CRYPTOGRAPHIE

Un système de cryptographie est une technique permettant de protéger une communication au moyen d'un code secret.

Il existe deux grands types de cryptographie :

- La cryptographie symétrique à clé secrète : la même clé est utilisée pour chiffrer et pour déchiffrer l'information. Une difficulté essentielle est de pouvoir transmettre de façon sécurisée la clé à son correspondant et de la conserver secrète. L'interception de cette clé par un tiers permet en effet de décoder les messages privés et permet aussi d'envoyer de faux messages. Le système utilisé dans l'antiquité par César en est un exemple très simple : les lettres étaient toutes toujours décalées de la même façon, la simple observation de la fréquence d'apparition des lettres permet assez facilement de retrouver le principe qui a servi à chiffrer le message.

- La cryptographie dissymétrique à clé publique : deux clés différentes servent à chiffrer et à déchiffrer. L'utilisateur possède deux clés : une clé publique et une clé privée. La clé publique est publiée (dans un annuaire), elle sert à chiffrer les messages destinés à l'utilisateur, mais l'utilisateur garde secrète sa clé privée qui lui sert à déchiffrer les messages qui lui sont envoyés. Ainsi tout le monde peut écrire des messages à l'aide de la clé publique, tout le monde peut les lire (sans les comprendre) mais l'utilisateur est le seul à pouvoir les déchiffrer à l'aide de sa clé privée. Le système RSA en est un exemple.

Il existe d'autres systèmes qui mélangent les techniques à clé symétriques et les techniques à clé publique. Un des plus récent est le système Cayley-Purser pour lequel un prix international a récompensé en 1999 Sarah Flannegan, jeune irlandaise de seize ans.

Des très grands chercheurs associés à la naissance de l'informatique étaient aussi des spécialistes de la cryptographie : Charles Babbage (1894), Alan Turing : il a été un des membres les plus importants du groupe de personnes réunies pendant la deuxième guerre mondiale pour comprendre le fonctionnement de la machine Enigma construite par l'armée allemande pour chiffrer ses messages et dont un exemplaire a été envoyé en Angleterre par des résistants ; déchiffrer le courrier de l'armée ennemie et lui envoyer des faux messages a joué un rôle important dans le déroulement du conflit.

Mais d'abord un peu de calcul…

1. Arithmétique modulaire

Si on prend une division dans les entiers, comme $\frac{253}{19}$, on obtient un quotient décimal, soit ici 13,315… dont la partie entière est 13. Avec Excel, pour obtenir 13 on tape « =ENT(253/19) » et on a 13.

Maintenant quand on enlève 13 fois le diviseur 19, on récupère le reste : « =253−19*ENT(253/19) », soit 6. Ce résultat peut être obtenu avec la fonction **modulo** : « =MOD(253 ;19) » redonne bien 6.

On dit que 6 est le reste de 253 **modulo** 19.

1. Réaliser les opérations précédentes avec Excel en utilisant la disposition suivante : on peut d'abord compléter ce tableau avec les fonctions Excel.

	A	B	C	D	E
1	Dividende	Diviseur	Quotient entier	Reste	Modulo
2	253	19	=….	=….	=….
3	1011	27	=….	=….	=….
4					

Comme évidemment le reste (parfois appelé *résidu*) est inférieur au diviseur (également appelé *module*) d, ce reste peut prendre toutes les valeurs comprises entre 0 et $d-1$ pour n'importe quel dividende D. On note F_d l'ensemble des restes modulo d, par exemple $F_5 = \{0, 1, 2, 3, 4\}$.

2. En considérant tous les restes modulo 7 puis modulo 12 et modulo 17, compléter le tableau du fichier *restes1.xls*.

L'idée de l'arithmétique modulaire est que l'on peut calculer avec les restes comme avec les nombres habituels mais avec quelques petites subtilités. Par exemple si on effectue des additions modulo 5, on a la table d'addition suivante :

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	(5 =) 0
2	2	3	4	(5 =) 0	(6 =) 1
3	3	4	(5 =) 0	(6 =) 1	(7 =) 2
4	4	(5 =) 0	(6 =) 1	(7 =) 2	(8 =) 3

Par exemple $3 + 4 = 7$ mais le reste modulo 5 de 7 est 2, donc dans la table on a 2.

Le même tableau peut se faire pour la multiplication :

×	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	(6 =) 1	(8 =) 3
3	0	3	(6 =) 1	(9 =) 4	(12 =) 2
4	0	4	(8 =) 3	(12 =) 2	(16 =) 1

3. Refaire des tableaux similaires pour les modules 7, 12 et 17 en complétant le fichier restes2. On expliquera au préalable la formule de la cellule C4 (pour quelles raisons les $ sont-ils placés de cette manière).

Un des aspects fondamentaux de ces questions est que l'on peut résoudre des équations dans F_d comme dans l'ensemble des nombres rationnels ou réels.

Par exemple on veut résoudre l'équation $3x+4=0$ dans F_5 :

$3x+4=0$: il faut se débarasser du 4, je regarde dans la table d'addition et je vois qu'en ajoutant 1 des deux côtés j'obtiens $3x+4+1=1 \Leftrightarrow 3x+5=1 \Leftrightarrow 3x=1$ puisqu'on calcule modulo 5.

Maintenant il faut que je divise par 3, ou encore que je multiplie par quelque chose des deux côtés de manière à isoler x. Je regarde dans la table de multiplication et je vois qu'en multipliant des deux côtés par 2 je fais apparaître 6, soit 1 modulo 5 à gauche : $3x=1 \Leftrightarrow 2\times 3x=2\times 1 \Leftrightarrow 6x=2 \Leftrightarrow x=2$. La solution est donc 2, ce que l'on peut vérifier : $3\times 2+4=6+4=10=0$ (modulo 5 toujours !)

4. Résoudre dans F_5 les équations suivantes : $4x+2=0$, $2x+1=0$, $24x+37=119$, $x^2+1=0$.

5. Ces résolutions d'équations sont possibles car chaque nombre a un opposé et un inverse dans F_5 : en regardant dans les tables on a immédiatement

Reste	Opposé	Inverse
0	0 (0+0=0)	
1	4 (1+4=0)	1 (1.1=1)
2	3 (2+3=0)	3 (2.3=6=1)
3	2 (3+2=0)	2 (3.2=6=1)
4	1 (1+4=0)	4 (4.4=16=1)

Compléter les colonnes opposé et inverse du fichier *restes2.xls* en vous servant des réponses du 3. Quel semble être le problème dans F_{12} ?

6. Résoudre dans F_7, dans F_{12} et dans F_{17} les équations suivantes : $6x+9=0$, $8x+7=0$, $24x+37=119$, $x^2+1=0$.

Le problème dans le cas du module 12 est lié à ce que 12 n'est pas un nombre premier : on démontre que si le module est un nombre premier p, alors tous les éléments de F_p ont un inverse comme on le voit pour 7 et 17.

2. Une première approche

La méthode employée par César telle que rapportée dans la Guerre des Gaules est la suivante :

Chaque lettre du message est remplacée par sa suivante dans l'alphabet.

AH J AIME TANT LES MATHEMATIQUES

devient alors

BI K BJNF UBOU MFT NBUIFNBUJRVFT

qui semble bien protéger notre message alors que ce n'est pas du tout le cas… : on peut assez facilement retrouver le texte original à partir de la fréquence des lettres dans la langue de départ (le français ici où le E et le A ont des fréquences importantes).

La parade est alors de faire un décalage variable : on choisit une « clef » comme le mot FRED et on code en décalage pour chaque lettre :

A	H	J	A	I	M	E	T	A	N	T	L	E	S	Texte à coder
65	72	74	65	73	77	69	84	65	78	84	76	69	83	Codage ASCII
F	R	E	D	F	R	E	D	F	R	E	D	F	R	Clef répétée
70	82	69	68	70	82	69	68	70	82	69	68	70	82	Codage ASCII de la clef
5	24	13	3	13	3	8	22	5	4	23	14	9	9	Somme des codages modulo 26
F	Y	N	D	N	D	I	W	F	E	X	O	J	J	Décodage ASCII

AH J AIME TANT LES MATHEMATIQUES devient alors FY N DNDI WFEX OJJ QDYYIPFKMTZVW, évidemment bien moins facile à décrypter !

Ceci dit on peut utiliser le codage affine pour une première approche : on utilise dans un exemple la table de multiplication de F_7 pour simplifier.

On donne une valeur aux lettres, on passe à l'inverse pour le module chois (7 ici) puis on recode en lettre. Par exemple

au codage : $\begin{matrix} A \to 1 = 1[7] \underbrace{\longrightarrow}_{inverse} 1[7] \to A \\ B \to 2 = 2[7] \underbrace{\longrightarrow}_{inverse} 4[7] \to D \end{matrix}$; le décodage est alors évident. On peut évidemment mixer avec la méthode de César ou un système à clé pour compliquer.

7. a. Quels sont les inconvénients du module 7 ? Quel module a-t-on intérêt à choisir au minimum ?

b. Coder grâce à cette méthode et au module 29 la phrase suivante :

AH J AIME TANT LES MATHEMATIQUES

c. Décoder, toujours avec le module 29, la phrase suivante : Q AIBRU PBRCBRUZ

Du point de vue cryptographique le système proposé n'est pas très fiable car on sait très bien et très facilement résoudre l'équation $ax \equiv b[p]$. N'importe quel ordinateur pourra décrypter votre message en moins de deux… Dommage…

Ceci dit si on reprend la question de la clef, c'est quand même le moyen le plus sûr de coder un message : on montre qu'en prenant une clef aléatoire de longueur égale au message et à condition de l'utiliser une seule fois, le message est indécryptable. Seulement il faut transmettre la clef à votre correspondant… et c'est ici que les ennuis commencent !

3. Le logarithme discret

La multiplication modulaire donne évidemment accès à la notion de puissance modulaire : reprenons le module 7 et regardons ce que donnent les diverses puissances des restes :

	exposant	1	2	3	4	5	6	7	8	9	10
reste	2	2	4	8	16	32	64	128	256	512	1024
	3	3	9	27	81	243	729	2187	6561	19683	59049
	4	4	16	64	256	1024	4096	16384	65536	262144	1E+06
	5	5	25	125	625	3125	15625	78125	390625	2E+06	1E+07
	6	6	36	216	1296	7776	46656	279936	2E+06	1E+07	6E+07

	exposant	1	2	3	4	5	6	7	8	9	10
reste	2	2	4	1	2	4	1	2	4	1	2
	3	3	2	6	4	5	1	3	2	6	4
	4	4	2	1	4	2	1	4	2	1	4
	5	5	4	6	2	3	1	5	4	6	2
	6	6	1	6	1	6	1	6	1	6	1

Dans le premier tableau le calcul est fait de manière brutale, on voit que les résultats deviennent énormes très vite et qu'ils vont dépasser rapidement les capacités de calcul de la machine. Dans le deuxième tableau les opérations sont faites en arithmétique modulaire et les résultats sont vraiment plus simples.

1. a. Refaire ces deux tableaux avec le tableur de sorte que les calculs soient réalisables même pour de grands modules et de grands exposants.

b. Regarder attentivement le deuxième tableau et émettre quelques conjectures.

c. Reprendre ce deuxième tableau en changeant de module : on prendra des modules premiers et des modules non premiers. Vos conjectures restent-elles valables dans chaque cas ?

2. Calculer 5^{32} , 6^{21} , 3^{2008} modulo 7 puis modulo 17.

On va s'intéresser au module 29 de nouveau.

3. a. Refaire un tableau semblable pour 29.

b. Que pouvez-vous dire de la colonne où se trouve l'exposant 28 ? Pensez-vous qu'il s'agit d'une propriété générale ?

c. Vu ce qu'on vient de découvrir en 2.b., il est inutile de considérer les exposants supérieurs à 28. Pourquoi ?

d. Regardez maintenant les lignes correspondant aux restes 2, 3, 4, 7, 17 et 28. Pouvez vous constater quelque chose ?

On a dans ces tableaux de nombreux théorèmes d'arithmétique très importants.

Le premier, appelé « **petit** » **théorème de Fermat** dit que si a et p sont premiers entre eux (a et p n'ont pas de diviseurs communs), alors $a^{p-1} \equiv 1[p]$: c'est ce que vous pouvez observer dans la colonne 6 du tableau ci-dessus. Souvent on prend pour p un nombre premier et pour a un nombre inférieur à p ; on est alors sûr que les conditions sont remplies.

Evidemment dans le tableau, lorsque l'exposant dépasse 6 on retrouve les mêmes résultats, il est donc inutile de calculer le tableau d'exposants au-delà de 6, ou de $p-1$ dans le cas général d'un module p.

Voir le fichier *exposants1.xls*

Une deuxième constatation est que suivant les lignes, donc suivant les restes considérés, les résultats coïncident avec tous les restes possibles ou pas : dans la ligne 2 les résultats sont 2, 4 et 1 alors que dans la ligne 3 les résultats sont 3, 2, 6, 4, 5, 1, soit tous les éléments de F_7 (sauf 0) mais dans le désordre. Un nombre tel que 3, pour lequel tous les restes sont atteints, est appelé **élément générateur** ou **racine primitive** du module 7.

Pour 7 on a donc deux racines primitives, 3 et 5,

pour 11 on a 2, 6 et 8,

pour 17 on a 3, 5, 6, 7, 10, 11, 12, 14,

pour 29 on a 2, 3, 8, 10, …

Voir le fichier *exposants2_generateurs.*

Les générateurs sont évidemment très importants, mais comme vous pouvez le voir ci-dessus leur apparition est assez aléatoire : il n'existe d'ailleurs pas de réel moyen de calcul permettant de déterminer les générateurs d'un module donné. On pourrait penser qu'en général 2, 3 ou 5 pourraient faire l'affaire, mais il n'y a rien d'assuré dans ce domaine ; par exemple le premier générateur de 409 est 22.

Réciproquement supposons que l'on connaisse le module, disons 29, le résultat, disons 19, et le générateur, disons 3, quel est le reste dont on est parti ? La question ici est d'arriver à résoudre l'équation d'inconnue x,

$$3^x \equiv 19[29].$$

C'est ce qu'on appelle le problème du **logarithme discret** (x est le logarithme cherché, il vaut 13 pour cet exemple car $3^{13} \equiv 19[29]$) et à part des méthodes exhaustives (exploration de tous les cas possibles) il est impossible à résoudre actuellement pour des modules premiers assez grands (300 chiffres) et des exposants de l'ordre de 100 chiffres.

4. Le système Diffie - Hellmann

Alors comment va-t-on utiliser cette grande difficulté pour la cryptographie ?

On fabrique d'abord un système de chiffrage à clef comme pour le code de César : dans ce système on aura deux correspondants Alice et Bob qui peuvent s'échanger leurs messages chiffrés. Il suffit donc qu'ils aient la clef de chiffrement ; le problème est alors uniquement de transmettre la clef de manière sécurisée. Par exemple dans un système de cartes de crédit le commerçant va demander une autorisation à un central lequel va renvoyer une autorisation chiffrée. Personne ne doit pouvoir déchiffrer cette autorisation, sinon cela voudrait dire que l'on peut la reproduire autant que l'on veut…

Alors ça se passe comme ça :

Au départ Alice et Bob sont d'accord sur un module p et un générateur g (module et générateur peuvent d'ailleurs être connus des attaquants potentiels).

Alice choisit a et envoie g^a à Bob ; Bob choisit b et envoie g^b à Alice ; la clef qui pourra alors être calculée par les deux et qui sera identique sera $\left(g^a\right)^b = g^{ab}$ calculée par Bob et $\left(g^b\right)^a = g^{ba}$ calculée par Alice.

Par exemple, module 29 , générateur 3 : Alice choisit 13, calcule $3^{13} = 19[29]$ et envoie donc 19, Bob choisit 17, calcule $3^{17} = 2[29]$ et envoie 2.

Alice calcule $(2)^{13} = 14[29]$, de même Bob calcule $(19)^{17} = 14[29]$. Nos deux amis ont donc la même clef, 14, mais Bob ne sait pas qu'Alice avait choisi 13 de même qu'Alice ne sait pas que Bob a choisi 17, et à moins de savoir résoudre les équations $3^x = 19[29]$ et $3^y = 2[29]$, il est impossible de trouver la clef finale.

C'est ce qu'on appelle des systèmes à clef publique : Alice peut envoyer 19 en clair, Bob envoyer 2 en clair, on ne peut pas trouver le résultat.

Le seul inconvénient de ce système est que le système est vulnérable à l'attaque de *l'homme du milieu* (voir http://fr.wikipedia.org/wiki/%C3%89change_de_cl%C3%A9s_Diffie-Hellman) mais les avantages sont nombreux, les calculs étant très faciles même avec de très grands nombres.

5. Le système RSA

Le système RSA, qui est utilisé actuellement pour sécuriser environ 85% des échanges mondiaux a été publié le 4 avril 1977 par Ronald Rivest, Adi Shamir et Leonard Adleman.

Le système RSA est un système de cryptographie dissymétrique à clé publique très sûr. La puissance du système RSA repose sur l'idée que tous les systèmes de cryptographie adaptés aux communications de masse peuvent être forcés mais on peut parvenir à une sécurité suffisante en rendant totalement irréaliste la quantité de travail qu'il faudrait fournir pour cela.

Il est fondé sur le fait qu'on ne connaît pas d'algorithme, exécutable en un temps raisonnable, capable de décomposer de très grands nombres (ayant plus d'une centaine de chiffres) en produit de facteurs premiers. Ainsi si on fait le produit de deux nombres premiers de plus de 100 chiffres chacun, personne ne peut décomposer le nombre obtenu sauf celui qui a fait le produit.

Le système RSA assure la confidentialité et l'intégrité de la correspondance, il permet de plus d'en établir l'authenticité.

La clé publique (qui est un produit de deux nombres premiers) du système RSA utilisé pour les cartes bancaires possède (possédait ?) 239 chiffres.

En 1998 onze équipes réparties dans le monde entier ont mis trois mois pour factoriser RSA-155 sur 300 ordinateurs différents (en séparant les calculs) (d'après Promenades Mathématiques de F. Laroche édition Ellipses). Pour augmenter la sécurité du système RSA il suffit d'augmenter le nombre de chiffres, mais il faudra adapter le matériel et peut-être attendre plus longtemps aux caisses des supermarchés…

5-1 : La méthode

On choisit p et q, deux nombres premiers distincts. On note n leur produit, appelé « **module de chiffrement** » : $n = pq$.

On calcule **l'indicatrice d'Euler** de n : $\varphi(n)=(p-1)(q-1)$.

On choisit e, un entier premier avec $\varphi(n)$, appelé « **exposant de chiffrement** ».

Comme e est premier avec $\varphi(n)$, il est, d'après le théorème de Bézout, inversible mod $\varphi(n)$, c'est-à-dire qu'il existe un entier d tel que $ed\equiv 1[\varphi(n)]$. d est l' « **exposant de déchiffrement** ».

Le couple (n,e) est appelé « **clef publique »**, alors que le couple (n,d) est appelé « **clef privée »**.

5-2 : Chiffrement du message

Si m est un entier inférieur à n représentant un message, alors le message chiffré sera représenté par $c\equiv m^e[n]$.

5-3 : Déchiffrement du message

Pour déchiffrer c, on utilise d, l'inverse de e modulo $\varphi(n)$ et on calcule $c^d[n]$.

On a alors $c^d[n]\equiv\left(m^e\right)^d[n]\equiv m^{ed}[n]$.

Comme $ed\equiv 1[\varphi(n)]\Leftrightarrow ed=1+k\varphi(n)=1+k(p-1)(q-1)$, $k\in\mathbb{N}$, on remplace :

$m^{ed}[n]\equiv m^{1+k(p-1)(q-1)}[n]\equiv m[p]\equiv m[q]$ car si m est premier avec p alors, d'après le petit théorème de Fermat,

$m^{p-1}\equiv 1[p]\Rightarrow\left(m^{p-1}\right)^{q-1}\equiv 1[p]\Rightarrow\left(m^{(p-1)(q-1)}\right)^k\equiv 1[p]\Rightarrow m^{1+k(p-1)(q-1)}\equiv m[p]$.

Si m n'est pas premier avec p, comme p est un nombre premier, cela signifie que m est multiple de p donc

$m^{1+k(p-1)(q-1)}\equiv 0[p]\equiv m[p]$. Un raisonnement analogue prouve la congruence modulo q.

L'entier $m^{1+k(p-1)(q-1)}-m\equiv 0[p]\equiv 0[q]$ est donc un multiple de p et de q.

Comme p et q sont premiers (et premiers entre eux), le lemme de Gauss permet d'affirmer que $m^{1+k(p-1)(q-1)}-m\equiv 0[pq]\equiv 0[n]$.

On a finalement $c^d\equiv m^{ed}[n]\equiv m^{1+k(p-1)(q-1)}[n]\equiv m[n]$.

Pour chiffrer un message, il suffit évidemment de connaître e et n.

En revanche pour déchiffrer, il faut d et n.

Pour calculer d à l'aide de e et n, il faut trouver l'inverse de e modulo $(p-1)(q-1)$ ce qui nécessite de connaitre les entiers p et q, c'est-à-dire la décomposition de n en facteurs premiers.

5-4 : L'utilisation

Authentification

Tous les éléments du message sont transformés en nombres (par exemple en utilisant le code ASCII) et le message est transformé en blocs de nombres (inférieurs à n).

Alice et Bob (prénoms traditionnels pour les personnes A et B) veulent communiquer secrètement. Alice utilise le système RSA : elle choisit deux nombres premiers p et q, elle calcule les nombres n, e et d, elle fait connaître le couple clé publique (n, e), et elle garde secret la clé secrète d qu'elle utilisera pour décoder les messages de Bob.

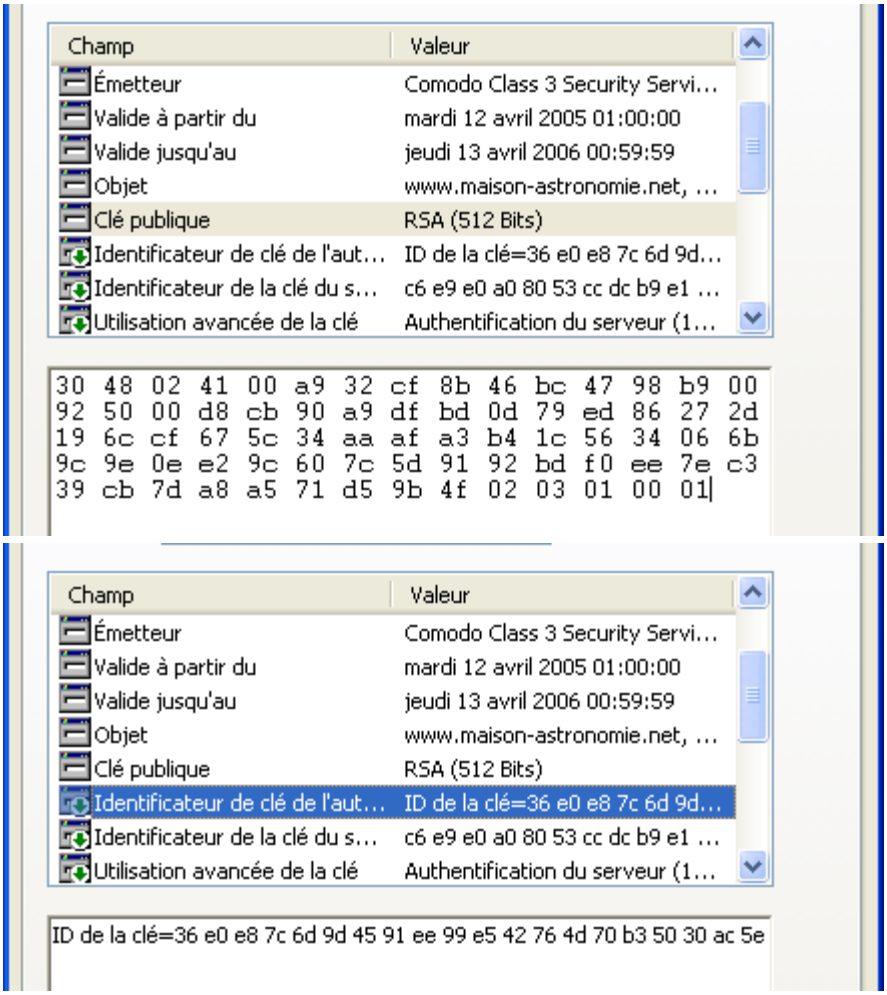
Bob utilise la clé publique (e, n) d'Alice pour chiffrer son message :	Message envoyé	Alice utilise sa clé privée (d, n) pour déchiffrer le message :
$c\equiv m^e[n]$	c	$c^d\equiv m^{ed}[n]\equiv m[n]$

Bob utilise aussi un système RSA et cela va permettre d'authentifier son message. Bob peut le signer en ajoutant à son message initial sa signature codée avec sa propre clé privée d' et Alice, à la fin du décodage avec sa clé privée d, utilisera la clé publique e' de Bob pour décoder la signature de Bob. Le message sera authentifié car seul Bob est capable de coder sa signature :

Signature de Bob	Bob utilise sa clé privée	Bob crypte avec la clé publique d'Alice	Alice décrypte avec sa clé privée	Alice décrypte avec la clé publique de Bob
s	$\sigma = s^{d'} [n]$	Σ	S	s

Un achat sur Internet

Un exemple trouvé lors d'un achat sur Internet. Il s'agit d'un **certificat** où apparaît le nombre *n* de la clé publique écrit en système hexadécimal (base 16) ainsi que l'identificateur de la clé de l'autorité qui a établi le certificat.



Utilisation de certificats dans la pratique d'après : www.cryptosec.org

Afin de s'assurer de l'appartenance d'une clé publique on utilisera souvent des certificats X.509v3, bien que PGP et GPG (autres formats, non signés par une Autorité) en soient des contre-exemples de poids. Le certificat contient une clé publique, et la signature d'une autorité reconnue par les protagonistes. Cette clé pourra aussi bien être une clé de chiffrement, une clé de vérification ou bien une clé qui remplira les deux fonctions.

D'après la revue Tangente (n°107 novembre-décembre 2005) : le Haut Comité Européen de lutte contre les codages illicites rappelle que la détention de nombres premiers supérieurs à 2^{128} est désormais soumise à autorisation. Il demande donc que tout détenteur d'un nombre strictement supérieur le fasse tester. Un test rapide portant sur le dernier chiffre permet d'éliminer 60% des suspects. Les 40% restants doivent impérativement subir un test de Miller-Rabin.