UFR Sciences et techniques
Licence renforcé de mathématiques

Rapport du projet personnel de recherche

Cryptologie

Coupeaux Bastien
2022-2024

Tuteur : M. Robert
Laboratoire : IMATH

Je tenais à remercier Mr Jean-Marc Robert pour l'aide, les connaissances ainsi que le temps qu'il m'a accordé pour mon projet personnel de recherche.

Table des matières

1 Introduction

En 404 avant Jésus-Christ, Lysandre de Sparte, général spartiate, vit un de ses soldats revenir suite à un long voyage depuis Sparte. Ce dernier était en sang et fit alors don à son général de sa ceinture. Lysandre comprit que cette ceinture était un message qu'il devait entourer autour de sa scytale afin d'en connaître le sens et apprit dès lors que Pharnabaze de Perse allait attaquer.

C'est ainsi qu'est racontée par Plutarque une des premières utilisations d'un des premiers dispositifs de cryptographie militaire connus : la scytale. Cette dernière est décrite par Plutarque comme un rouleau autour duquel on entoure un bout de papier ou une lanière de cuir, préparé au préalable sur un rouleau de dimension identique et incompréhensible sans support. Ainsi, une fois le manuscrit enroulé autour du support, le texte écrit sur le manuscrit devient alors compréhensible.

Depuis la scytale, les systèmes cryptographiques ont grandement évolué : en effet, si les premiers systèmes comme la scytale n'étaient pas très complexes à déchiffrer (il suffisait, en l'occurrence, pour cette dernière de simplement posséder un support de dimension identique), de nouveaux systèmes bien plus complexes à "casser" sont venus les remplacer avec notamment le développement des mathématiques. L'apparition de ces nouvelles méthodes de cryptage fait alors lien au besoin essentiel pour l'Homme qui est de pouvoir communiquer un message à quelqu'un sans que personne d'autre puisse comprendre le sens de ce dernier. La cryptologie a alors au départ simplement un but de confidentialité.

Ce n'est que très récemment, avec le développement de l'informatique, que la cryptologie a acquis de nouveaux objectifs qui sont d'assurer l'intégrité des données et d'assurer l'authenticité.

La cryptologie se fragmente en deux branches :
- La cryptographie, l'écriture secrète, qui consiste en l'étude et en la mise en place de systèmes cryptographiques.
- La cryptanalyse, l'analyse de la cryptographie, qui consiste en la recherche de méthodes afin de pouvoir décrypter les systèmes mis en place par la cryptographie.

La cryptographie, malgré la multitude de systèmes cryptographiques existants, fonctionne toujours selon le même principe : on choisit une information appelée "texte clair" que l'on souhaite communiquer et on va la chiffrer à partir d'une "fonction de chiffrement" dont l'inverse, appelée "fonction de déchiffrement", permet de déchiffrer aisément le message une fois chiffré.

La cryptanalyse, quant à elle, va tenter de "casser" les systèmes mis en place par la cryptographie à partir de différentes méthodes mathématiques que l'on évoquera plus tard.

Nous allons d'abord effectuer la cryptanalyse et la cryptographie de deux systèmes cryptographiques assez simples avant de parler en détail de la sécurité des systèmes cryptographiques et de la méthode de chiffrement la plus utilisée à l'heure actuelle, le chiffrement AES.

2 Cryptographie et cryptanalyse de systèmes “simples”

2.1 Chiffrement par substitution : Le Chiffrement de César

Lorsqu'une personne souhaite s'intéresser à la cryptologie, l'un des premiers systèmes auquel cette dernière doit se confronter est une méthode de chiffrement utilisée par l'imperator Jules César et qui porte son nom : le chiffrement de César. Ce système constitue un cas particulier d'une grande famille de chiffrement : le chiffrement par décalage.

2.1.1 Cryptographie de ce système

Le chiffrement de César est un cas particulier de chiffrement par décalage dont la clé de chiffrement vaut 3. On s'intéresse à la façon dont fonctionne ce type de système cryptographique.

Le principe de chiffrement est le suivant : on va décaler l'alphabet d'un certain nombre de caractères puis remplacer chaque lettre du texte clair par les lettres du “nouvel” alphabet obtenu par décalage. Cela peut sembler assez complexe mais en réalité il n'en est rien :

On va d'abord commencer par attribuer à chaque lettre de l'alphabet un chiffre allant de 0 pour A à 25 pour Z. Ensuite, on va choisir un nombre qui constitue la clé de notre chiffrement et donc le décalage de notre alphabet. Enfin, on va additionner cette clé aux différents nombres associés à chaque lettre puis on réduira le tout modulo 26. Il ne nous restera alors plus qu'à réécrire le texte à partir de la nouvelle correspondance.

Lettre	A	B	C	D	...	Y	Z
Chiffre	0	1	2	3	...	24	25

TABLE 1 – Tableau de correspondance entre lettre et chiffre sans clé

Mathématiquement, tout cela se traduit par la fonction de chiffrement suivante : soit $K \in [0; 25]$ [1], la clé choisit. Soit $x \in [0; 25]$, le nombre associé à la lettre que l'on souhaite chiffrer. La fonction de chiffrement s'écrit alors :

$$\boxed{C(x) = (x + K) \pmod{26}} \tag{1}$$

On définit alors aisément une façon de déchiffrement si l'on possède la clé : on prend le nombre associé à la lettre chiffrée, on lui soustrait la valeur de la clé choisie et on associe le résultat modulo 26.

1. Le fait que les valeurs possibles pour la clé sont comprises entre 0 à 25 résulte du fait que modulo 26, choisir pour clé 0 ou 26 est équivalent par exemple.

La fonction de déchiffrement associée est alors définie de la sorte : soit $K \in [0; 25]$, la clé choisie. Soit $y \in [0; 25]$, le nombre associé à la lettre chiffrée. La fonction de déchiffrement s'écrit alors :

$$\boxed{D(y) = (y - K) \pmod{26}} \tag{2}$$

2.1.2 Cryptanalyse de ce système

Comme l'indique le nom, le chiffrement de César date de l'époque de Jules César et comme pour la plupart des systèmes de cette époque, on sait comment déchiffrer facilement ce type de chiffrement. On va s'intéresser à deux méthodes permettant de "casser" le chiffrement de César.

La recherche exhaustive de clés

La première méthode est la plus simple des deux que l'on va étudier. En effet, cette méthode propose simplement de tester toutes les valeurs de clés possibles étant donné qu'il n'y en a que 26. Ainsi, le déchiffrement se fait assez rapidement et nous assure de réussir sur n'importe quel type de texte, qu'il soit long ou court. Malgré cela, même si cette méthode fonctionne bien pour les alphabets ayant peu de caractère comme l'alphabet latin qui n'a que 26 lettres, elle devient dès lors beaucoup plus fastidieuse pour des alphabets ayant plus de caractère comme l'alphabet japonais qui en comprend 46 par exemple. C'est pour cela que l'on va notamment étudier la seconde méthode qui est généralement bien plus rapide pour casser le chiffrement de César.

La méthode statistique

Cette méthode de déchiffrement se base sur un constat simple : certaines lettres apparaissent plus que d'autres dans certaines langues. C'est notamment le cas du français et de l'anglais où la lettre "e" prédomine avec près de 15 % de fréquence d'apparition dans les textes.

Lettre	E	A	I	S	N
Fréquence d'apparition	12,1%	7,11%	6,59%	6,51%	6,39%

TABLE 2 – Fréquence d'apparitions des cinq lettres les plus courantes en français.

Ainsi, on va commencer par calculer la fréquence d'apparition de chacune des lettres du texte chiffré et en faire un tableau que l'on va comparer avec les fréquences d'apparition que l'on possède dans la langue respective du texte.

Si la lettre "z" apparaît alors fréquemment dans le texte chiffré, il est fortement probable que cette lettre remplace la lettre "e". Pour obtenir la clé potentielle, il suffit alors de faire le calcul :

$$\boxed{K = positionalphabet(z) - positionalphabet(e)} \quad (3)$$

Enfin, il suffit de tester la clé et de voir si elle fonctionne ; sinon on réitère avec la seconde lettre ayant la plus grande fréquence d'apparition, ...

Finalement, bien que le chiffrement de César, et plus généralement, le chiffrement par décalage , soient des méthodes importantes dans l'histoire de la cryptologie, elles sont aujourd'hui dépassées et remplacées par des méthodes plus efficaces comme le chiffrement AES qui utilisent encore en partie cette méthode avec l'opérateur XOR.

2.2 Chiffrement par permutation : Transposition par colonnes

Les chiffrements par permutation forment, avec les chiffrements par substitution, les deux grandes familles de "chiffrement simple". Ils sont souvent plus pénibles que les chiffrements par substitution à déchiffrer, mais bien plus simples à mettre en place, comme nous allons le voir.

2.2.1 Cryptographie de ce système

Tout au long de cette partie, on va illustrer comment fonctionne le chiffrement à partir d'un exemple où l'on voudra chiffrer la suite de caractères : "CRYPTOGRAPHYISSO-COOL".

On commence d'abord par choisir un texte que l'on souhaite chiffrer puis on choisit ensuite la clé que l'on va utiliser pour chiffrer. Ici, on va choisir la clé : "MYKEY". Ensuite, on compte le nombre de lettres dans le mot que l'on souhaite chiffrer et dans notre clé, on les notera respectivement **m** et **n**. Dans notre exemple, $\mathbf{m} = 20$ et $\mathbf{n} = 5$.

On va maintenant construire un tableau où la première ligne contiendra notre mot clé, la seconde comprendra des chiffres correspondant à l'ordre alphabétique de chacune des lettres de notre mot et les lignes suivantes serviront simplement à ranger les lettres de notre mot afin de le chiffrer. La figure 1 nous donne le tableau associé à notre exemple :

Une fois l'étape du tableau accomplie, il ne nous reste plus qu'à récolter les colonnes dans l'ordre établi au préalable. Notre exemple donne alors, après chiffrement : "PASOY-RIOCOHORGYCTPSL".

M	Y	K	E	Y
3	4	2	1	5
C	R	Y	P	T
O	G	R	A	P
H	Y	I	S	S
O	C	O	O	L

FIGURE 1 – Tableau de chiffrement pour la transposition par colonnes

2.2.2 Cryptanalyse de ce système

Dans cette section, nous allons nous intéresser aux différents moyens de déchiffrer la méthode de transposition par colonnes. Cette méthode de transposition est très complexe à déchiffrer sans aucun indice et réside notamment dans le fait de tâtonner en cherchant une juxtaposition de deux colonnes, ce qui peut être très pénible. C'est pour cela que nous allons rapidement évoquer l'utilisation de la méthode force brute avant d'étudier une attaque précise vis-à-vis de ce chiffrement : l'attaque à clé de chiffrement connue.

La méthode force brute

Comme pour le chiffrement de César, la méthode force brute (qui correspond à la recherche exhaustive de clés) est également applicable à la transposition par colonne où nous allons tout simplement tester toutes les permutations possibles pour le texte. Par rapport au chiffrement de César, cette méthode n'est souvent pas utilisée pour déchiffrer car si le chiffrement de César possédait peu de valeurs de clé possibles, la transposition par colonnes peut par contre arriver rapidement à un grand nombre de permutations. En effet, pour des textes courts, la méthode est assez efficace mais pour des textes plus longs, nous ne possédons souvent pas la puissance de calcul nécessaire permettant de tester toutes les permutations possibles. C'est pourquoi nous préférerons souvent déchiffrer à partir de certains angles d'attaque très précis qui nous assureront d'obtenir à coup sûr le texte clair.

L'attaque à clé de chiffrement connue

C'est l'attaque qui permet de déchiffrer le plus facilement la méthode de transposition par colonnes. Considérons que l'attaquant possède notre texte chiffré et notre clé de chiffrement.

Dans ce cas, ce dernier va tout d'abord compter le nombre de lettres dans notre clé et notre texte chiffré. Soient **m** le nombre de lettres de notre texte chiffré et **n** le nombre de lettres de notre clé, la hauteur des colonnes nécessaire pour retrouver notre texte clair est :

$$\boxed{m \pmod{n} = h} \tag{4}$$

Une fois ceci fait, il va attribuer un numéro à chacune des lettres de notre clé, comme précédemment, selon leur ordre d'apparition dans l'alphabet. Il ne lui reste alors plus qu'à disposer **h** lettres du mot chiffré sous chacune des colonnes et il retrouve ainsi le texte clair.

3 Chiffrement AES et sécurité

Lors de la partie précédente, nous avons présenté des méthodes de cryptologie simples et faciles à comprendre, permettant à n'importe qui de saisir les principes de la cryptologie et de s'amuser avec ces derniers. Mais, la cryptologie moderne est différente et bien plus complexe que ce que nous avons vu : en effet, le rapide développement des mathématiques, mais également de l'informatique a mis à genoux la totalité des systèmes cryptographiques n'assurant alors plus la sécurité des messages transmis.

C'est pourquoi de nouvelles méthodes de communication sécurisées ont été mises en place visant à mieux protéger les données, de sorte que même les ordinateurs et machines les plus puissants ne puissent les obtenir. C'est de cette notion de sécurité des données dont nous parlerons dans cette seconde partie, avant d'aborder le chiffrement le plus élaboré et sûr existant de nos jours : le chiffrement AES.

3.1 Protection des données et confidentialité parfaite

Avec les percées technologiques effectuées au cours des derniers siècles, de nombreux problèmes mathématiques jusqu'alors irrésolus ont pu être élucidés. La cryptologie s'en est alors retrouvée directement affectée étant donné que la plupart des systèmes qui existaient reposaient essentiellement sur des opérations supposées "trop complexes" à résoudre.

Mais, en 1949, Claude Shannon publia un article intitulé " Communication Theory of Secrecy Systems " qui changea à jamais la cryptologie. Dans cet article, il introduisit une nouvelle définition de la sécurité d'un système cryptographique : la confidentialité parfaite.

DÉFINITION : *On dit qu'il y a confidentialité parfaite si l'attaquant n'obtient aucune information sur le texte clair à partir du texte chiffré.*

Cette définition peut également se présenter sous forme de probabilités et pour cela, on définit la notion de probabilité conditionnelle :

DÉFINITION : *Soit $B \subset \Omega$, un évènement tel que $P(B)>0$. Soit $A \subset \Omega$, la probabilité de A sachant B se définit alors par :*

$$\boxed{P(A|B) = \frac{P(A \cap B)}{P(B)}} \tag{5}$$

Cette définition nous permet alors d'obtenir la définition suivante de la confidentialité parfaite :

DÉFINITION : *Un système cryptographique assure une confidentialité parfaite si et seulement si $P(X|Y) = P(X)$ et si toutes les clés de chiffrement sont équiprobables, pour tout texte clair X et tout texte chiffré Y.*

Certains systèmes cryptographiques que l'on a vus dans 2 assurent cette confidentialité sous certaines conditions : en effet, le chiffrement par décalage et donc le chiffrement de César assure la confidentialité parfaite à condition de choisir une nouvelle clé pour chaque lettre. Mais ces systèmes ont des problèmes notamment au niveau de la gestion des clés et c'est pourquoi un tel système est, pour le moment, un idéal.

3.2 Le chiffrement AES

Le chiffrement AES est l'algorithme de chiffrement le plus utilisé au monde. En tant que chiffrement par bloc, il sépare le texte que l'on souhaite chiffrer en plusieurs blocs de même taille. On étudie, dans cette section, comment fonctionne cette méthode de chiffrement et on s'interroge sur sa capacité à résister à différents types d'attaques.

3.2.1 Notions nécessaires au chiffrement AES

L'écriture binaire

L'écriture binaire permet d'écrire n'importe quel nombre décimal comme une suite de 1 et de 0. Pour écrire en binaire, il suffit de faire des divisions successives par 2 jusqu'à ce que le quotient soit nul puis de récupérer tous les restes du dernier au premier et de les placer les uns à la suite des autres.

$$14 = 2 \times 7 + 0 \quad 7 = 2 \times 3 + 1$$
$$3 = 2 \times 1 + 1 \quad 1 = 2 \times 0 + 1$$
$$14 = 1110$$

FIGURE 2 – Algorithme de conversion de 14 en binaire

Conversion binaire-hexadécimal

En plus de l'écriture binaire, le chiffrement AES utilise aussi l'écriture hexadécimale. Cette écriture comprend 16 caractères : les dix premiers sont les chiffres allant de 0 à 9 et les six derniers correspondent aux six premières lettres de l'alphabet. Cette écriture s'obtient de la même façon que l'écriture binaire, mais on effectue des divisions par 16.

Pour passer de l'écriture binaire à l'écriture hexadécimale (et inversement), on utilise un tableau d'analogie présenté comme suit :

Hexadécimal	0	1	2	3	...	E	F
Binaire	0000	0001	0010	0011	...	1110	1111

TABLE 3 – Tableau d'analogie entre écriture binaire et hexadécimal

L'opérateur XOR

La fonction OU exclusif, souvent appelée XOR, est un opérateur logique très utilisé en cryptologie grâce à sa facilité d'utilisation et d'implémentation. On l'utilise souvent en

tant qu'étape intermédiaire au sein d'un chiffrement, comme dans le cas de l'AES par exemple.

Ce dernier est un simple chiffrement modulo 2 : considérons deux suites de bits A et B, si on souhaite "xorer" ces deux suites, il suffit de faire une addition modulo 2 de chaque élément des suites A et B.

3.2.2 Protocole de chiffrement

Dans notre cas, afin de simplifier la présentation de cette méthode de chiffrement, on va s'intéresser au chiffrement AES avec des blocs et des clés de taille 128 bits (chaque bloc et chaque clé sont alors composés de 4 lignes et 4 colonnes, c'est-à-dire que chaque case contient 8 bits).

3d	3f	5f	49
2c	57	e9	1f
a8	2b	1e	76
15	91	62	c8

FIGURE 3 – Exemple d'un "bloc-texte"

La première étape de cette méthode consiste en la génération aléatoire d'une clé de même taille et respectant le même schéma que les blocs. On répète ensuite 10 fois les 5 étapes suivantes pour chacun des blocs issus de notre texte :

SubBytes

Chaque case de notre bloc comporte une information écrite en hexadécimal. Dans cette étape, on substitue une à une chaque case de notre bloc à l'aide d'une table de substitution donnée appelée "S-BOX". Chaque case étant composée d'une suite de 2 caractères, pour substituer, il suffit de choisir la ligne correspondant à l'élément de droite et la colonne à celui à gauche.

		y														
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

FIGURE 4 – S-BOX

ShiftRows

Dans cette étape, on va permuter les cases de notre bloc de la façon suivante : on commence par faire passer la case de gauche de la seconde ligne, tout à droite. On fait ensuite pareil pour les deux cases de gauche de la troisième ligne, et idem pour les trois cases de gauche de la dernière ligne.

MixColumns

Cette étape est la plus complexe car elle fait intervenir des notions d'algèbre linéaire ainsi que plusieurs étapes de conversion. Elle consiste en un produit matriciel de chacune des colonnes par une matrice donnée dans le champ de Galois GF(2^8).

$$\begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \tag{6}$$

FIGURE 5 – Matrice donnée dans le cas du chiffrement AES 128

DÉFINITION : GF(2^8) *est un corps fini contenant 256 polynômes de degré inférieur ou égal à 7 à coefficients dans* $\mathbb{F}_2$*, un corps ne contenant que les éléments 0 et 1.*

Afin de pouvoir calculer le produit matrice-colonne, il faut d'abord commencer par convertir chacun des éléments de la matrice et de la colonne de l'hexadécimal vers le binaire. Ensuite, on convertit nos éléments du binaire en polynômes.

Pour passer du binaire aux polynômes, on possède la représentation suivante : soient a_0, a_1, a_2, a_3, a_4, a_5, a_6, $a_7 \in \{0,1\}$ et $a_7a_6a_5a_4 \quad a_3a_2a_1a_0$, un nombre binaire. La conversion de ce nombre en polynôme s'écrit alors :

$$\boxed{a_7a_6a_5a_4 \quad a_3a_2a_1a_0 \Rightarrow a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0} \tag{7}$$

On effectue ensuite la somme des produits de polynômes entre les éléments de la matrice et de la colonne. On obtient ainsi des polynômes de degré inférieur ou égal à 8. Comme on considère uniquement des coefficients égaux à 0 ou 1, tous les coefficients a_i devant les x^i sont remplacés par 0 si a_i est un multiple de 2, par 1 sinon (cela signifie que l'addition correspond en réalité à l'opérateur XOR). De plus, s'il y a x^8 dans le polynôme, le théorème des polynômes irréductibles nous dit alors :

$$\boxed{x^8 = x^4 + x^3 + x + 1} \tag{8}$$

Il ne reste alors plus qu'à retransformer les polynômes obtenus en binaire puis en hexadécimal pour obtenir le résultat.

AddRoundKey

"AddRoundKey" est la dernière étape où l'on va modifier notre bloc avant de changer de clé de chiffrement. On commence par convertir chaque case de notre bloc et de notre clé de l'hexadécimal vers le binaire. Ensuite, on "xore" chaque élément de notre bloc et de notre clé case par case. Enfin, on convertit de nouveau chaque résultat obtenu du binaire vers l'hexadécimale afin de retrouver notre bloc composé de cases en hexadécimal.

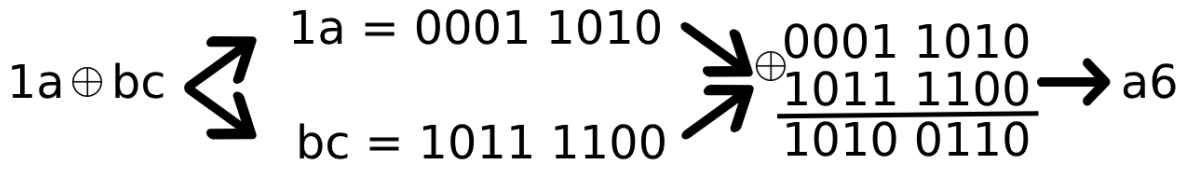

FIGURE 6 – Exemple de calcul d'une case

Clé de ronde

Enfin, avant de répéter le processus que l'on vient de présenter, on change de clé afin de renforcer la sécurité du chiffrement. Cette clé est créée à partir de la clé utilisée dans l'étape précédente de la façon suivante : on commence par extraire la dernière colonne de la clé utilisée pendant le tour qui précède cette étape. On fait ensuite passer l'élément en tête de cette même colonne en bas de cette dernière. On reprend maintenant le tableau évoqué dans SubBytes et on substitue un à un les éléments de cette colonne.

Enfin, on pose C_i, la 1ère colonne de la seconde clé. Les colonnes 2, 3 et 4 de cette seconde clé correspondent donc aux indices $i+1$, $i+2$ et $i+3$. On notera respectivement C_{i-4}, C_{i-3}, C_{i-2} et C_{i-1}, les 4 colonnes de la première clé, C_{i-4} étant la colonne la plus à gauche. Pour obtenir C_i, il faut "xorer" la colonne que l'on vient de modifier dans les étapes précédentes avec C_{i-4} ainsi qu'une constante de ronde qui varie selon le tour auquel on se situe. Pour les autres colonnes de la clé qu'on souhaite générer, il suffit de "xorer" la colonne qui les précède avec respectivement C_{i-3}, C_{i-2} et C_{i-1}.

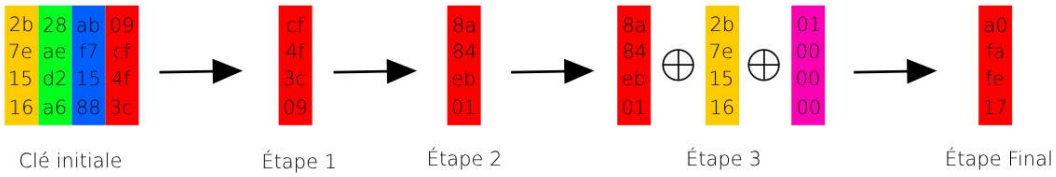

FIGURE 7 – Génération de la première colonne de la clé

La génération des clés de ronde se fait généralement avant de commencer le chiffrement afin d'éviter de potentiels problèmes dans le processus. Ces clés de ronde constituent le cœur de la sécurité de l'AES qui repose en partie sur l'utilisation de plusieurs clés.

3.2.3 Un système de chiffrement "incassable" ?

Ce qui fait que le chiffrement AES est le système cryptographique le plus utilisé de nos jours n'est pas sa simplicité (bien que ce fût un critère lorsqu'il fut mis en place) mais son universalité et sa capacité à ne pas être cassée. En effet, la puissance de calcul dont on dispose à l'heure actuelle n'est pas suffisante et nécessiterait plusieurs années pour décrypter ne serait-ce qu'un seul système.

Les seules façons que l'on a trouvées permettant potentiellement de vaincre le chiffrement AES dépendent de si le système est défaillant et laisse fuir des informations. En effet, en considérant des angles d'attaque particuliers comme l'attaque par réutilisation de clés qui exploitent le fait qu'une même clé de chiffrement est utilisée pour plusieurs "bloc-texte" et peut révéler des schémas exploitables par l'attaquant, on peut venir à bout de l'AES. Mais, pour un système AES parfaitement configuré, on ne connaît, à l'heure actuelle, aucune méthode envisageable pour déchiffrer.

Le cryptage par l'AES valide même le concept de confidentialité parfaite sous une condition particulière : si on décide de chiffrer un seul bloc. Si on ne chiffre qu'un seul bloc, il est impossible d'obtenir une information sur le texte clair à partir du texte chiffré. En réalité, la confidentialité parfaite est facile à obtenir pour une méthode de chiffrement à partir du moment où on ne chiffre qu'une donnée. Le chiffrement de César et même la transposition par colonnes vérifient ce principe à partir du moment où l'on ne chiffre qu'une seule lettre, par exemple.

C'est pourquoi il faut faire attention lorsque l'on aborde le principe de confidentialité parfaite car il faut bien considérer le cas général et non pas des cas particuliers comme évoqué ci-dessus.

C'est pourquoi le chiffrement AES ne satisfait pas, en général, le principe de confidentialité parfaite. En effet, si l'on possédait une puissance de calcul infinie ou si une découverte était faite du jour au lendemain, on pourrait en théorie parvenir à casser le chiffrement AES. C'est ce qu'ont prouvé des chercheurs du laboratoire Microsoft Research qui sont parvenus à trouver une faille permettant de découvrir directement la première clé utilisée pour le chiffrement.

Le seul problème de leur méthode réside dans le temps nécessaire à sa mise en œuvre. En effet, bien que leur méthode ait été reconnue par les créateurs de l'AES eux-mêmes, elle ne peut pas être envisagée car, selon l'un des créateurs de la faille, même si on disposait de plusieurs millions d'ordinateurs bien plus puissants que ceux dont on dispose de nos jours, il faudrait plusieurs années pour retrouver seulement la clé de chiffrement.

Néanmoins, les preuves apportées par les chercheurs de Microsoft ainsi que l'apparition de nouvelles technologies telles que l'ordinateur quantique soulèvent de véritables questions vis-à-vis de la sécurité des méthodes chiffrement modernes. L'ordinateur quantique a permis l'invention d'algorithmes comme celui de Shor en 1994, qui permet de factoriser des grands nombres qui est ce sur quoi repose la sécurité du chiffrement RSA, par exemple.

La question selon laquelle la sécurité chiffrement AES pourrait, elle aussi, se retrouver impactée par ces nouvelles techniques se pose alors même si présentement, nous sommes encore loin de voir une technologie casser le chiffrement AES. Les ordinateurs quantiques accélèrent grandement la rapidité de calcul par exemple mais ne sont pas assez puissants pour casser l'AES.

Les seules méthodes permettant de casser le chiffrement AES n'étant donc pas envisageables à l'heure actuelle, les utilisateurs de cette méthode de chiffrement peuvent, pour le moment, considérer que leurs données sont en sécurité.

4 Conclusion

Le développement des technologies ou de prodigieuses découvertes dans le domaine de la cryptanalyse pourrait bien remettre en question nos méthodes de chiffrement modernes. Le chiffrement AES, reposant sur des opérations simples mais difficiles à remettre bout à bout, pourrait se retrouver inutile face aux futurs progrès.

Au même titre que le chiffrement de César et que la transposition par colonne, le chiffrement AES pourrait alors se trouver inefficace et ne serait utile que pour comprendre comment fonctionne la cryptologie. Cette possibilité ne semble pas si éloignée avec l'amélioration des ordinateurs quantiques pouvant résoudre des systèmes mathématiques toujours plus complexes. Poussés par des processeurs de plus en plus rapides et l'intelligence artificielle, les ordinateurs quantiques ne sont plus si loin de l'exploit que serait de briser le chiffrement AES.

Il faudrait alors réinventer de nouvelles méthodes de chiffrement comme on le fait à chaque fois. La possibilité d'avoir un jour un système cryptographique assurant une confidentialité parfaite devient alors utopique car, s'il existe des méthodes de chiffrement qui assurent théoriquement cette confidentialité, telles que le chiffrement de Vernam, elles sont confrontées à de nombreux problèmes tels que la gestion, le stockage ou encore la destruction de clés par exemple.

C'est pourquoi on tente encore et encore de renforcer le chiffrement AES afin de retarder le plus possible cette échéance. On augmente par exemple la taille des clés de chiffrement passant de 128 bits à 192 bits et même à 256 bits actuellement. On pourrait même envisager des clés de taille plus grande dans le futur.

Le gouvernement américain a même mis en place un concours afin de trouver une nouvelle méthode de chiffrement ainsi qu'un nouveau système de signature capable de résister au futur avènement des ordinateurs quantiques afin de renforcer la solidité du chiffrement AES et donc la protection des données.

La course entre la cryptologie et les avancées technologiques semble donc plus que jamais d'actualité et semble assurer un avenir certain à la cryptologie dans notre société nécessitant toujours plus de protection pour ses données.

5 Bibliographie

- Douglas Stinson." Cryptographie, theorie et pratique, 2nd edition " (6 octobre 2003)
  ISBN : 978-2-7117-4800-6
  Editeur : Vuibert

- Simon Singh." Histoire des codes secrets ". (1999)
  ISBN Poche : 978-2253150978

- Plutarque. " Vie de Lysandre ". (4 octobre 2016)
  ISBN 13 : 978-1539320203

- Laurent Poinsot." Chap. IV : La Théorie de Shannon - partie 1 : La confidentialité parfaite ". Université Paris 13 - Institut Galilée
  https ://lipn.univ-paris13.fr/ poinsot/save/INFO%203/Cours/Chap4.pdf

- " Histoire de la cryptologie ". Wikipédia, (17 février 2024)
  https ://fr.wikipedia.org/wiki/Histoire_de_la_cryptologie

- " Comprendre les grands principes de la cryptologie et du chiffrement ". Commission Nationale de l'Informatique et des Libertés, (25 octobre 2016)
  https ://www.cnil.fr/fr/comprendre-les-grands-principes-de-la-cryptologie-et-du-chiffrement

- " Cryptographie par transposition- Transpositions rectangulaires ". Bibm@th.net
  https ://www.bibmath.net/crypto/index.php ?action=affiche&quoi=ancienne/transposition

- Didier Müller. " Transpositions rectangulaires ". Apprendre-en-ligne.net, (29 juillet 2023)
  https ://www.apprendre-en-ligne.net/crypto/transpo/rectangulaire.html

- " AES Animation ". Cryptool
  https ://legacy.cryptool.org/en/cto/aes-animation

- Jean-Max Dutertre. " AES 128 bits ". Université de Saint-Étienne, (2011) https ://www.emse.fr/˜dutertre/documents/synth_AES128.pdf

- " Qu'est-ce que le chiffrement de César ? ". Futura-Sciences, (17 novembre 2017) https ://www.futura-sciences.com/sciences/questions-reponses/mathematiques-quest-ce-chiffrement-cesar-8032/

- Didier Müller. " Chiffre de César". Apprendre-en-ligne.net, (13 juin 2023) https ://www.apprendre-en-ligne.net/crypto/cesar/index.html

- Jean-Marc Robert. " Chiffrement par bloc". (1 février 2013)

- " De la taille des clés dans un système cryptographique ". Bibm@th.net https ://www.bibmath.net/crypto/index.php=action=affiche&quoi=chasseur/key

- Kenny Paterson. " Key Reuse : Theory and Practice ". Royal Holloway, University of London https ://crypto.stanford.edu/RealWorldCrypto/slides/kenny.pdf

- " NIST Announces First Four Quantum-Resistant Cryptographic Algorithms ". NIST https ://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms