Agrégation externe de mathématiques, session 2017
Épreuve de modélisation, option C : Calcul Formel

(*public2018-C2*)

Résumé : On considère quelques attaques contre le cryptosystème RSA.

Mots clefs : arithmétique des entiers

► *Il est rappelé que le jury n'exige pas une compréhension exhaustive du texte. La présentation, bien que totalement libre, doit être organisée et le jury apprécie qu'un plan soit annoncé en préliminaire. L'exposé doit être construit en évitant la paraphrase et mettant en lumière les connaissances, à partir des éléments du texte. Il doit contenir des illustrations informatiques réalisées sur ordinateur, ou, à défaut, des propositions de telles illustrations. Des pistes de réflexion, indicatives et largement indépendantes les unes des autres, vous sont proposées en fin de texte.*

Dans ce texte, on note $\mathbf{N}$ l'ensemble des entiers positifs ou nuls et $\mathbf{Z}$ l'anneau des entiers relatifs. Nous notons $m|n$ si m divise n. Pour tout entier $n > 0$, nous notons $\varphi(n)$ le cardinal du groupe multiplicatif $(\mathbf{Z}/n\mathbf{Z})^*$. On note également $\lfloor x \rfloor$ la partie entière d'un nombre réel x.

1. Cadre et motivation

La méthode de cryptographie la plus utilisée actuellement, est la méthode RSA de Rivest, Shamir et Adleman. Rappelons rapidement son principe. Pour recevoir des messages chiffrés de Bob, Alice choisit deux nombres premiers distincts p et q et un nombre entier e premier avec le produit $\varphi(n) = (p-1)(q-1)$. Elle calcule alors $n = pq$ et d l'inverse de e dans $\mathbf{Z}/\varphi(n)\mathbf{Z}$. Elle transmet alors à Bob sa clef publique, le couple (n, e) et conserve sa clef privée d. Pour chiffrer un message μ, que nous supposerons entier de $[0, n-1]$, et que nous identifierons avec sa classe dans $\mathbf{Z}/n\mathbf{Z}$, Bob calcule $c = \mu^e$. Pour déchiffrer le message c et retrouver μ, Alice n'a plus qu'à calculer c^d dans $\mathbf{Z}/n\mathbf{Z}$.

La sécurité de cette méthode RSA repose

— en ce qui concerne la sécurité d'un message, sur la difficulté d'extraire des racines e-ièmes dans $\mathbf{Z}/n\mathbf{Z}$;

— en ce qui concerne la sécurité de la clef privée, sur la difficulté de trouver les facteurs d'un produit de deux nombres premiers.

Nous étudions dans la suite plusieurs attaques contre le cryptosystème RSA correspondant à de mauvais choix de clés ou des utilisations malencontreuses du système. Nous considérerons qu'une attaque est réaliste si son coût est au plus de l'ordre d'un polynôme en $\log n$.

2. RSA pour la diffusion large de messages

Le coût du chiffrement RSA est directement lié au choix de l'exposant de chiffrement e (plus précisément, à sa taille et au nombre de "1" figurant dans sa décomposition en base 2). La tentation est donc grande de prendre e petit (typiquement $e = 3$) pour faciliter le travail de l'émetteur (nous verrons plus loin que le choix alternatif d petit, pour faciliter le travail du récepteur, est encore plus dangereux).

Supposons que r utilisateurs $A_1, \ldots, A_r$ aient comme clé publique RSA $(n_1, e), \ldots, (n_r, e)$, où nous supposerons les n_i premiers entre eux deux-à-deux. Si un message μ commun leur est envoyé par un expéditeur donné, et qu'on observe les r messages chiffrés $c_1, \ldots, c_r$, il est alors possible de retrouver $\mu^e \bmod n$, où $n := n_1 \cdot n_2 \cdots n_r$.

Proposition 1. *Si $\mu < \min_{1 \leqslant i \leqslant r} n_i$ (c'est l'hypothèse que l'on a fait à l'origine sur l'encodage du message), et si $e \leqslant r$, on peut retrouver μ à partir de $c_1, \ldots, c_r$ en effectuant un nombre d'opérations polynomial en* $\log(n)$.

Démonstration. Comme les n_i ont été supposés premiers entre eux deux-à-deux, on utilise le théorème des restes chinois, puis on calcule une racine e-ème dans $\mathbb{N}$. □

Un choix de e très petit est donc dangereux dans le contexte actuel d'utilisation de plus en plus large de la cryptographie ; tout en essayant, en général, de garder e petit, on tend à se protéger de l'attaque ci-dessus en utilisant des valeurs comme $e = 257$ ou surtout $e = 65537$.

3. Attaques contre RSA par itération

Comme souvent en cryptographie, même si la sécurité de RSA est dans l'absolu très bonne, certains choix de clé sont particulièrement maladroits, et il importe de les connaître pour pouvoir les éviter.

Un exemple particulier de clés faibles sont les e tels que l'ordre de e dans $(\mathbf{Z}/\varphi(n)\mathbf{Z})^*$ est petit. Dans ce cas, notons δ cet ordre : on a donc $e^\delta \equiv 1 \mod \varphi(n)$. Il suffit alors à un pirate observant $c = \mu^e \bmod n$ d'itérer $\delta - 1$ fois l'opération de chiffrement RSA sur c pour retrouver μ.

Si e est choisi au hasard, on peut contrôler la probabilité de cet événement ; nous nous limitons ici au cas plus simple (qui correspond à l'utilisation de RSA en pratique) où $p - 1 = 2p_1$, $q - 1 = 2q_1$, avec p_1 et q_1 premiers que nous supposerons différents de 2. Notons que p_1 et q_1 sont distincts puisque p et q le sont.

Proposition 2. *Sous ces hypothèses, la probabilité qu'un élément tiré au hasard dans $(\mathbf{Z}/\varphi(n)\mathbf{Z})^*$ soit d'ordre $\leqslant \delta$ est majorée par $2\delta^3/\varphi(\varphi(n))$.*

Démonstration. Avec les hypothèses de la proposition, $(\mathbf{Z}/\varphi(n)\mathbf{Z})^*$ est isomorphe au groupe $\mathbf{Z}/2\mathbf{Z} \times \mathbf{Z}/(p_1 - 1)\mathbf{Z} \times \mathbf{Z}/(q_1 - 1)\mathbf{Z}$.

Si l'on compte le nombre de triplets (x_1, x_2, x_3) de $\mathbf{Z}/2\mathbf{Z} \times \mathbf{Z}/(p_1-1)\mathbf{Z} \times \mathbf{Z}/(q_1-1)\mathbf{Z}$ solution de $k(x_1, x_2, x_3) = 0$, on trouve qu'il y en a au plus $2k^2$, avec égalité si k est pair et divise $\operatorname{pgcd}(p_1 - 1, q_1 - 1)$, et le résultat s'ensuit. □

Cette probabilité est en particulier très faible si l'on considère que la puissance de calcul du pirate, ici la plus grande valeur de δ qui lui est accessible, est de l'ordre d'une puissance de $\log n$. Néanmoins, pour se protéger de cette attaque il est préférable de calculer l'ordre de e et de vérifier qu'il est assez grand ; si la factorisation de $\varphi(\varphi(n))$ est connue cela peut se faire au moyen de la remarque suivante :

Proposition 3. *Soit G un groupe, et $x \in G$ un élément de G. L'ordre de x est égal à τ si et seulement si*

— $x^\tau = 1$,

— *et pour tout r premier divisant τ, $x^{\tau/r} \neq 1$.*

4. Partage de module RSA

Une autre faiblesse historique de RSA est la complexité de la génération de clés – il faut arriver à construire de grands nombres premiers, et multiplier de grands entiers. Pour simplifier cette étape, on a pu envisager le déploiement de RSA à large échelle de la manière suivante : une autorité centrale déterminait un couple de nombres premiers (p,q), et transmet à chaque utilisateur U le triplet $(n = p\cdot q, e_U, d_U)$ vérifiant

$$e_U \cdot d_U = 1 \bmod \varphi(n). \tag{1}$$

Notons que le module n est commun à tous les utilisateurs, et que la connaissance du triplet est suffisante pour déchiffrer les messages.

Nous montrons maintenant qu'il est possible pour un utilisateur particulier de déduire de son triplet (n,e,d) la factorisation de n, et donc de déchiffrer tous les messages destinés aux autres utilisateurs. Commençons par noter que cela est aisé dans le cas particulier où $ed-1 = \varphi(n)$. En effet, dans un tel cas, on connaît à la fois $n = pq$ et $\varphi(n) = (p-1)(q-1)$, on peut donc calculer $p+q$ et lorsque l'on connaît pq et $p+q$ on peut en déduire p et q.

Dans le cas général, la détermination de la factorisation de n repose d'abord sur :

Lemme 1. *Soit $x \in \mathbf{Z}/n\mathbf{Z}$ tel que $x \neq \pm 1 \bmod n$ et $x^2 = 1 \bmod n$. Alors* $\operatorname{pgcd}(x-1,n)$ *est un diviseur non trivial de l'entier n.*

Écrivons maintenant $\omega := ed-1$ sous la forme $2^s \cdot t$, avec t impair. Soit $y \in \mathbf{Z}/n\mathbf{Z} - \{0\}$ choisi de façon aléatoire. On calcule $\operatorname{pgcd}(y,n)$; si par hasard ce n'est pas 1, on a terminé. Dans le cas contraire, $y \in (\mathbf{Z}/n\mathbf{Z})^*$.

Calculons maintenant $u = y^t \bmod n$. Si $u = 1$, on tire un nouveau y. Sinon, il existe un $1 \leqslant j \leqslant s$ minimal tel que $u^{2^j} = 1 \bmod n$, et si $u^{2^{j-1}} \neq -1 \bmod n$, le lemme précédent nous dit qu'on a factorisé n.

Évaluons maintenant nos chances de réussite.

Lemme 2. *La probabilité que $u = 1 \bmod n$ est au plus* $1/4$.

Démonstration. Soit $\epsilon_1 \in \mathbf{Z}/n\mathbf{Z}$ tel que $\epsilon_1 = 1 \bmod p$, $\epsilon_1 = -1 \bmod q$. Si maintenant y est tel que $y^t = 1 \bmod n$, on vérifie que ce n'est pas le cas de $\epsilon_1 y, -y, -\epsilon_1 y$. □

Une idée analogue permet d'étudier la probabilité que $u^{2^{j-1}} = -1 \bmod n$:

Proposition 4. *La probabilité que* $u^{2^{j-1}} = -1 \bmod n$ *est au plus* $1/2$.

Démonstration. Supposons que $u^{2^{j-1}} = -1 \bmod n$. On pose $p-1 = 2^{k_1} t_1$, et $q-1 = 2^{k_2} t_2$, avec t_1, t_2 impairs. Supposons, sans perte de généralité, que $k_2 \leqslant k_1$; observons qu'alors $j \leqslant k_1$, et soit $\varepsilon \in (\mathbf{Z}/n\mathbf{Z})^*$ tel que

— $\varepsilon \bmod p$ est d'ordre exactement 2^j dans $(\mathbf{Z}/p\mathbf{Z})^*$;

— $\varepsilon = 1 \bmod q$.

On vérifie alors que $y' = y\varepsilon$ est tel que $u' = y'^t \neq 1 \bmod n$, $u'^{2^j} = 1 \bmod n$, et $u'^{2^{j-1}} \neq \pm 1 \bmod n$. □

En moyenne, en $O(1)$ tirages aléatoires de y, la stratégie décrite ci-dessus fournit donc une factorisation de n, et on vérifie que les calculs sont très réduits – à vrai dire pas notablement plus chers que d'effectuer un chiffrement RSA ! L'idée du partage de module est donc à proscrire absolument.

5. Optimisation du déchiffrement

Nous avons exclu dans la première partie l'optimisation consistant à accélérer le chiffrement en choisissant e petit. On peut, à l'inverse, tenter d'accélérer le déchiffrement en choisissant d petit.

Il se trouve que ce choix est encore plus dangereux, comme nous le voyons maintenant.

5.1. *Étude d'un algorithme de type Euclide*

Étant donné des entiers a, b premiers entre eux avec $b > 0$, nous allons donner un algorithme, lié à l'algorithme d'Euclide étendu, pour trouver tous les couples d'entiers premiers entre eux (u, v) avec $v > 0$ tels que

$$\left|\frac{a}{b} - \frac{u}{v}\right| < \frac{1}{2v^2}. \tag{2}$$

Dans ce qui suit, nous supposons que $0 < a < b$, ce qui est vérifié dans le cas qui nous intéresse.

Nous posons $r_0 = a$, $r_1 = b$ et définissons des familles $(r_0, \ldots, r_{n+1})$ et $(q_1, \ldots, q_n)$ de la façon suivante : on suppose r_{i-1} et r_i définis. Si $r_i = 0$, on pose $n = i-1$, sinon on effectue la division euclidienne de r_{i-1} par r_i, en notant q_i le quotient et r_{i+1} le reste, si bien que $r_{i-1} = q_i r_i + r_{i+1}$. On pose alors $u_1 = v_0 = 1$, $u_0 = v_1 = 0$ et, pour $i \in \{1, \ldots, n\}$,

$$u_{i+1} = q_i u_i + u_{i-1} \quad \text{et} \quad v_{i+1} = q_i v_i + v_{i-1}.$$

On démontre aisément que

$$r_i = (-1)^i (a v_i - b u_i) \tag{3}$$

pour $i \in \{0, \ldots, n+1\}$. Soit $i \in \{0, \ldots, n\}$; on a

$$\begin{pmatrix} v_i & v_{i+1} \\ u_i & u_{i+1} \end{pmatrix} = \prod_{j=1}^{i} \begin{pmatrix} 0 & 1 \\ 1 & q_j \end{pmatrix}.$$

Il en résulte que $v_i u_{i+1} - v_{i+1} u_i = (-1)^i$. De la relation (3), on déduit alors, pour $i \in \{2,\dots,n\}$, que $u_i/v_i < a/b$ si i est pair et $a/b < u_i/v_i$ dans le cas contraire. En particulier,

$$\left|\frac{a}{b} - \frac{u_i}{v_i}\right| \leqslant \frac{1}{v_i v_{i+1}}.$$

Proposition 5. *Soient u et v des entiers premiers entre eux avec $v > 0$ qui vérifient la condition* (2)*. Alors il existe $i \in \{1,\dots,n+1\}$ tel que $u = u_i$ et $v = v_i$.*

Esquisse de la preuve. Comme $a/b = u_{n+1}/v_{n+1}$, on peut supposer que $u/v \neq a/b$. Alors $|u/v - a/b| \geqslant 1/(vb)$, donc $v < b/2$. Comme la famille $(v_1,\dots,v_{n+1})$ est croissante, on peut choisir un entier m appartenant à $\{1,\dots,n\}$ tel que $v_m \leqslant v < v_{m+1}$. Si $u/v - a/b$ et $u_m/v_m - a/b$ sont de même signe, on a la relation

$$\left|\frac{uv_m - vu_m}{vv_m}\right| < \max\left(\frac{1}{2v^2}, \frac{1}{v_m v_{m+1}}\right)$$

ce qui entraîne que $u = u_m$ et $v = v_m$. Dans le cas contraire, on a la relation

$$\left|\frac{uv_m - vu_m}{vv_m}\right| < \frac{1}{2v^2} + \frac{1}{v_m v_{m+1}}$$

et il en résulte que $v_{m+1} < 2v$. D'un autre côté, pour $m < n$, on a la relation

$$\left|\frac{uv_{m+1} - vu_{m+1}}{vv_{m+1}}\right| < \max\left(\frac{1}{2v^2}, \frac{1}{v_{m+1} v_{m+2}}\right)$$

qui entraîne l'inégalité $2v < v_{m+1}$. Celle-ci vaut également si $m = n$ et on obtient une contradiction. □

5.2. *Retour à RSA*

Supposons maintenant qu'on a les inégalités $p < q < 2p$, $e < n$ et $d < \frac{1}{3}\sqrt[4]{n}$. Par construction, $\varphi(n) \mid ed-1$; soit $k \in \mathbf{N}$ tel que $ed-1 = k\varphi(n)$. Mais, avec les hypothèses faites, $n - 3\sqrt{n} < \varphi(n) < n-1$ et on obtient pour n assez grand les inégalités

$$\left|\frac{e}{n} - \frac{k}{d}\right| < \frac{3k}{d\sqrt{n}} < \frac{1}{2d^2}.$$

En vertu du paragraphe précédent, il s'ensuit que l'on peut alors trouver d et k, et même une factorisation de n à l'aide du paragraphe 4.

Suggestions et pistes de réflexion

► *Les pistes de réflexion suivantes ne sont qu'indicatives et il n'est pas obligatoire de les suivre. Vous pouvez choisir d'étudier, ou non, certains des points proposés, de façon plus ou moins approfondie, mais aussi toute autre question à votre initiative. Vos investigations comporteront une partie traitée sur ordinateur et, si possible, des représentations graphiques de vos résultats. À défaut, si vos illustrations informatiques n'ont pas abouti, il est conseillé d'expliquer ce que vous auriez souhaité mettre en œuvre.*

— Donner une majoration du temps nécessaire pour trouver un facteur d'un nombre entier par une méthode naïve.
— Estimer le coût du chiffrement d'un message par RSA.
— Illustrer la méthode du paragraphe 2 sur un exemple.
— Expérimenter la méthode du paragraphe 4 sur un exemple.
— Expliquer les choix $e = 3, 257, 65537$.
— Compléter les preuves d'assertions du texte.
— Comment déterminer l'ordre d'un élément d'un groupe si la factorisation du cardinal du groupe n'est pas connue ?
— Pour de petites valeurs de p, q, expérimenter autour de la proposition 2 ; la borne a-t-elle l'air fine ?