14 octobre 2015

Chapitre 1

Exemples

1.1 Syntaxe

Supposons qu'Alice (A) veut recevoir un message $m \in M$ provenant de Bob (B) en utilisant un canal authentique (" un attaquant ne peut remplacer l'envoi de A à B") [1]. Mais, on suppose la présence d'un attaquant ou adversiare, Oscar (O), qui inspecte le canal de communication. A peut engendrer deux clés pk (clé publique) et sk (clé secrète) et B peut envoyer $c = Enc_{pk}(m) \in C$, un message chiffré c en utilisant un algorithme de chiffrement publique Enc_{pk}.

Le receveur A peut déchiffrer par $Dec_{sk}(c)$ en utilisant un algorithme de déchifrement Dec_{sk} sachant qu'elle possède une clé privée sk.
On exige donc la propriété de correction qui est ici :

$$m = Dec_{sk}(Enc_{pk}(m)), \forall m \in M \tag{1.1}$$

Comme convention si $\mathcal{A}$ est un algorithme probabiliste qui reçoit l'entrée x, alors on écrit $y \leftarrow \mathcal{A}(x)$ ou $\mathcal{A}(x) \rightarrow y$ pour dire que y reçoit la sortie de $\mathcal{A}$. Si $\mathcal{A}$ est détermiste, on écrit alors $y := \mathcal{A}(x)$.

Formellement on a la définition d'un cryptosystème :

Définition 1 *Un schéma cryptographique à clé publique est un 3-uplet d'algorithmes probabilistes s'exécutant en temps polynomial,*

$$\prod = (Gen, Enc, Dec)$$

tel que :

1. on supose pour le moment qu'il n'y a pas d'erreur dans la transmission et que l'attaquant n'agit pas par subtitution de symboles transitants. Donc il est passif.

1. *$Gen(1^n) \to (pk, sk)$. L'algorithme Gen, en recevant un paramètre de sécurité n engendre les paramètres du système, en particulier la clé publique pk et la clé secrète sk de tailles au moins n bits.*
2. *$Enc_{pk}(m) \to c$ où $m \in M$. L'algorithme de cryptage Enc_{pk} reçoit le message clair m et sort le chiffré c.*
3. *Dec_{sk} est un algorithme déterministe qui reçoit le chiffré c et sort soit un messahe m' ou un symbole $\perp$ pour signifier qu'il y a déffailance.*

Le cryptosystème $\prod$ doit vérifier la correction ($m' = m$?) : la probabilité pour $m \in M$,

$$Pr(Dec_{sk}(Enc_{pk})(m) = m) \tag{1.2}$$

est très grande. Si le cryptosystème $\prod$ est déterministe, on remplace l'eq. (2) par (1).

1.2 RSA

Soit $N > 0$ un entier. On note par $\mathbb{Z}_N^\times = \mathbb{Z}_N^* = (\mathbb{Z}/N\mathbb{Z})^*$ le groupe des unités de $\mathbb{Z}/N\mathbb{Z}$.

On sait que $\mid \mathbb{Z}_N^* \mid = \phi(N)$ avec

$$x \in \mathbb{Z}_N^* \Leftrightarrow pgcd(x, N) = 1.$$

Donc $(\mathbb{Z}_N^*, \cdot)$ est un groupe pour la multiplication.
On présente ici RSA ("non probabiliste et une version qui n'est pas sécurisé contre certaines attaques") pour illustrer un cryptosystème déterministe.

- *$Gen(1^n)$: n étant le paramètre de sécurité.*
  Trouver p, q premiers sur n bits tels que $N = pq$. On a $\phi(N) = (p-1)(q-1)$.
  Choisir $e \in \mathbb{Z}$ tel que $pgcd(e, \phi(N)) = 1$;
  Calculer $d := e^{-1} \pmod{\phi(N)}$;
  $pk := (N, e)$; $sk := (N, d)$.
- *$Enc(m, pk)$, où $m \in \mathbb{Z}_N^*$.*
  Calculer le chiffré $c := m^e \pmod{N}$.
- *$Dec(c, sk)$: calcule $m' := c^d \pmod{N}$.*

Pour la correction, on doit vérifier que $m' = m$. En effet :
$d \equiv e^{-1} \pmod{\phi(N)} \Rightarrow \exists k \in \mathbb{Z}, d = e^{-1} + k\phi(N)$. *Donc*
$c \equiv m^e \pmod{N} \Rightarrow c^d \equiv m^{ed} \pmod{N}$
$\equiv m^{e(e^{-1}+k\phi(N))} \pmod{N}$
$\equiv m.m^{ek\phi(N)} \pmod{N}$
$\equiv m(m^{\phi(N}})^{ek} \pmod{N}$
$\equiv m \pmod{N}$,
d'après le théorème d'Euler car $m \in \mathbb{Z}_N^*$.

On peut utiliser aussi pour la correction le résultat suivant un peu plus général [KL]. Soit G un groupe fini de cardinal $m > 1$. Soit $e > 1$ un entier. Soit l'application $f_e : G \to G$ définie par $f_e(g) = g^e$. Si $PGCD(e, m) = 1$, alors l'application est une permutation de G. En plus, si $d = e^{-1} \pmod{m}$, alors $f_d = f_e^{-1}$.

Exemple $p = 11$, $q = 23$,$e = 3$,$N = 253$,$\phi(N) = 220$,$d = 147$, *le message* $m = 0111001 (= 57) \in \mathbb{Z}_{253}^*$
cryptage : $250 := 57^3 \pmod{253}$
Décryptage : $57 := 250^{147} \pmod{253}$

1.3 Protocole Diffie-Hellman (DH) d'échange de clés

Alice et Bob veulent construire une clé secrète commune. Soit $\mathcal{G}$ un algorithme probabiliste polynomial en n qui reçoit 1^n et sort G (un groupe cyclique) de cardinal q avec la longueur $long_2(q) = n$, en représentation binaire et $g \in G$ engendrant $G =< g >$. On dit que g est primitif ou une racine primitive. On a donc $\mathcal{G}(1^n) \to (G, g, q)$. (G, g, q) sont des paramètres publiques.

Si X un ensemble, la notation $x \longleftarrow X$ signifie ici que x est pris aléatoirement dans X, alors que $x := f$ est l'affectation de f à x. Le protocole DH est le suivant.

Début.
$A : x \leftarrow \mathbb{Z}_q$;
$h_1 := g^x$;
A envoie h_1 *à* B;

$B : y \leftarrow \mathbb{Z}_q$;
$h_2 := g^y$;
B *envoie* h_2 *à* A;
$A : k_A := h_2^x$;
$B : k_B := h_1^y$;
Fin.

A la fin, on a $k_B = h_1^y = (g^x)^y = g^{xy} = k_A$. *Ainsi* A *et* B *ont construit une clé secrète commune. Un adversaire* O *peut trouver le secret commun s'il sait résoudre éfficacement (i.e. en temps polynomial) le problème suivant qu'on appelle le problème du logarithme discret (PLD).*
PLD : étant donné un élément $h \in G$,*d'un groupe cyclique* G *de cardinal* q, *d'élément primitif* g ($G =< g >$), *trouver* $x \in \mathbb{Z}_q$ *tel que* $h = g^x$.

1.4 El Gamal

Si X *un ensemble, la notation* $x \longleftarrow X$ *signifie ici que* x *est définit aléatoirement dans* X, *alors que* $x := f$ *c'est l'affectation de* f *à* x.

On décrit un cryptosystème El Gamal, qui s'inspire du protocole de DH, qui est aussi basé sur la difficulté de résoudre le PLD.

- $Gen(1^n)$ *: on obtient* G *un groupe cyclique engendré par* g *avec* $q = |G|$ *et* $n =$ *longueur de* q *en binaire. Ainsi* $Gen(1^1) \longrightarrow (G, g, q)$.
  $x \leftarrow \mathbb{Z}_q$, $h := g^x \in G$
  $pk := (G, g, q, h)$ *clé publique ;* $sk := (G, q, g, x)$ *clé secrète.*

- $Enc(pk, m) \rightarrow c$ *: où* c *est le chiffré,* m *message* $\in G$.
  $y \leftarrow \mathbb{Z}_q$; $c := (g^y, h^y \cdot m) = (c_1, c_2)$.
  noter que le message m *est masqué par* h^y *et aussi le choix aléatoire de* y *pour crypter.*
- $Dec(sk, (c_1, c_2)$ *: calcule* $m' = c_2/c_1^x$

Pour la correction, montrons que $m = m'$. *On a :*

$$m' = \frac{c_2}{c_1^x} = \frac{h^y \cdot m}{(g^y)^x} = \frac{g^{xy}}{g^{xy} \cdot m} = m.$$

1.5 Cryptosystème de Goldwasser-Micali (GM)

GM est construit à partir des propriétés des résidus quadratiques, voir le chapitre précédent pour quelques preuves et propriétés du symbole de

Legendre qui est défini modulo un premier. On aura besoin par la suite du symbole de Jacobi qui utilise un module N composé. La particularité du cryptosystème GM est qu'il permet de cypter un bit. En plus, ce système est montré sécurisé contre les attaques cpa (chosen plaintext attack), voir [KL].

1.5.1 Symbole de Jacobi

Soit G un groupe. un élément $y \in G$ est RQ (résidu quadratque), s' il existe un $x \in G$ tel que $x^2 = y$: x est une racine carrée de y. Sinon, y est un NRQ (non résidu quadratque). On rappelle quelques propriétés et natations en prenant $G = \mathbb{Z}_p$ où $p > 2$ est premier, avec des indices convenables :
1) l'équation $y = x^2 \pmod{p}$ ou n'admet aucune solution (i.e. $y \in NRQ_p$)) ou a exactement deux solutions.
2) $|RQ| = |NRQ| = \frac{p-1}{2}$
3) Soit $x \in \mathbb{Z}_p^*$. Le symbole de Legendre est $(\frac{x}{p}) = 1$ si $x \in RQ_p$, -1 sinon. On a

$$\left(\frac{x}{p}\right) \equiv x^{\frac{p-1}{2}} \pmod{p}. \tag{1.3}$$

4) $\forall x, x' \in RQ, \forall y, y' \in NRQ_p$,

$$(xx' \pmod{p}) \in RQ_p,$$
$$(yy' \mod p)) \in RQ_p,$$
$$(xy \pmod{p}) \in NQR_p,$$

grâce à la relation importante :

$$\left(\frac{xy}{p}\right) = \left(\frac{x}{p}\right)\left(\frac{y}{p}\right).$$

Maintenant, on suppose que $N = pq$ où p, q premiers >2 et que le groupe $G = \mathbb{Z}_N^*$ le groupe multiplicatif des unités modulo N. On sait , par le théorème des restes chinois, que $\mathbb{Z}_N^* \simeq \mathbb{Z}_p^* \times \mathbb{Z}_q^*$ (isomorphisme de groupes multiplicatifs), avec la correspondance bijective $y \longleftrightarrow (y_p, y_q) = (y \pmod{p}, y \pmod{q})$. On note

$$RQ_N = \{y \in \mathbb{Z}_N^* : y \text{ est un RQ modulo } N\}.$$

Proposition 1 *$y \in \mathbb{Z}_N^*$ est résidu quadratique modulo N, si et seulement si $y_p \in RQ_p$ et $y_q \in RQ_q$.*

On déduit de la propostion1 que chaque $y \in \mathbb{Z}_N^*$ a 4 racines carrés ou aucune. Si on considère que $y \longleftrightarrow (y_p, y_q)$ et que $y \in RQ_N$ avec x_p (resp. x_q) racine carrée de y_p (resp. y_q), alors y a quatre racines carrées qui sont :

$$(x_p, x_q), (x_p, -x_q), (-x_p, x_q), (-x_p, -x_q).$$

On a aussi la fraction suivante des RQ :

$$\frac{|RQ_N|}{|\mathbb{Z}_N^*|} = \frac{|RQ_p||RQ_q|}{|\mathbb{Z}_N^*|} = \frac{\frac{p-1}{2}\frac{q-1}{2}}{(p-1)(q-1)} = \frac{1}{4}. \tag{1.4}$$

On définit le symbole de Jacobi, en fonction de celui de Legendre comme suit. Soit $x \in \mathbb{Z}_N^*$,

$$\left(\frac{x}{N}\right) = \left(\frac{x}{p}\right) \cdot \left(\frac{x}{q}\right).$$

On définit

$$J_N^1 \quad (resp. \quad J_N^{-1}) = \{x \in \mathbb{Z}_N^* : \left(\frac{x}{N}\right) = 1 \quad (resp. \quad -1)\}.$$

Donc si x est un RQ modulo N, alors on a $\left(\frac{x}{N}\right) = 1$ et $x \in J_N^1$. Mais il se peut que $\left(\frac{x}{N}\right) = 1$ et x n'est pas dans RQ_N. C'est possible si $\left(\frac{x}{p}\right) = \left(\frac{x}{q}\right) = -1$. D'où on introduit l'ensemble

$$NRQ_N^1 = \{x \in \mathbb{Z}_N : x \in NRQ_N \quad et \quad \left(\frac{x}{N}\right) = 1\},$$

des éléments de $\mathbb{Z}_N$ dont le symbole de Jacobi est égal à 1.

Proposition 2 *$N = pq$, où p, q premiers >2. Alors :*

- *1) $|J_N^1| = \frac{1}{2}|\mathbb{Z}_N^*|$*
- *2) $RQ_N \subseteq J_N^1$*
- *3) $\frac{1}{2}|J_N^1| = |Rq_N| = |NRQ_N^1|$*
- *4) $x, y \in \mathbb{Z}_N^*, \left(\frac{xy}{N}\right) = \left(\frac{x}{N}\right)\left(\frac{y}{N}\right)$*

– *5)* $\forall x, x' \in QR_N, \forall y, y' \in NRQ_N^1$, *on a*

$$(xx' \mod N) \in RQ_N,$$

$$(yy' \mod N) \in RQ_N,$$

$$(xy \mod N) \in NRQ_N^1.$$

1.5.2 Description du cryptosystème GM

Le système GM crypte des messages d'un bit basé sur l'hypothèse que, si la factorisation de N *est inconnue, décider si* $x \in \mathbb{Z}_N^*$ *est résidu quadratique ou non est difficile.*
Les clés publique et privée sont : $pk = N$ *;*$sk = (p, q)$ *où* $N = pq$*, avec une modification décrite ci-dessous.*
-Le bit '0' est chiffré par un élément aléatoire de RQ_N *et le bit '1' par un élément aléatoire de* NRQ_N^1*.*
-Le chiffré c *est déchiffré en utilisant* sk *pour décider si* c *est* RQ *ou non modulo* N*. Noter ici que si on connait la factorisation de* N*, alors il est facile de décider si* c *est* RQ *ou non, en utilisant la proposition 1 et aussi l'équation (3) pour calculer le symbole de Legendre modulo un premier.*

⇒ ***Choix d'un élément aléatoire de*** RQ_N*.*
On veut choisir $y \leftarrow RQ_N$*. Soit* $x \leftarrow \mathbb{Z}_N^*; y := x^2 \mod N$*. Donc* $y \in RQ_N$*. Alors, on a* $\forall \widehat{y} \in QR_N$, $Pr[y = \widehat{y}] = \frac{1}{|QR_N|}$*.*
⇒ ***Choix d'un élément aléatoire de*** NRQ_N^1*.*
Le receveur A *connait le factorisation de* $N = pq$ *et exécute* $z \leftarrow NRQ_N^1$ *et inclut* z *dans* pk*, au moment de l'exécution de l'algorithme probabiliste* $Gen(1^n)$*. L'expéditeur* B *, alors peut chiffrer par* $x \leftarrow \mathbb{Z}_N^*$ *et* $y := (zx^2 \mod N)$*. Donc* $y \in NRQ_N^1$*, d'après le point 5) de la proposition 2.*

Schéma GM

– $Gen(1^n)$ *: donne* $(N, p, q), N = pq$ *et* $long_2(p) = long_2(q) = n$ *en bits ;* $z \leftarrow NRQ_N^1$*.*
$pk := (N, z)$ *et* $sk = (p, q)$*.*
– $Enc(pk, m) : m \in \{0, 1\}$ *un bit.*
$x \leftarrow \mathbb{Z}_N^*$*; sortir* $c := z^m \cdot x^2 \mod N$ *.*

– $Dec(sk, c)$: *si* $c \in RQ_N$, *sortir '0' sinon '1'*

Noter que la correction est vérifiée vu que le receveur, connaissant la factoriasation $N = pq$, *peut décider efficacement si un élément de* $\mathbb{Z}_N$ *est* RQ *ou non.*

Bibliographie

[BS] Eric Bach and Jeffrey Shallit. Algorithmic Number Theory, Volume I : Efficient Algorithms. Published by MIT Press, August 1996.

[FLA] Daniel E. Flath. Introduction to number theory. Wiley, 1988.

[KL] Jonathan Katz, Yehuda Lindell. Introduction to Modern Cryptography : Principles and Protocols. Chapman and Hall/CRC, 2007.

[MUR] M. Ram Murty. Problems in Algebraic Number Theory. Springer, 2004.

[NI] A. Nitaj. Notes de Cours cryptographie avec Maple. 2013.

[STI] Douglas Stinson. Cryptography : Theory and Practice. CRC Press, Third edition, 2002.