

# Cryptographie : les Maths à la rescousse!

Roland Gillard \*

Vendredi 20 et samedi 21 octobre à 14h30

## Table des matières

<b>1</b>	<b>Codage par décalage</b>	<b>2</b>
<b>2</b>	<b>Décalage variable : clef</b>	<b>2</b>
<b>3</b>	<b>Xor</b>	<b>2</b>
<b>4</b>	<b>Standards cryptographiques</b>	<b>3</b>
4.1	D.E.S. (Data Encryption Standard) . . . . .	3
4.2	A.E.S (Advanced Encryption Standard) . . . . .	3
<b>5</b>	<b>Clef publique !</b>	<b>4</b>
5.1	R.S.A. . . . .	4
5.2	El Gamal . . . . .	5
<b>6</b>	<b>Mathématiques contemporaines : Courbes elliptiques, ...</b>	<b>5</b>
<b>A</b>	<b>Méthode pour calculer rapidement <math>x^n</math></b>	<b>7</b>
<b>B</b>	<b>Pour en savoir plus</b>	<b>8</b>
B.1	Articles : . . . . .	8
B.2	Livres : . . . . .	8
B.3	Liens Internet : . . . . .	8

---

\*Institut Fourier, UJF

## 1 Codage par décalage

On remplace chaque lettre par une autre suivant la table :

A	B	C	D	...	V	W	X	Y	Z
D	E	F	F	...	Y	Z	A	B	C

qui montre un décalage de 3 sur la droite exemple BONJOUR A TOUS devient en ignorant les blancs :

ERQMRXUDWRXV

Ce procédé aurait été utilisé par Jules César!

Le problème est que le décalage est facile à trouver car on connaît les fréquences des lettres dans les textes courants

## 2 Décalage variable : clef

On choisit une clef par exemple 1526 et on décale la première lettre de 1, la deuxième de 5, la troisième de 2, et la quatrième de 6 et on recommence :

BONJOURATOUS devient en décalant de

152615261526 :

CTPPPATGUTWY

Cette méthode s'appelle Chiffrement de Vigenère

(Blaise Vigenère a vécu au XVI<sup>e</sup> siècle)

## 3 Xor

Pour le décalage variable, il est commode d'utiliser une table des lettres par exemple le codage ASCII où A=65, ..., Z=90, ce qui permet de faire le codage par ordinateur. Si on veut aller plus loin, on se dit que dans les mémoires d'ordinateur tout est codé sous forme de 0 ou de 1 (valeur d'un «bit»). On introduit alors l'opération de Xor (ou exclusif), notée  $\oplus$  et qui est donnée par :

$$0 \oplus 0 = 0$$

$$1 \oplus 0 = 1$$

$$0 \oplus 1 = 1$$

$$1 \oplus 1 = 0$$

On remarque que  $(x \oplus y) \oplus y = x \oplus (y \oplus y) = x \oplus 0 = x$ .

On peut donc coder un peu comme plus haut une suite de 0 et 1 à l'aide d'une clef K en faisant des Xor chiffre par chiffre.

Si  $K = 1101$ ,

à coder	0	1	0	0	0	1	1	0	1	1	0	0	0	1	0	1	0	0	1
clef K	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0
sortie s	1	0	0	1	1	0	1	1	0	0	0	1	1	0	0	0	1	1	1

La remarque permet de retrouver le message initial :

$K \oplus s$	0	1	0	0	0	1	1	0	1	1	0	0	0	1	0	1	0	0	1
--------------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

## 4 Standards cryptographiques

### 4.1 D.E.S. (Data Encryption Standard)

Employé depuis 1977, il utilise une clef  $K$  de 56 bits, clef secrète qui est utilisée aussi bien pour crypter que pour décrypter.

#### Description

Le texte est découpé en blocs de 64 bits. Chacun des blocs est codé ainsi :

1. On fait une permutation  $P$  sur les bits de ce bloc et on le découpe en 2 moitiés  $G_0$  et  $D_0$ .

2. On répète 16 fois (pour  $i=0, \dots, 15$ ) un codage (codage réalisé dans un «cycle») utilisant une sous-variante  $K_i$  de  $K$ .

3. On finit en faisant la permutation inverse  $P^{-1}$  de celle du début.

Description du  $i^e$  cycle :

$$\boxed{G_i \mid D_i}$$

↓

$$\boxed{G_{i+1} = D_i \mid D_{i+1} = G_i \oplus f(D_i, K_i)}$$

Ici  $f$  est une fonction compliquée, construite de façon à masquer la clef  $K_i$ . Dans l'autre sens on trouve :

$$D_i = G_{i+1}, G_i = D_{i+1} \oplus f(G_{i+1}, K_i)$$

Le procédé est donc invariant en retournant à la fois le temps (remplacer  $i$  par  $16 - i$  et la gauche et la droite (échanger  $G_i$  et  $D_i$  pour tout  $i$ ).

### 4.2 A.E.S (Advanced Encryption Standard)

Appel du NIST : 12 septembre 1997

Admission des candidatures : 15 retenues le 20 aout 1998

Finalistes : MARS, RC6, Rijndael, Serpent, and Twofish

Choix terminal (2 octobre 2000) : Rijndael conçu par 2 belges (Joan Daemen et Vincent Rijmen)

[How is that pronounced? If you're Dutch, Flemish, Indonesian, Surinamer or South-African, it's pronounced like you think it should be. Otherwise, you could pronounce it like "Reign Dahl", "Rain Doll", "Rhine Dahl". We're not picky. As long as you make it sound different from "Region Deal".]

Ce procédé utilise 10 répétitions d'un cycle comportant chacune 4 opérations sur le bloc (sauf la dernière qui n'en a que 3) :

\* Une transformation affine dans l'espace vectoriel de dimension 8 sur  $F_2$ .

\* Un décalage des lignes après représentation du bloc en tableau rectangulaire

\* Multiplication modulo  $x^4 + 1$  par  $03X^3 + 01X^2 + 01x + 02$  des colonnes considérées comme des polynômes sur  $F_{256}$  qui est un ensemble à 256 éléments

muni d'une addition d'une multiplication et une division par les éléments non nuls. Les éléments sont numérotés en base 16 (les quatre premiers sont donc 00, 01, 02 et 03 qui interviennent ci-dessus). Ce genre de structure est très utilisé en mathématiques depuis Galois (1811-1832).

\* Xor par une sous-clef dépendant du numéro du cycle.

## 5 Clef publique !

### 5.1 R.S.A.

Diffie et Hellman en 1976, ont eu une idée révolutionnaire : utiliser des clefs distinctes pour le cryptage et le décryptage. Une des deux pouvant donc être **publique**. Rivest, Shamir et Adleman ont trouvé en 1978 une méthode pratique de codage illustrant ce principe.

La méthode utilise de l'arithmétique assez simple ainsi qu'un constat empirique : il est très difficile de factoriser un gros nombre : si on choisit deux nombres premiers  $p$  et  $q$ , les retrouver à partir de la simple donnée de leur produit

$$n = pq$$

demande un temps considérable de calcul.

Un mot d'arithmétique :

On dit que deux entiers  $a$  et  $b$  sont égaux  $\pmod n$  si leur différence est divisible par  $n$ .

Condition équivalente : la division de  $a$  et  $b$  par  $n$  donnent le même reste.

Exemple : 2, 12 et 92 sont égaux  $\pmod{10}$

Tous peuvent s'écrire  $2 + 10k$  avec  $k$  entier :

$k$  est 0, 1 ou 9.

Si  $x$  est un entier, on a

$$x^t = x \pmod n$$

si

$$t = 1 \pmod f$$

avec

$$f = (p - 1)(q - 1) .$$

Le principe est d'utiliser des couples d'entiers  $(c, d)$  avec

$$cd = 1 \pmod n .$$

$$x^{cd} = x \pmod n .$$

On diffuse des listes de  $(n, d)$  attribués à certains abonnés. On cache les compléments  $c$ .

Le DES est beaucoup plus rapide que R.S.A. (plus de 1000 fois)... mais souvent on utilise R.S.A. pour transmettre la clef  $K$ !

Supposons que Bob veuille envoyer un message  $x$  à Alice. Il la cherche dans l'annuaire et découvre que ses clefs sont  $(n, d)$  : il calcule donc  $y = x^d \pmod n$  et envoie à sa destinataire Alice le message  $y$ . Pour le décoder Alice calcule  $y^c \pmod n$  puisque

$$y^c = x^{cd} = x \pmod n .$$

Exemple :  $n = 29.37 = 1073$ ,  
 $f = 28.36 = 1008$   
 $c = 605, d = 5, cd = 3025 = 3.1008 + 1$ .

messages $x$	021	514	101	521
sortie $x^d$	263	234	048	424

En effet  $21^5 = 263 \pmod{1073}$   
 $263^{605} = 21 \pmod{1073}$ .

Une variante très importante fonctionne dans l'autre sens et donne lieu à la notion de **signature** : si Bob veut prouver son identité, il n'a qu'à coder son nom avec sa clef secrète  $c$  tout le monde pourra vérifier que c'est bien lui en élevant son message à la puissance  $d \pmod{n}$ .

Maintenant s'il veut réserver sa signature à Alice, il peut combiner en utilisant d'abord son  $c$  puis le  $d$  d'Alice. Celle-ci procède en sens inverse en utilisant d'abord son  $c$  puis le  $d$  de Bob .

## 5.2 El Gamal

Ce procédé repose sur la difficulté d'inverser l'opération puissance. Il est facile de calculer  $y = x^a$ , mais connaissant  $x$  et  $y$  il est long de trouver  $a$  ( $a$  s'appelle le logarithme de  $y$  en base  $x$ ).

Le procédé utilise des tables publiques de  $(p, \alpha, \beta)$  et des nombres secrets  $k$  et  $a$ , avec  $\beta = \alpha^a$ .

Bob choisit  $p, \alpha, \beta$  publics correspondant à sa destinataire Alice. Pour coder le message  $x$  : il choisit encore un nombre auxiliaire  $k$  (secret) et on diffuse un message codé double :  $(y_1, y_2) = (\alpha^k, x\beta^k)$ .

Pour décoder, il suffit à Alice d'utiliser  $a$  (qu'elle seule connaît) car, en remarquant que

$$(\alpha^k)^a = \alpha^{ak} = (\alpha^a)^k = \beta^k,$$

elle peut retrouver  $x$  en calculant

$$y_2(y_1)^{-a} = x\beta^k(\alpha^k)^{-a} = x !!$$

Le procédé de codage est public : Bob peut l'employer avec les  $p, \alpha$  et  $\beta$  d'Alice qui seule connaît le  $a$  correspondant. Alice n'a pas besoin de connaître  $k$ .

Exemple :  $p = 2579, \alpha = 2, a = 765, \beta = 2^{765} \pmod{2579} = 949$ .

Message  $x = 1299$ , clef de codage  $k = 853$ .

$y_1 = 2^{853} \pmod{2579} = 435$

$y_2 = 1299.949^{853} \pmod{2579} = 2396$

Donc Bob transmet  $(435, 2396)$  et Alice retrouve son message par :  $2396.435^{-765} = 1299 \pmod{2579}$ .

## 6 Mathématiques contemporaines : Courbes elliptiques, ...

Le principe est d'observer que pour effectuer le codage El Gamal, on n'a besoin que d'une opération (la multiplication) avec des contraintes permettant

les calculs ci-dessus : il suffit d'avoir un **groupe**, notion introduite aussi par Galois.

Un tel exemple de groupes a été mis sous les feux des projecteurs par la preuve de Wiles du théorème de Fermat : il s'agit des courbes elliptiques, objet énormément étudié : un prix d'un million de dollars vient d'être institué pour qui résoudra un des principaux problèmes ouverts à leur sujet (la conjecture de Birch et Swinnerton-Dyer)

Exemple de courbe elliptique :

$$y^2 = x^3 + x + 6$$

avec  $x$  et  $y$  entiers mod 11 :

si  $P = (x_1, y_1)$  et  $Q = (x_2, y_2)$  leur produit  $P.Q$  est le point  $(x_3, y_3)$  donné par

$$\begin{aligned}x_3 &= \lambda^2 - x_1 - x_2 \\y_3 &= \lambda(x_1 - x_3) - y_1\end{aligned}$$

avec

$$\lambda = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1} & \text{si } P \neq Q \\ \frac{3x_1^2 + 1}{2y_1} & \text{si } P = Q \end{cases} .$$

## A Méthode pour calculer rapidement $x^n$

On considère un triplet  $(u, v, m)$  :

- valeur initiale  $(u, v, m) = (1, x, n)$ ;
- on le remplace par
  - \*  $(u, v^2, m/2)$  si  $m$  est pair
  - \*  $(uv, v^2, (m-1)/2)$  sinon

et on recommence jusqu'à obtenir  $m = 1$ .  
-La puissance recherchée est alors  $uv$ .

Exemple :  
Calcul de  $2^6$  :  $(1, 2, 6) \rightarrow (1, 4, 3) \rightarrow (4, 16, 1)$  :  
 $2^6 = 4.16 = 64$

Cet algorithme est difficile à comprendre!  
Le mieux est de **prouver** qu'il marche  
en introduisant  $y = uv^m$  ;  
cette quantité passe de  $x^n$  à  $uv \dots$   
mais reste fixe à chaque étape :

$$\begin{cases} uv^m = u(v^2)^{m/2} & \text{si } m \text{ est pair,} \\ uv^m = uv(v^2)^{(m-1)/2} & \text{sinon,} \end{cases}$$

mais  $m$  diminue donc finit par valoir 1.

Exercice : le modifier pour avoir en plus l'écriture  
de  $n$  en base 2.

## B Pour en savoir plus

### B.1 Articles :

Pour la science N° 267 janvier 2000 pp. 104-110 : Cryptologie R.S.A 20 ans après

Eureka N° 58H pp. 70-77 : Octets sataniques

La recherche HS N° 2 aout 99 pp. 21-29 : Le théorème de de Fermat enfin démontré

Gazette des mathématiciens N° 85 pp. 9-23 : Les standards cryptographiques du XXI ième siècle, F. Leprévost

Notices AMS N° 47-3, p. 341-349, Data Encryption Standard

Notices AMS N° 47-4, p.450-459 Advanced Encrytion Standard

### B.2 Livres :

J. Stern : La science du secret, O. Jacob, 1998

D. Stinson : Cryptographie, théorie et pratique Int. Thomson Pub France 1996.

N. Koblitz : Algebraic aspects of cryptology : Algorithms & computation in Math. vol 3, Springer 1998

### B.3 Liens Internet :

Un site général : <http://www.crypto.com/>

A.E.S : <http://csrc.nist.gov/encryption/aes/>

Rinjndael : <http://www.esat.kuleuven.ac.be/rijmen/rijndael/>

Biblio générale : <http://www.counterpane.com/biblio/>

Un serveur de clefs : <http://www.keyserver.net>

Tout sur les nombres premiers :  $2^{6972593} - 1$  est-il toujours le plus gros avec ses plus de deux millions de chiffres ?

<http://www.utm.edu/research/primes/>

Comment gagner 1 million de dollars :

[http://www.claymath.org/prize\\_problems/birchsd.htm](http://www.claymath.org/prize_problems/birchsd.htm)

Ma page personnelle :

<http://www-fourier.ujf-grenoble.fr/~gillard/>

-oOo-