École normale supérieure **2011-2012**
Département d'informatique

Introduction à la cryptologie
TD n° 5 : Cryptographie asymétrique 2/3

Un nombre composé n est dit de Carmichael si il est pseudo-premier de Fermat en base a pour tout entier $a > 0$ premier avec n.

Exercice 1.

1. Montrer qu'un nombre de Carmichael est nécessairement impair.

Soient n un nombre de Carmichael et p un facteur premier de n.

2. Montrer que p^2 ne divise pas n.
3. Montrer que $p-1$ divise $n-1$. On pourra considérer une racine primitive a modulo p et montrer que $a^{n-1} = 1 \bmod p$.
4. Réciproquement, montrer que si n est un entier composé sans facteur carré, et tel que pour tout entier p divisant n, $p-1$ divise $n-1$ alors n est un nombre de Carmichael.
5. En déduire qu'un nombre de Carmichael a au moins trois diviseurs premiers.

Exercice 2.

1. Démontrer le critère d'Euler
2. Montrer que si p est premier alors $\mathbb{Z}_{p^t}^*$ est cyclique pour tout entier $t \geq 1$.
3. Montrer que si $n \geq 3$ n'est pas premier alors pour (au moins) la moitié des $a \in \mathbb{Z}_n^*$, nous avons
$$\left(\frac{a}{n}\right) \not\equiv a^{(n-1)/2} \bmod n.$$
4. En déduire pour tout entier T un algorithme qui, prenant en entrée un entier n, retourne Composé ou Premier en $O(T \log^3 n)$ opérations binaires, de sorte que
   – si l'algorithme retourne Composé, alors n est toujours un nombre composé ;
   – si l'algorithme retourne Premier, alors la probabilité que n soit composé est inférieure à 2^{-T}.

Exercice 3. Soient N un module RSA, $2 < e < N$ un entier premier avec $\varphi(N)$ et $2 < d < \varphi(N)$ l'inverse de e modulo $\varphi(N)$. Montrer qu'il existe un algorithme polynomial probabiliste qui, étant donnés N, e et d, retourne la factorisation de N.

Exercice 4.

1. Montrer que si l'on dispose des chiffrés RSA c et c' d'un clair aléatoire m et d'un clair lié $m + r$, où $0 < r < N$ est connu, pour une clé publique $(N, e = 3)$, alors on peut retrouver m en temps polynomial.
2. Montrer comment généraliser cette approche pour tout entier e.
3. **Application.** Utiliser la méthode de la question précédente pour retrouver le message m vérifiant $c = m^{17} \bmod N$ et $c' = (m+1)^{17} \bmod N$ avec

$$\begin{aligned} N &= 47502685232865341825439992464725145700424182999231011547935 93 \\ c &= 19356218805125223063783713929395487370916847718680080264316 26 \\ c' &= 10114248818546991018461882489677552339876583926018473780350 75. \end{aligned}$$

Exercice 5 (Factorisation par divisions successives)**.** Factoriser l'entier

$$n = 3148240326296492491829836036538028522262397298543021512290073$$

par divisions successives.

Exercice 6 (Factorisation par la méthode Fermat)**.** Factoriser l'entier

$$n = 443363497731795997718971635197891857229652767733117521088186 1$$

par la méthode de Fermat.

Exercice 7 (Factorisation par la méthode $p-1$ de Pollard)**.** Factoriser l'entier

$$n = 11782768142027158401743290352232730332534494805066532395654586 3$$

par la méthode $p-1$ de Pollard.

Exercice 8 (Extraction de racine carrée modulo p)**.** Soit p un nombre premier.

1. Nous supposons que $p \equiv 3 \bmod 4$. Donner un algorithme de complexité $O(\log^3 p)$ opérations binaires qui, étant donné $\alpha \in \{1, \ldots, p-1\}$ tel que $\left(\frac{\alpha}{p}\right) = 1$, retourne $\beta \in \{1, \ldots, p-1\}$ tel que $\beta^2 \equiv \alpha \bmod p$.

Nous supposons désormais que $p \equiv 1 \bmod 4$. Posons $p = 2^h m + 1$ avec m impair.

2. Donner un algorithme probabiliste qui étant donné p retourne un élément γ de $\{1, \ldots, p-1\}$ tel que $\left(\frac{\gamma}{p}\right) = -1$ en temps espéré $O(\log^2 p)$ opérations binaires. Montrer que pour un tel γ, l'élément $\delta = \gamma^m$ engendre l'unique sous-groupe d'ordre 2^h de $\mathbb{Z}_p^*$.
3. Soit $\alpha \in \{1, \ldots, p-1\}$ tel que $(\alpha|p) = 1$. Montrer que α^m appartient au sous-groupe engendré par δ et en utilisant un algorithme de logarithme discret, en déduire un algorithme pour calculer une racine carré de α^m modulo p.
4. Conclure en donnant un algorithme permettant de calculer les racines carrés de α en temps $O((\log p)^3 + h \log(h) \log(p)^2)$.

Exercice 9 (Extraction de racine carrée modulo p^ℓ)**.** Soit p un nombre premier impair et $\ell \geq 2$ un entier.

1. Montrer qu'un entier x premier avec p vérifie $x^2 \equiv 1 \bmod p^\ell$ si et seulement si $x \equiv \pm 1 \bmod p^\ell$.
2. Montrer qu'un entier x premier avec p est un carré modulo p si et seulement si x est un carré modulo p^ℓ.
3. Donner un algorithme pour calculer une racine carrée de x modulo p^ℓ.
   Indication. On pourra commencer par calculer une racine carrée de x modulo p puis calculer récursivement une racine carrée de x modulo p^{i+1} à partir d'une racine carrée de x modulo p^i pour tout entier $i \in \{1, \ldots, \ell - 1\}$.

Exercice 10 (Extraction de racine carrée modulo N)**.** Soit N un entier dont la décomposition en facteur premier est $N = q_1^{f_1} \ldots q_d^{f_d}$ où $q_i \in \mathbb{P}$ sont des nombres premiers et $f_i \geq 1$ pour $i \in \{1, \ldots, d\}$).

1. Montrer qu'un entier x est un carré modulo N dès que tous les symboles de Legendre $\left(\frac{x}{q_j}\right)$ sont égaux à 1 pour $j \in \{1, \ldots, d\}$.
2. Montrer qu'un tel carré modulo N a exactement 2^d racines carrées.
3. Montrer que s'il existe un algorithme $\mathcal{A}$ capable d'extraire des racines carrées dans $\mathbb{Z}_N$ en temps τ, alors il existe un algorithme $\mathcal{B}$ qui retourne un diviseur propre de N en temps espéré $O(\tau \cdot (1 - 2^{1-d}))$.

Exercice 11 (Sécurité des protocoles de signature de De Jonge et Chaum)**.** Nous considérons deux variantes du schéma de signature RSA où le module $N = pq$ est le produit de deux nombres premiers p et q dits *forts* (*i.e.* $p = 2p' + 1$ et $q = 2q' + 1$ où p' et q' sont premiers).

1. Supposons que la signature d'un message impair $m \in \mathbb{Z}_N^*$ est l'entier $\sigma \in \mathbb{Z}_N^*$, s'il existe, tel que $\sigma^m = m \bmod N$. Montrer que ce schéma n'est pas résistant à une contrefaçon universelle sous une attaque à deux messages choisis.
2. Supposons que la signature d'un message $m \in \mathbb{Z}_N^*$ est l'entier $\sigma \in \mathbb{Z}_N^*$, s'il existe, tel que $\sigma^{2m+1} = m \bmod N$. Montrer que ce schéma n'est pas résistant à une contrefaçon universelle sous une attaque à deux messages choisis.

Indication. On pourra chercher deux messages m_1 et m_2 (dépendants de m) et deux entiers a et b tels que $\sigma = \sigma_1^a/\sigma_2^b$ soit une signature valide de m (où σ_1 et σ_2 sont les signatures de m_1 et m_2).

Exercice 12 (Sécurité du protocole de signature de Boyd)**.** Nous considérons un protocole de signature numérique où la clé publique est un couple d'entiers (N, g) et la clé secrète est un entier r telles que

(i) N est le produit de deux nombres premiers distincts p et q (*i.e.* N est un module RSA) ;

(ii) r est un diviseur premier de $p-1$;

(iii) g est un élément d'ordre r dans $\mathbb{Z}_N^*$.

Nous notons k la taille en bits des nombres premiers p et q et ℓ la taille en bits de l'entier r. La signature σ d'un entier m de taille ℓ est la racine m-ème de g dans $\mathbb{Z}_N^*$ (*i.e.* $\sigma^m = g$).

1. Montrer que si r ne divise pas $q-1$, alors la connaissance de (N, g) permet de factoriser l'entier N. Nous supposerons dans toute la suite de l'exercice que :

   (iv) r divise $q-1$;

2. Proposer un algorithme probabiliste qui, prenant en entrée deux entiers k et ℓ, retourne un triplet (N, g, r) vérifiant les propriétés (i)–(iv) et comparer la complexité de l'algorithme de signature avec celle de la signature RSA classique.

3. **Bris total.**

   (a) Donner un algorithme de complexité $O(2^{\ell/2})$ opérations dans le groupe $\mathbb{Z}_N^*$ permettant de retrouver r à partir de la donnée publique (N, g).

   (b) Montrer que si r est connu alors il est possible de factoriser N en $O(N^{1/4}/r)$ opérations dans le groupe $\mathbb{Z}_N^*$.
   Indication : en notant $p = xr + 1$ et $q = yr + 1$ et $(N-1)/r = ur + v$ avec $0 \leq v < r$, on pourra utiliser un algorithme de logarithme discret pour retrouver la « retenue » c définie par $x + y = v + cr$ et montrer que sa connaissance est suffisante pour retrouver p et q.

4. **Contrefaçon universelle.**

   (a) Montrer qu'il existe un algorithme polynomial qui prenant en entrée N et un entier m premier avec r, retourne un entier γ tel que $m\gamma \equiv 1 \mod r$. En déduire une contrefaçon universelle sous une attaque à clé seule contre le schéma de signature de Boyd.
   Nous supposerons désormais que :

   (ii') r est un diviseur *composé* de $p-1$ de taille ℓ ;

   et qu'il est difficile de calculer un multiple de r.

   (b) Montrer que la connaissance de la signature de deux messages m_1 et m_2 premiers entre eux permet de calculer la signature du message produit $m = m_1 m_2$ (et réciproquement). En déduire une contrefaçon universelle sous une attaque à deux messages choisis contre le schéma de signature de Boyd.
   Nous supposerons désormais que :

   (v) La signature σ d'un message $m \in \{0,1\}^*$ est la racine $H(m)$-ième de g (*i.e.* $\sigma^{H(m)} = g$) où $H : \{0,1\}^* \to \{0,1\}^\ell$ est une fonction de hachage cryptographique (dans la suite nous supposerons que H se comporte comme une fonction aléatoire).

5. **Contrefaçon existentielle.**
   Montrer que la connaissance d'un ensemble de messages $\{m, m_1, \ldots, m_t\}$ vérifiant :
   – $H(m) = a_1 \cdot a_2 \cdots a_t$ où les a_i sont des entiers deux à deux premiers entre eux ;
   – a_i divise $H(m_i)$ pour $i \in \{1, \ldots, n\}$
   est suffisante pour monter une contrefaçon existentielle sous une attaque à messages choisis. En déduire que le schéma n'est pas résistant aux contrefaçons existentielles lorsque ℓ est significativement plus petit que k.