les calculs d'indices, ont échoué **[57, 33]**, en raison de la faible nombre de générateurs (ou de points de petite hauteur) dans les groupes de Mordell-Weil.

Notre compréhension de l'arithmétique des courbes sur $\mathbb{Q}$ et sur les corps de nombres se fonde sur la théorie des courbes sur les corps finis. La fonction zeta (locale) $Z(C,t)$ d'une courbe C sur $\mathbb{F}_p$

$$Z(C,t) = \exp\left(\sum_{n=1}^{\infty} \frac{|C(\mathbb{F}_{p^n})|}{n} t^n\right),$$

est une fonction génératrice logarithmique qui code les nombres de points $|C(\mathbb{F}_{p^n})|$ sur toutes les extensions finies de $\mathbb{F}_p$. La fonction zeta d'une courbe elliptique $E/\mathbb{F}_p$ prend la forme

$$Z(E,t) = \frac{1 - a_p t + t^2}{(1-t)(1-pt)},$$

où $a_p = p + 1 - |E(\mathbb{F}_p)|$, et la fonction zeta de la droite projective $\mathbb{P}^1$ est

$$Z(\mathbb{P}^1,t) = \frac{1}{(1-t)(1-pt)}.$$

Ce sont des cas particuliers que Weil a généralisés dans ses conjectures célèbres.

Si E est une courbe elliptique sur $\mathbb{Q}$, et E_p sa réduction à $\mathbb{F}_p$, posons

$$L_p(E,s) = \frac{Z(E_p, p^{-s})}{Z(\mathbb{P}^1, p^{-s})} = 1 - a_p p^{-s} + p^{2-s},$$

et définissons la fonction L de E comme suit

$$L(E,s) = \prod L_p(E,s)^{-1}.$$

Après de vastes expériences numériques, Birch et Swinnerton-Dyer **[8]** ont formulé des conjectures remarquables sur le comportement de cette fonction L globale. Elles affirment que l'ordre de son pôle en $s = 1$ est égal au rang r, et donnent une description précise du résidu d'ordre r de $L(E,s)$ en 1. La résolution par Wiles **[63]** (et Taylor et Wiles **[59]**) du dernier théorème de Fermat (plus précisément, de la conjecture de Taniyama-Shimura) établit que $L(E,s)$ provient d'une forme modulaire. Néanmoins, on n'a pu démontrer jusqu'à présent que des cas particuliers de la conjecture de Birch et Swinnerton-Dyer, et même l'équivalence entre l'ordre du pôle (appelé *rang analytique*) et le rang

de Mordell-Weil reste ouverte. Il reste à voir si les mathématiques développées pour résoudre cette question notoire auront un impact sur les humbles applications cryptographiques des courbes elliptiques sur les corps finis.

5. Perspectives

Après ce tour d'horizon des principales constructions arithmétiques en cryptographie à clef publique à la fin du XXème siècle, on peut se demander quelles sont les chances de survie du sujet au delà de la première moitié du XXIème siècle. Si on croit au progrès constant des ordinateurs et des algorithmes, les systèmes à base de courbes elliptiques devraient supplanter les systèmes classiques, RSA et ElGamal. Pour battre des ordinateurs et des algorithmes toujours plus puissants, il suffit d'allonger les clefs. Chaque allongement linéaire améliore la sécurité de manière sous-exponentielle, dans le cas de RSA, et exponentielle, pour les systèmes à courbes elliptiques. Le cryptographe garde toujours l'avantage, mais, pour un même niveau de sécurité, les courbes elliptiques sont plus efficaces, et cela de plus en plus.

L'arrivée des ordinateurs quantiques change la donne. Manin [**41**, pp. 12–15] et Feynman [**21**] ont vu des signes annonçant que l'ordinateur quantique pourrait être plus puissant que la machine de Turing. Shor [**56**] a trouvé des algorithmes en temps polynômiaux pour la factorisation des entiers et le problème du logarithme discret sur un ordinateur quantique hypothétique (il s'agit d'un modèle de complexité bien défini par Deutsch [**16**]). Ce dernier algorithme s'applique aussi au problème du logarithme discret sur les courbes elliptiques (voir Proos and Zalka [**52**]). Dans le monde post-quantique, RSA ainsi que la cryptographie à courbes elliptiques s'effondreront, alors qu'on ne connaît pas d'attaque, dans le modèle de complexité quantique, contre la cryptographie à base de réseaux (ainsi que certains systèmes utilisant des codes correcteurs d'erreurs, entre autres, voir Bernstein [**7**]). Attendons de voir si un ordinateur quantique fiable sera réalisé de notre vivant, et quelles en seront les conséquences pratiques.

Références

[1] M. Agrawal, N. Kayal et N. Saxena, 'PRIMES is in P', *Ann. Math.*, **160** (2004), 781–793.

[2] M. Ajtai et C. Dwork, 'A public-key cryptosystem with worst-case/average-case equivalence', *Proc. 29th ACM Symp. Theory Comp.*, ACM (1997), 284–293.

[3] E. Artin, 'Quadratische Körper im Gebiete der höheren Kongruenzen. II. Analytischer Teil', *Math. Zeitschrift*, **19** (1924), 207–246.

[4] R. C. Baker, G. Harman et J. Pintz, 'The difference between consecutive primes. II', *Proc. Lond. Math. Soc.*, **83** (2001), 532–562.

[5] R. Balasubramanian et N. Koblitz, 'The improbability that an elliptic curve has subexponential discrete log problem under the Menezes-Okamoto-Vanstone algorithm', *J. Crypto.*, **11** (1998), 141–145.

[6] R. Barbulescu, P. Gaudry, A. Joux, E. Thomé, 'A quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic', *Proc. Eurocrypt'14*, *Lect. Notes Comput. Sci.*, **8441**, Springer-Verlag (2014), 1–16.

[7] D. J. Bernstein, 'Introduction to post-quantum cryptography', *Post-quantum cryptography*, Springer (2009), 1–14.

[8] B. Birch et P. Swinnerton-Dyer, 'Notes on Elliptic Curves (II)' *J. Reine Angew. Math.*, **65** (1965), 79–108.

[9] D. Boneh et G. Durfee, 'Cryptanalysis of RSA with private key d less than $N^{0.292}$', *IEEE Trans. Inf. Theory*, **46** (2000), 1339–1349.

[10] D. Boneh, A. Joux et P. Q. Nguyen, 'Why textbook ElGamal and RSA encryption are insecure', *Proc. Asiacrypt'00*, *Lect. Notes Comp. Sci.*, **1976**, Springer-Verlag (2000), 30–43.

[11] Z. Brakerski, A. Langlois, C. Peikert, O. Regev et D. Stehlé, 'Classical hardness of learning with errors', *Proc. 45th Symp. Theory Comp.*, ACM (2013), 575–584.

[12] D. Coppersmith, 'Small solutions to polynomial equations, and low exponent RSA vulnerabilities', *J. Crypto.*, **10** (1997), 233–260.

[13] R. Cramer et V. Shoup, 'Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack', *SIAM J. Comp.*, **33** (2003), 167–226.

[14] R. Crandall et C. Pomerance, *Prime numbers*: *A computational perspective*, Springer-Verlag, 2005.

[15] P. Deligne, "La conjecture de Weil. I", *Publ. Math. IHES*, **43** (1974), 273–307.

[16] D. Deutsch, 'Quantum theory, the Church-Turing principle and the universal quantum computer', *Proc. Royal Soc. London A*, **400** (1985), 97–117.

[17] C. Diem, 'On the discrete logarithm problem in elliptic curves', *Compos. Math.*, **147** (2011), 75–104.

[18] C. Diem, 'On the discrete logarithm problem in elliptic curves, II', *Algebra and Number Theory*, **7** (2013), 1281–1323.

[19] W. Diffie et M. E. Hellman, 'New directions in cryptography', *IEEE Trans. Inf. Theory*, **22** (1976), 644–654.

[20] T. Elgamal, 'A public-key cryptosystem and a signature scheme based on discrete logarithms', *Proc. Crypto'84*, *Lect. Notes Comp. Sci.*, **196**, Springer-Verlag (1985), 10–18.

[21] R. P. Feynman, 'Simulating physics with computers', *Int. J. Theor. Physics*, **21** (1982), 467–488.

[22] K. Ford, 'The distribution of integers with a divisor in a given interval', *Ann. Math.*, **168** (2008), 367–433.

[23] E. Fujisaki, T. Okamoto, D. Pointcheval et J. Stern, 'RSA-OAEP is secure under the RSA assumption', *J. Crypto.*, **17** (2004), 81–104.

[24] P. Gaudry, F. Hess, et N. Smart, 'Constructive and destructive aspects of Weil descent', *J. Crypto.*, **15** (2002), 19–46.

[25] O. Goldreich, S. Goldwasser et S. Halevi, 'Public-key cryptosystems from lattice reduction problems', *Lect. Notes Comp. Sci.*, **1294**, Springer-Verlag (1997), 112–131.

[26] A. Granville, 'Harald Cramér and the distribution of prime numbers', *Harald Cramér Symposium (Stockholm, 1993)*, *Scand. Actuar. J.*, **1** (1995), 12–28.

[27] A. Granville, 'Smooth numbers: Computational number theory and beyond', *Algorithmic Number Theory*: *Lattices, Number Fields, Curves, and Cryptography*, Cambridge University Press (2008), 267–322.

[28] A. Grothendieck, 'The cohomology theory of abstract algebraic varieties', *Proc. Int. Congress Math. (Edinburgh 1958)*, Cambridge University Press, 1960, 103–118.

[29] H. Hasse, 'Zur Theorie der abstrakten elliptischen Funktionenkörper. I, II & III", *J. Reine Angew. Math.*, **175** (1936), I, 55–62, II, 69–88, III, 193–208.

[30] J. Håstad, 'Solving simultaneous modular equations of low degree', *SIAM J. Comp.*, **17** (1988), 336–341.

[31] A. Hildebrand et G. Tenenbaum, 'Integers without large prime factors', *J. Théor. Nombres Bordeaux*, **5** (1993), 411–484.

[32] J. Hoffstein, J. Pipher et J. H. Silverman, 'NTRU: A ring based public key cryptosystem', *Lect. Notes Comp. Sci.*, **1433**, Springer-Verlag (1998), 267–288.

[33] M. J. Jacobson, N. Koblitz, J. H. Silverman, A. Stein, E. Teske, 'Analysis of the Xedni Calculus Attack', *Designs, Codes and Cryptography*, **20** (1999), 41–64.

[34] A. Joux, 'A new index calculus algorithm with complexity $L(1/4 + o(1))$ in very small characteristic', Cryptology ePrint Archive, 2013/095 (2013).

[35] N. Koblitz, 'Elliptic curve cryptosystems', *Math. Comp.*, **48** (1987), 203–209.

[36] N. M. Korobov, 'Estimates of trigonometric sums and their applications' (russe), *Uspehi Mat. Nauk*, **13** (1958), 185–192.

[37] A. K. Lenstra, H. W. Lenstra et L. Lovász, 'Factoring polynomials with rational coefficients', *Math. Annalen*, **261** (1982), 515–534.

[38] A. K. Lenstra et H. W. Lenstra, Jr. (eds.), *The Development of the Number Field Sieve*, *Lect. Notes Math.*, **1554**, Springer-Verlag, 1993.

[39] V. Lyubashevsky, C. Peikert et O. Regev, 'On ideal lattices and learning with errors over rings', *J. ACM*, **60** (2013), Article 43.

[40] V. Lyubashevsky, C. Peikert et O. Regev, 'A toolkit for ring-LWE cryptography', *Proc. Eurocrypt'13*, *Lect. Notes Comp. Sci.*, **7881**, Springer-Verlag (2013), 35–54.

[41] Yu. I. Manin, 'The computable and the non-computable' (russe : 'Vychislimoe i nevychislimoe'), Sovetskoe Radio, 1980.

[42] J. Maynard, 'Small gaps between primes', *preprint*, 2013, `http://arxiv.org/abs/1311.4600/`.

[43] B. Mazur, 'Modular curves and the eisenstein ideal', *Publ. Math. IHES*, **47** (1977), 33–186

[44] A. Menezes, T. Okamoto and S. A. Vanstone, 'Reducing elliptic curve logarithms to logarithms in a finite field', *IEEE Trans. Inf. Theory*, **39** (1993), 1639–1646.

[45] R. C. Merkle et M. E. Hellman, 'Hiding information and signatures in trapdoor knapsacks', *IEEE Trans. Inf. Theory*, **24** (1978), 525–530.

[46] D. Micciancio et O. Regev, 'Lattice-based cryptography', *Post-Quantum Cryptography*, Springer-Verlag (2009), 147–191.

[47] V. S. Miller, 'Uses of elliptic curves in cryptography', *Lect. Notes Comp. Sci.*, **218**, Springer-Verlag (1986), 417–426.

[48] L. J. Mordell, 'On the rational solutions of the indeterminate equations of the third and fourth degrees', *Proc. Cam. Phil. Soc.*, **21** (1922), 179–192.

[49] P. Q. Nguyen, 'Public-key cryptanalysis', *Recent Trends in Cryptography*, *Contemp. Math.*, **477**, Amer. Math. Soc. (2009), 67–120.

[50] A. M. Odlyzko, 'Discrete logarithms in finite fields and their cryptographic significance', *Proc. Eurocrypt'84*, *Lect. Notes Comp. Sci.*, **209**, Springer-Verlag (1985), 224–314.

[51] A. M. Odlyzko, 'The rise and fall of knapsack cryptosystems', *Cryptology and Computational Number Theory*, *Proc. Symp. in Appl. Math.*, **42**, Amer. Math. Soc. (1990), 75–88.

[52] J. Proos et C. Zalka, 'Shor's discrete logarithm quantum algorithm for elliptic curves', *Quantum Information & Computation*, **3** (2003), 317–344.

[53] O. Regev, 'On lattices, learning with errors, random linear codes, and cryptography', *J. ACM*, **56** (2009), Article 34.

[54] R. Rivest, A. Shamir et L. M. Adleman, 'A method for obtaining digital signatures and public-key cryptosystems', *Commun. ACM*, **21** (1978), 120–126.

[55] R. Schoof, 'Elliptic curves over finite fields and the computation of square roots mod p.' *Math. Comp.*, **44** (1985), 483–494.

[56] P. W. Shor, 'Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer.', *SIAM J. Comp.*, **26** (1997), 1484–1509.

[57] J. H. Silverman, 'The Xedni Calculus And The Elliptic Curve Discrete Logarithm Problem', *Designs, Codes and Cryptography*, **20** (1999), 5–40.

[58] T. Tao, E. Croot III et H. Helfgott, 'Deterministic methods to find primes', *Math. Comp.* **81** (2012), 1233–1246.

[59] R. Taylor et A. Wiles, 'Ring-theoretic properties of certain Hecke algebras', *Ann. Math.*, **141** (1995), 553–572.

[60] I. M. Vinogradov, 'A new estimate for $\zeta(1 + it)$' (russe), *Izv. Akad. Nauk SSSR, Ser. Mat.*, **22** (1958), 161–164.

[61] A. Weil, 'Numbers of solutions of equations in finite fields', *Bull. Amer. Math. Soc.*, **55** (1949), 497–508.

[62] M. J. Wiener, 'Cryptanalysis of short RSA secret exponents', *IEEE Trans. Inform. Theory*, **36** (1990), 553–558.

[63] A. Wiles, 'Modular elliptic curves and Fermat's Last Theorem', *Ann. Math.*, **141** (1995), 443–551.

[64] Y. Zhang, 'Bounded gaps between primes', *Ann. Math.*, **179** (2014), 1121–1174.

D. Kohel, Aix Marseille Université, CNRS, Centrale Marseille, I2M, UMR 7373, 13453 Marseille, France • *E-mail :* `david.kohel@univ-amu.fr`

I. E. Shparlinski, Department of Pure Mathematics, University of New South Wales, Sydney, NSW 2052, Australia • *E-mail :* `igor.shparlinski@unsw.edu.au`