Fiche méthode : chiffrement symétrique

Alice veut envoyer un message à Bob. Pour garantir la confidentialité, elle décide de chiffrer son message avant de le transmettre. Pour simplifier, elle n'utilise que les 32 caractères donnés dans le tableau ci-dessous :

Le chiffrement **symétrique** s'effectue à l'aide d'une clé K que seuls Alice et Bob connaissent. Ici la clé est un mot binaire de longueur six : **K = 110110**

Principe d'un chiffrement symétrique (Vigenère) :

- chaque caractère du message est converti en un mot binaire de longueur 5, comme indiqué dans le tableau
- chaque bit du message binaire **M** ainsi obtenu est mis en face d'un des bits de la clé **K**, répétée autant de fois que nécessaire
- l'opération de chiffrement consiste à calculer le **xor** bit à bit

a	0	0	1	1
b	0	1	0	1
a **xor** b	**0**	**1**	**1**	**0**

- le message chiffré **C** s'obtient après le caclul du xor en convertissant chaque mot de 5 bits en un caractère, comme indiqué dans le tableau.

Exemple : le message ALAN est chiffré ZAWU

Message M	A	L	A	N	
bits de M	00001	01100	00001	01110	
bits de K	**11011**	01101	10**11**0	**110**11	0110
bits de C	11010	00001	10111	10101	
Message chiffré :C	Z	A	W	U	

Déchiffrement : le déchiffrement s'opère exactement comme le chiffrement, avec la même clé K

Message chiffré :C	Z	A	W	U	
bits de C	11010	00001	10111	10101	
bits de K	**11011**	**0**1101	10**110**	**110**11	0110
bits de M	00001	01100	00001	01110	
Message M	A	L	A	N	

Remarque : la méthode de chiffrement et la méthode de déchiffrement utilisent la même clé. C'est pourquoi on parle de chiffrement **symétrique**.

0		00000
1	A	00001
2	B	00010
3	C	00011
4	D	00100
5	E	00101
6	F	00110
7	G	00111
8	H	01000
9	I	01001
10	J	01010
11	K	01011
12	L	01100
13	M	01101
14	N	01110
15	O	01111
16	P	10000
17	Q	10001
18	R	10010
19	S	10011
20	T	10100
21	U	10101
22	V	10110
23	W	10111
24	X	11000
25	Y	11001
26	Z	11010
27	.	11011
28	,	11100
29	!	11101
30	?	11110
31	:	11111

Fiche méthode : chiffrement asymétrique

Alice souhaite envoyer à Bob le mot binaire **K = 110110** dont l'écriture décimale est : **M = 54**.

Pour cet envoi, Alice et Bob vont utiliser un chiffrement asymétrique.

Principe d'un chiffrement asymétrique (RSA) :

- Bob génère deux clés (le détail du calcul des clés n'est pas présenté ici) :
  - une **clé publique** (e,n) qu'il envoie à Alice : (9,69)
  - une **clé privée** (d,n) qu'il garde secrète : (5,69)
- **Alice** chiffre l'entier M=54 avec la **clé publique** en calculant **C = M**e % n** soit ici :
  C = 54**9 % 69 = 3904305912313344 % 69 = 9 puisque $54^9 = 69 \times 56584143656715 + 9$
- Alice envoie le nombre chiffré C = 9 à Bob.
- **Bob** le déchiffre avec sa **clé privée** en calculant **D = C**d % n** soit ici :

  D = 9**5 % 69 = 59049 % 69 = 54 puisque $9^5 = 59049 = 69 \times 855 + 54$

Ainsi, Bob peut connaître l'entier 54 choisi par Alice, sans que cet entier ait été envoyé dans un de leurs messages.

En pratique, l'entier n utilisé dans ce type d'algorithme est le produit de deux très grands nombres premiers. Ainsi, il est impossible de retrouver la clé privée à partir de la seule clé publique en un temps raisonnable.

Ici $n = 69 = 3 \times 23$ est bien le produit de deux nombre premiers, mais il serait très rapide de trouver la clé privée (5,69) à partir de la clé publique (9,69), ou de tenter une attaque de type « force brute » en essayant toutes les clés possibles.