CRYPTOGRAPHIE ET ARITHMÉTIQUE

Maurice MIGNOTTE [1]
ULP Strasbourg

1 Introduction

Larousse nous dit : **Cryptographie.** (du grec *kruptos* caché et *graphein* écrire). Écriture secrète au moyen d'abréviations ou de signes convenus entre deux personnes.

Par extension, ce terme désigne la science du codage et du déchiffrement des messages codés. Souvent les caractères du texte original sont transformés, pour leur transmission, en chiffres, d'où la synonymie qui s'est établie entre *chiffre* et crytographie.

Les systèmes de codage fondamentaux sont de deux types : *transposition* et *substitution*. La transposition est une modification de l'ordre des lettres du texte en clair. Par contre, la substitution respecte l'ordre des symboles du texte initial, mais opère une traduction de ce texte, souvent symbole par symbole, mais aussi parfois groupe de caractères par groupe de caractères.

Pour renforcer la sécurité, on est souvent amené à *surchiffrer* les messages codés. La clef de surchiffrement utilisée à cet effet est assez simple et changée très souvent.

L'utilisation des calculateurs permet de réaliser des codages très sophistiqués assurant une grande sécurité du secret des communications. Mais — en sens inverse — les ordinateurs modernes ont une telle puissance de calcul qu'ils permettent aux spécialistes de décoder des codes extrêmement subtils.

Cet exposé est surtout orienté vers les applications de l'arithmétique, en particulier de l'arithmétique élémentaire, à la crytographie. On présentera donc la méthode RSA, du nom de ses inventeurs R. Rivest, A. Shamir et L. Adleman [R-S-A]. Nous donnerons aussi quelques informations sur des méthodes plus récentes. Cependant, dans une première partie très brève, nous présentons la crytographie avant notre ère.

2 La crytographie avant notre ère

Il semble que la méthode la plus ancienne "d'écriture cachée" ait été la suivante : pour envoyer un message secret on prenait un esclave auquel on rasait le cuir chevelu, on écrivait alors le message sur son crâne, on mettait l'esclave dans une prison discrète en attendant que ses cheveux repoussent, ensuite on envoyait l'esclave chez le destinataire, ce dernier lui rasait à nouveau le crâne et pouvait donc prendre connaissance du message. Cette méthode a été utilisée il y a plus de vingt-cinq siècles. De nos jours, les impératifs de temps sont tels qu'elle a été abandonnée.

Chez les Grecs, à une époque très ancienne, on trouve l'utilisation de la *scythalle*. Il s'agit du procédé suivant : l'expéditeur et le destinataire possèdent deux bâtons identiques (ils partagent une même clef secrète), le matériel pour la communication

est une lanière de cuir ; pour envoyer un message codé l'expéditeur enroule la lanière autour de son bâton et écrit le message sur cette lanière parallèlement à la longeur du bâton, quand il reçoit la lanière le destinataire l'enroule autour de son propre bâton et peut donc retrouver le message initial. Le lecteur pourra vérifier que par ce procédé, les lettres du message initial sont permutées quand on regarde la lanière déroulée. Avec le vocabulaire introduit plus haut, on a donc effectué une transposition du message.

À l'époque de Jules César les romains utilisaient la méthode suivante que l'on retrouvera en décodant le petit message qui suit.

Soit donc le message

YHQL YLGL YLFL.

On essaie de remplacer chaque lettre par sa précédente dans l'alphabet (par permutation circulaire), jusqu'à obtenir un texte significatif. Ici on trouve successivement

XGPK XKFK XKEK,
WFOJ WJEJ WJDJ,
VENI VIDI VICI,

qui correspond bien sûr au message caché. Ce code — appelé code de Jules César — consiste donc à translater chaque lettre d'un nombre fixé de positions, ici de trois positions. C'est l'ancêtre des méthodes de substitution, c'est à dire des méthodes dans lesquelles on transforme chaque caractère (ou de façon plus compliquée, chaque groupe d'un nombre fixé de caractères).

Le lecteur intéressé par ces méthodes traditionnelles trouvera une présentation très lisible dans l'ouvrage de Sinkov [S].

3 Systèmes cryptographiques à clef publique

Introduction

Ce schéma a été introduit par Diffie et Hellman [D-H]. On considère un ensemble d'individus i, j, k, ..., qui veulent communiquer entre eux. Mais, lorsque j envoie un message à i, seul i doit pouvoir le déchiffer.

À chacun de ces individus i correspondent deux procédures, l'une E_i qui est publique, l'autre D_i, secrète, connue seulement de i (en principe!). La liste des procédures E_i figure tout simplement sur l'équivalent d'un annuaire.

Comme les auteurs américains, attribuons des prénoms, Alice, Bob, Eve, ... aux individus désirant communiquer. Si Alice veut envoyer un message m à Bob elle procédera ainsi : elle consulte l'annuaire pour trouver la procédure, disons E_B, qui indique le codage pour Bob, elle calcule $m' = E_B(m)$ et l'envoie à Bob ; pour déchiffrer m' celui-ci calcule $D_B(m')$.

On considère en plus que Alice et Bob communiquent en présence d'une espionne, Eve, qui veut connaître les messages échangés et éventuellement les modifier au passage. Les problèmes modernes de la cryptographie sont les suivants :

– **Confidentialité** : Un message envoyé par Alice à Bob ne doit pouvoir être lu par personne d'autre.

– **Authentification** : Bob doit pouvoir vérifier que c'est bien Alice qui lui a envoyé le message.

– **Intégrité** : Bob doit pouvoir vérifier que le message que lui a envoyé Alice n'a pas été modifié durant la communication.

– **Non-répudiation** : Il doit être impossible qu'Alice puisse éventuellement prétendre qu'elle n'a pas envoyé le message.

On voit donc que dans ce schéma, on a quitté la situation classique de la crytographie où les personnes échangeant des messages se faisaient mutuellement confiance. C'est le progrès !

Une situation comme la précédente se produit par exemple pour les achats via Internet. Supposons qu'Alice veuille acheter quelque chose à Bob par ce moyen. Elle envoie à Bob son numéro de carte de crédit et les détails pour l'achat. Elle demande une communication confidentielle à ce sujet. D'autre part, Bob doit savoir si le message provient bien d'Alice. Tous deux doivent pouvoir vérifier que l'intégrité du message a été respectée. Enfin Bob exige qu'Alice ne puisse pas prétendre un jour qu'elle n'a jamais effectué cette commande. Ainsi la transaction a lieu entre deux personnes qui — mutuellement — ne se font pas confiance. On peut noter que cette situation ne se présente pas dans certains réseaux privés, comme celui des banques, où l'intégrité des communications est assurée par des moyens "hardware", et où les personnes se font mutuellement confiance.

Le domaine envisagé ici semble nécessiter l'utilisation de systèmes crytographiques à clef publique, tandis que celui des banques peut fonctionner (et fonctionne en fait) avec le système usuel des clefs secrètes symétriques.

Une autre remarque s'impose : les systèmes actuels à clef secrète restent beaucoup plus rapides que les systèmes à clef publique et permettent donc des communications avec des débits beaucoup plus élevés que les systèmes à clef publique.

La méthode RSA

Nous présenterons brièvement cette méthode qui est désormais très connue. Elle repose sur des propositions arithmétiques élémentaires toutes connues de Fermat. De manière précise elle utilise le lemme suivant.

LEMME . Soit n un entier sans facteur carré, $n = p_1 \cdots p_r$, où les p_i sont des nombres premiers distincts. Soit aussi k un entier congru à 1 modulo le produit $(p_1 - 1) \cdots (p_r - 1)$. Alors tout entier x vérifie

$$x^k \equiv x \pmod{n}.$$

Démonstration .— Grâce au théorème d'Euclide et au fait que les p_i sont distincts, il suffit de vérifier que l'on a $x^k \equiv x \pmod{p_i}$ pour chaque i, mais ceci n'est rien d'autre que le petit théorème de Fermat.

Nous sommes maintenant en mesure de présenter la méthode RSA. À chaque individu i on associe des entiers e_i, d_i et n_i, où e_i et n_i sont publics tandis que i est le seul à connaître d_i. Les messages m envoyés à i sont des entiers modulo n_i (ce qui n'ôte rien à la généralité de la méthode). Les procédures de codage et de décodage,

E_i et D_i, sont respectivement définies par

$$E_i(m) = m^{e_i}, \quad D_i(c) = c^{d_i}.$$

Chaque entier n_i est le produit de deux nombres premiers p_i et q_i distincts. Pour que le décodage soit correct on doit avoir $D_i \circ E_i = Id$; pour ceci, d'après le lemme, il suffit que l'on ait

$$e_i d_i \equiv 1 \pmod{(p_i - 1)(q_i - 1)}.$$

D'autre part, c'est un truc bien connu des informaticiens (et déjà mis en évidence par Legendre) que le calcul de x^k nécessite au plus $O(\log k)$ multiplications. Ceci prouve que les temps de calcul des fonctions de codage et de décodage sont polynomiaux en fonction de $\log n$, donc possibles pour de très grandes valeurs de n.

En ce qui concerne la sécurité de cette méthode, il semble qu'elle soit exactement liée à la difficulté de factoriser l'entier n. Malgré les progrès spectaculaires des méthodes de factorisation, en choisissant les p_i et q_i plus grands que 10^{70} on doit encore avoir une excellente sécurité pour quelques années.

4 Cryptographie basée sur des groupes

Dans cette section on considère un groupe fini commutatif fixé G, d'ordre N. On considère souvent que G est cyclique et on désigne par g un de ses générateurs (fixé). Le problème du *logarithme discret* est le suivant : on se donne $h \in G$ et on cherche le plus petit entier x (s'il existe) tel que

$$h = g^x.$$

Il est important, en crytographie, que ce problème soit difficile.

Échange de clefs de Diffie-Hellman

Alice et Bob veulent avoir une clef commune, qui pourrait par exemple être utilisée pour une communication codée pour un algorithme symétrique tel que le DES (Data-Encrypton-Standard, le système sans doute le plus utilisé actuellement), ceci sans faire circuler la clef sur le réseau. Pour ce faire on peut prendre un groupe G fixé, d'ordre N, et un élément d'ordre élevé g de ce groupe. Ils procédent alors comme suit :

- Alice choisit un entier a au hasard dans l'intervalle $[1, N-1]$ (qu'elle garde secret) et transmet g^a à Bob,
- Bob choisit un entier b au hasard dans l'intervalle $[1, N-1]$ (qu'il garde secret) et transmet g^b à Alice,
- De son côté Alice calcule $(g^b)^a$, tandis que Bob calcule $(g^a)^b$. Ils ont alors la même clef g^{ab}, qui n'a pas circulé sur la ligne.

Pour la plupart des groupes, on pense que découvrir la clef à partir de la connaissance de G, g, g^a et g^b est aussi difficile que de savoir résoudre le problème du logarithme discret.

Chiffrement de El-Gamal

Alice souhaite envoyer un message $m \in G$ à Bob, dont la clef publique est (g, h) avec $h = g^x$, où x est secret. Alice choisit un entier $k \in [1, N-1]$ puis calcule et envoie à Bob la paire $(a, b) = (g^k, h^k m)$.

Ensuite Bob effectue le calcul

$$ba^{-x} = h^k m g^{-kx} = g^{kx-kx} m = m,$$

qui lui permet donc de retrouver le message initial m.

Signature de El-Gamal

On considère cette fois un message $m \in \mathbb{Z}/N\mathbb{Z}$ que Bob veut envoyer signé. Comme plus haut, on suppose qu'il possède une clef publique $h = g^x$, avec x secret. De plus on suppose que f est une bijection donnée entre G et $\mathbb{Z}/N\mathbb{Z}$. La procédure est la suivante :

- Bob choisit un entier $k \in [1, N-1]$ et calcule $a = g^k$,
- puis Bob calcule une solution $b \in \mathbb{Z}/N\mathbb{Z}$ de la congruence

$$m \equiv x f(a) + bk \pmod{N}$$

et il envoie la paire (a, b) et le message m à Alice,

- il ne reste plus à Alice qu'à vérifier la relation

$$h^{f(a)} a^b = g^{xf(a)+kb} = g^m$$

qui atteste que Bob est bien l'expéditeur.

DSA

Le sigle DSA correspond à *Digital Signature Algorithm*, qui est devenue un standard.

On reprend la situation précédente dont le cas présent n'est qu'une variante. Comme avant, Bob calcule $a = g^k$, puis — cette fois — une solution b à la congruence

$$m \equiv -x f(a) + bk \pmod{N}$$

et envoie a, b et m à Alice. Cette dernière calcule

$$u = mb^{-1} \pmod{N} \quad \text{et} \quad v = f(a)b^{-1} \pmod{N}$$

puis $w = g^u h^v$ et vérifie que

$$w = g^{mb^{-1}} g^{vx} = g^{mb^{-1}+vx} = g^{kbb^{-1}} = a.$$

L'avantage de cette procédure sur celle de El-Gamal est qu'elle comporte deux exponentiations dans G au lieu de trois.

À dire vrai, les procédures effectives de DSA utilisent en plus une fonction de *hashing* qui comprime le message initial afin de le rendre illisible.

Codage de Massey-Omura

Ici Alice veut envoyer un message $m \in G$ à Bob. La procédure est la suivante

- Alice choisit un entier x premier avec N et envoie $a = m^x$ à Bob,
- Bob choisit un entier y, lui aussi premier avec N, et envoie $b = a^y = m^{xy}$ à Alice,
- Alice calcule x' tel que $xx' \equiv 1 mod N$ et envoie à Bob l'élément $a' = b^{x'} = m^{xyx'} = m^y$,
- enfin Bob calcule $a'^{y'} = m^{yy'} = m$, où y' vérifie $yy' \equiv 1 mod N$, ainsi Bob retrouve le message initial m.

On pourra noter que ce procédé n'utilise aucune clef publique, ni ne partage une clef secrète, par contre chacun des deux acteurs utilise une clef personnelle secrète. On parle tout naturellement d'algorithme à "double-clef".

Choix du groupe

Pour que les opérations de codage et décodage soient réalisables en un temps rapide il est nécessaire que les calculs dans le groupe G soient relativement simples. Il faut par contre que le logarithme discret ne se calcule pas facilement dans G, ce qui exclut par exemple tous les groupes donnés sous la forme $\mathbb{Z}/N\mathbb{Z}$. Pour ces raisons, les premiers groupes choisis ont été les groupes $\mathbf{F}_q^*$, où $\mathbf{F}_q$ désigne un corps fini. Mais — dans ce cas — on connaît un algorithme sous-exponentiel pour calculer le logarithme discret, ce qui impose de prendre de très grandes valeurs de q. C'est à cause de cela que, dès le milieu des années 80, Miller et Koblitz ont proposé d'utiliser pour G le groupe des points d'une courbe elliptique sur un corps fini (voir [B-S-S]), dans ce cas on ne connait pas de méthode sous-exponentielle pour calculer le logarithme discret et on peut donc travailler avec des valeurs de q beaucoup moins grandes. Pour une discussion plus détaillée et pour l'étude d'autres exemples de groupes, nous renvoyons le lecteur à l'ouvrage de Blake, Seroussi et Smart [B-S-S].

Références

[B-S-S] I.F. Blake, G. Seroussi, N.P. Smart .— Elliptic curves in Cryptography, Cambridge University Press, 1999.

[D-H] W. Diffie, M. Hellman .— New directions in cryptography, I.E.E.E. Trans. Inform. Theory IT–22, Nov. 1976, No. 6, p. 644–654.

[R-S-A] R.L. Rivest, A. Shamir, L. Adleman .— A method for obtaining digital signatures and Publi-Key cryptosystems, Com. A.C.M., Fev. 1978, v. 21, No. 2, p. 120–126.

[S] A. Sinkov .— Elementary cryptanalysis, The Math. Ass. of america, New Math. Library, Washington, 1979.