Cryptographie

Préparation agrégation, option C

Révision du 03/06

Ce texte présente quelques mises en oeuvre de la méthode de cryptographie RSA et de variantes ainsi que quelques attaques. Une bonne partie de ce texte est un condensé de
`http://iml.univ-mrs.fr/~rolland/rr/lycees/concret1.pdf` (Robert Rolland)

1 RSA

Du nom de ses inventeurs Ron Rivest, Adi Shamir et Len Adleman.

1.1 Principe

Soit $n = pq$ le produit de deux nombres premiers, alors le groupe multiplicatif des inversibles de $\mathbb{Z}/n\mathbb{Z}$ possède $\phi(n) = (p-1)(q-1)$ éléments, donc si a est premier avec n et d et e sont inverses l'un de l'autre dans $\mathbb{Z}/\phi(n)Z$, on a

$$(a^d)^e \pmod n = a^{de} \pmod n = a^{1+\alpha\phi(n)} \pmod n = a \pmod n$$

On observe que cette identité reste encore vraie si a n'est pas premier avec n. En conséquence on peut retrouver $a \pmod n$ connaissant $b = a^d \pmod n$ en calculant $b^e \pmod n$. La fonction de codage consiste à transformer le texte en une suite de nombres modulo n, à calculer les puissances d-ièmes de ces nombres modulo n, le décodage à calculer les puissances e-ièmes de ces nombres modulo n (ou inversement, car d et e jouent un role symétrique).

L'intérêt de RSA est que n et e peuvent être publié sans que d ne soit calculable en un temps raisonnable, pourvu que n soit assez grand (par exemple 1024 bits), parce qu'on ne connait pas d'algorithme de factorisation efficace. On peut donc signer un message en utilisant sa clef privée d et en transmettant un entier a dépendant du message codé en a^d. N'importe qui pourra décoder la signature car e est public. On peut transmettre un message destiné à un seul destinataire en le codant avec la clef publique du destinataire ($a \to a^e$), seul celui-ci sera en mesure de le décoder avec sa clef privée d.

1.2 Attaques triviales

Si le message transmis m est petit et si e est petit (par exemple $e = 3$) alors il peut se produire que $m^e < n$, on peut alors calculer m par extraction de racine cubique. Si les messages possibles m sont dans un ensemble de cardinal petit (par exemple si on prend m le code ASCII des caractères du message), on peut construire l'ensemble des valeurs possibles de m^e et décrypter le texte transmis en comparant le texte chiffré à cette table. En pratique, il faut donc disposer d'un nombre important de valeurs de message à transmettre (on peut par exemple grouper plusieurs lettres, en rajoutant des espaces au besoin, et utiliser la représentation en base 256), rendre impossible la transmission de 0 et 1, éviter les m ayant un facteur commun avec n, ne pas choisir une clef publique (ni privée !) trop petite.

1.3 Attaque par fraction continue

Fonctionne si la clef privée d est petite et si p et q sont du même ordre de grandeur :

$$p > q, \quad p < 2q$$

Le principe consiste à calculer les réduites de e/n. Soit $k \in]0, d[$ tel que

$$ed = 1 + k\phi(n) = 1 + k(n + 1 - p - q) = kn + 1 + k(1 - p - q)$$

On divise par dn :

$$\frac{e}{n} = \frac{k}{d} + \frac{1 + k(1 - p - q)}{dn}$$

donc :

$$\begin{aligned} |\frac{e}{n} - \frac{k}{d}| &= \frac{k(p + q - 1) - 1}{dn} \\ &\leq \frac{k(p + q)}{nd} \\ &\leq \frac{kq(\frac{p}{q} + 1)}{nd} \\ &\leq \frac{3kq}{nd} \\ &\leq \frac{3k}{\sqrt{n}d} \\ &< \frac{3}{\sqrt{n}} \end{aligned}$$

Si $3/\sqrt{n} < 1/(2d^2)$, alors les résultats connus sur les fractions continues permettent de conclure que k/d est une réduite de e/n. Il suffit de les calculer et voir si on a $m^{de} = m \pmod{n}$.

2 Signature avec ombre des cartes bancaires

2.1 Principe

On choisit p et q tels que $\phi(n)$ ne soit pas divisible par 3. On pose par exemple $e = 3$. Le numéro de carte est un nombre I codé sur k bits, on construit le nombre $J = I2^k + I$. J est signé par le calcul de $A = J^d \pmod n$. On stocke sur la carte bancaire I et A. Lorsqu'on saisit le code secret sur un terminal, la carte transmet au terminal les valeurs de I et A. Le terminal calcule alors J et $A^3 \pmod n$ et vérifie que $J = A^3$. Une personne désirant créer une carte bancaire pirate devra donc fournir I et A tel que $A^3 = (I + 2^k I) \pmod n$. On peut bien sur calculer le cube d'un nombre modulo n mais la probabilité de tomber sur un nombre ombré est infime (on peut aussi construire un nombre ombré mais le calcul de sa racine cubique modulo n est impraticable).

2.2 Attaque : calcul de n s'il n'est pas publié.

Si on connait deux couples I_1, A_1 et I_2, A_2, alors `gcd(A1^3-J1,A2^3-J2)` est un multiple kn de n avec k très probablement petit. En divisant par les petits premiers ce pgcd, on trouve très probablement n.

3 Échange de clefs de Diffie-Hellman

Le principe : on se donne un premier p, et un élément primitif α de $\mathbb{Z}/p\mathbb{Z}^*$. Chacun des deux interlocuteurs qui veulent se mettre d'accord sur une clef choisit secrètement un entier n_1, n_2 inférieur à l'ordre du groupe. Ils envoient alors à l'autre interlocuteur $\alpha^{n_1} \pmod p$ ($\alpha^{n_2} \pmod p$). La clef commune est déduite de $\alpha^{n_1 n_2} \pmod p$.

Ce principe nécessite de pouvoir construire un élément primitif a de $\mathbb{Z}/p\mathbb{Z}^*$. Il faut donc tester que $a^{(p-1)/d} \neq 1 \pmod p$ pour tout diviseur premier d de $p-1$. En pratique, on ne sait pas forcément factoriser $p-1$, on inverse donc le problème, on part d'un grand nombre premier q et on construit les entiers $2kq + 1$ pour k suffisamment petit pour être factorisé. On s'arrête dès que $2kq + 1$ est premier. Le théorème de Dirichlet sur la densité des nombres premiers permet de montrer qu'il faut en moyenne $\ln(p)$ essais avant succès, où p est la taille du premier souhaité de cette forme.

4 Le cryptosystème ElGamal

(D'après `wikipedia.fr`). Cet algorithme tire son nom de son inventeur Taher Elgamal. L'algorithme fonctionne comme suit :

- Alice calcule $h = g^x$, $g \in \mathbb{Z}_p$ pour un grand nombre premier p, et divulgue sa clé publique (p, g, h). La valeur x est sa clé privée.
- Si Bob veut envoyer un message à Alice, il convertit d'abord son message sous la forme d'un nombre $m \in \mathbb{Z}_p$.
- Bob génère un nombre entier k aléatoirement et calcule $c1 = g^k$ et $c_2 = m \cdot h^k$. Il envoie $(c1, c2)$ à Alice.

– Alice peut reconstruire le message initial m en calculant c_2/c_1^x. On remarque en effet que :

$$\frac{c_2}{c_1^x} = \frac{m \cdot h^k}{g^{xk}} = \frac{m \cdot g^{xk}}{g^{xk}} = m$$

L'intérêt de ce cryptosystème est qu'un même message n'est pas codé deux fois de suite de la même facon. Le prix à payer est qu'on envoie en gros deux fois plus d'informations qu'avec un cryprotsystème comme RSA.

Il n'est pas obligatoire d'utiliser $\mathbb{Z}_p$. Tout groupe cyclique convient.

La sécurité de ElGamal repose sur la difficulté de calculer le logarithme discret dans le groupe cyclique choisi. Si G est un groupe cyclique d'ordre q, étant donné (g, g^a, g^b) pour un générateur g de G et a, b deux nombres aléatoires compris entre 0 et $q-1$, l'élément $g^{a\,b}$ doit être comme un élement aléatoire de G.

5 Suggestions de développement

– Tests de primalité, de pseudo-primalité, génération de clefs.
– Mise en oeuvre de RSA, ou de ElGamal et d'une application.
– Mise en oeuvre d'une attaque (par exemple fraction continue).