LE CHIFFREMENT PAR COURBES ELLIPTIQUES

1. Qu'est ce qu'une courbe elliptique ?

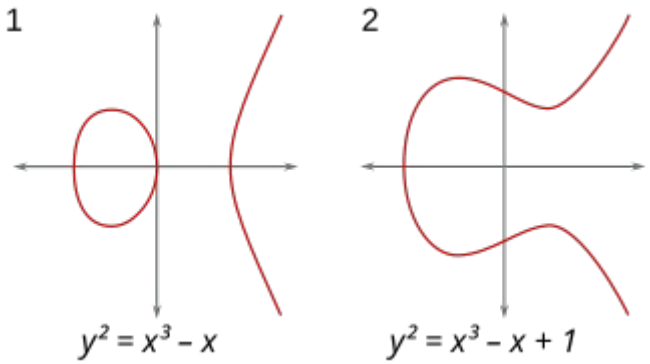
Les courbes elliptiques sont des fonctions de type : $y^2 = f(x)$ où f(x) est un polynôme en x^3

En cryptographie, on utilise des courbes elliptiques de la forme :

$$y^2 = x^3 + ax + b$$

Remarquons qu'il n'y a pas de x^2 . Les coefficients a et b sont des nombres réels. Selon le choix de ces coefficients, les graphes peuvent avoir deux formes possibles.

Exemples :



On voit que dans le graphe 1, l'équation a trois racines réelles distinctes (-1, 0, et +1) et que dans le graphe 2, elle n'a qu'une seule racine réelle.

2. Utilisation des courbes elliptiques en cryptographie :

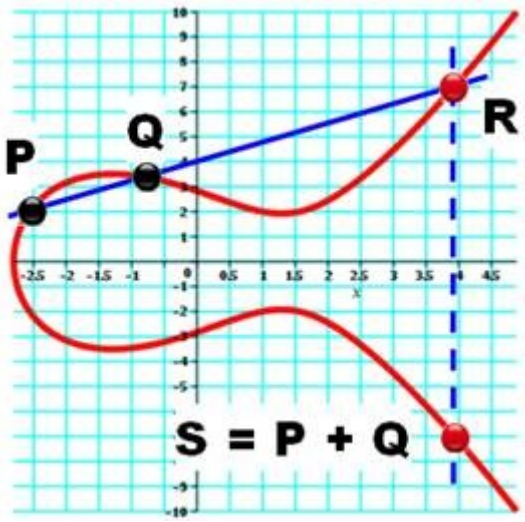
Cette fiche a pour but de présenter de façon simple le chiffrement par courbes elliptiques. Nous resterons dans un contexte général et ne tiendrons pas compte des nombreux cas particuliers présentés par ces courbes. Nous essaierons de rester au niveau de mathématiques du lycée.

2.1 Addition de deux points sur une courbe elliptique :

Pour commencer, on définit l'addition de deux points de la courbe. L'addition s'effectue de la manière suivante : on choisit deux points P et Q sur une courbe elliptique. Ces deux points sont définis par leurs coordonnées $P(x_1, y_1)$ et $Q(x_2,y_2)$.

Deux points P et Q sur une courbe elliptique forment une droite qui coupe toujours la courbe en un troisième point. Soit R ce point. Le symétrique du point R par rapport à l'axe des x, appelons-le S, **est défini comme la somme des points P et Q**. Les points sont définis par leurs coordonnées.

Cette définition peut paraître un peu étrange, regardons un graphe :



On a donc $S(x_3, y_3) = P(x_1, y_1) + Q(x_2,y_2)$

Pour ne pas alourdir cette présentation, nous verrons en détail l'aspect mathématique de cette addition dans l'annexe 1. Pour l'instant prenons cette addition telle qu'elle est.

Sur une courbe elliptique, de même que l'on peut additionner deux points, on peut également multiplier un point par un nombre entier.

2.2 Multiplication d'un point par un entier :

Puisque l'on a défini l'addition, continuons et définissons la multiplication d'un point par un nombre entier d.

Si P est un point sur la courbe, on peut calculer 2P, 3P, 4P etc. en utilisant ce qu'on appelle la *multiplication scalaire*. Concrètement **on répète le principe de l'addition** : P + P = 2P, puis 2P + P = 3P, puis 3P + P = 4P et ainsi de suite…

Le résultat de cette multiplication est un point S. Là encore nous verrons tout cela en détail en annexe avec des exemples chiffrés et des graphes. Le but ici est de comprendre les principes.

En résumé, nous avons donc un point **P** sur une courbe elliptique, un nombre entier **d** et un point **S** sur cette courbe tel que :

$$\mathbf{S = d \times P}$$

Et nous en arrivons enfin au principe de chiffrement :

3. Principe de chiffrement :

Le chiffrement repose sur le fait que si l'on connaît un point P sur une courbe elliptique et un autre point S tel que S = dP, il est extrêmement difficile de trouver d à partir des coordonnées de P et de S.

Ce chiffrement, qui est **asymétrique**, s'effectue donc comme suit :

- Alice et Bob se mettent d'accord, publiquement, sur une courbe elliptique ainsi que sur un point P (x_1, y_1) situé sur la courbe, ce point P étant également connu publiquement.

- Alice choisit secrètement un nombre entier d_A et envoie à Bob les coordonnées du point d_AP.

- Bob choisit secrètement un nombre entier d_B et envoie à Alice les coordonnées du point d_BP.

- Alice peut calculer $d_A(d_BP)$ et Bob peut calculer $d_B(d_AP)$, **c'est à dire $(d_A*d_B)P$ qui est leur clé commune.**

Si Ève, qui espionne Alice et Bob, a intercepté les échanges, elle connaît l'équation de la courbe, le point P, d_AP et d_BP. Mais elle ne peut pas calculer d_A et d_B, et donc le produit $(d_A*d_B)P$ qui est la clé commune.

Le calcul de d_A en connaissant P et le produit d_AP s'appelle résoudre le *logarithme discret* sur une courbe elliptique. Si les nombres d_A et d_B sont suffisamment grands, on ne peut pas les calculer avec les performances actuelles des ordinateurs.

À l'image du chiffrement RSA, le chiffrement par courbes elliptiques, **asymétrique** également, repose sur un problème arithmétique non calculable.

ANNEXE 1

NOMBRES RÉELS ET CORPS FINIS

1. Courbes elliptiques sur les nombres réels (définition mathématique générale)

Les courbes elliptiques ont d'abord été étudiées dans le cadre des mathématiques pures, sur des corps tels que les **nombres réels R**. Ce sont des objets géométriques qui possèdent une structure algébrique particulière, souvent définis par une équation de la forme :

$$y^2 = x^3 + ax + b$$

Dans ce cadre, les courbes elliptiques sont représentées comme des objets continus, avec une courbe lisse qui peut être visualisée graphiquement. Travailler sur les nombres réels ou complexes permet d'étudier les propriétés géométriques et algébriques de ces courbes (points d'inflexion, tangentes, symétries, etc.).

Cependant, en matière de cryptographie, **travailler avec des nombres réels n'est pas sécurisé ni pratique** pour plusieurs raisons :

- Les nombres réels impliquent une précision infinie, ce qui est impossible à gérer en informatique.
- Les algorithmes de chiffrement nécessitent des opérations discrètes pour éviter les attaques basées sur des approximations ou des erreurs de calcul.

Ainsi, bien que la définition mathématique des courbes elliptiques soit souvent donnée dans le contexte des nombres réels, **ce n'est pas un cadre approprié pour les applications cryptographiques**.

2. Courbes elliptiques sur un corps fini $\mathbf{F_p}$

En cryptographie, pour rendre les courbes elliptiques utilisables et sécurisées, on les utilise sur un **corps fini,** souvent noté $\mathbf{F_p}$ (F pour Field = champ fini en anglais) et où p est un nombre premier. Travailler sur un corps fini signifie que toutes les opérations (addition, multiplication, etc.) se font avec des entiers **modulo p**.

Par exemple, l'équation de la courbe elliptique devient :

$$y^2 \equiv x^3 + ax + b \pmod p$$

Ce passage des nombres réels à un corps fini présente plusieurs avantages :

1. **Sécurité** : Les courbes elliptiques sur les corps finis offrent une grande difficulté à résoudre certains problèmes mathématiques sous-jacents (comme le problème du logarithme discret elliptique), ce qui les rend particulièrement adaptées pour la cryptographie. Les solutions aux problèmes cryptographiques dans un corps fini sont beaucoup plus difficiles à calculer que dans $\mathbb{R}$ surtout avec des nombres très grands.
2. **Calculs discrets** : Dans un corps fini, tous les calculs sont discrets et bornés. Cela permet d'éviter les problèmes d'approximation liés aux nombres réels et d'utiliser des algorithmes efficaces pour les opérations de chiffrement et de déchiffrement.
3. **Efficacité** : Les opérations sur les corps finis peuvent être implémentées efficacement sur des ordinateurs, car elles se réduisent à des opérations arithmétiques simples (addition, multiplication) modulo un nombre premier p. Cela est particulièrement important dans les systèmes cryptographiques modernes, où la vitesse et la précision des calculs sont critiques.

3. Exemple concret d'utilisation d'un corps fini

Si on utilise la courbe elliptique définie par l'équation :

$$y^2 = x^3 + ax + b$$

et que l'on choisit de travailler sur le corps fini $\mathbb{F}_p$, cela signifie que l'on va "restreindre" les valeurs possibles de x et y aux nombres entiers compris entre 0 et $p-1$, en appliquant des opérations modulo p. Par exemple, si $p=17$, alors x et y seront des entiers entre 0 et 16, et toutes les additions, multiplications et divisions seront effectuées modulo 17. Cet exemple sera traité en détail dans l'annexe 2.

Pourquoi cette distinction est-elle importante ?

- **Contexte mathématique général** : Les courbes elliptiques définies sur les nombres réels servent surtout à étudier les propriétés générales des courbes et à comprendre leur structure géométrique. C'est un contexte important pour la recherche théorique en mathématiques.
- **Contexte cryptographique** : En cryptographie, on travaille sur des **corps finis** pour garantir que les opérations soient discrètes, efficaces et sécurisées. Les courbes sur les corps finis sont celles qui résistent aux attaques cryptographiques.

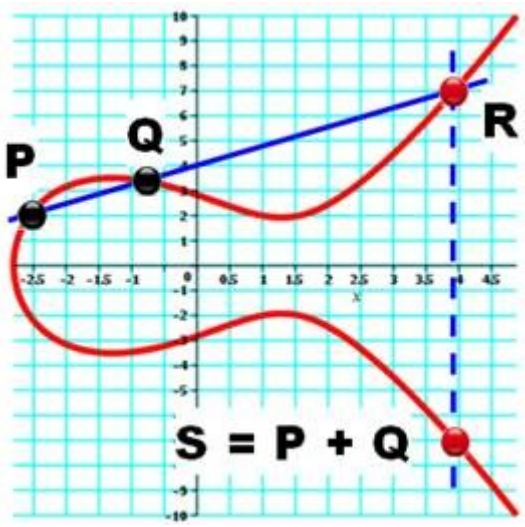
ANNEXE 2

LES OPÉRATIONS D'ADDITION ET DE MULTIPLICATION DE POINTS SUR UNE COURBE ELLIPTIQUE

Nous allons d'abord travailler sur les nombres réels. Nous avons vu que sur une courbe elliptique, on peut additionner deux points ou multiplier un point par un nombre entier. Commençons par l'addition de deux points :

1. Addition de deux points :

Voici un graphe pour visualiser :



Pour additionner deux points P + Q, on considère le point R qui est le point d'intersection de la droite passant par P et Q avec la courbe elliptique. Puis on prend le symétrique de R par rapport à l'axe des x, ce qui détermine le point S. On démontre que S = P +Q.

Nous allons expliquer dans ce paragraphe **l'aspect mathématique théorique** (voir annexe 1)**.** Puis, dans le paragraphe 2, qui présente la multiplication d'un point par un nombre, nous prendrons un exemple chiffré. Nous avons donc comme données :

- une courbe elliptique $y^2 = x^3 + ax + b$

- une droite $y = kx + m$ qui coupe la courbe en 3 points, P, Q et R

Les nombres a, b, k et m sont connus. On a défini les coordonnées de P et Q, et nous cherchons celles du point R et du point S.

On pose P = (x_1, y_1) , Q = (x_2, y_2) , R = (x_3, y_3) et S = (x_4, y_4)

et la pente de la droite est k = $\frac{y2-y1}{x2-x1}$

Le point R (x_3, y_3) étant à une intersection de la courbe et de la droite, on commence par déterminer la valeur de x_3 à partir des coordonnées connues des points P et Q qui se trouvent sur la courbe et sur la droite. Après un calcul assez long, on obtient :

$$\mathbf{x_3 = k^2 - x_1 - x_2}$$

Nous avons donc déterminé la valeur de x_3 à partir des données connues k, x_1 et x_2

Puis on calcule la valeur de y_3 :

La pente k de la droite peut s'écrire également k = $\frac{y3-y1}{x3-x1}$

On a donc $k (x_3 - x_1) = y_3 - y_1$

Soit $\mathbf{y_3 = k (x_3 - x_1) + y_1}$

Pour terminer, nous obtenons donc les coordonnées du point S = P + Q, sachant que par définition le point S (x_4, y_4) est le symétrique de R par rapport à l'axe des abscisses.

Nous avons donc $\mathbf{x_4 = x_3}$ et $\mathbf{y_4 = - y_3 = k (x_1 - x_3) - y_1}$

En conclusion, on peut écrire le résultat de l'addition :

$$\mathbf{P (x_1,y_1) + Q (x_2,y_2) = S(x_4,y_4)}$$

Nous verrons dans le paragraphe suivant un exemple concret avec des chiffres.

2. Multiplication d'un point P par un nombre entier

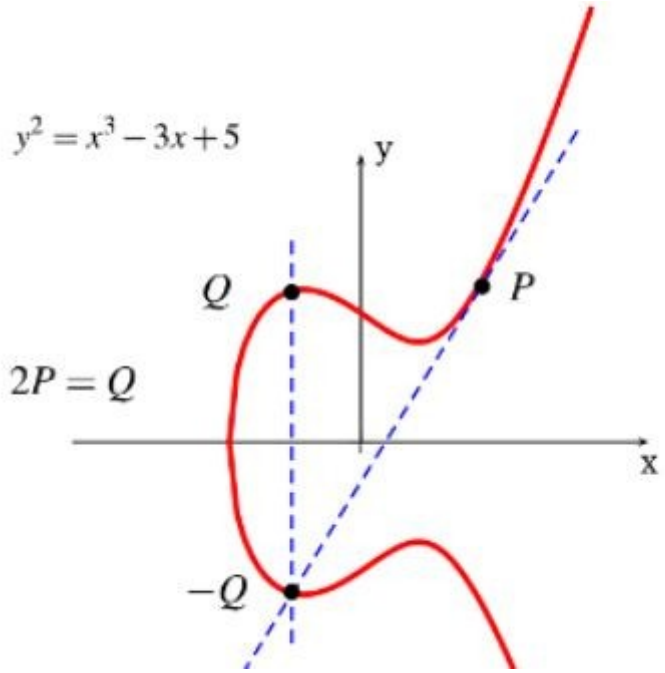
Nous allons maintenant voir **un exemple sur un corps fini** (voir annexe 1) avec des petits nombres.

Nous allons donc multiplier le point P par un nombre. Cette multiplication s'effectue selon le principe de l'addition : on additionne P + P = 2.P, puis 2P + P = 3.P, puis 3 P + P = 4.P etc.

Dans l'addition, on additionne deux points P et Q. Mais ici, on n'a qu'un seul point P. Comment faire ?

Pour ajouter P à lui-même , on trace la tangente à la courbe en P, et on considère que cette tangente touche deux fois la courbe au point P. On calcule les coordonnées de la troisième intersection, puis celles du point symétrique, comme dans l'addition. On obtient ainsi P + P = 2P.

Graphe :



Sur ce graphe, on a tracé la tangente en P à la courbe. Elle recoupe la courbe en un point – Q. Le symétrique de ce point – Q par rapport à l'axe des abscisses est le point Q. On a donc Q = 2.P.

L'équation de la courbe dans l'exemple chiffré n'est pas celle du graphe. Le graphe est simplement là pour visualiser les calculs.

Exemple chiffré en travaillant sur un corps fini:

Les données sont :

- une courbe elliptique $\mathbf{y^2 = x^3 + 2x + 2 \bmod 17}$, donc **a = 2 et b = 2**
- un point générateur **P = (5,1)**
- on effectue les calculs modulo p, avec p nombre premier, et on choisit **p = 17**

2.1 Doublement du point P :

On commence par doubler P (5,1). Comme dans le cas de l'addition, il faut déterminer le pente de la droite. La pente k de la tangente en P est donnée par :

$$k = \frac{3x2+a}{2y1} \quad \text{modulo p}$$

Il faut lire $(3 x_1^2 + a) / (2y_1)$, où x_1 et y_1 sont les coordonnées du point P (5,1)

On effectue le calcul : $k = (3*5^2 + 2) / (2*1) \bmod 17 = \frac{77}{2} \bmod 17$

Ici il faut calculer l'inverse modulaire de 2 mod 17, et on trouve 9.

Si l'on est peu familier avec la notion d'inverse modulaire, voir la fiche sur le chiffrement RSA.
Par ailleurs, le site dcode.fr permet de calculer l'inverse modulaire rapidement sur des grands nombres : https://www.dcode.fr/inverse-modulaire

On a donc k = 77*9 mod 17 = 693 mod 17 = 13

La pente de la tangente est **k = 13 mod 17**

On calcule ensuite les coordonnées du nouveau point avec les mêmes formules que l'addition :

$x_2 = k^2 - 2x_1 \bmod 17$, soit

$x_2 = 13^2 - 2*5 \bmod 17 = (169 - 10) \bmod 17 = 159 \bmod 17 = 6 \bmod 17$

x_2 = 6 mod 17

Puis on calcule y_2 toujours de la même façon que dans l'addition :

$y_2 = k(x_1 - x_2) - y_1 \pmod{17}$, soit

$y_2 = 13*(5-6) - 1 \pmod{17}$

$= -14 \pmod{17}$

y_2 = 3 mod p soit en définitive **Q = 2.P = (6, 3)**

2. 2 Suite de la multiplication : calcul de 3.P

On veut continuer le processus de multiplication, c'est-à-dire calculer 3P. Pour ce faire, on applique les règles de l'addition de deux points, c'est à dire dire que P + 2P = 3P. On va donc définir un point S tel que S = P + Q = P + 2P = 3P

On effectue donc l'addition des 2 points P (5,1) et Q (6,3)

La pente de la droite P, Q est donnée par

$$k = \frac{1-3}{5-6} \bmod 17, \text{ soit}$$

Or (1-3) = -2 et (5 -6) = -1 , soit $\frac{-2}{-1} = 2 \bmod 17$

$k = 2 \bmod 17$

À partir de k, on calcule $x_3 = k^2 - x_1 - x_2 \bmod 17$

x_3 = (2^2 -5 -6) mod 17 = (- 7) mod 17 = **10 mod 17**

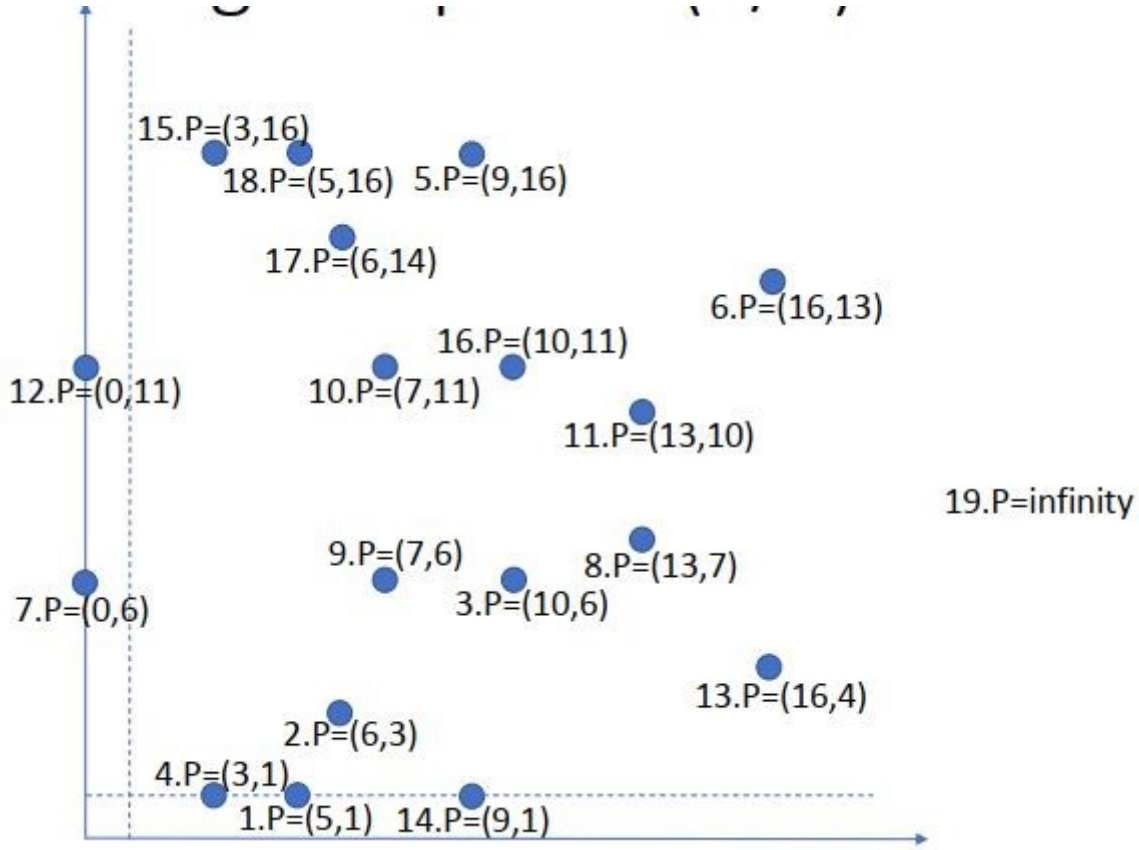
Puis on calcule $y_3 = k(x_1 - x_3) - y_1$ (mod 17)

y_3 = 2(5-10) – 1 mod 10 soit (-11) mod 17 = **6 mod 17**

Au final on a S = 3.P = (10,6)

On a vu dans le corps principal de la fiche que d'une façon générale, on avait S = dP. Dans la réalité, le chiffre d est très grand et la difficulté pour casser ce chiffre réside donc dans la difficulté de calculer d en connaissant P et S. De plus, il y a deux nombres d, celui d'Alice et celui de Bob.

Le tableau suivant permet de visualiser la position des points 4.P, 5.P, 6.P , etc. jusqu'à 18.P à partir de P (5,1), 2.P (6,3) et 3.P (10,6) :



Ce tableau est tiré du cours du professeur Alexandre Guitton, qui est téléchargeable librement sur Internet :

https://perso.isima.fr/~alguitto/index.html (voir support de cours, courbes elliptiques V2).

Le point 19.P = l'infini correspond à une droite verticale, parallèle à l'axe des y et qui coupe donc la courbe en 2 points. Le 3ème point est renvoyé à l'infini. Ce cas particulier n' a pas été traité pour ne pas alourdir la fiche.

On pourra également si on le souhaite effectuer des calculs sur le site :

https://andrea.corbellini.name/ecc/interactive/modk-add.html

Ce site permet d'effectuer des additions et des multiplications de points sur l'ensemble R des nombres réels ou sur un champ fini F_p en choisissant la courbe elliptique, le nombre premier p et les coordonnées du point P. Bien qu'en anglais, son utilisation est très simple et les graphiques qui s'affichent instantanément permettent de bien comprendre et de voir la différence entre un calcul avec les nombres réels et un calcul avec un corps fini.

3. Conclusion :

Dans les chiffrements par courbes elliptiques, il faut bien voir que dans la réalité, la clé privée d_A d'Alice et la clé privée d_B de Bob sont des nombres entiers qui ont généralement une taille d'environ 256 bits.

Comme on l'a vu au début de la fiche et dans cette annexe, le principe de chiffrement repose sur le fait qu'après l'échange de leurs clés d_A et d_B, Alice et Bob ont pu calculer une clé commune :

$$K = d_A * d_B * P$$

K est donc un point sur la courbe elliptique représenté par ses coordonnées x_K et y_K. Chacune de ces coordonnées est un entier de 256 bits. Le total des deux coordonnées représente donc 512 bits.

En pratique, souvent, seules les coordonnées x_K ou des dérivés de x_K sont utilisées afin de produire une clé symétrique plus petite. Par exemple, Alice et Bob peuvent utiliser une clé de chiffrement AES de 256 bits en utilisant x_K comme clé initiale pour dériver des clés, puisque AES utilise des clés dérivées à partir de la clé initiale (voir fiche sur le chiffrement AES).

Pour se faire une idée de la taille de ces clés, rappelons qu'un nombre de 256 bits en binaire représente un nombre entier d'environ 77 chiffres en décimal et de 64 chiffres en hexadécimal.

*****
***
*