FACULTÉ DES MATHÉMATIQUES ET INFORMATIQUES

Département de Mathématiques

Mémoire de Fin D'etude

Présenté pour l'obtention du diplôme de **Master**

Domaine :Mathématiques et Informatique

Filière :Mathématiques

Spécialité :Mathématique Discréte

Par

Lakel Djilali

Président

Promotion:2014/2015

Remerciements

Je remercie tout d'abord mon Dieu qui m'a donné la force pour terminer ce modeste travail.

Je tiens à remercier mon promoteur Mr le professeur Abdemadjid BOUDAOUD pour la confiance qu'il m'a témoignée en me proposant ce sujet, ses encouragements et sa patience.

Mr Lahcen LADJLET qui ma aussi aider avec ses conseils, les discussions scientifiques qu'il a su générer, ses remarques et ses suggestions qui m'ont permis de finaliser ce modeste travail. Je leurs transmet ma reconnaissance et ma plus profonde gratitude.

Je remercie aussi tous les membres du Jury les professeurs (Abdelaziz AMROUNE et Dhaouadi MIHOUBI) pour l'honneur qu'ils m'ont fait, en acceptant de juger ce modeste travail.

Je ne peux pas clôturer mes remerciements sans se retourner vers les êtres qui me sont les plus chers ; ma famille qui ont eu un rôle essentiel et continu dans ma réussite.

Merci

Notations :

G	disige le groupe
H	disigne le sous groupe
$\|E\|$	le cardinal de de l'ensemble E
Ord_g	ordre de l'élément g
Z/nZ	le groupe quotient
$(Z/nZ)^*$	le groupe inversible
$\overline{x}$	la classe de congruence de l'élément x
φ	fonction d'indicatrice d'Euler
$\sum$	symbôle de sommation
$\prod$	symbôle de produit
$log_b\ n$	logarithme en base b de n
$\lfloor n \rfloor$	le plus grand entier inférieur ou égal à n

Table des matières

Introduction

Ce travail dont le titre est " **Congruence et cryptographie à clé publique** " s'inscrit dans le domaine de la théorie des nombres considéré par beaucoup de mathématiciens comme la reine des mathématiques. La théorie des nombres est l'un des rares domaines des mathématiques dont la plupart des problèmes peuvent être compris par tous, ou des moins par tous qui sont familiers avec les notions de base d'algèbre, de combinatoire et d'analyse. Parmi les nombreuses applications de cette branche, on s'est aperçu depuis 1977 que la théorie des nombres peut avoir des applications très importantes en cryptographie. C'est dans cette optique que notre travail vient selon le plan suivant.

Le premier chapitre, présente les concepts de base de l'arithmétique, à savoir la divisibilité, structures algébriques.

Le deuxième chapitre est consacré à l'étude des congruences. Après ceci on donne les fameux théorèmes qui sont exprimés en fonction de la congruence, tels que: Théorème de Wilson, d'Euler, de Fermat et de reste chinois. De plus, nous illustrons ces théorèmes par des applications.

Le troisième chapitre s'intéresse, comme nous avons l'annoncé ci-dessus, à la cryptographie publique où nous allons examiner quelques systèmes comme les chiffrements RSA, puis Rabin, finalement El Gamal. Nous donnons quelques applications de chaque système ainsi que l'étude de son efficacité.

Chapitre 1

Élements d'arithmétique

L'objectif de ce chapitre est de rappeller quelque notions sur la divisibilité et les resultats élémentaires d'arithmétique et de théorie des congruences. Cette théorie conduit naturellement à introduire l'anneau Z/nZ ainsi que le groupe de ses éléments inversibles, nous rappelons la construction de ces objets et précisons leurs principales propriétés.

1.1 Notions sur la divisibilité

1.1.1 Division euclidienne

Théorème 1.1.1 *Soit a un entier et $b > 1$ un entier strictement positif. Alors il existe un couple unique (q, r) vérifiant la double condition :*

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

Définition 1.1.1 *On dit qu'un entier $p \geq 2$ est premier lorsqu'il possède pour seuls diviseurs positifs 1 et lui-même.*

Théorème 1.1.2 (*théorème d'Euclide*) *Il ya une infinité de nombres premiers.*

Preuve. supposons que $p_1, \ldots, p_n$ l'ensemble fini de tout les nombres premiers. Considérons l'entier

$$N = p_1 \cdots p_n + 1$$

$N > 1$, il résulte que N est divisible par un certain nombre premier p. Si $p = p_i$ pour $i = 1, .., n$, puis p divise $N - p_1 \cdots p_n = 1$, ce qui est absurde.

Par conséquent, $p \neq p_i$. Cela signifie que, pour un ensemble fini de nombres premiers, il existe toujours un premier qui n'appartient pas à l'ensemble, et donc les nombres premiers est infinie. ■

Théorème 1.1.3 (*Théorème fondamental de l'arithmétique*) *Tout entier $a > 1$ s'écrit de façon unique*

$$(D) \qquad a = p_1^{\alpha_1} p_2^{\alpha_2} \cdots p_n^{\alpha_n},$$

Où α_i sont des entiers positifs, et les entiers p_i sont premiers et vérifient

$$p_1 < p_2 < \cdots < p_n$$

Exemple 1.1.1 $60 = 2^2 \cdot 3 \cdot 5.$

1.1.2 PGCD

Nous définisons le plus grand commun diviseur de deux entiers.

Définition 1.1.2 *un diviseur commun à a et b est un entier qui divise à la fois a et b.*

Théorème 1.1.4 *parmi tous les diviseurs communs à deux entiers a et b, qui ne sont pas tous les deux nuls, il en existe un qui est plus grand que tous les autres. On le note $pgcd(a, b)$ où (a, b) et on l'appelle le plus grand commun diviseur de a et b.*

Plus généralement, on définit le plus grand commun diviseur des entiers $a_1, a_2, ..., a_k$ de la façon suivante. Si un, au moins, des a_i n'est pas nul, $pgcd(a_1, a_2, ..., a_k)$ est le plus grand entier positif qui divise tous les a_i. Si tous les a_i sont nuls, $pgcd(a_1, a_2, ..., a_k) = 0$.

Exemple 1.1.2 *le plus grand commun diviseur de* 18 *et* 30 *est* 6, *on écrit* $pgcd(18, 30) = (18, 30) = 6$.

Définition 1.1.3 (*ppcm*) *Si a et b sont deux nombres entiers, il existe un plus petit entier ≥ 0 qui est multiple commun de a et de b. Cet entier sera notée $ppcm(a;b)$ et appelée le plus petit commun multiple de a et b.*

Corollaire 1.1.1 *pour tous a, b, n l'équation $ax + by = n$ possède des solutions entières x et y si seulement si $pgcd(a, b)$ divise n.*

Exemple 1.1.3 *le corrolaire dit que l'équation*

$$3x + 4y = 123$$

a une solution, parce que $pgcd(3, 4) = 1$.

1.1.3 Algorithme d'Euclide

Soient a et b deux entiers tels que $a > b \geq 0$.

1. Si $b = 0$ alors $pgcd(a, b) = pgcd(a, 0) = a$.
2. Si $b \neq 0$ alors $pgcd(a, b) = pgcd(b, r)$, où r est le reste de la division euclidienne de a par b.

L'algorithme d'Euclide consiste à réitérer la $2^{éme}$ formule jusqu'à ce que l'on tombe sur un reste nul. Dans ce cas, on applique la 1^{er} formule.

Exemple 1.1.4 $pgcd(96, 81) =?$

$$96 = 81 \cdot 1 + 15 \quad donc \quad pgcd(96, 81) = pgcd(81, 15)$$

$$81 = 15 \cdot 5 + 6 \quad donc \quad pgcd(81, 15) = pgcd(15, 6)$$

$$15 = 6 \cdot 2 + 3 \quad donc \quad pgcd(15, 6) = pgcd(6, 3)$$

$$6 = 3 \cdot 2 + 0 \quad donc \quad pgcd(6, 3) = pgcd(3, 0) = 3$$

Ainsi :

$$pgcd(96, 81) = 3.$$

Algorithme 1.1.1 *Algorithme d'Euclide en langage C.*

```
euclid (int a, int b, int pgcd)
begin
  int r
  a=|a|
  b=|b|
        while (b! = 0)
              r = a mod b
              a = b
              b = r
        end while
   pgcd = a
end
```

Lemme 1.1.1 *(Gauss) Soit $a, b, c \in N$. Si*

$$c|ab \quad et \quad pgcd(c,b) = 1, \ alors \quad c|a.$$

Preuve. Comme $pgcd(c,b) = 1$, il existe deux entiers x et y tels que $cx + by = 1$. En multipliant cette égalité par a, il vient : $acx + aby = a$. Or, $c|ab$ et

$c|ac$, donc $c|\ (acx + aby)$. En d'autres termes: $c|a$. ■

1.1.4 Algorithme d'Euclide étendu

L'algorithme d'Euclide étendu permet de déterminer $d = pgcd(a,b)$ ainsi que deux entiers x et y vérifiant

$$d = ax + by.$$

on écrit

$$\begin{cases} r_0 = a \\ x_0 = 1, \\ y_0 = 0, \end{cases} \quad \begin{cases} r_1 = b, \\ x_1 = 0, \\ y_1 = 1. \end{cases} \quad \text{et} \ \ \forall k \geq 1, \begin{cases} r_{k+1} = r_{k-1} - r_k q_k, \\ x_{k+1} = x_{k-1} - x_k q_k, \\ y_{k+1} = y_{k-1} - y_k q_k. \end{cases}$$

jusqu'à obtenir un reste nul.

Si r_n est le dernier reste non nul, on a $\begin{cases} d = r_n, \\ x = x_n, \\ y = y_n. \end{cases}$.

Exemple 1.1.5 *Choisissons* $a = 100$ *et* $b = 35$. *les valeurs* r_k, q_k, x_k *et* y_k *sont indiquées dans le tableau suivant.*

k	*0*	*1*	*2*	*3*	*4*
r_k	*100*	*35*	*30*	***5***	*0*
q_k		*2*	*1*	*6*	
x_k	*1*	*0*	*1*	*1*	*7*
y_k	*0*	*1*	*2*	*3*	*20*

Nous trouvons que $n = 3$ *et que* $pgcd(100, 35) = 5 = -1 \cdot 100 + 3 \cdot 35$.

Algorithme 1.1.2 *Algorithme d'Euclide étendu en langage C.*

Xeuclide (int a, int b, int pgcd, int x, int y)
begin
int q, r, xx, yy, $sign$
int $xs[2]$, $ys[2]$
// les coefficients sont initialisés
$xs[0] = 1 \quad xs[1] = 0$
$ys[0] = 0 \quad ys[1] = 1$
$sign = 1$
// Tant que b n'est pas 0, on remplace a par b, et b par $(a \ mod \ b)$
// On met à jour les coefficients x et y.
while $(b! = 0)$
$r = a \ mod \ b$
$q = \lfloor a/b \rfloor$
$a = b$
$b = r$
$xx = xs[1]$
$yy = ys[1]$
$xs[1] = q \cdot xs[1] + xs[0]$
$ys[1] = q \cdot ys[1] + ys[0]$
$xs[0] = xx$
$ys[0] = yy$
$sign = -sign$
end while
// Calcule final des coefficients.
$x = sign \cdot xs[0]$
$y = -sign \cdot ys[0]$
// détermination de $pgcd(a, b)$
$pgcd = a$
end

1.1.5 Théoréme de Bézout

Corollaire 1.1.2 (*Identité de Bézout*) *: Soient a et b sont des entiers non nuls et d leur pgcd. Il existe deux entiers u et v tels que :*

$$au + bv = d.$$

Exemple 1.1.6 *Chercher deux entiers u et v tels que* $96u + 81v = 3$.

On part de l'avant-dernière ligne dans l'algorithme d'Euclide, puis on remonte successivement jusqu'à la 1^{re} formule:

$$\begin{aligned} 3 &= 15 - 6 \cdot 2 \\ &= 15 - (81 - 5 \cdot 15) \cdot 2 \\ &= -2 \cdot 81 + 11 \cdot 15 \\ &= -2 \cdot 81 + 11 \cdot (96 - 81) \\ &= -13 \cdot 81 + 11 \cdot 96 \end{aligned}$$

Donc :

$$11 \cdot 96 - 13 \cdot 81 = 3.$$

Théorème 1.1.5 (*Théorème de Bézout*) *Soient a et b sont des entiers non nuls. a et b sont premiers entre eux si et seulement s'il existe deux entiers u et v tels que :*

$$au + bv = 1.$$

1.2 Structure algébrique

1.2.1 Opérations

Une loi de composition interne $*$, ou opération $*$ sur un ensemble E est donc tout simplement une application de $E \times E$ dans E, que l'on note

$$(x, y) \to x * y.$$

Définition 1.2.1 *Une opération $*$ sur un ensemble E est dite*

1. Associative, si
$$\forall(x,y,z) \in E \times E \times E, (x * y) * z = x * (y * z).$$
2. Posséder un élément neutre, s'il existe un élément $e \in E$ vérifiant
$$\forall x \in E,\ e * x = x * e = x.$$
3. Commutative, si
$$\forall(x,y) \in E \times E, x * y = y * x.$$
4. posséder un élément symétrique, ou est inversible (en notation multiplicative) ou possède un opposé (en notation additive), s'il existe un élément $y \in E$ vérifiant
$$x * y = y * x = e$$
En notation multiplicative, on dit alors que y est l'inverse de x et on note $y = x^{-1}$
En notation additive, on dit que y est l'opposé de x, on note $y = -x$.

1.2.2 Groupes

Définition 1.2.2 *Un groupe est la donnée d'un ensemble G muni d'une opération interne possédant les propriétés suivantes.*

$$\begin{cases} \textit{associative} \quad \textit{i.e} & \forall(x,y,z) \in G^3, (x * y) * z = x * (y * z). \\ \textit{possède un élément neutre} \quad \textit{i.e.} & \forall x \in G, e * x = x * e = x. \\ \textit{tout élément de } G \textit{ admet un inverse} \quad \textit{i.e.} & x * y = y * x = e \end{cases}$$

Si de plus l'opération est commutative on dit que le groupe est commutatif ou abélien.

Exemple 1.2.1 *$(Z, +)$ est un groupe commutatif.*

Proposition 1.2.1 *Soit G un groupe, et soit x et y deux éléments de G, alors*
$$(xy)^{-1} = y^{-1}x^{-1}$$

Preuve.

$$(xy)(y^{-1}x^{-1}) = x(yy^{-1})x^{-1} = x1x^{-1} = xx^{-1} = 1.$$

■

Définition 1.2.3 *Un groupe G est fini si l'ensemble G est fini. Le nombre d'éléments de G est alors appelé ordre du groupe G.*

Exemple 1.2.2 *le groupe G_n de l'ensemble des permutations $\{1, 2, ..., n\}$ est un groupe fini d'ordre $n!$.*

Définition 1.2.4 *Soit G un groupe. Une partie H de G est un sous-groupe de G si les conditions suivantes sont réalisées*

1. $\forall (x, y) \in H \times H, \ \ xy \in H$.
2. $1 \in H$.
3. $\forall x \in H, \ \ x^{-1} \in H$.

Exemple 1.2.3 *Z est un sous-groupe de Q, qui est un sous-groupe de R, lequel est un sous groupe de C.*

Proposition 1.2.2 *Avec les notations ci-dessus, H est un sous-groupe de G si et seulement si*

1. $H \neq \emptyset$.
2. $\forall (x, y) \in H \times H, \ \ xy^{-1} \in H$.

Preuve. Il existe $x \in H$ d'après 1. il résulte alors de 2. que $xx^{-1} = 1 \in H$.

On en déduit que pour tout $y \in H$, $1 \cdot y^{-1} = y^{-1} \in H$.

Il en résulte que si $\forall (x, y) \in H \times H$, donc $x(y^{-1})^{-1} = xy \in H$. ■

Définition 1.2.5 *Soit g un élément d'un groupe G. S'il existe un entier strictement positif n tel que $g^n = e$, alors on peut choisir n minimal avec cette propriété. On dit alors que g est un élément d'ordre n, et on le note $n = ord_g$.*

S'il n'existe pas d'entier strictement positif n tel que $g^n = e$, on dit que g est un élément d'ordre infini.

Théorème 1.2.1 (*Lagrange*) *Si G est un groupe fini et H est un sous-groupe de G, alors l'ordre de H divise l'ordre de G.*

Preuve. (*voir* [1]) . ■

Définition 1.2.6 *Un groupe G est cyclique s'il est fini et s'il existe un élément $g \in G$, appelé générateur de G, tel que $G = \langle g \rangle$, ce qui équivaut à l'égalité*

$$ord_g = ord_G.$$

On rappelle également que tout groupe cyclique est commutatif.

1.2.3 Anneaux Z/nZ

Définition 1.2.7 *Un anneau est la donnée d'un ensemble A muni de deux opérations, une addition et une multiplication, vérifiant*

1. $(A, +)$ est un groupe commutatif, d'élément neutre noté 0.

2. La multiplication est associative et possède un élément neutre noté 1, appelé élément unité.

3. La multiplication est distributive par rapport à l'addition, c'est-à-dire

$$\forall (x, y, z) \in A^3, \begin{cases} x(y+z) = xy + xz \\ (y+z)x = yx + zx \end{cases}$$

Si la multiplication est commutative, on dit que l'anneau A est commutatif.

Exemple 1.2.4 $(Z, +, \cdot), (Q, +, \cdot), (R, +, \cdot), (C, +, \cdot)$ *Sont des anneaux commutatifs.*

Définition 1.2.8 (*congruence*) *Soit n un entier positif. On dit que deux entiers a et b sont congrus modulo n si leur différence $(b - a)$ est multiple de n, c'est à dire si $(b - a) \in nZ$. Cette relation est notée*

$$a \equiv b \ (mod\ n) \quad ou\ bien \quad a \equiv b\ mod\ n$$

La notion de congruence modulo n a été introduite par Gauss.

Exemple 1.2.5 $12 \equiv 2 \ (mod\ 5)$

Proposition 1.2.3 *Si a, b, c, d, m et n sont des entiers relatifs,*

- $a \equiv a \ (mod\ m)$,
- *si* $a \equiv b \ (mod\ m)$, *alors* $b \equiv a \ (mod\ m)$,
- *si* $a \equiv b \ (mod\ m)$ *et* $b \equiv c \ (mod\ m)$, *alors* $a \equiv c \ (mod\ m)$,
- *si* $a \equiv b \ (mod\ m)$ *et* $c \equiv d \ (mod\ m)$, *alors* $a + c \equiv b + d \ (mod\ m)$,
- *si* $a \equiv b \ (mod\ m)$ *et* $c \equiv d \ (mod\ m)$, *alors* $ac \equiv bd \ (mod\ m)$,
- *si* $a \equiv b \ mod\ m$ *et si* n *est un entier positif, alors* $a^n \equiv b^n \ (mod\ m)$.

On vérifie que la relation $\equiv$ est une relation d'équivalence sur Z compatible avec l'addition et la multiplication, c'est-à-dire que si $a \equiv a' \ (mod\ n)$ et si $b \equiv b' \ (mod\ n)$ alors

$$a + b \equiv a' + b' (mod\ n) \ \text{ et } \ ab \equiv a'b' (mod\ n).$$

pour tout entier a, on note $a\ mod\ n$ la classe de congruence (ou la classe résiduelle) de a *modulo* n, et s'il n'y a pas d'ambiguité, on utilisera aussi la notion $\overline{a}$. Autrement dit,

$$a\ mod\ n = \overline{a} = \{x \in Z : x \equiv a \ (mod\ n)\}.$$

On note Z/nZ le quotient de Z avec cette relation d'équivalence.

Pour $\overline{a}, \overline{b} \in Z/nZ$, on pose $\overline{a} + \overline{b} = \overline{a + b}$ *et* $\overline{a}\overline{b} = \overline{ab}$.

On vérifie alors que $(Z/nZ, +, \cdot)$ est un anneau commutatif, dite l'anneau des classes de congruence (des classes résiduelles). De plus, le groupe $(Z/nZ, +)$ est l'unique groupe cyclique à n éléments, il est engendré par $\overline{1}$, à savoir

$$Z/nZ = \{\overline{0}, \overline{1}, ..., \overline{(n-1)}\}.$$

Définition 1.2.9 *Soit $\overline{x}$ un élément de Z/nZ, on dit que $\overline{x}$ est inversible si et seulement si il existe $\overline{y} \in Z/nZ$ tel que $\overline{xy} = \overline{1}$.*

Alors cet élément $\overline{y}$ est unique et s'appelle l'inverse de x et on le note $\overline{x}^{-1}$.

L'ensemble des éléments inversibles de Z/nZ est noté $(Z/nZ)^*$.

Chapitre 2

Arithmétique des congruences

2.1 Équation de Congruence

Le théorème suivant est l'un des outils les plus utiles et importantes en théories élémentaires des nombres [2].

Théorème 2.1.1 *Soit m, a, b des entiers avec $m \geq 1$. Soit $d = (a, m)$ le plus grand commun diviseur de a et m. La congruence*

$$ax \equiv b \ (mod \ m) \tag{2.1.1}$$

admet une solution si et seulement si

$$b \equiv 0 \ (mod \ d).$$

Si $b \equiv 0 \ (mod \ d)$, alors la congruence (2.1.1) admet exactement d solutions qui sont deux à deux incongrues modulo m.

En particulier, si $(a, m) = 1$, alors pour tout entier b la congruence (2.1.1) a une solution unique modulo m.

Preuve. Soit $d = (a, m)$. Congruence (2.1.1) a une solution si et seulement s'il existe des entiers x et y tels que

$$ax - b = my$$

ou, de manière équivalente,

$$b = ax - my.$$

Cela est possible si et seulement si

$$b \equiv 0 \ (mod\ d).$$

Si x et x_1 sont solutions de (2.1.1) , alors

$$a(x_1 - x) = ax_1 - ax \equiv b - b \equiv 0 \ (mod\ m)$$

et ainsi

$$a(x_1 - x) = mz$$

pour un entier z. Si d est le plus grand commun diviseur de a et m,

$$(a/d, m/d) = 1 \ \text{ et } \ (a/d)(x - x_1) = (m/d)z$$

Par lemme (1.1.1) m/d divise $x_1 - x$. Alors

$$x_1 = x + im/d \qquad \text{pour un entier } i,$$

$$x_1 \equiv x \ (mod\ m).$$

De plus, chaque x_1 entier de cette forme est une solution de (2.1.1) . x_1 entier congru à x *modulo* m/d est congru à $x + im/d$ modulo m, $\forall i = 0, 1, ...d - 1$. ■

Exemple 2.1.1 1. *soit*

$$6x \equiv 7 \ (mod\ 8) \qquad i.e. \qquad \overline{6}x = \overline{7} \quad dans \ \ Z/8Z.$$

On cherche toutes les solutions revient à chercher toutes couples (x, y) *telles que*

$$6x + 8y = 7$$

Il n'y en aucune car $pgcd(6, 8) = 2$ *ne divise pas* 7.

2. *Considérons la congruence*

$$35x \equiv -14 \ (mod\ \ 91) \tag{2.1.2}$$

est résoluble puisque $(35, 91) = 7$ *et comme* $-14 = -2 \cdot 7$ *D'où*

$$-14 \equiv 0 \ (mod \ \ 7).$$

Alors l'équation possède 7 *solutions, et équivalente à la congruence*

$$5x \equiv -2 \ (mod \ 13) \tag{2.1.3}$$

Qui a l'unique $x \equiv 10 \ (mod \ 13)$. *Chaque solution de 2.1.2 satisfait* $x \equiv 10 \ (mod \ 13)$, *et ainsi un ensemble complet de solutions qui sont deux à deux incongru 91 est :*

$$\{10, 23, 36, 49, 62, 75, 88\}.$$

Lemme 2.1.1 *Soit* p *un nombre premier. Alors,*

$$x^2 \equiv 1 \ (mod \ p) \Longleftrightarrow x \equiv \pm 1 \ (mod \ p).$$

Preuve. Si $x \equiv \pm \ 1 \ (mod \ p)$, alors $x^2 \equiv 1 \ (mod \ p)$.

Inversement, si $x^2 \equiv 1 \ (mod \ p)$, alors p divise $x^2 - 1 = (x-1)(x+1)$, donc p doit diviser $x - 1$ ou $x + 1$.

D'où

$$x \equiv \pm \ 1 \ (mod \ p).$$

■

2.2 L'indicatrice d'Euler

Définition 2.2.1 *Soit* n *un entier positif, l'indicatrice d'Euler de* n, *notée* $\varphi(n)$, *est définie comme étant égale au nombre des entiers* k *vérifiant*

$$(1 \leq k \leq n) \qquad et \qquad pgcd(k, n) = 1 \tag{(1)}$$

Notons que pour tout entier positif n, *on a* $pgcd(1, n) = 1$, *ce qui fait que* $\varphi(n) \geq 1$.

Exemple 2.2.1 *On trouvera quelques valeurs de* $\varphi(n)$ *dans le tableau suivant*

n	1	2	3	4	5	6	7	8	9	10	...
$\varphi(n)$	1	1	2	2	4	2	6	4	6	4	...

Théorème 2.2.1 *Soit* m *et* n *entiers premiers positifs. Pour chaque entier* c *il existe des entiers uniques* a *et* b *tels que*

$$0 \leq a \leq n-1,$$

$$0 \leq b \leq m-1,$$

et

$$c \equiv ma + nb \ (mod \ \ mn). \tag{2.2.1}$$

De plus, $(c, mn) = 1$ *si et seulement si* $(a, n) = (b, m) = 1$ *dans la représentation* (2.2.1).

Exemple 2.2.2 *Nous pouvons représenter les classes de congruence modulo* 6 *de la forme linéaire des combinaisons de* 2 *et* 3,

comme suit:

$$0 \equiv 0 \cdot 2 + 0 \cdot 3 \ (mod \ 6),$$

$$1 \equiv 2 \cdot 2 + 1 \cdot 3 \ (mod \ 6),$$

$$2 \equiv 1 \cdot 2 \cdot 3 + 0 \ (mod \ 6),$$

$$3 \equiv 0 \cdot 2 + 1 \cdot 3 \ (mod \ 6),$$

$$4 \equiv 2 \cdot 2 + 0 \cdot 3 \ (mod \ 6),$$

$$5 \equiv 1 \cdot 2 \cdot 3 + 1 \ (mod \ 6).$$

Définition 2.2.2 *Une fonction arithmétique* f *est dite multiplicative si pour toutes les paires premiers positifs* m *et* n

$$f(mn) = f(m) \cdot f(n)$$

Théorème 2.2.2 *La fonction d'Euler est multiplicatif. De plus*

$$\varphi(m) = m \prod\nolimits_{(p|m)} (p - 1/p)$$

Preuve. *Soit* $(m,n)=1$. *Il y a* $\varphi(mn)$ *classes de congruence dans l'anneau* Z/mnZ *qui sont relativement premier avec* mn.

D'après le théorème (2.2.1), *tous les classes des congruences modulo* mn *peut être écrit de manière unique sous la forme* $ma+nb+nb+mnZ$, *où* a *et* b *sont des entiers tels que*

$$0 \leq a \leq n-1 \quad et \quad 0 \leq b \leq m-1$$

En outre, la classe de congruence $ma+nb+mnZ$ *est premier avec* mn *si et seulement si* $(b,m)=(a,n)=1$. *Alors il existe* $\varphi(n)$ *entiers* $a \in [0, n-1]$, *qui sont relativement premier avec* n, *et* $\varphi(m)$ *entiers* $b \in [0, m-1]$ *relativement premier à* m, *il se ensuit que*

$$\varphi(mn) = \varphi(m) \cdot \varphi(n)$$

Donc la fonction d'Euler φ *est multiplicative.*

Si $m_1, ..., m_k$ *sont deux à deux premiers positifs, alors*

$$\varphi(m_1 \cdots m_k) = \varphi(m_1) \cdots \varphi(m_k).$$

En particulier, si $m = p_1^{r_1} \cdots p_k^{r_k}$ *la factorisation de* m, *où* $p_1, \ldots, p_k$ *sont des premiers distincts, et* $r_1, \ldots, r_k$ *sont des entiers positifs, alors*

$$\varphi(m) = \prod_{i=1}^{k} \varphi\left(p_i^{r_i}\right) = \prod_{i=1}^{k} p_i^{r_i}\left(1-\frac{1}{p_i}\right) = m\prod_{(p|m)}(1-\frac{1}{p})$$

■

Exemple 2.2.3 *On a* $7875 = 3^2 \cdot 5^3 \cdot 7$

et

$$\varphi(7875) = \varphi(3) \cdot \varphi(5^3) \cdot \varphi(7) = (9-3)(125-25)(7-1) = 3600.$$

Théorème 2.2.3 *Pour tout entier positif* m, *on a*

$$\sum_{(d/m)} \varphi(d) = m$$

Exemple 2.2.4

$$\sum_{(d/10)} \varphi(d) = \varphi(1) + \varphi(2) + \varphi(5) + \varphi(10)$$

$$= 1 + 1 + 3 + 5$$

$$= 10.$$

Corollaire 2.2.1 *le groupe multiplicatif de résidus mod p est cyclique d'ordre* $p-1$. *Il a exactement* $\varphi(p-1)$ *générateurs.*

Définition 2.2.3 *Un entier a dont la classe résiduelle* $a+pZ$ *engendre le groupe multiplicatif* $(Z/pZ)^*$ *s'appelle une racine primitive mod p.*

Exemple 2.2.5 *Pour* $p = 13$, *nous avons* $p-1 = 12$. *Le théorème* (2.2.2) *donne* $\varphi(12) = 4$. *Il existe donc quatre racine primitives mod* 13; *ce sont* $2, 6, 7,$ *et* 11.

2.3 Système de congruences (restes chinois)

Ce problème trouve sa solution dans le théorème chinois [10]; les ⟨Chinois de l'antiquité⟩ , savaient résoudre le type de problème suivant :

"Mon panier peut contenir au plus 100 œufs. Si je le vide par 3 œufs à la fois, il en reste 1, Si je le vide par 5 œufs à la fois, il en reste 2, et si je le vide par 7 œufs à la fois, il en reste 3. Combien ai-je d'œufs ? "

Dans le langage des congruences, le problème des œufs que nous avons donné en exemple consiste à trouver les x au plus égaux à 100 pour lesquel

$$x \equiv 1 \ (mod\ 3) \qquad (1)$$

$$x \equiv 2 \ (mod\ 5) \qquad (2)$$

$$x \equiv 3 \ (mod\ 7) \qquad (3)$$

Pour que x soit une solution de (1), x doit être de la forme $x = 1 + 3k$.

L'image de cette valeur dans $Z/5Z$ est $\overline{(1+3k_1)}$ par (2), c'est aussi $\overline{2}$, donc $\overline{1} + \overline{3k_1}$=$\overline{2}$.

D'où $\overline{3k_1} = \overline{2}$ dans $Z/5Z$, c'est-a-dire que k_1 est de la forme k_1=2+5k_2. D'où

$$x = 1 + 3k = 1 + 3(2 + 5k_2) = 7 + 15k_2$$

L'image de cette valeur dans $Z/7Z$ est $\overline{k_2}$ mais aussi $\overline{3}$ par (3). Il s'ensuit que k_2 est de la forme $k_2 = 3 + 7k_3$. D'où

$$x = 7 + 15(3 + 7k_3) = 52 + 105k_3$$

Donc pour que x satisfasse le système $(1), (2)$ et (3), il doit être de la forme $52 + 105k$.

Comme d'autre part

$$\overline{(52 + 105k)} = \begin{cases} \overline{1} \ sur \ Z/3Z \\ \overline{2} \ sur \ Z/5Z \\ \overline{3} \ sur \ Z/7Z \end{cases}$$

$\{52 + 105k, k \in Z\}$ est l'ensemble des solutions du système de congruences.

La réponse au problème est donc 52 œufs.

Théorème 2.3.1 (*théorème des restes chinois*) *Soit $m_1, m_2, \ldots, m_k$ une suite d'entiers positifs deux à deux premiers entre eux. Alors le système de congruences :*

$$\begin{cases} x \equiv a_1 \ (mod \ m_1) \\ x \equiv a_2 \ (mod \ m_2) \\ \qquad ... \\ x \equiv a_k \ (mod \ m_k) \end{cases}$$

a une solution unique $x \ mod \ (M = m_1 \cdot m_2 \cdots m_k)$,

$$x = a_1 M_1 y_1 + a_2 M_2 y_2 + \cdots + a_k M_k y_k$$

avec $M_i = M/m_i$ et $y_i M_i = 1 \ (mod \ m_i)$

Preuve. Démontrons le théorème dans le cas $k = 2$. (Une récurrence permet de le démontrer dans le cas général). On veut résoudre le système suivant :

$$(S) : \begin{cases} x \equiv a_1 \ (mod \ m_1) \\ x \equiv a_2 \ (mod \ m_2) \end{cases}$$

Supposons que x soit solution du système (S) alors la première équation nous donne l'existence d'un entier k tel que

$$x = km_1 + a_1$$

et la seconde congruence s'écrit alors :

$$a_1 + km_1 \equiv a_2 \ (mod\ m_2)$$

i.e. $k \equiv (a_2 - a_1)m_1' \ (mod\ m_2)$ où m_1' désigne un inverse de m_1 modulo m_2 et m_1' existe, car m_1 et m_2 sont premiers entre eux. Ainsi, si l'on pose :

$$a = (a_2 - a_1)m_1m_1' + a_1$$

on a bien

$$x \equiv a \ (mod\ m_1m_2).$$

Réciproquement, si $x \equiv a \ (mod\ m_1m_2)$ avec $a = (a_2 - a_1)\ m_1m_1' + a_1$, on a bien :

$$x \equiv a_1 \ (mod\ m_1)$$

et

$$x \equiv a_2 \ (mod\ m_2)$$

■

Exemple 2.3.1 *Soit le système des congruences du problème précédent* $(œufs)$ *:*

$$\begin{cases} x \equiv 1 \ (mod\ 3) & (1) \\ x \equiv 2 \ (mod\ 5) & (2) \\ x \equiv 3 \ (mod\ 7) & (3) \end{cases}$$

On pose $M = 3 \times 5 \times 7 = 105$

$$M_1 = 105/3 = 35 \qquad y_1 \times 35 \equiv 1 \ (mod\ 3) \qquad y_1 = 2$$

$$M_2 = 105/5 = 21 \qquad y_2 \times 21 \equiv 1 \ (mod\ 5) \qquad y_2 = 1$$

$$M_3 = 105/7 = 15 \qquad y_3 \times 15 \equiv 1 \ (mod\ 7) \qquad y_3 = 1$$

$$x \equiv 1 \times 35 \times 2 + 2 \times 21 \times 1 + 3 \times 15 \times 1 \equiv 157 \ (mod\ 105)$$

$$x \equiv 52 \ (mod\ 105)$$

Algorithme 2.3.1 *Chinois des restes en langage C.*

```
chrPrecomp (int modules [ ], int nbrmodules, int module, int multp [ ] )
begin
  int i, m, M, inverse, pgcd, y
      module = 1;
        for (i = 0; i < nbrmodules; i = i + 1)
              module = module · modules [i]
        end for
        for (i = 0; i < nbrmodules; i = i + 1)
              m = modules [i];
              M = module / m;
              Xeuclide (M, m, pgcd, inverse, y);
              multp [i] = inverse · M;
        end for
end
chr (int modules [ ], int nbrmodules, int résultat, int x [ ] )
begin
    int multp [nbrmodules]
    int résultat=0
    int modulus, i
    chrPrecomp ( modules, nbrmodules, module, multp )
        for (i = 0; i < nbrmodules; i = i + 1)
              résultat = (résultat + multp [i] · x [i])
       end for
end
```

Exemple 2.3.2 *Quand les modulos ne sont pas premiers entre eux.*

Soit le système de congruence suivant :

$$\begin{cases} x \equiv 1 \ (mod\ 6) \\ x \equiv 4 \ (mod\ 15) \end{cases}$$

$$x \equiv 1 \ (mod\ 6) \Longleftrightarrow \begin{cases} x \equiv 1 \ (mod\ 2) \\ x \equiv 1 \ (mod\ 3) \end{cases}$$

et

$$x \equiv 4 \ (mod\ 15) \Longleftrightarrow \begin{cases} x \equiv 1 \ (mod\ 3) \\ x \equiv 4 \ (mod\ 5) \end{cases}$$

Donc

$$\begin{cases} x \equiv 1 \ (mod\ 6) \\ x \equiv 4 \ (mod\ 15) \end{cases} \Longleftrightarrow \begin{cases} x \equiv 1 \ (mod\ 2) \\ x \equiv 1 \ (mod\ 3) \\ x \equiv 4 \ (mod\ 5) \end{cases}$$

Application : Un phare émet un signal jaune toutes les 15 secondes et un signal rouge toutes les 28 secondes.On aperçoit le signal jaune 2 secondes après minuit et le rouge 8 secondes après minuit. A quelle heure verra-t-on pour la première fois les deux signaux émis en même temps ? [11].

On commence par mettre en équation le problème. On note x les temps, en secondes, depuis minuit, où les deux phares sont allumés au même moment. À l'aide des données du problème, on peut dire que x est solution du système :

$$(s) \quad \begin{cases} x \equiv 2(mod\ 15) \\ x \equiv 8(mod\ 28) \end{cases}$$

On cherche le plus petit entier naturel x solution de ce système. On remarque que 15 et 28 sont premiers entre eux, on peut alors appliquer le théorème des restes chinois pour trouver les solutions de ce système. Cherchons alors une relation de Bézout entre 15 et 28.

On a par l'algorithme d'Euclide

$$\begin{aligned} 28 &= 15 \cdot 1 + 13; \\ 15 &= 13 \cdot 1 + 2; \\ 13 &= 2 \cdot 6 + 1. \end{aligned}$$

En remontant les calculs, on obtient :

$1 = 13 - 2 \cdot 6 = 13 - 6 \cdot (15 - 13) = -6 \cdot 15 + 7 \cdot 13 = -6 \cdot 15 + 7 \cdot (28 - 15)$

$= -13 \cdot 15 + 7 \cdot 28.$

Ainsi, la relation : $-13 \cdot 15 + 7 \cdot 28 = 1$ est une relation de Bézout entre 15 et 28. On pose alors

$$a_1 = 2,\ a_2 = 8,\ m_1 = 15,\ m_2 = 28,\ u = -13 \ \ et \ \ v = 7.$$

On calcule alors

$$a = (a_2 - a_1)m_1u + a_1 = a_2m_1u + a_1m_2v = (8-2) \times 15 \times (13) + 2 = -1168$$

et on peut déduire que

$$x \equiv a \ (mod\ m_1m_2)$$

(i.e: $x \equiv -1168\ (mod\ 15 \times 28)$). Ainsi,

$$x \equiv -1168\ (mod\ 420).$$

Les solutions de ce système sont alors les entiers congrus à -1168 *modulo* 420, c'est-à-dire les entiers de la forme : $420k - 1168$, $k \in Z$. On cherche le plus petit entier naturel qui est solution de ce système.

Pour $k = 2$, $420k - 1168 < 0$, mais pour $k = 3$, $420k - 1168 > 0$ et on a :

$$420 \times 3 - 1168 = 92.$$

Donc le plus petit entier naturel solution de ce système est 92. Ainsi, les deux phares seront allumés au même moment, pour la première fois, $1^{\min}$ *et* 32^s après minuit.

Proposition 2.3.1 *Il est une application importante de théorème du reste chinois pour le problème de la résolution des équations diophantiennes de la forme*

$$f(x_1, ..., x_k) \equiv 0\ (mod\ m),$$

où $f(x_1, ..., x_k)$ est un polynôme avec des coefficients entiers en plusieurs Variables. Cette équation est résoluble modulo m si il existe des entiers $a_1, ..., a_k$ tels que

$$f(a_1, ..., a_k) \equiv 0\ (mod\ m).$$

Le théorème des restes chinois nous permet de réduire la question de la solvabilité de cette congruence modulo m au cas particulier des puissances modulo p^r. Pour plus de simplicité, nous considérons des polynômes en une seule variable [2].

Théorème 2.3.2 *Soit*

$$m = p_1^{r_1} \cdots p_k^{r_k}$$

la factorisation de l'entier positif m.

Soit $f(x)$ un polynôme à coefficients entiers. La congruence

$$f(x) \equiv 0 \ (mod\ m)$$

est résoluble si et seulement si

$$f(x) \equiv 0 \ (mod\ p_i^{r_i})$$

sont résoluble pour tout $i = \overline{1 \cdot k}$.

Exemple 2.3.3 *Considérez la congruence*

$$f(x) = x^2 - 34 \equiv 0 \ (mod\ 495)$$

on a $495 = 3^2 \cdot 5 \cdot 11$, *il suffit de résoudre les congruences*

$$f(x) = x^2 - 34 \equiv x^2 + 2 \equiv 0 \ (mod\ 9),$$

$$f(x) = x^2 - 34 \equiv x^2 + 1 \equiv 0 \ (mod\ 5),$$

et

$$f(x) = x^2 - 34 \equiv x^2 - 1 \equiv 0 \ (mod\ 11).$$

Ces congruences ont les solutions

$$f(5) \equiv 0 \ (mod\ 9)$$

$$f(2) \equiv 0 \ (mod\ 5)$$

et

$$f(1) \equiv 0 \ (mod\ 11).$$

Par le théorème des restes chinois, il existe un entier a tel que

$$a \equiv 5 \ (mod\ 9),$$

$$a \equiv 2 \ (mod\ 5),$$

$$a \equiv 1 \ (mod\ 11)$$

La résolution de ces congruences, nous obtenons

$$a \equiv 122 \ (mod\ 495).$$

Nous pouvons vérifier que

$$f(122) = 122^2 - 34 = 14850 = 30 \cdot 495,$$

et donc

$$f(122) \equiv 0 \ (mod\ 495).$$

2.4 Théorème de Wilson

Théorème 2.4.1 $(Wilson)$ *Si p est premier, alors*

$$(p-1)! \equiv -1 \ (mod\ p).$$

Preuve. Vrai pour $p = 2$ et $p = 3$, car $1! \equiv -1 \ (mod\ 2)$ et $2! \equiv -1 \ (mod\ 3)$.

Soit $p \geq 5$. D'après le théorème (2.1.1), chaque entier a de $\{1, 2, ..., p-1\}$ existe un entier unique $a^{-1} \in \{1, 2, ..., p-1\}$ tel que

$$aa^{-1} \equiv 1 \ (mod\ p)$$

D'après le lemme (2.1.1), $a = a^{-1}$ si et seulement si $a = 1$ ou $a = p - 1$.

Par conséquent, on peut partitionner $p - 3$ chiffres dans $\{2, 3, ..., p-2\}$ en $(p-3)/2$ couples d'entiers $\{a_i, (a_i)^{-1}\}$ tel que

$$a_i \ a_i^{-1} \equiv 1 \ (mod\ p) \quad \text{pour } i = 1, ..., (p-3)/2.$$

Alors

$$\begin{aligned}(p-1)! &\equiv 1 \cdot 2 \cdot 3 \cdots (p-2)(p-1) \\ &\equiv (p-1) \prod_{i=1}^{(P-3)/2)} a_i \ a_i^{-1}\end{aligned}$$

$$\equiv (p-1)$$

$$\equiv -1 \ (mod\ p).$$

■

Exemple 2.4.1 $(5-1)! = 4! \equiv 24 \equiv -1 \ (mod\ 5)$.

Remarque 2.4.1 *L'inverse de théorème de Wilson est aussi vrai. C'est-à-dire si p est un entier positif avec* $(p-1)! \equiv -1 \ (mod\ p)$. *Alors p est premier.*

Voila, quelques applications sur le théorème de Wilson [1].

Corollaire 2.4.1 *Soit p un nombre premier et soit k un entier,* $0 < k < p$. *Alors*

$$(k-1)!(p-k)! \equiv (-1)^k \ (mod\ p).$$

Preuve. D'après le théorème de Wilson,

$$(p-k)! = (p-1)\cdots(p-(k-1))(p-k)! \equiv (-1)^{k-1}(k-1)!(p-k)!$$

$$\equiv -1 \ (mod\ p),$$

et en multipliant par $(-1)^{k-1}$, on obtient $(-1)^{k-1}\cdot\ (-1)^{k-1}(k-1)!(p-k)! \equiv (-1)^{k-1}(-1)$ $(mod\ p) = (-1)^k \ (mod\ p)$. ■

Corollaire 2.4.2 *Soit p un nombre premier et soit r un entier tel que* $1 \leq r \leq p$. *Si*

$$(-1)^r r! \equiv 1 \ (mod\ p)$$

alors

$$(p-r-1)! \equiv -1 \ (mod\ p).$$

et on déduit de ce résultat que

$$259! \equiv -1 \ (mod\ \ 269)$$

Preuve. D'après le théorème de Wilson.

$$(p-1)! = (p-1)(p-2)\cdots(p-r)(p-r-1)! \equiv (-1)^r r!(p-r-1)!$$

$$\equiv -1 \ (mod\ p),$$

Puisque $(-1)^r r! \equiv 1 \ (mod\ p)$, on obtient le résultat.

Pour la deuxième partie, on a $259! = (269-9-1)$, $r = 9$. Il suffit de remarque que $(-1)^9\ 9! \equiv 1 \ (mod\ 269)$. ■

Corollaire 2.4.3 *soit p un nombre premier de la forme $4n+1$, alors*

$$[(2n)!]^2 \equiv -1 \ (mod\ p).$$

De plus générale, si p est un nombre premier et si $m+n = p-1$, $m \geq 0, n \geq 0$, alors

$$m!n! \equiv (-1)^{m+1} \ (mod\ p).$$

Un résultat analogue a celui-ci a été obtenu au corrolaire . On déduit de cette dernière formule que

$$\left(\frac{p-1}{2}!\right)^2 \equiv (-1)^{m+1} \ (mod\ p).$$

Preuve. D'après le théorème de Wilson, on a $(4n)! \equiv -1 \ (mod\ p)$, auquel cas

$$(4n)(4n-1)\cdots[4n-(2n-1)](2n)! \equiv -1 \ (mod\ p).$$

Comme

$$4n = p-1 \equiv -1 \ (mod\ p)$$

on a

$$4n-1 \equiv -2 \ (mod\ p),$$

et donc

$$4n-2 = p-3 \equiv -2 \ (mod\ p),$$

de sorte que

$$4n-(2n-1) \equiv -2n \ (mod\ p),$$

et le résultat suit.

Pour la généralisation, on a, d'après le théorème de Wilson,

$$(m+n)! \equiv -1 \ (mod\ p)$$

et alors

$$(m+n)(m+n-1)\cdots[m+n-(n-1)]m! \equiv -1 \ (mod\ p) \qquad (*)$$

Comme

$$m+n = p-1 \equiv -1 \ (mod\ p)$$

et

$$m+n-1 \equiv -2 \ (mod\ p),$$

et ainsi de suite pour enfin avoir

$$m+n-(n-1) \equiv -n \ (mod\ p),$$

on substituant dans $(*)$, on trouve

$$(-1)^n m!n! \equiv -1 \ (mod\ p) \qquad (**)$$

Puisque $m+n$ est pair, la deuxième relation du corrolaire est démontrée.

Enfin la dernière congruence s'obtient en posant $m = n = (p-1)/2$ dans $(**)$. ■

2.5 Théorème de Fermat et théorème d'Euler

Nous allons nous intéresser aux puissances successives d'un entier $a \ (mod\ m)$ lorsque $(a, m) = 1$. Commençons par un exemple.

Exemple 2.5.1 *Prenons $m = 7$ et considérons l'anneau $Z/7Z$. Les unités dans cet anneau sont toutes les classes non nulles. Si l'on calcule les puissances de 2, on trouve que*

$$2^2 = 4 \equiv 4 \ (mod\ 7) \qquad 2^3 = 8 \equiv 1 \ (mod\ 7)$$

Ce qui donne, en d'autre termes que $\overline{2}^3 = \overline{1}$ dans $Z/7Z$.

On trouve encore que

$$3^2 = 9 \equiv 2 \ (mod\ 7) \qquad 3^3 = 27 \equiv -1 \ (mod\ 7)$$

$$3^4 = -1 \cdot 3 \equiv 4 \ (mod\ 7) \qquad 3^5 = 4 \cdot 3 \equiv 5 \ (mod\ 7)$$

$$3^6 = 15 \equiv 1(mod 7)$$

Ce qui montre que $\overline{3}^6 = \overline{1}$ *dans* $Z/7Z$ *et* 6 *est le plus petit entier positif* n *tel que* $\overline{3}^n = \overline{1}$.

On peut montrer qu'il existe toujours un entier n *tel que*

$$a^n \equiv 1 \ (mod\ m).$$

Proposition 2.5.1 *Soient* a *et* m *des entiers tels que* $(a, m) = 1$ *et* $m \geq 1$. *Alors il existe un entier* n *tel que*

$$a^n \equiv 1 \ (mod\ m).$$

Définition 2.5.1 (*ordre d'un entier* $(mod\ m)$) *Soit* m *un entier* > 1 *et* a *un entier tel que* $(a, m) = 1$. *Le plus petit entier positif* n *tel que*

$$a^n \equiv 1 \ (mod\ m)$$

est appelé l'ordre de a *modulo* m.

Exemple 2.5.2 *les exemples ci-dessus ont montré que l'ordre de* $2(mod\ 7)$ *est égal à* 3 *alors que l'ordre de* $3(mod\ 7)$ *vaut* 6.

L'entier 1 *est évidemment toujours d'ordre* 1 *quelque soit* m.

Théorème 2.5.1 (*Théorème Euler*) *Soit* a *et* m *deux entiers tels que* $m > 1$, *et* $(a, m) = 1$. *Alors*

$$a^{\varphi(m)} \equiv 1 \ (mod\ m).$$

Preuve. Soit $\{r_1, ..., r_{\varphi(m)}\}$ un ensemble de résidus modulo m. Comme $(a, m) = 1$, nous avons $(ar_i, m) = 1$ pour $i = 1, ..., \varphi(m)$. Par conséquent, pour tout $i \in \{1, ..., \varphi(m)\}$, il existe $\sigma(i) \in \{1, ..., \varphi(m)\}$ tel que

$$ar_i \equiv r_{\sigma(i)}(mod\ m).$$

En outre, $ar_i \equiv ar_j(modm)$ si et seulement si $i = j$, et donc σ est une permutation de l'ensemble $\{1, ..., \varphi(m)\}$ et $\{ar_1, ..., ar_{\varphi(m)}\}$ est aussi un ensemble réduit de résidus modulo m. Il se ensuit que

$$a^{\varphi(m)} r_1 r_2 \cdots r_{\varphi(m)} \equiv (ar_1)(ar_2)(ar_{\varphi(m)}) \ (mod\ m)$$

$$\equiv r\sigma(1)\ r\sigma(2) \cdots r\sigma(\varphi(m)) \ (mod\ m)$$

$$\equiv r_1 \cdot r_2 \cdot r_{\varphi(m)} \ (mod\ m).$$

Divisant par $r_1 \cdot r_2 \cdot r_{\varphi(m)}$, nous obtenons

$$a^{\varphi(m)} \equiv 1(mod\ m).$$

■

Proposition 2.5.2 *Soit a un entier ≥ 2 et soit $m \in N$. Si*

$$(a, m) = (a - 1, m) = 1$$

Alors

$$1 + a + a^2 + \cdots + a^{\varphi(m)-1} \equiv 0 \ (mod\ m)$$

Preuve. *Puisque $(a, m) = 1$, alors d'après le théorème d'Euler, on a*

$$a^{\varphi(m)-1} \equiv 0 \ (mod\ m)$$

alors

$$a^{\varphi(m)-1} = (a - 1)\left(a^{a^{\varphi(m)-1}} + a^{a^{\varphi(m)-2}} + \cdots + a + 1\right)$$

et puisque $(a - 1, m) = 1$, donc

$$1 + a + a^2 + \cdots + a^{\varphi(m)-1} \equiv 0 \ (mod\ m)$$

■

Théorème 2.5.2 (*petit théoréme de Fermat*) *: Soit a un entier et p un premier ne divisant pas a. Alors*

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

Par ailleurs,

$$a^p \equiv a \ (mod\ p)$$

Preuve. C'est une conséquence immédiate du théorème d'Euler en sachant que

$$\varphi(p) = p - 1$$

lorsque p est premier. ■

2.6 Exponentiation rapide

Le théoréme de Fermat montre que l'entiere $x \equiv a^{\ell(m)-1} \ mod \ m$ est solution de la cogruence $ax \equiv 1 \ mod \ m$. Pour rendre efficace cette nouvelle méthode de résolution de cette congruence, il faut pouvoir calculer rapidement les puissances $mod \ m$.

Nous allons décrire un algorithme efficace permettant de calculer les puissances dans un groupe G. Cet algorithme et ses variantes sont des ingrédients centraux dans de nombreux protocoles cryptographiques.

Soient $g \in G$ et e un entier positif. Notons $e = \sum_{i=0}^{k} e_i 2^i$ le développement binaire de e. Observons que les coefficients e_i valent 0 au 1.

Par conséquent,

$$g^e = g^{\left(\sum_{i=0}^{k} e_i 2^i\right)} = \prod_{i=0}^{k} (g^{2^i})^{e_i} = \prod_{\substack{e_i=1 \\ 0 \leq i \leq k}} g^{2^i}$$

Cette formule donne l'idée suivante pour calculer g^e.

1. Calculer les carrés successifs g^{2^i}, $0 \leq i \leq k$.
2. Déterminer g^e comme le produit des g^{2^i} pour lesquels $e_i = 1$.

Observons que:

$$g^{2^{i+1}} = (g^{2^i})^2$$

par conséquent, $g^{2^{i+1}}$ peut être calculé à partir de g^{2^i} par une simple élévation au carré [7].

Exemple 2.6.1 *Pour déterminer* 6^{73} *mod* 100*, nous écrivons le développement binaire de l'expossant*:

$$73 = 1 + 2^3 + 2^6$$

Puis nous calculons les carrés successifs de 6 :

$$\begin{aligned} 6^2 &= 36, \\ 6^{2^2} &= 36^2 \equiv -4 \ mod\ 100, \\ 6^{2^3} &\equiv 16 \ mod\ 100, \\ 6^{2^4} &\equiv 16^2 \equiv 56 \ mod\ 100, \\ 6^{2^5} &\equiv 56^2 \equiv 36 \ mod\ 100, \\ 6^{2^6} &\equiv -4 \ mod\ 100. \end{aligned}$$

Donc

$$6 \cdot 6^{2^3} \cdot 6^{2^6} \equiv 6 \cdot 16 \cdot (-4) \equiv 16 \ mod\ 100.$$

Nous n'avons calculé que 6 carrés et 2 produit dans $(Z/100Z)^*$.

Si nous avions calculé 6^{73} *mod* 100 *comme* $6 \cdot 6 \cdots 6$ *mod* 100, *il aurait fallu* 72 *multiplications modulo*100.

Algorithme 2.6.1 *Exponentiation rapide en langage C.*

```
pow (élément du groupe base, entier exposant, élément du groupe résultat )
begin
    résultat=1
          while (exposant > 0)
             if   (Parité (exposant) == false)
                résultat = résultat · base
                base = base · base
                exposant = exposant/2
          end while
end
```

Chapitre 3

Cryptographie à clé publique

À l'heure de l'explosion des nouvelles technologies de l'information et de la communication, la Cryptographie est aujourd'hui essentielle pour le développement du commerce électronique, des cartes à puce, de la téléphonie mobile, et particulièrement cruciale dans le secteur bancaire, elle est devenue une discipline à deux facettes multiples qui concerne un public de plus en plus important.

La cryptographie traite de la transmission confidentielle de données. C'est l'étude des méthodes permettant de transmettre des messages sous forme déguisée, de telle sorte que seuls les destinataires autorisés soient capables de les lire. Le message à envoyer est appelé message ou texte en clair et, sous sa forme déguisée, message chiffré ou cryptogramme. Le codage est une transformation mathématique particulière, en général bijective, et la fiabilité de la plupart des cryptosystèmes modernes dépend essentiellement de la difficulté de cette transformation, dans le sens où le retour en arrière pour retrouver le message en clair nécessiterait, d'un éventuel indiscret, des moyens très coûteux.

Un chiffrement à clé publique (ou bien dite asymétrique) est un cryptage où l'algorithme de chiffrement n'est pas le même que celui de déchiffrement, et où les clés utilisées sont différentes. L'intérêt est énorme : il n'y a plus besoin de transmettre la clé à son destinataire, il suffit de publier librement les clés de cryptage. N'importe qui peut alors crypter un message, mais seul son destinataire, qui possède la clé de décodage, pourra le lire.

F1: schéma de cryptage publique.

3.1 RSA

Le cryptosystéme RSA, du nom de ses concepteurs, Ron Rivest, Adi Shamir et Leonard Adleman, est le premier algorithme de chiffrement asymétrique. Il a été découvert en 1977 au Massachusetts Institute of Technology.

3.1.1 Fabrication des clés

Expliquons comment Bernard fabrique ses deux clés RSA, la clé publique et la clé privée.

Bernard génère, au hasard et indépendamment, deux grands nombres premiers p *et* q puis il calcule le produit

$$n = p \cdot q$$

il choisit aussi un entier e avec

$$1 < e < \varphi(n) = (p-1)(q-1) \qquad \text{et} \qquad pgcd(e, \varphi(n)) = 1$$

On doit noter que e est toujours impair puisque $p-1$ est pair. Bernard calcule un entier d avec

$$1 < d < (p-1)(q-1) \quad \text{et} \quad d \cdot e \equiv 1 \; mod \; (p-1)(q-1) \tag{3.1.1}$$

Puisque $pgcd(e,(p-1)(q-1)) = 1$, le nombre d existe et on peut le calculer par l'algorithme d'Euclide étendu.

La clé publique de Bernard est le couple (n,e). Sa clé privée est d.

Le nombre n s'appelle le module. e s'appelle l'*exposant de chifffrement*, et d s'appelle l'*exposant de déchifffrement.* Il faut noter que la clé secrète d peut étre calculée à partir de l'exposant de chiffrement e si les facteurs premiers p et q de n sont connus. Par conséquent, si l'attaquant, Oscar, est capable de trouver la décomposition en facteurs premiers de n, il peut facilement trouver la clé secrète d de Bernard. Donc les facteurs p et q doivent étre choisis pour rendre la factorisation de n infaisable.

Exemple 3.1.1 *Bernard choisit* $p = 11$ *et* $q = 23$. *Alors* $n = 253$ *et* $(p-1)(q-1) = 10 \cdot 22 = 4 \cdot 5 \cdot 11 = 220$.

le plus petit e possible est $e = 3$, *puisque* $pgcd(3,220) = 1$.

L'algorithme d'Euclide étendu conduit à $d = 147$.

3.1.2 Chiffrement

Nous expliquons d'abord comment chiffrer les message avec le système RSA, puis nous montrons comment RSA peut étre utilisé pour le chiffrer par bloc.

Dans la premiére variante, l'espace des messages en clair est constitué de tout les entiers m avec

$$0 \leq m < n$$

un message en clair m est chiffré en calculant le cryptogramme

$$c = m^e \ mod \ n \tag{3.1.2}$$

Si on connaît une clé publique (n,e), on peut chiffré. Pour rendre le chiffrement efficace, on utilise l'exponentiation rapide.

Exemple 3.1.2 *Comme dans l'exemple précedent, soit* $n = 253$ *et* $e = 3$. *Alors l'espace des messages en clair est* $\{0,1,...,252\}$. *En chiffrant l'entier* $m = 165$, *on obtient*

$$165^3 \ mod \ 253 = 110.$$

Remarque 3.1.1 *Pour écrire des textes, nous avons besoin des symboles d'un alphbet. Par alphabet nous désignons un ensemble fini non vide* $\sum$. *Les éléments de* $\sum$ *sont appelés les symboles ou les lettres.*

Si k *est un entier non négatif,* $\sum^n$ *est l'ensemble de tous les mots de longueur* n *sur* $\sum$.

Maintenant nous montrons comment utilisé le RSA pour en faire un chiffrement par bloc.

Nous utilisons l'alphabet $\sum = Z/NZ = Z_N = \{0, 1, ..., N-1\}$ pour un certain entier positif N. Nous posons

$$k = \lfloor log_N\ n \rfloor \tag{3.1.3}$$

un mot $m_1 \cdot m_2 \cdots m_k \in \sum^k$.

correspond à l'entier

$$m = \sum_{i=1}^{k} m_i N^{k-i}$$

$$0 \leq m \leq (N-1) \sum_{i=1}^{k} N^{k-i} = N^k - 1 < n$$

Nous identifierons un mot de $\sum^k$ avec l'entier dont il est la représentation binaire.

Le bloc m est chiffré en calculant $c = m^e\ mod\ n$ et l'entier c est écrit en base N.

Le développment $N - addique$ de c a au plus la longueur $k+1$. Par conséquant, nous pouvons écrire

$$c = \sum_{i=0}^{k} c_i N^{k-i} \qquad c_i \in \sum,\ 0 \leq\ i \leq k$$

Le bloc chiffré est

$$c = c_1 \cdot c_2 \cdots c_k.$$

Exemple 3.1.3 *Nous continuons l'exemple* (3.1.1). *Soit* $\sum = \{0, A, B, C\}$*; nous faisons l'identification*

0	*A*	*B*	*C*
0	*1*	*2*	*3*

Avec le module $n = 253$, *nous avons* $k = \lfloor log_4\ 253 \rfloor = 3$. *C'est la longueur des blocs de message clair. La longueur des blocs de cryptogramme est* 4. *Nous chiffrons le bloc de message* ABB.

il correspond au bloc 122*, qui à son tour, correspond à l'entier*

$$m = 1 \cdot 4^2 + 2 \cdot 4^1 + 2 \cdot 4^0 = 26$$

cet entier est chiffré en

$$c = 26^3 mod\ 253 = 119$$

nous écrivons c *en base* 4*, ce qui donne*

$$c = 1 \cdot 4^3 + 3 \cdot 4^2 + 1 \cdot 4 + 3 \cdot 1$$

et finalement, le bloc cryptogramme est $ACAC$*.*

3.1.3 Déchiffrement

Le déchiffrement de RSA est basé sur le théorème suivant:

Théorème 3.1.1 *Soit* (n, e) *une clé publique RSA et soit* d *le clé privée qui lui correspond. Alors*

$$(m^e)^d\ mod\ n = m$$

Pour n'import quel entier m *avec* $0 \leq m < n$*.*

Preuve. Puisque $ed \equiv 1 mod (p-1)(q-1)$, il existe un entier l avec

$$ed = 1 + l(p-1)(q-1)$$

Par conséquant

$$(m^e)^d = m^{ed} = m^{1+l(p-1)(q-1)} = m(m^{(p-1)})^{(q-1)l} \equiv m\ mod\ p$$

si p n'est pas un diviseur de m, cette congruence résulte du théorème (2.5.2) de Fermat. Sinon, elle est vraie parce que les deux membres de la congruence sont $0\ mod\ p$. De façon analogue, on a

$$(m^e)^d \equiv m\ mod\ q$$

Et, parce que p et q sont des nombres premiers ditincts, ces deux congruences de donnent

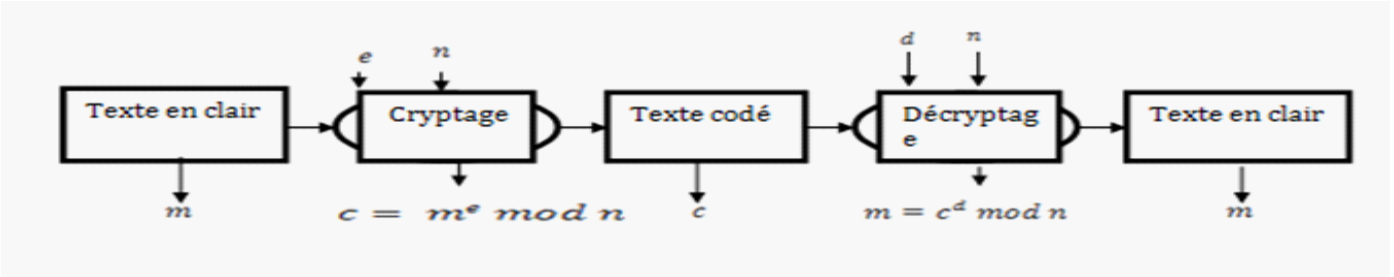
$$(m^e)^d \equiv m\ mod\ n$$

L'assertion résulte alors de fait que $0 \leq m < n$.

Le cryptogramme c ayant été calculé, le théoréme montre que le message en clair m peut étre retrouvé en calculant

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

Ce qui monter que le systéme RSA est bien un cryptosystéme, pour chaque fonction de chiffrement, il existe une fonction de déchiffrement. ■



F2: schéma de systéme RSA.

Exemple 3.1.4 *Nous concluons les exemples précedantes. Nous avons choisi* $n = 253$¸ $e = 3$ *et* $d = 147$¸ *puis nous avons calculé le cryptogramme* $c = 110$. *Alors en calculant*

$$110^{147} \ mod \ 253 = 165$$

Nous retrouvons¸ *en clair*¸ *le message qui a donné ce cryptograme.*

Remarque 3.1.2 *Tout d'abord, il faut un message préalablement codé. On doit donc transformer en nombres le message d'origine, par exemple en utilisant la valeur ASCII (American Standard Code for Information Interchange) de chaque lettre ou encore en remplaçant chaque lettre par son rang dans l'alphabet* [*voir* [6]].

Application: Voici un exemple de l'utilisation de RSA, avec des petits nombres [13].

Ali souhaiterait envoyer le message suivant à Samir : «**msila** ». Ces deux amis vont donc crypter leurs échanges avec la méthode RSA.

Ali a choisi

$$p = 37 \quad et \quad q = 43$$

Il en déduit

$$n = 37 \times 43 = 1591$$

et

$$\varphi(1591) = (37 - 1) \cdot (43 - 1) = 1512.$$

Il choisit ensuite

$$e = 19$$

qui est premier avec 1512.

L'inverse de 19 *modulo* 1512 est

$$d = 955$$

Ali peut donc maintenant publier sa clé publique $(e = 19, n = 1591)$.

Ali va utiliser ce clé pour crypter son message, mais il doit avant tout convertir son *texte* en une suite de nombres.

Il choisit (par exemble) le code **ASCII**.

En ASCII « msila » devient:

lettre	m	s	i	l	a
code ASCII	109	115	105	101	97

Il suffit à Ali de coder chaque nombre comme expliqué ci-dessus. Il obtient :

$$\begin{aligned}
\text{m} &\longrightarrow 109^{19} \,(mod\ 1591) = 483 \\
\text{s} &\longrightarrow 115^{19} \,(mod\ 1591) = 1410 \\
\text{i} &\longrightarrow 105^{19} \,(mod\ 1591) = 1338 \\
\text{l} &\longrightarrow 101^{19} \,(mod\ 1591) = 1174 \\
\text{a} &\longrightarrow 97^{19} \,(mod\ 1591) = 930
\end{aligned}$$

lettre	m	s	i	l	a
en ASCII	109	115	105	101	97
chiff	483	1410	1338	1174	930

Ali envoie cette suite de nombres à samir

$$\{483, 1410, 1338, 1174, 930\}$$

qui va le décrypter avec sa clé d. Il va pouvoir retrouver le message original :

$$\begin{aligned} 483^{955} \ (mod \ \ 1591) &\equiv 109 \\ 1410^{955} \ (mod \ \ 1591) &\equiv 115 \\ 1338^{955} \ (mod \ 1591) &\equiv 105 \\ 1174^{955} \ (mod \ \ 1591) &\equiv 101 \\ 930^{955} \ (mod \ \ 1591) &\equiv 97 \end{aligned}$$

En recodant en ASCII, Samir va pouvoir lire le message de son ami "msila".

déchiff	109	115	105	101	97
lettre	m	s	i	l	a

3.1.4 Efficacité

Le chiffrement au moyen du RSA demande de faire une exponentiation modulo n. Plus petit est l'exposant de chiffrement, plus le chiffrement est réalisé efficacement. Cependant, les petits exposants de chiffrement ouvrent la porte à une attaque sur petit exposant, et des contre-mesures spéciales sont nécessaires.

Le déchiffrement RSA demande aussi une exponentiation modulo n, mais l'exposant de déchiffrement doit étre aussi grand que n. Des petits exposants de déchiffrement d peuvent étre efficacement calculés à partir de (n, e) correspondantes.

Le déchiffrement RSA peut étre accéléré si le théorème chinois des restes est utilisé.

Voici Alice veut déchiffré le cryptogramme c. Sa clé privée RSA est d. Elle calcule

$$m_p = c^{d \ mod \ p-1} \ mod \ p \qquad m_q = c^{d \ mod \ q-1} \ mod \ q$$

Puis elle calcule un entier $m \in \{0, 1, ..., n-1\}$ tel que

$$m \equiv m_p \ mod \ p \qquad m \equiv m_q \ mod \ q$$

Ce m est le message en clair qui a été chiffré. Pour trouver m, elle utilise l'algorithme d'Eulide étendu et elle obtient deux entiers y_p et y_q tels que

$$y_p p + y_q q = 1$$

Alors

$$m \equiv (m_p y_q q + m_q y_p p) \ mod\ n$$

Parce que les coefficients $y_q q \ mod\ n$ et $y_q q \ mod\ n$ sont indépondants des cryptogrammes, ils pourront étre pré-calculés.

Exemple 3.1.5 *Pour accélérer le déchiffrement de l'exemble* (3.1.4) , *Alice calcule*

$$m_p = 119^{147} \ mod\ 11 = 4 \qquad m_q = 119^{147} \ mod\ 23 = 3$$

puis $y_p = -2$ *et* $y_q = 1$. *Alors*

$$m \equiv (4 * 23 - 3 * 2 * 11) \ mod\ 253 = 26.$$

3.1.5 Multiplicativité

Soit (n, e) une RSA. Quand deux messages m_1 et m_2 sont chiffrés avec cette clé, cela donne

$$c_1 = m_1^e \ mod\ n$$

$$c_2 = m_2^e \ mod\ n$$

Le produit des cryptogrammes est

$$c = c_1 c_2 \ mod\ n = (m_1 m_2)^e \ mod\ n$$

Celui qui connaît les cryptogrammes c_1 et c_2 peut calculer le chiffrement de $m = m_1 m_2$ sans connaître ce message en clair.

Il faut restreindre l'espace des messages en clair d'une certaine forme seront acceptés. On peut, par example, s'arranger pour que le premier et le dernier octet d'un message en clair soient indentiques. Il devient alors extrêmement improbaple que le produit $m_1 m_2$ de deux messages en clair valables soit lui aussi valable. Par conséquent, si Alice reçoit le chiffrement de $m = m_1 m_2$, elle rejette le message en clair m.

3.2 Chiffrement de Rabin

La securité du systéme de Rabin, qui est présenté dans cette section, elle aussi basé sur la difficulté de factorisé les entiers. Mais contrairement à RSA, on peut montrer que celui qui peut casser le systéme de Rabin efficacement peut tout aussi efficacement factoriser les entiers.

3.2.1 Fabrication des clés

Alice choisit au hasard deux grands nombres premiers p et q avec

$$p \equiv q \equiv 3 \ mod \ 4.$$

La fabrication des nombres premiers se fait comme dans les sections précedantes, excepté les propriétés des congruences qui sont en plus. Ces propriétés servent à rendre le déchiffrement efficace. Mais, comme nous allons voir plus loin, le systéme de Rabin fonctionnerait aussi sans elles. Alice calcule

$$n = p \cdot q.$$

Sa clé publique est n. Sa clé privée est la paire (p, q).

3.2.2 Chiffrement

Comme dans le systéme RSA, l'espace des messages en clair est l'ensemble $\{0, ..., n-1\}$. Pour chiffré le message en clair $m \in \{0, ..., n-1\}$, Bernard utilise la clé publique n d'Alice et calcule le cryptogramme

$$c = m^2 \ mod \ n$$

Comme RSA, le systéme de Rabin peut étre utilisé pour implémenter un chiffrement par bloc.

3.2.3 Déchiffrement

Alice reconstitue le message en clair m en extrayant la racine carreé modulo n du cryptogramme c, ce qu'il fait de la façon suivante. Elle calcule

$$m_p = c^{(p+1)/4} \ mod \ p$$

$$m_q = c^{(q+1)/4} \ mod \ q$$

Alors $\pm m_p + pZ$ sont les deux racines carreé de $c + pZ$ dans Z/pZ, et $\pm m_q + qZ$ sont les deux racines carreé de $c + qZ$ dans Z/qZ . Cette méthode pour calculer les racines carreés de $c \ mod \ p$ et $mod \ q$ fonctionne seulement parce que p et q sont tous les deux congrus à 3 mod 4. Si ce n'était pas vrai, le calcul de ces racines carreé serait plus difficile, bien qu'encore possible en temps polynomial. Maintenant Alice peut calculer les 4 racines carreés de $c + nZ$ dans Z/nZ en utilisant le théorème chinois des restes. Cela ressemble au déchiffrement de RSA au moyen du théorème chinois des restes [7].

En utilisant l'algorithme d'Euclide étendu, Alice détermine des coefficients y_p, $y_q \in Z$

$$y_p p + y_q q = 1$$

Puis elle calcule

$$r = (y_p \ p \ m_q + y_q \ q \ m_p) \ mod \ n \qquad \text{et} \qquad s = (y_p \ p \ m_q - y_q \ q \ m_p) \ mod \ n$$

Il est facile de vérifier que $\pm r$, $\pm s$ sont les 4 racines carreé de $c \ mod \ n$ dans l'ensemble $\{0, 1, ..., n-1\}$. Une de ces racines carreé est forcément m, le message originel.

Exemple 3.2.1 *Alice utilise les nombres premiers* $p = 11$ *et* $q = 23$*, donc* $n = 253$*.*

Bernard chiffre le message $m = 158$*. En calculant*

$$c = m^2 \ mod \ n = 170$$

Alice détermine $y_p = -2$ *et* $y_q = 1$*. Elle obttient les racines carreé*

$$m_p = c^{(p+1)/4} \ mod \ p = c^3 \ mod \ p = 4$$

$$m_q = c^{(q+1)/4} \ mod \ q = c^6 \ mod \ q = 3$$

Ensuite, elle calcule

$$\begin{aligned} r &= (y_p \ p \ m_q + y_q \ q \ m_p) \ mod \ n = -2 \cdot 11 \cdot 3 + 23 \cdot 4 \ mod \ n = 26 \\ s &= (y_p \ p \ m_q - y_q \ q \ m_p) \ mod \ n = -2 \cdot 11 \cdot 3 - 23 \cdot 4 \ mod \ n = 95 \end{aligned}$$

Il en résulte que les racines carreé de 170 *mod* 253 dans $\{1, ..., 252\}$ sont $26, 95, 158, 227$ et qu'une d'elles d'origine en clair.

3.2.4 Efficacité

Dans le systéme de Rabin, le chiffrement ne demande qu'une élévation au carré, c'est pourquoi ce chiffrement est plus efficace que le chiffrement RSA, même avec le plus petit possible exposant de chiffrement 3. Le déchiffrement dans le systéme de Rabin est aussi coûteux que le déchiffrement RSA amélioré par le théorème chinois des restes. Il demande une exponentiation *mod* p, une autre *mod* q, et une application du théorème chinois des restes [*voir* [7]].

3.3 Échange de clés selon Diffie-Hellman

Dans cette section, nous décrivons le protocole de Diffie et Hellman permettant d'échangrr des clé secrètes sur des canaux qui ne sont pas sûrs. Par lui-même, ce protocole n'est pas un cryptosystéme à clés publiques, mais il est à la bas du systéme ELGAMAL, qui sera décrit dans la prochaine section.

la sécurité du systémé d'échange de clés de Diffie-Hellman ne repose pas sur le problème de la factorisation des entieres mais sur le problème du logarithme discret (le problème DL) [5].

3.3.1 Logarithmes Discretes

Soit p un nombre premier. Nous savons que le groupe $(Z/pZ)^*$ est cyclique d'ordre $p-1$. Soit g une racine primitive *mod* p. Alors pour tout entier $A \in \{1, 2, ..., p-1\}$ il existe un expossant $a \in \{0, 1, 2, ...p-1\}$ avec

$$A \equiv g^a \ mod \ p$$

cet exposant a s'applle le *logarithme discret* de A dans la base g. Nous écrivons $a = log_g A$.

Le calcule du logarithrme discret est considéré comme difficile parce qu'aucun algorithme efficace permettant de le retrouver n'est connu. À l'inverse; il n'existe pas de preuve que ce problème soit définitivement difficile.

Exemple 3.3.1 *Soit* $p = 13$. *Une racine primitive modulo* 13 *est* 2. *Le logarithme discret dans la base* 2 *de chaque entier de* $\{1, 2, ..., 12\}$ *est donné dans le tableau suivant:*

A	1	2	3	4	5	6	7	8	9	10	11	12
$log_2\ A$	0	1	4	2	9	5	11	3	8	10	7	6

3.3.2 Échange de clés

Le protocole de Diffie-Hellman fonctionne de la façon suivante. Alice et Bernard veulent se mettre d'accord sur une clé secrète commune mais ils n'ont à leur disposition qu'un canal de communication qui n'est pas sûr. Pour commencer, ils se mettent d'accord sur un grand nombre premier p et un entier g avec $2 \leq g \leq p-1$ choisi pour que l'ordre de $g\ mod\ p$ soit suffisamment élevé. Les nombres p et g peuvent étre connus publiquement, ce qui fait que Bernard et Alice peuvent utiliser leur canal de communication qui n'est pas sûr pour cet agrément.

Maintenant Alice choisit un entier $a \in \{0, 1, ..., p-2\}$ au hasard, qu'elle garde secret. Elle calcule

$$A = g^a\ mod\ p$$

et envoie le resultat A à Bernard. Bernard choisit un entier $b \in \{0, 1, ..., p-2\}$ au hasard. Il calcule

$$B = g^b\ mod\ p$$

et envoie le resultat à Alice. Il garde aussi son exposant b secret. Pour obtenir la clé secrète commune, Alice calcule

$$B^a\ mod\ p = g^{ab}\ mod\ p$$

et Bernard calcule

$$A^b \ mod \ p = g^{ab} \ mod \ p$$

Leur clé commune est

$$k = g^{ab} \ mod \ p.$$

Exemple 3.3.2 *Soit* $p = 17$ *et* $g = 3$. *Alice choisit* $a = 7$, *calcule* $g^a \ mod \ p = 11$, *et envoie le resultat,* $A = 11$, *à Bernard.*

Bernard choisit $b = 4$, *calcule* $g^b \ mod \ p = 13$, *et envoie le resultat,* $B = 13$, *à Alice.*

Alice calcule

$$B^a \ mod \ p = 4$$

Bernard calcule

$$A^b \ mod \ p = 4$$

Leur clé commune est 4.

3.4 Chiffrement El-Gamal

Le cryptosystéme ElGamal est étroitement lié au protocole d'échange de clés de Diffie-Hellman . Sa sécurité repose, elle aussi, sur la difficulté de résoudre le problème de Diffie-Hellman dans $(Z/pZ)^*$.

3.4.1 Fabrication de clé

Alice choisit un nombre premier p et une racine primitive $g \ mod \ p$. Ensuite, elle choisit un exposant au hasard $a \in \{0, 1, ..., p-2\}$ et calcule

$$A = g^a \ mod \ p$$

La clé publique d'Alice est (p, g, A); sa clé secrète est l'exposant a. L'entier A était la partie de la clé venant d'Alice dans le protocole de Diffie-Hellman. Cette partie de la clé est fixée dans le cryptosystème Elgamal.

3.4.2 Chiffrement

L'espace des message en clair est l'ensemble $\{0, 1, ..., p-1\}$. Pour chiffrer un message en clair m, Bernard se procure la clé publique authentique (p, g, A) d'Alice.

Il choisit un entier au hasard $b \in \{0, 1, ..., p-2\}$ et calcule

$$B = g^b \ mod \ p$$

Le nombre B était la partie de la clé du protocole de Diffie-Hellman venant de Bernard. Bernard détermine

$$c = A^b m \ mod \ p$$

En d'autre terme, Bernard chiffre le message m en multipliant par la clé de Diffie-Hellman. Le cryptogramme Elgamal complet est (B, c).

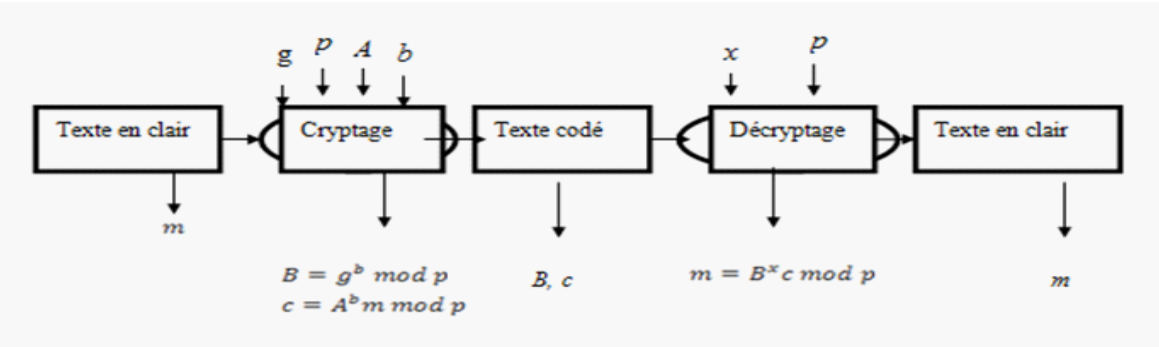
3.4.3 Déchiffrement

Alice reçoit le cryptogramme (B, c). Elle connaît sa clé secrète a. Pour reconstituer le message en clair m, elle divise c par la clé de Diffie-Hellman $B^a \ mod \ p$. Afin d'éviter les inversions $mod \ p$, elle détermine d'abord l'exposant $x = p - 1 - a$. Puisque $1 \leq a \leq p-2$, on a $1 \leq x \leq p-2$. Puis elle calcule

$$m = B^x c \ mod \ p$$

C'est bien le message en clair, comme le montre le calcule suivant :

$$B^x c \equiv g^{b(p-1-a)} A^b m \equiv (g^{p-1})^b (g^a)^{-b} A^b m \equiv A^{-b} A^b m \equiv m \ mod \ p$$



F3: schéma de système ElGamal

Exemple 3.4.1 *Alice choisit* $p = 23$, $g = 7$, $a = 6$, *et calcule*

$$A = g^a \ mod \ p = 4.$$

Sa clé publique est $(p = 23, g = 7, A = 4)$. *Sa clé secrète est* $a = 6$.
Bernard chiffre $m = 7$. *Il choisit* $b = 3$, *et calcule*

$$B = g^b \ mod \ p = 21$$

et

$$c = A^b m \ mod \ p = 11.$$

Le cryptogramme est $(B, c) = (21, 11)$.
Alice retrouve m *en calculant*

$$\begin{aligned} B^{p-1-6} c \ mod \ p &= 7 \\ &= m. \end{aligned}$$

3.4.4 Efficacité

Le déchiffrement d'ElGamal, comme le déchiffrement du RSA, demande une exponentiation modulaire. Nous allons voir que le module doit étre de la même taille dans les deux systèmes. le théoreme chinois des restes, cependant, n'accélère pas le déchiffrement du protocole ElGamal [7].

Le chiffrement ElGamal demande deux exponentiations modulaires: le calcule

$$A^b \ mod \ p \qquad \text{et celui} \qquad B = g^b \ mod \ p.$$

. Le chiffrement du RSA ne demande qu'une seule exponentiation modulaire. Mais les exponentiations pour le chiffrement ElGamal sont indépendantes des messages en clair qui sont en train d'être chiffrés.

Par conséquent, ces exponentiation peuvent être faites séparément comme des pré-calculs. De la sorte, le chiffrement ne demande plus qu'une seule multiplication modulaire et devient beaucoup plus efficace que le chiffrement du RSA.

Exemple 3.4.2 *Comme dans l'exemple précedent, la clé publique d'Alice est* $(p = 23, g = 7, A = 4)$. *Sa clé secrète est* $(a = 6)$. *Pour son pré-calcule, Bernard choisit* $b = 3$ *et calcule*

$$B = g^b \ mod \ p = 21 \qquad et \qquad K = A^b \ mod \ p = 18.$$

Plus tard, Bernard chiffre $m = 7$. *Pour cela, il calcule simplement* $c = K * m \ mod \ 23 = 11$.

Le cryptogramme est $(B, c) = (21, 11)$. *Une fois encore,Alice retrouve le message en clair* m, *en calculant*

$$B^{p-1-6} \ c \ mod \ p = 7 = m.$$

3.4.5 Généralisation

Un avantage important du systéme ElGamal est le fait qu'il peut être implémenté dans n'importe quel groupe cyclique. Les seules obligations sont que les calculs dans le groupe soient efficaces et que le probléme de Diffie-Hellman soit difficile. En particulier, le calcul des logarithmes discréts dans le groupe doit être infaisable, sinon le probléme de Diffie-Hellman serait facile résoudre [*voir* [7]] .

Conclusion

Ce mémoire concerne l'application de la congruence dans le domaine de la cryptographie. Nous avons présenté quelques systèmes comme les chiffrements RSA, puis Rabin, finalement El Gamal. L'intérêt du travail présenté dans ce mémoire est c'est que Le principe de chiffrement et déchiffrement est presque le même, c'est-à-dire une fois que l'on a compris les méthodes exposées on pourra comprendre d'autre méthode de cryptage. Ainsi, acquérir le contenu de ce travail signifie que l'on a appris de bons outils dans le domaine de la cryptographie.

Bibliographie

[1] Armel Mercier et Jean-Marie De koninck, 1001 problèmes en théorie classique des nombres, Ellipses, 2003.

[2] A. Boudaoud. Cours $2^{éme}$ Mastre Discréte "théorie des nombres". Université de M'sila, 2014.

[3] B. Nathanson Melvyn. Elementary Methods in Number Theory. Springer, 2000.

[4] David A.Santos. Elementary Number Theory Notes. Springer, 2004.

[5] G. Dubertret. Initiation à la cryptographie, Vuibert, 2012.

[6] H.Schyns. Cours Mathématique-Chiffrement RSA, 2008.

[7] Johannes Buchmann. Introduction à la cryptographie. Dunod, 2006.

[8] J. Itard. Arithmétique et théorie des nombres. Payout, 2001.

[9] Michel Demazure. Cour d'algébre. Cassini. 1997.

[10] Pierre Damphousse. Découvrir l'arithmétique. Ellipses, 2000.

[11] P. Didien et Y. Bernard. Arithmétique. Springer-Verlag 2013.

[12] W.Edwin Clarck. Elementary Number Theory. University of South Florida, 2003.