Problèmes d'arithmétique pour la cryptographie

Robert Rolland

29 Septembre 2002

1 La division euclidienne

1.1 La division euclidienne dans $\mathbb{Z}$

Soient a et b deux entiers. Nous supposons b non nul. Alors il existe un et un seul couple d'entiers (q, r) tels que

$$a = bq + r \ \ et \ \ 0 \leq r < |b|.$$

1) Montrer que si ce couple existe il est unique.

En ce qui concerne l'existence nous allons donner deux algorithmes différents produisant le couple (q, r). Nous supposerons dans les deux cas que a et b sont des entiers > 0. Les autres cas s'en déduisent facilement.

1.1.1 Algorithme d'euclide

$B := b$;
$R := a$;
$Q := 0$;
tant que $R \geq B$ *faire*
  début
    $R := R - B$;
    $Q := Q + 1$;
  fin ;

2) Montrer que cet algorithme se termine et qu'à la fin Q contient le quotient q et R le reste r.

1.1.2 Version binaire

On calcule avant toute chose un entier $n \geq 0$ tel que $2^n b > a$.

$R := a$;
$Q := 0$;
$N := n$;
$Aux := 2^n b$;
tant que $N > 0$ *faire*
  début
    $Aux := Aux/2$;
    $N := N - 1$;
    si $R < Aux$
      alors $Q := 2 * Q$
      sinon début
          $Q := 2 * Q + 1$;
          $R := R - Aux$;
        fin;
  fin;

3) Montrer que cet algorithme se termine et qu'à la fin Q contient le quotient q et R le reste r.

1.2 Le plus grand commun diviseur

Soient a et b deux entiers dont l'un au moins est non nul. Il existe un plus grand entier > 0 qui soit diviseur commun de a et de b. Cet entier sera noté $pgcd(a, b)$ et appelé le **plus grand commun diviseur** de a et b.

4) Montrer que les diviseurs communs de a et b sont les diviseurs de $pgcd(a, b)$.

1.2.1 Algorithme d'Euclide

$R0 := |a|$;
$R1 := |b|$; $(b \neq 0)$
tant que $R1 > 0$ *faire*

début
$R := Reste\_Division(R0, R1)$;
$R0 := R1$;
$R1 := R$;
fin ;

5) Montrer qu'en sortie $R1 = 0$, et $R0 = pgcd(a, b)$.

1.2.2 Algorithme d'Euclide étendu

Si $pgcd(a, b) = d$, il existe deux entiers u et v tels que $ua + vb = d$.

Là encore nous supposerons que $a \geq 0$ et $b > 0$. Le cas général s'en déduit.
Voici un algorithme (**algorithme d'Euclide étendu**, adaptation de l'algorithme précédent) qui permet de trouver explicitement un couple (u, v) qui convient.

$R0 := a$; $\quad (a \geq 0)$
$R1 := b$; $\quad (b > 0)$
$U0 := 1$;
$U1 := 0$;
$V0 := 0$;
$V1 := 1$;
tant que $R1 > 0$ *faire*
début
$Q := Quotient\_Division(R0, R1)$;
$R := Reste\_Division(R0, R1)$;
$U := U0 - Q * U1$;
$V := V0 - Q * V1$;
$R0 := R1$;
$R1 := R$;
$U0 := U1$;
$U1 := U$;
$V0 := V1$;
$V1 := V$;
fin ;

6) Montrer qu'en sortie $R0 = pgcd(a, b)$, $U0 = u$ et $V0 = v$.

7) Faire le calcul avec $a = 325$, $b = 145$.

1.2.3 Quelques résultats

8) Montrer que a et b sont premiers entre eux si et seulement s'il existe u et v tels que $au + bv = 1$ (théorème de Bezout).

9) Montrer que si c divise ab et si c est premier avec a alors c divise b (lemme d'Euclide Gauss).

10) Soient a et b deux entiers dont l'un au moins est non nul, et d leur plus grand commun diviseur. Trouver toutes les solutions de $au + bv = d$.

2 Les classes résiduelles modulo n

2.1 Les éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$

Rappelons les notations usuelles concernant les congruences. On dit que x est congru à y modulo n et on note

$$x \equiv y \quad (n),$$

lorsque $x - y$ est un multiple de n, c'est-à-dire

$$x = y + kn.$$

La congruence est une relation d'équivalence et l'ensemble des classes d'équivalence est noté $\mathbb{Z}/n\mathbb{Z}$. Dans chaque classe il y a un représentant x et un seul tel que $0 \leq x < n$. Ainsi on peut considérer que les éléments de $\mathbb{Z}/n\mathbb{Z}$ sont $0, 1, 2, \cdots, n-1$. L'addition et la multiplication dans $\mathbb{Z}/n\mathbb{Z}$ se font en additionnant et en multipliant dans Z puis en réduisant modulo n. On notera

$$x = y \ \ mod \ \ n,$$

l'unique élément x congru à y modulo n et tel que $0 \leq x < n$. (attention dans de nombreux langages informatiques la fonction mod ne renvoie pas tout à fait cela quand le nombre y est négatif).

2.1.1 éléments inversibles

1) Montrer que x est inversible dans $\mathbb{Z}/n\mathbb{Z}$ si et seulement si x est premier avec n.

2) Montrer que $\mathbb{Z}/n\mathbb{Z}$ est un corps si et seulement si n est premier.

2.1.2 Le théorème des restes chinois

3) Montrer que si m et n sont premiers entre eux alors la condition

$$\begin{cases} a \equiv b & (m) \\ a \equiv b & (n) \end{cases}$$

est équivalente à

$$a \equiv b \quad (mn).$$

Soient m et n premiers entre eux. On cherche toutes les solutions de

$$\begin{cases} x \equiv a & (m) \\ x \equiv b & (n) \end{cases}$$

On considére u et v tels que $um + vn = 1$.

4) Montrer que

$$x = bum + avn$$

est une solution.

5) Trouver toutes les solutions.

6) Résoudre

$$\begin{cases} x \equiv 3 & (7) \\ x \equiv 5 & (16) \end{cases}.$$

7) Montrer que $\mathbb{Z}/nm\mathbb{Z}$ est isomorphe à $\mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/n\mathbb{Z}$.

2.1.3 Le petit théorème de Fermat

8) Soit p un nombre premier, montrer que si $1 \leq k \leq p-1$ alors

$$C_p^k \equiv 0 \quad (p).$$

9) Montrer que pour tout a

$$a^p \equiv a \quad (p).$$

10) Montrer que si a est premier avec p alors (petit théorème de Fermat)

$$a^{p-1} \equiv 1 \quad (p).$$

11) Soit $n = pq$ où p et q sont deux nombres premiers. Montrer que pour tout a et tout k on a

$$a^{k(p-1)(q-1)+1} \equiv a \quad (n).$$

3 La fonction d'Euler

3.1 Les éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$

Notons $(\mathbb{Z}/n\mathbb{Z})^*$ le groupe multiplicatif des éléments inversibles de $\mathbb{Z}/n\mathbb{Z}$. Nous noterons $\Phi(n)$ le nombre des éléments de $(\mathbb{Z}/n\mathbb{Z})^*$. On posera en outre $\Phi(1) = 1$. (fonction d'Euler).

1) Calculer $\Phi(p)$ lorsque p est un nombre premier.

2) Soit p un nombre premier. Trouver le nombre de multiples de p de l'intervalle $[1 \cdots p^\alpha]$. En déduire la valeur de $\Phi(p^\alpha)$.

3) Montrer que si m et n sont premiers entre eux alors

$$\Phi(mn) = \Phi(m)\Phi(n).$$

4) Montrer que pour tout $n > 1$

$$\phi(n) = n \prod_i (1 - \frac{1}{p_i}),$$

où les p_i sont les facteurs premiers de n.

5) Soit a un élément de $(\mathbb{Z}/n\mathbb{Z})^*$ (c'est-à-dire a premier avec n). Notons $r_1, \cdots, r_{\Phi(n)}$ tous les éléments de $(\mathbb{Z}/n\mathbb{Z})^*$. Montrer que

$$(ar_1)(ar_2)\cdots(ar_{\Phi(n)}) \equiv r_1 r_2 \cdots r_{\Phi(n)} \quad (n).$$

En conclure que

$$a^{\Phi(n)} \equiv 1 \quad (n).$$

On note

$$D = \left\{ \frac{1}{n}, \frac{2}{n}, \cdots, \frac{n}{n} \right\}$$

et pour tout d divisant n

$$D_d = \left\{ \frac{k}{d} \mid (k,d) = 1 \; et \; 1 \leq k \leq d \right\}.$$

6) Montrer que les D_d forment une partition de D.

7) En conclure que

$$n = \sum_{d|n} \Phi(d).$$

3.2 Calcul d'une puissance

Soit $n \geq 1$ un entier. On veut calculer a^n (a est par exemple dans $\mathbb{Z}/n\mathbb{Z}$ ou dans $\mathbb{R}$, $\mathbb{C}$,...). On considère l'algorithme suivant

$A := a$;
$N := n$;
$R := 1$;
tant que $N > 0$ *faire*
  si N *pair*
    alors début
      $A := A * A$;
      $N := N/2$;
    fin
    sinon début
      $R := R * A$;
      $N := N - 1$;
    fin.

8) Montrer que cet algorithme se termine et qu'en sortie R contient a^n.

9) Calculer le nombre de tours de boucle en fonction de la longueur du développement binaire de n. En conclure la complexité de cet algorithme.

10) Soit a un élément inversible de $\mathbb{Z}/n\mathbb{Z}$. Indiquer comment calculer a^{-1} grâce à la fonction d'Euler. Comparer la complexité de cette méthode avec celle de l'algorithme d'Euclide étendu.

4 Générateurs de $(\mathbb{Z}/p\mathbb{Z})^*$ (p premier) - Logarithme discret

Dans toute la suite p est un nombre premier et $(\mathbb{Z}/p\mathbb{Z})^*$ est le groupe multiplicatif des éléments non nuls du corps $\mathbb{Z}/p\mathbb{Z}$.

4.1 Ordre d'un élément

On rappelle que d'après le petit théorème de Fermat pour tout $a \in (\mathbb{Z}/p\mathbb{Z})^*$

$$a^{p-1} = 1.$$

Soit $a \in (\mathbb{Z}/p\mathbb{Z})^*$, il existe donc un plus petit entier $e > 0$ tel que $a^e = 1$.
L'entier e est appelé l'ordre de a.

1) Calculer les ordres de tous les éléments de $(\mathbb{Z}/7\mathbb{Z})^*$.

2) Montrer que e divise $p-1$ (on pourra effectuer la division euclidienne de $p-1$ par e).

3) Montrer que si e_a est l'ordre de a et si e_b est l'ordre de b et si e_a et e_b sont premiers entre eux, alors l'ordre de ab est $e_a e_b$. Pour cela on pourra utiliser les questions suivantes:

a) Montrer que l'ordre k de ab divise $e_a e_b$.

b) Montrer que l'ordre k de ab vérifie

$$a^{ke_b} = 1.$$

En conclure que k est multiple de e_a et pour la même raison de e_b et donc que $k = e_a e_b$.

4) Montrer que si a est d'ordre e_a et b d'ordre e_b alors il existe un élément d'ordre $ppcm(e_a, e_b)$. En conclure que si e est le plus petit commun multiple des ordres des éléments de $(\mathbb{Z}/p\mathbb{Z})^*$ alors il existe un élément d'ordre e.

5) Montrer que pour tout $a \in (\mathbb{Z}/p\mathbb{Z})^*$ on a $a^e = 1$. En conclure que $e = p-1$ et donc qu'il existe un élément α tel que

$$(\mathbb{Z}/p\mathbb{Z})^* = \{1, \alpha, \cdots, \alpha^{p-2}\}.$$

Les éléments qui ont cette propriété sont appelés éléments primitifs.

4.2 Répartition des éléments suivant leur ordre

6) Montrer que si a est d'ordre e, tout autre élément d'ordre e est nécessairement de la forme a^i avec $1 \leq i \leq e-1$. Montrer alors que a^i est d'ordre $ppcm(i, e)/i$.

7) Montrer que si a est un élément d'ordre e alors a^i est d'ordre e si et seulement si i est premier avec e. Dans ce cas il y a $\Phi(e)$ éléments d'ordre e.

8) Montrer que pour tout diviseur e de $p-1$ il y a exactement $\Phi(e)$ éléments d'ordre e.

9) Proposer un algorithme pour trouver un élément primitif.

4.3 Les logarithmes discrets

Soit α un élément primitif de $(\mathbb{Z}/p\mathbb{Z})^*$. Tout élément a de $(\mathbb{Z}/p\mathbb{Z})^*$ s'écrit donc

$$a = \alpha^k.$$

Le nombre k est le logarithme discret de a à base α.

10) Classer tous les éléments de $(\mathbb{Z}/19\mathbb{Z})^*$ suivant leur ordre.

11) Trouver un élément primitif α de $(\mathbb{Z}/19\mathbb{Z})^*$ ainsi que les logarithmes discrets à base α de tous les éléments de $(\mathbb{Z}/19\mathbb{Z})^*$.

5 Problèmes arithmétiques difficiles

5.1 Nombres premiers

On sait (petit théorème de Fermat) que si n est premier, pour tout a premier avec n on a

$$a^{n-1} \equiv 1 \ (n).$$

La réciproque est fausse, il existe des nombres n qui ne sont pas premiers et pour lesquels on a aussi pour tout a premier avec n l'égalité $a^{n-1} \equiv 1 \ (n)$. Ces nombres sont appelés pseudo-premiers ou encore nombres de Carmichael. Il existe une infinité de nombres de Carmichael.

1) Décomposer 561 en facteurs premiers.

2) Montrer que pour tout a premier avec 561

$$a^{560} \equiv 1 \ (3),$$

$$a^{560} \equiv 1 \ (11),$$

$$a^{560} \equiv 1 \ (17).$$

En conclure que pour tout a premier avec 561

$$a^{560} \equiv 1 \ (561).$$

3) Soit n un entier > 1. Montrer que les trois conditions a) b) c) suivantes sont équivalentes

a) Le nombre n est sans facteurs carrés et $p - 1$ divise $n - 1$ pour tout facteur premier p de n.

b) Pour tout entier a on a

$$a^n \equiv a \quad (n).$$

c) Pour tout entier a premier avec n on a

$$a^{n-1} \equiv 1 \quad (n).$$

4) Montrer que tout nombre de Carmichael est le produit d'au moins trois nombres premiers impairs distincts.

5.2 Racines carrées

5) Soit p un nombre premier de la forme $4t-1$. Soit $n = a^2$ dans $\mathbb{Z}/p\mathbb{Z}$. Montrer que $n^{\frac{p+1}{4}}$ est une racine carrée de n.

6) On dispose d'un algorithme (cf. cours) pour trouver une racine carrée d'un nombre carré dans $\mathbb{Z}/p\mathbb{Z}$ pour tout nombre premier p. Comment calculer une racine carrée d'un nombre carré dans $\mathbb{Z}/n\mathbb{Z}$ où n est le produit de deux nombres premiers p et q et où la décomposition $n = pq$ est connue?

5.3 Logarithmes discrets

La construction de la table de logarithmes complète pour $(\mathbb{Z}/p\mathbb{Z})^*$ demande de l'ordre de p opérations et demande une place mémoire de taille d'ordre p. Voici un algorithme (Baby-step, Giant-step) dû à Shanks qui réduit le temps et la place à l'ordre $\sqrt{p}\log(p)$.

On veut chercher le logarithme discret x à base α d'un $y \in (\mathbb{Z}/p\mathbb{Z})^*$. On choisit un m et on écrit $x = mq + r$. Alors $\alpha^{mq} = y\alpha^{-r}$. On construit la table des α^{mq} (Giant-step) et la table des $y\alpha^{-r}$ (Baby-step).

7) Indiquer en détail comment procéder, en particulier comment choisir m, comment construire les tables, comment s'en servir, et faire le calcul de la complexité.

6 Algorithme de Shank pour les racines carrées

6.1 Cas simple

1) Soit p un nombre premier de la forme $4t-1$. Soit $n=a^2$ dans $\mathbb{Z}/p\mathbb{Z}$. Montrer que $n^{\frac{p+1}{4}}$ est une racine carrée de n.

6.2 Algorithme de Shank

2) On suppose maintenant que p est de la forme $4t+1$. On écrit alors

$$p-1=2^s t$$

avec t impair et $s \geq 2$.
Soit $n=a^2$ dans $\mathbb{Z}/p\mathbb{Z}$. On suppose qu'on connaît un m qui ne soit pas un résidu quadratique modulo p. Posons alors

$$z=m^t.$$

a) Montrer que z vérifie

$$z^{2^{s-1}} \equiv -1 \ \ (p).$$

b) On pose

$$B=n^t, \quad X=n^{\frac{t+1}{2}}, \quad Y=z, \quad R=s-1.$$

et on effectue l'algorithe suivant

tant que $R \geq 1$ *faire*
début
si $B^{2^{R-1}} \equiv 1 \ (p)$
alors
début
$Y = Y^2$;
$R = R - 1$;
fin
sinon
début
$B = BY^2$;
$X = XY$;
$Y = Y^2$;
$R = R - 1$;
fin
fin ;

Montrer qu'en sortie, X contient une racine carrée de n.

Remarquons que cet algorithme donne le résultat pourvu qu'on ait tiré au sort au début un m qui ne soit pas un résidu quadratique. Ceci donne naissance à un algorithme de Las Vegas.

7 Nombre de pas de l'algorithme d'Euclide

7.1 La suite de Fibonacci

On définit la suite $(F_n)_{n\in\mathbb{N}}$ par récurrence grâce aux relations

$$F_0 = 0, \quad F_1 = 1, \tag{1}$$

$$F_{n+1} = F_n + F_{n-1} \tag{2}$$

1) Interprétations matricielles.

a) Montrer que pour tout entier $n \in \mathbb{N}$ on a

$$\begin{pmatrix} F_{n+1} \\ F_n \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

b) On pose aussi $F_{-1} = 1$. Montrer que pour tout $n \in \mathbb{N}$ on peut écrire

$$\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n.$$

c) En conclure que

$$F_{n+1}F_{n-1} - F_n^2 = (-1)^n.$$

2) Calcul de F_n.

a) Chercher les solutions de (2) de la forme r^n. On donnera les deux solutions $r = r_1$ et $r = r_2$ en fonction de $\phi = \frac{1+\sqrt{5}}{2}$ (nombre d'or).
Déterminer toutes les solutions de (2).

b) Calculer F_n en fonction de ϕ. En déduire que

$$F_n = Round\left(\frac{1}{\sqrt{5}}\phi^n\right).$$

3) Calculs de $pgcd$.

a) Montrer que pour tout $n \geq 0$ et tout $m \geq 1$

$$F_{n+m} = F_{n+1}F_m + F_nF_{m-1}.$$

b) Montrer que F_{n+1} et F_n sont premiers entre eux.

c) Montrer que

$$pgcd(F_{m+n}, F_m) = pgcd(F_m, F_n).$$

c) Montrer que

$$pgcd((F_m, F_n) = F_{pgcd(m,n)}.$$

7.2 Le théorème de Lamé

1) Prouver maintenant le résultat suivant (théorème de Lamé)

Theorem 7.2.1 *Soient x et y deux entiers vérifiant $0 < y < x$. Notons d leur $pgcd$. Si l'algorithme d'Euclide de calcul du $pgcd$ s'arrête au bout de n pas alors*

$$x \geq dF_{n+2}, \quad y \geq dF_{n+1}.$$

Indication: On pourra raisonner par récurrence.

2) Que se passe-t-il pour $x = F_{n+2}$ et $y = F_{n+1}$?

7.3 Calcul du nombre de pas de l'algorithme d'Euclide

1) Montrer que

$$F_{n+1} \geq \frac{1}{\sqrt{(5)}}\phi^{n+1}\left(1-\left(\frac{1}{\phi}\right)^4\right).$$

2) En conclure que

$$n \leq \frac{log(F_{n+1})}{log(\phi)} + C,$$

puis que

$$n \leq \frac{log(y)}{log(\phi)} + C$$

où C est une constante.

Le nombre de pas de l'algorithme d'euclide est donc linéaire en fonction de la taille $log(y)$ de l'entrée y.

8 Complement sur RSA

8.1 Modules ayant 2 facteurs

Soit $n = pq$ le produit de deux nombres premiers. On pose :

$$\lambda(n) = lcm(p-1, q-1).$$

Soit $3 \leq e < n$ premier avec $\lambda(n)$. On calcule d, d_p, d_q tels que :

$$\begin{aligned} de &\equiv 1 \quad (\lambda(n)), \\ d_p e &\equiv 1 \quad (p-1), \\ d_q e &\equiv 1 \quad (q-1). \end{aligned}$$

Soit m un entier tel que $0 \leq m < n$. Définissons

$$c = m^e \mod n.$$

8.1.1 Une première façon de calculer m à partir de c

1) Montrer que pour tout a et tout k

$$a^{1+k\lambda(n)} \equiv a \quad (n).$$

2) En conclure que :

$$m = c^d \mod n.$$

8.1.2 Une deuxième façon de retrouver m à partir de c

Calculons :

$$m_1 = c^{d_p} \mod p,$$
$$m_2 = c^{d_q} \mod q,$$

et q_{inv} tels que :

$$qq_{inv} \equiv 1 \quad (p).$$

Posons maintenant :

$$h = (m_1 - m_2).q_{inv} \mod p,$$

3) Montrer que :

$$m \equiv m_1 \quad (p),$$
$$m \equiv m_2 \quad (q).$$

4) Montrer que :

$$m = m_2 + qh.$$

8.2 Modules ayant $s > 2$ facteurs

Nous supposons maintenant que :

$$n = r_1 r_2 \cdots r_s,$$

où les r_i sont des nombres premiers distincts.

Nous posons :

$$\lambda(n) = lcm(r_1 - 1, \cdots, r_s - 1).$$

Soit $3 \leq e < n$ premier avec $\lambda(n)$. Nous calculons d_i tel que :

$$d_i e \equiv 1 \quad (r_i - 1).$$

Soit m un entier tel que $0 \leq m < n$. Définissons :

$$c = m^e \mod n.$$

On calcule pour chaque i :

$$m_i = c^{d_i} \mod r_i.$$

5) Montrer que pour chaque i :

$$m \equiv m_i \quad (r_i).$$

5) Calculons pour $i \geq 2$, t_i de telle sorte que :

$$t_i . r_1 \cdots r_{i-1} \equiv 1 \quad (r_i).$$

En utilisant plusieurs fois la méthode de la sous-section précédente avec $p = r_i$ et $q = r_1 \cdots r_{i-1}$, décrire un algorithme pour retrouver m.

9 Une attaque de RSA

Nous utilisons un système RSA pour signer des messages. Nous nous préoccupons seulement de la partie qui applique la fonction RSA à un message haché m.

Un utilisateur dispose donc d'un module $n = pq$ (où p et q sont deux nombres premiers grands). Il dispose d'une clé publique e et d'une clé privée d de telle sorte que

$$pgcd(e, \phi(n)) = 1,$$

$$ed \equiv 1 \quad (\phi(n)),$$

où $\phi(n) = (p-1)(q-1)$ est la valeur de la fonction d'Euler en n.

On supposera que le message haché m ($0 < m < n$)à signer est premier avec n (sinon c'est catastrophique car en prenant le pgcd de m et n on obtient un facteur de n)
L'utilisateur doit alors calculer la signature s à joindre au message

$$s = m^d \mod n.$$

Afin d'accélérer un peu les calculs il utilise la méthode qui suit.
Soient d_p et d_q définis respectivement par

$$d_p = d \; mod \; (p-1),$$
$$d_q = d \; mod \; (q-1).$$

a) Montrer que

$$d_p e \equiv 1 \quad (p-1),$$
$$d_q e \equiv 1 \quad (q-1).$$

b) Montrer que

$$s \equiv m^{d_p} \quad (p),$$
$$s \equiv m^{d_q} \quad (q).$$

c) Expliquer comment calculer s une fois qu'on a calculé

$$s_p = m^{d_p} \mod p$$

et

$$s_q = m^{d_q} \mod q.$$

d) On suppose que pour une raison ou une autre, lors du calcul de

$$s_q = m^{d_q} \mod q$$

une erreur se produit si bien qu'au lieu de calculer s on calcule s' tel que

$$s' \equiv m^{d_p} \quad (p),$$
$$s' \not\equiv m^{d_q} \quad (q).$$

Montrer que

$$(s')^e \equiv m \quad (p),$$
$$(s')^e \not\equiv m \quad (q).$$

En déduire que

$$pgcd((s')^e - m, n) = p.$$

Ainsi grâce au message m (dont le destinataire dispose), à la signature erronée s' et aux données publiques n et e le destinataire peut calculer la factorisation de n.

10 Une attaque de la signature ElGamal

10.1 La signature El Gamal

Soit p un nombre premier grand. On rappelle que le groupe multiplicatif $(\mathbb{Z}/p\mathbb{Z})^*$ est cyclique. Soit g un générateur de ce groupe (ainsi tout élément de $(\mathbb{Z}/p\mathbb{Z})^*$ est une puissance de g). Alice choisit un x tel que $0 \leq x \leq p-2$ et calcule

$$y = g^x \ mod \ p.$$

La clé publique d'Alice est (p, g, y), sa clé privée est (p, g, x).

Lorsque Alice veut signer un message M, grâce à une fonction de hachage publique h bien choisie elle calcule $m = h(M)$ avec $0 \leq m \leq p-1$. Puis elle prend au hasard k tel que $1 \leq k \leq p-2$ et k premier avec $p-1$. Elle calcule $r = g^k \ mod \ p$ et $s = k^{-1}(m - rx) \ mod \ (p-1)$. Le message signé est (M, r, s).

Bob qui reçoit le message signé calcule aussi $m = h(M)$ puis $v = g^m \ mod \ p$ et $w = y^r r^s \ mod \ p$.

a) Montrer que si c'est bien Alice qui a signé alors $r < p$ et $v = w$. Ainsi Bob acceptera dans ce cas la signature.

10.2 Une attaque de cette signature dans un cas particulier

On suppose dans toute la suite que $p = 4u + 1$ et que $p - 1 = gt$. Puisque g divise $p-1$ on sait qu'il existe un sous groupe cyclique H d'ordre g du groupe cyclique $(\mathbb{Z}/p\mathbb{Z})^*$. On suppose en outre qu'on sait calculer le logarithme discret dans ce sous groupe H.

b) Montrer que g^t est un élément primitif de H.

c) Montrer qu'il est possible de calculer un z tel que $g^{tz} = y^t$.

d) Montrer que $g^{\frac{p-1}{2}} \equiv -1 \quad (p)$ puis que $t^{\frac{p-3}{2}} \equiv g \quad (p)$.

Un adversaire, voulant imiter la signature d'Alice calcule $m = h(M)$ puis prend $r = t$ et $s = 1/2(p-3)(m - tz)\ mod\ (p-1)$.

e) Montrer que (M, r, s) est accepté par Bob comme signature d'Alice.