Master – Le chiffrement

Le chiffrement de César

César, chiffrait ses messages pour ses communications en utilisant le principe du décalage des lettres. Cette méthode assez basique est l'une des premières technologies de chiffrement.

FONCTIONNEMENT

Pour chiffrer un message, A devient D, B devient E, C devient F,…
Pour déchiffrer , il suffit de décaler les lettres dans l'autre sens, D se déchiffre en A, E en B,…

Exemple
On prend un bloc de 4 et un décalage de 3
VENI VIDI VICI devient **YHQL YLGL YLFL**

On peut changer la donne en prenant un bloc de 2 et un décalage de 3
YH QL YL GL YL FL

> Cet algorithme est facile à craquer, il y a 26 clés possibles, on peut donc tester manuellement toutes les clés .

Le ROT 13

Ce système ressemble au chiffre de César, mais utilise une rotation avec un décalage de 13 soit **C13(x)= x + 13**

bonjour devient **obawbhe**

> Le problème avec cet exemple est que si l'on applique 2 fois le chiffrement, on obtient le déchiffrement.

Le chiffrement de Vigenère

Au lieu de décaler les lettres de manière égale, on associe à chaque lettre une autre lettre (sans ordre fixe ou règle générale)

Exemple

bonjour donne avec la clé (CLE) **dzrlzyt**

Prendre les premières lettres du message b (=2) et de la clé C (=3) et on les ajoute 2+3=5. On note la valeur et on continue avec la lettre suivante du message o (=15) et la lettre suivante de la clé L (=11) : 15+11=26 etc. Une fois à la fin de la clé, on recommence au début.

Le code de Vigenere utilise donc des clés plus longues que César et ainsi chaque lettre peut alors être codée de plusieurs façons.

> Ce type de chiffrement est assez simple à craquer, car les substitutions sont toujours les mêmes. En effet, le Z sera toujours équivalent à la même lettre dans un même message. Cela veut dire que pour décrypter le message, au départ on à 26 possibilités pour une lettre mais une fois la correspondance trouvée, on n'a plus que 25 possibilités pour la deuxième et 24 pour la troisième et ainsi de suite.

La machine Enigma

Pour mettre en place cette machine qui a servi à chiffrer les messages des allemands pendant la guerre, l'astuce a consisté à prendre chaque lettre et la remplacer par une autre, puis opérer une substitution qui change d'une lettre à l'autre, un peu comme dans le chiffre de Vigenère.

Elle est basée sur 3 grands principes :

Le tableau de connexions

il permet d'échanger des paires de l'alphabet, deux à deux, au moyen de fiches. Comme il y a 6 fiches on peut donc échanger 12 lettres.

Les rotors

Un rotor est une permutation où chaque lettre en entrée correspond une autre lettre en sortie. Les rotors sont cylindriques et à chaque saisie de lettre, le premier rotor tourne d'un cran et la permutation est changée. Par exemple, le rotor transforme E en C, puis il se décale d'un cran et devient D vers B, ce qui veut dire que lorsque la prochaine lettre est tapée, le rotor transforme cette fois E en B.

Le réflecteur

Au bout des 3 rotors on revient en arrière. Lorsqu'on permute une dernière fois les lettres (2/2), et on les fait repasser par les rotors et le tableau.

Techniques classiques du chiffrement

La confusion

Cette méthode sert à cacher la relation entre le clair et le chiffré.
Elle s'obtient par des opérations (non linéaires) de substitution de sous blocs.

La diffusion

Elle sert à cacher la redondance dans le message , elle évite de repérer la fréquence des caractères. En d'autres termes, les statistiques de la sortie doivent donner le moins possible d'informations sur l'entrée.

Le XOR

	0	1
0	0	1
1	1	0

Chiffrement à clé symétrique

Le principe du chiffrement par flot est de chiffrer une suite de caractères (ou octets) un à la fois, à l'aide d'une transformation qui varie au fur et à mesure du texte.

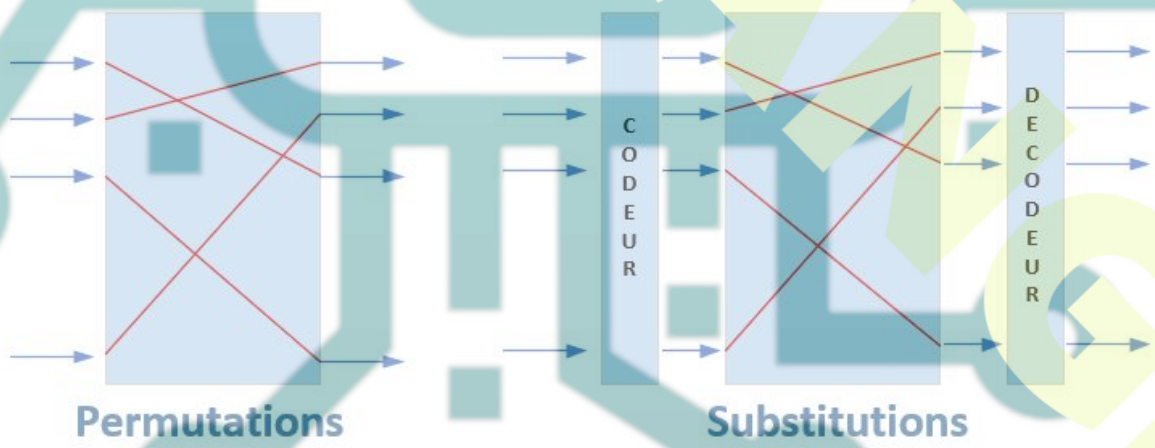
Schématiquement, le message est écrit sous forme d'une succession de bits m1…m*n*, où le flot de clés est aussi une succession de bits k1…k*n* et où l'opération de chiffrement est le OU exclusif, **ci=mi⊕ki.**

Dans un **système de chiffrement par blocs**, chaque texte clair est découpé en blocs de même longueur puis chiffré bloc par bloc (typiquement des blocs de 64 ou 128 bits)

Schématiquement, un chiffrement itératif par blocs est le suivant :
pour chaque bloc, on itère r fois la fonction interne F

A chacun des r tours, la fonction F est paramétrée par une clef **Ki(1≤i≤r)**, et la fonction du tour i peut être notée **FKi**.

Le principe de la fonction F est d'utiliser une combinaison de substitutions et de permutations.



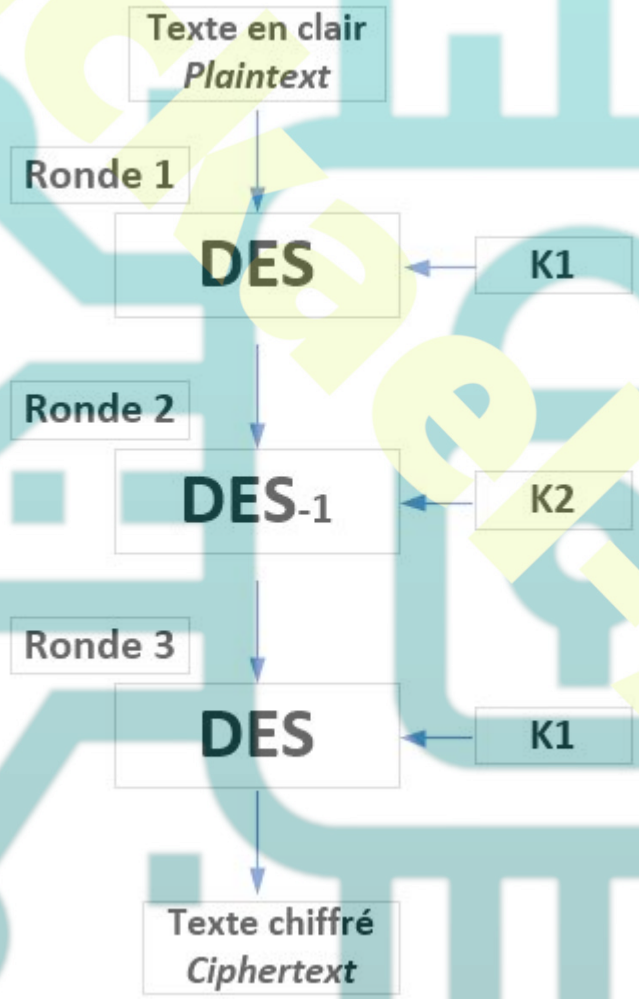
Présentation de DES et triple DES

DES prend en entrée un texte clair de 64 bits, une clé de 64 bits également (dont 8 bits de parité, donc seuls 56 sont utilisables) et retourne un texte chiffré de 64 bits. Le

déchiffrement se fait avec la même clé, mais en renversant l'ordre des blocs.

Le fonctionnement du 3 DES

Le premier DES utilise une clé de 56 bits, il y a donc 2^{56} cas possibles. C'est pareil pour le deuxième DES, sauf que qu'il faut le multiplier au premier cas, soit un total de 2^{112} possibilités.



Présentation de RC4

RC4 est chiffrement octet par octet (flot) qui utilise un générateur de bits pseudo aléatoires dont le résultat est combiné avec le texte en clair via une opération XOR.

Deux étapes sont nécessaires pour la mise en œuvre du chiffrement.
La première consiste en la création de la clé (algorithme KSA) qui permet de générer 2

tableaux de 256 octets avec une boucle qui donne en sortie la fonction (**S**).
S: {0,…,N-1}→{0,…,N-1} N valant généralement 256

La boucle se construit comme *suit :*
j := (j + S[i] + K[i mod L]) mod 256
intervertir *(S[i], S[j])*

La deuxième consiste à générer une suite aléatoire d'octets (algorithme PRGA) qui qui sont ajoutés aux caractères du message.

L'octet aléatoire est de la forme *S[(S[i] + S[j]) mod 256]*
L'octet chiffré est de la forme *ci=mi⊕ S[ji] (soit octet de chiffrement XOR octet du message)*

NB. La clé secrète à un état qui évolue dans le temps et les valeurs i et j doivent être initialisées à 0 en début de boucle.

Présentation de AES

L'AES (Advanced Encryption Standard) est un standard de chiffrement symétrique destiné à remplacer le DES (Data Encryption Standard).

Le développement de l'AES a été initié par le NIST (National Institue of Standards and Technology).

Spécifications

L'AES est un standard utilisant un algorithme de chiffrement symétrique par blocs.
Il supporte différentes combinaisons (longueur de clé/longueur de bloc) de 128-128, 192-128 et 256-128 bits.

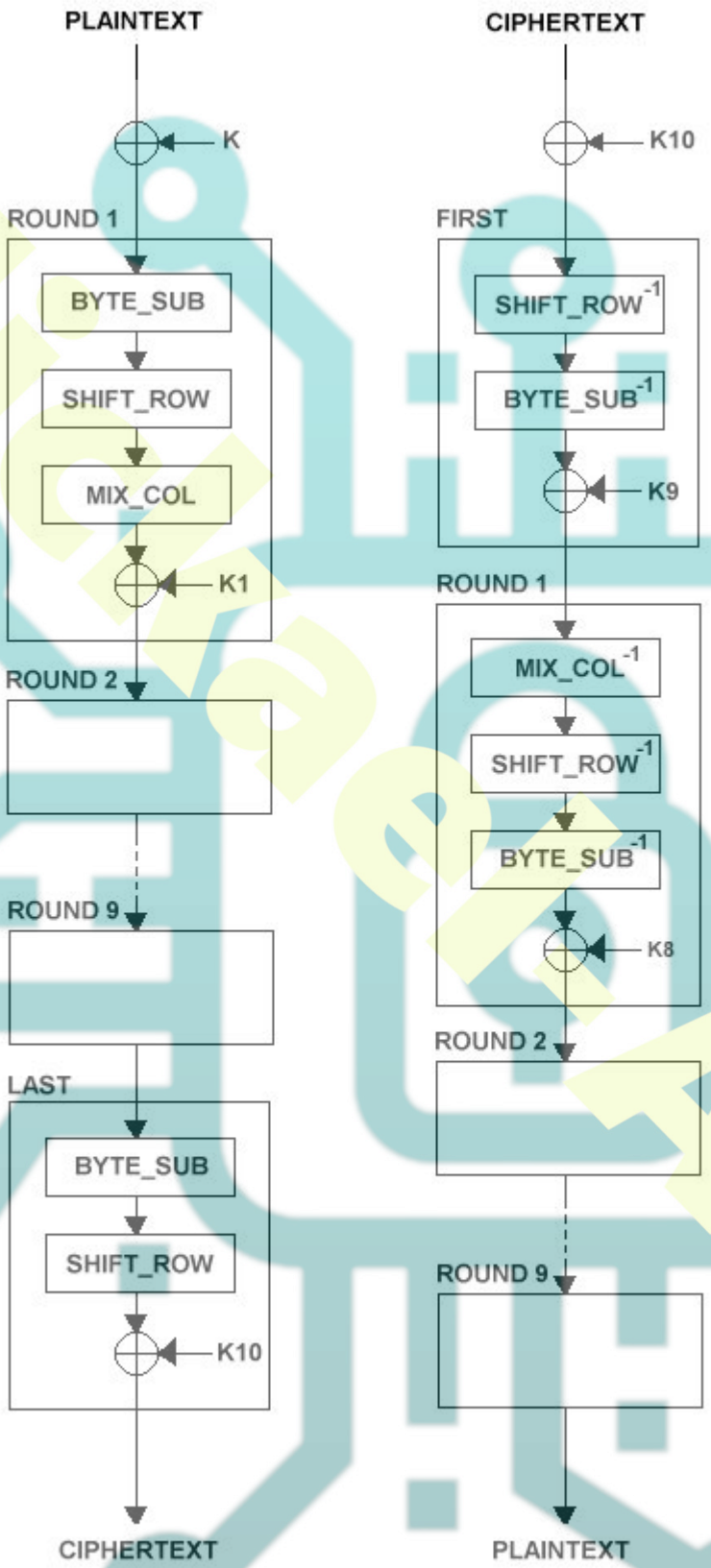
Cela donne une combinaison de clés d'environ 1000 fois supérieur au DES, soit pour un décryptage de DES d'une seconde, il faudrait plus de mille milliards d'années pour décrypter AES, mais tout ceci n'est que théorique.

Actuellement, AES est utilisé dans le 4G et pour le WIFI (WPA2) notamment.

Le fonctionnement

Le déchiffrement consiste à appliquer les opérations inverses, dans l'ordre inverse et avec des sous-clés également dans l'ordre inverse.

- BYTE_SUB (Byte Substitution) – fonction non-linéaire opérant indépendamment sur chaque bloc à partir d'une table de substitution.
- SHIFT_ROW – fonction opérant des décalages en prenant à l'entrée en 4 blocs de 4 octets et opère des décalages vers la gauche de 0, 1, 2 et 3 octets pour les morceaux 1, 2, 3 et 4 respectivement.
- MIX_COL – fonction qui transforme chaque octet d'entrée en une combinaison linéaire d'octets d'entrée.
- L'opération de OU exclusif (XOR) est appliquée.
- Kn est une sous-clé calculée à partir de la clé principale K.



Exemple

si je chiffre bonjour avec la clé 0123456789123456789 le résultat obtenu sera

5a7eaa98eed832ca682b39cbacedb463

Présentation de RSA

Cet algorithme crée par Rivest, Shamir et Adleman est un algorithme de chiffrement asymétrique à clé publique et privée utilisé dans le cadre des échanges Internet.

Il sert notamment dans les certificats numériques pour le chiffrement des échanges de clés de session, qui elles utilisent le chiffrement symétrique, et dans le cadre de la signature numérique.

Le RSA est fondé sur la difficulté de factoriser des grands nombres et la fonction à sens unique utilisée est une fonction "puissance".

Génération des clés

Choisir p et q, deux nombres premiers distincts n=pq
Calculer $n=p*q$
Calculer l'indicatrice d'Euler de n $\varphi(n) =(p-1)\ (q-1)$
On choisit un exposant e tel que $PGCD(e,\varphi(n)\ ) =1$
On calcule l'inverse d de e module $\varphi(n)$ soit $d*e$ correspond à $1 \pmod{\varphi(n)}$

Exemple
NB. l'exemple n'est présent que pour comprendre la base du système. Le système utilise des nombres premiers de très grande valeur (plus de 100 chiffres)

p=3 et q=5
$n=p*q \Rightarrow n=2*7 \Rightarrow$ **n=15**
$\varphi(n)=8$

On choisit e=**11** ce qui implique que PGCD(3,8)=1

On applique l'algorithme d'Euclide étendu pour calculer les coefficients de Bézout correspondant au PGCD (e et $\varphi(n)$)

L'algorithme de recherche :
0×11 + 1×8 = 8
1×11 + -1×8 = 3
-2×11 + 3×8 = 2
3×11 + -4×8 = 1
d'où le résultat : 3×11 + -4×8 = 1 avec u = 3 et v = -4.

L'inverse du e modulo φ(n) est **d = 3**

Clé publique

Elle est composée de 2 nombres e=11 et n=8
On met la clé privée de coté d=3 et on oublie le reste p,q et φ(n)

Chiffrement du message

Le message est un entier m , tel que 0≤ m <n

On envoie par exemple **m=C**

Au préalable, on transforme en chiffre un texte en remplaçant chaque lettre par son rang dans l'alphabet ou par son code ASCII. Puis on découpe le message chiffré en blocs de même longueur représentant chacun un nombre plus petit que n. Le regroupement par blocs évite que la lettre A, par exemple, porte toujours la même valeur, car en français la fréquence des lettres est connue et donc facile à retrouver.

Par exemple, si le message était abracadabra en le transposant en chiffre correspondant à sa position dans l'alphabet on aurait 1 2 18 1 3 1 4 1 2 18 1 le 1 arrive très souvent. Si je regroupe par blocs inférieurs à n par exemple 3 cela donne 121 813 142 181

On récupère la clé publique : n et e avec laquelle on calcule le message chiffré.

x=m^e mod(n) soit 2^11 mod(8) soit **x=177147**

Déchiffrement du message

On reçoit le message x et on le déchiffre avec d

m correspond à **x^d mod(n)** soit **3** donc **C** qui est bien le message de départ.