

I. Modélisation1) Chiffrement "one time pad" (ou chiffrement de Vernam)

$$\begin{array}{c|cc} \oplus & 0 & 1 \\ \hline 0 & 0 & 1 \\ 0 & 0 & 1 \\ \hline 1 & 1 & 0 \end{array}$$

$$C = M \oplus K$$

$$M = 010110011 \quad K = 1101110100 \quad C = 100000111$$

Propriété: Si l'attaquant connaît C et que tous messages M ont la même probabilité alors il n'apprend rien sur K

Le one time pad est sûr et est un chiffrement parfait et inconditionnellement sûr (car le chiffre ne donnera aucune info sur M) impossible

↳ Shannon 1948

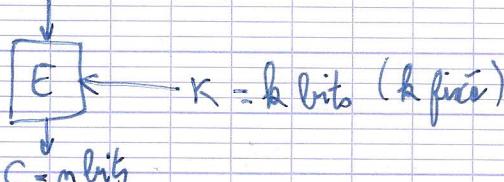
$$\begin{array}{ccc} \overbrace{M_1}^{\text{M}_1} & \overbrace{M_2}^{\text{M}_2} & \\ \overbrace{K_1}^{\text{K}_1} & \overbrace{K_2}^{\text{K}_2} & \\ \overbrace{C_1 = M_1 \oplus K_1} & & \\ \overbrace{C_2 = M_2 \oplus K_2} & \xrightarrow{\quad \quad \quad 0 \quad \quad \quad} & \\ \overbrace{C_1 \oplus C_2 = M_1 \oplus M_2 \oplus K_1 \oplus K_2} & \xrightarrow{\quad \quad \quad 0 \quad \quad \quad} & = M_1 \oplus M_2 \end{array}$$

Théorème de Shannon (1948):

Un système de chiffrement ne peut être parfait que si $|K| \geq |M|$

2) Algorithmes de chiffrement par blocs.

m bits (n fixé)



$$E: \{0,1\}^m \times \{0,1\}^k \rightarrow \{0,1\}^m$$

$$(M, K) \mapsto E_K(m)$$

Propriétés dont on a besoin:

- Pour tout M et pour tout K , $E_K(M)$ soit calculable, "en temps raisonnable".
- Pour tout K , E_K doit être bijective:
 - $E_K(M) = E_K(M') \Rightarrow M = M'$ (injective)
 - $\forall C \ L \in E_K(M)$ (surjective)
- $D_K = (E_K)^{-1}$ doit être raisonnable à calculer.

Sécurité d'un algorithme de chiffrement.

Objectifs: 1) Trouver K

2) Déchiffrer C

qu'est ce que Charlie connaît?

- attaques à clair connus : charlie connaît un ou plusieurs couples (M, C) où $C = E_K(M)$
- attaques à clair choisi : charlie peut choisir un ou plusieurs messages clairs M_1, M_2, \dots et obtenir le chiffré
- attaques à chiffré choisi : charlie peut choisir un ou plusieurs chiffrés C_1, C_2, \dots et obtenir le clair.

↳ exigence maximale de sécurité: étant donné un attaquant qui peut choisir M_1, M_2, \dots et obtenir les clairs C_1, C_2, \dots et choisir C'_1, C'_2, \dots et obtenir M'_1, M'_2, \dots il doit être calculatoirement difficile déchiffrer un chiffré $C'' \notin \{C_1, C_2, \dots, C'_1, C'_2, \dots\}$

2018

$\left. \begin{array}{l} 2^{80} \text{ opérations élémentaires infaisable (dans le monde civil)} \\ 2^{128} \text{ opérations élémentaires (même pour des gouvernements)} \end{array} \right\}$

2^{65} opérations faisables en 2018

$$2^k > 2^{80} \text{ ou bien } 2^k > 2^{128}$$
$$\Leftrightarrow k > 80 \quad \Leftrightarrow k > 128$$

Loi de Moore: La puissance de calcul double tous les 18 mois.
Pour résister jusqu'en 2068 $\rightarrow 2^{128+33} = 2^{161}$

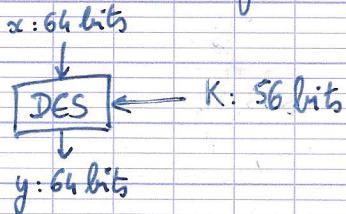
II. DES (Data Encryption Standard)

1) Historique

1975: appel à candidatures NBS (National Bureau of Standards)

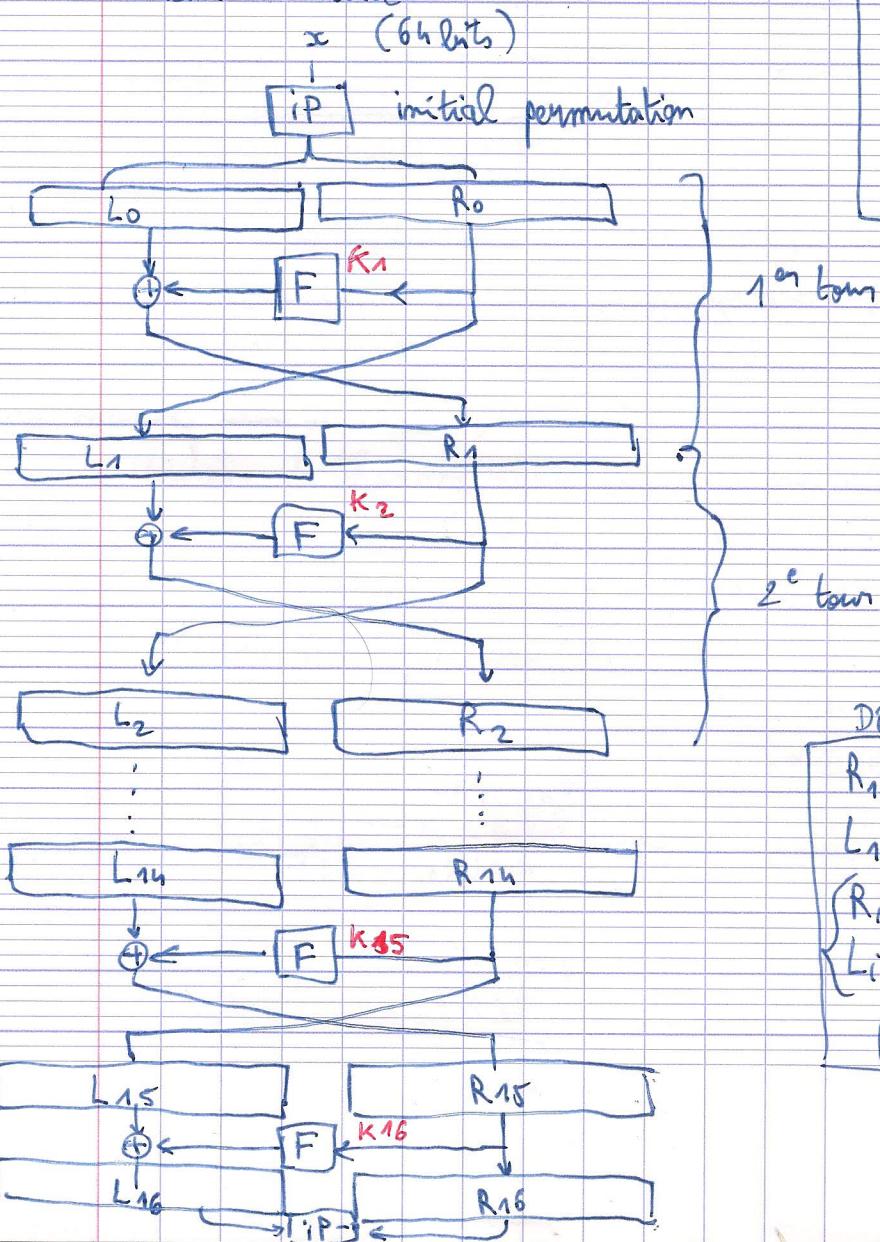
↳ IBM : Lucifer

↳ modifications → DES (1976)



2) Description

Schema de Feistel



$$\begin{cases} K \rightarrow K_1, K_2, \dots, K_{16} \\ (48) \quad (48) \quad (48) \end{cases}$$

Chiffrement

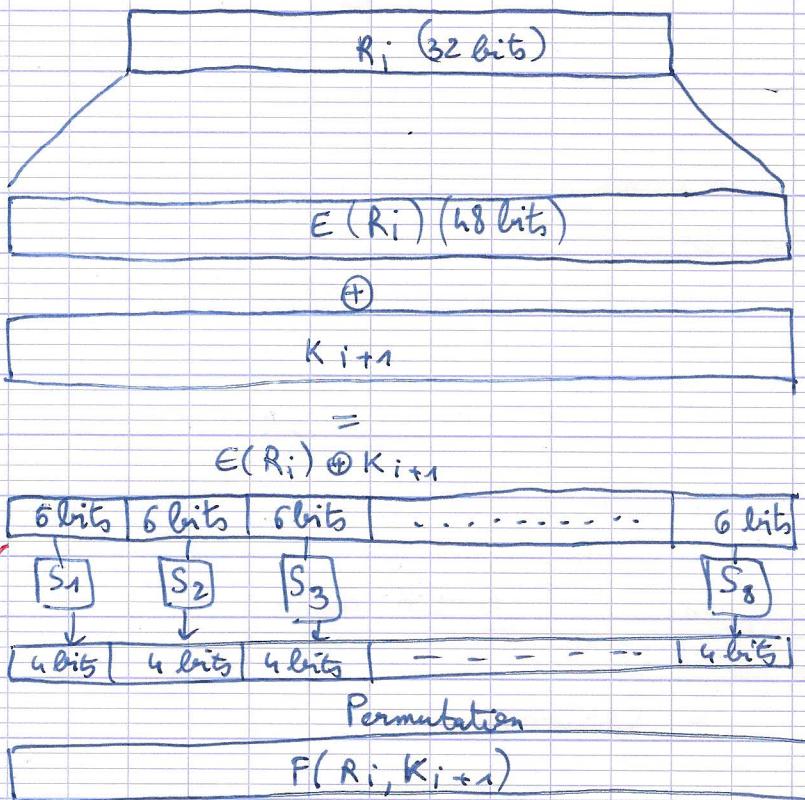
$$\begin{cases} R_{i+1} = F(R_i, K_{i+1}) \oplus L_i \\ L_{i+1} = R_i \end{cases} \text{ pour } 0 \leq i \leq 14$$

$$\begin{cases} R_{16} = R_{15} \\ L_{16} = L_{15} \oplus F(R_{15}, K_{16}) \end{cases}$$

Déchiffrement

$$\begin{cases} R_{15} = R_{16} \\ L_{15} = F(R_{16}, K_{15}) \oplus L_{16} \\ R_i = L_{i+1} \\ L_i = R_{i+1} \oplus F(L_{i+1}, K_{i+1}) \end{cases} \text{ pour } 0 \leq i \leq 14$$

Fonction F. $F(R_i, K_{i+1})$



$S_1 \rightarrow 4 \times 6$ bits

Cryptanalyse différentielle meilleure que attaque exhaustive.
 \hookrightarrow tables S_i sont optimales contre la cryptanalyse différentielle.

\triangle Remarque: À part S_1, S_2, \dots, S_8 toutes les autres fonctions utilisées par le DES sont linéaires.

$$c = a \oplus b$$

$$\downarrow$$

$$(c_1, c_2, \dots, c_{32})$$

$$a = (a_1, \dots, a_{32})$$

$$b = (b_1, \dots, b_{32})$$

$$c_1 = a_1 \oplus b_1$$

Si tout était linéaire:

$$x = (x_1, \dots, x_m)$$

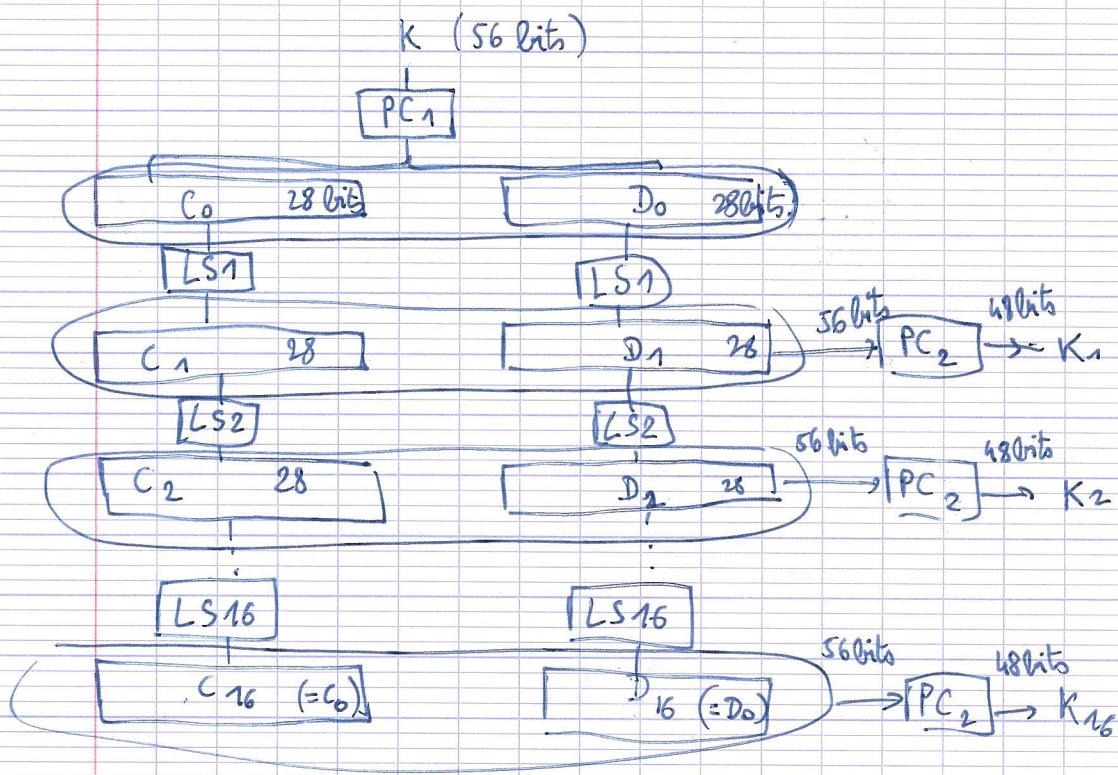
$$\downarrow$$

$$E \leftarrow K = (K_1, \dots, K_R)$$

$$\downarrow$$

$$y = (y_1, \dots, y_m)$$

$$\text{exemple: } y_1 = x_2 \oplus x_5 \oplus x_{11} \oplus x_{33} \oplus K_2 \oplus K_{13} \oplus K_{41}$$

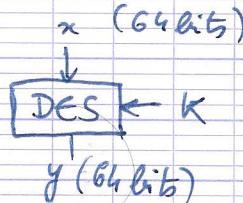
Dérivation des sous clés.

LS_i → rotation vers la gauche de :

- 2 crans si : $i \notin \{1, 2, 9, 16\}$
- 1 cran si : $i \in \{1, 2, 9, 16\}$

3) Taille de clé.

$$|K| = 56 \text{ bits}$$



Attaque par recherche exhaustive \rightarrow complexité = $O(2^{56})$

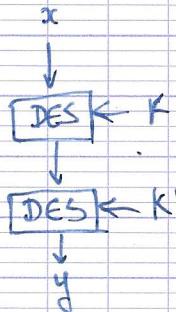
$$y = \text{DES}_K(x)$$

inconnue

DES cané \rightarrow 2000

Idee 1 pour continuer à utiliser le DES :

↳ Double DES



$$\text{Recherche exhaustive : } y = \text{DES}_{K'}(\text{DES}_K(x))$$

$$\hookrightarrow \text{complexité } O(2^{56 \times 2}) = O(2^{112})^{\text{incomm}}$$

• Méthode "meet in the middle"

$$y = \text{DES}_{K'}(\text{DES}_K(x))$$

$$\Leftrightarrow \text{DES}_{K'}^{-1}(y) = \text{DES}_K(x)$$

K	$\text{DES}_K(x)$	K'	$\text{DES}_{K'}^{-1}(y)$
0 - 0	00	0 - 0	00
0 - 1	01	0 - 01	01
1 - 0	10	1 - 0	10
1 - 1	11	1 - 01	11

Complexité :

$$-\text{méthode naïve : } O(2^{56}) \times O(2^{56}) = O(2^{112})$$

$$-\text{méthode liste triée : } O(2 \cdot 2^{56}) = O(2^{56}) \text{ Bonne méthode tri } O(n \log n)$$

$$\text{tri } \rightarrow O(2^{56} \log 2^{56}) \approx O(2^{61})$$

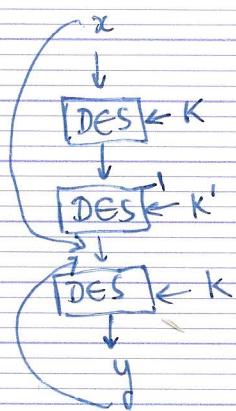
Idee 2 : Triple DES

$$y = \text{DES}_K(\text{DES}_{K'}^{-1}(\text{DES}_K(x)))$$

Remarque : Si $K = K'$ on retrouve DES_K

Sécurité contre un attaquant qui cherche à retrouver la clé :

Recherche exhaustive. $O(2^{112})$



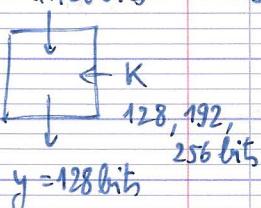
Attaque meet in the middle : $O(2^{12})$

(k, k')	$\text{DES}_{k'}^{-1}(\text{DES}_k(x))$

k	$\text{DES}_k^{-1}(y)$

III. AES (Advanced Encryption Standard)

$x = 128 \text{ bits}$



1998 : appel à candidatures lancé par le NIST (avant NBS)
 ↳ 64 candidats.

Performance : plus rapide que Triple DES

2000 : 5 candidats sélectionnés

2001 : 1 vainqueur = Rijndael (Rijmen, Daemen) → AES.

1) Calculs sur les octets.

$$01011011 \leftrightarrow 0x5B \leftrightarrow 91_{10} \leftrightarrow X^6 + X^4 + X^3 + X + 1$$

$$10111011 \leftrightarrow 0xB3 \leftrightarrow 187_{10} \leftrightarrow X^7 + X^5 + X^4 + X^3 + X + 1$$

$$\text{Addition } \oplus 11100000 \leftrightarrow 0xE0 \leftrightarrow 240_{10} \leftrightarrow X^7 + X^6 + X^5$$

$$\cancel{X^8 + X^4 + X^3 + X + 1} \text{ mod. } (X^8 + X^4 + X^3 + X + 1)$$

$$\text{Multiplicat. } X^{13} + X^{11} + X^{10} + X^9 + X^7 + X^6 + X^5 + X^4 \quad \Leftrightarrow X^5 + X^4 + X + 1$$

$$+ X^{11} + X^9 + X^8 + X^7 + X^5 + X^4$$

$$+ X^{10} + X^8 + X^7 + X^6 + X^4 + X^3$$

$$+ X^8 + X^6 + X^5 + X^4 + X^2 + X$$

$$+ X^2 + X^5 + X^4 + X^3 + X + 1$$

$$= X^{13} + X^8 + X^6 + X^5 + X^2 + 1$$

$$00110011 \leftrightarrow 33_{10} \leftrightarrow 55_{10}$$

$$X^{13} + X^8 + X^6 + X^5 + X^2 + 1 \quad | \quad X^8 + X^4 + X^3 + X + 1$$

$$- (X^{13} + X^9 + X^8 + X^6 + X^5) \quad | \quad \cancel{(X^5) + (X)}$$

$$\textcircled{1} \quad X^9 + X^2 + 1$$

$$- (X^9 + X^5 + X^4 + X^2 + X)$$

$$\textcircled{2} \quad X^5 + X^4 + X + 1$$

→ résultat

$$\begin{aligned}
 &= \{0, X, \dots, X^7 + X^6 + X^5 + X^4 + X^3 + X^2 + X + 1\} \\
 &= \{00, \dots, FF\} \\
 &= \{0, 1, \dots, 255\}
 \end{aligned}$$

$$E = \{\text{octets}\} \quad |E| = 256 = 2^8 = 16^2$$

$(E, +, \times)$ est un corps commutatif
 $\Rightarrow (E, +)$ groupe commutatif.

↳ élément neutre $0 = 00000000$

↳ $\forall x, x+0 = 0+x = x$ tout élément admet un inverse

↳ associativité: $\forall x, y, z \quad (x+y)+z = x+(y+z)$

↳ commutativité: $\forall x, y \quad x+y = y+x$

$\Rightarrow (E \setminus \{0\}, \times)$ groupe commutatif \rightarrow existence d'un inverse

$$\Rightarrow \text{Distributivité} \quad x \times (y+z) = (x \times y) + (x \times z)$$

$$(x+y) \times z = (x \times z) + (y \times z)$$

Propriété: $\forall x \in E \setminus \{0\}, \exists y \in E \setminus \{0\} / x \times y = y \times x = 1$

Preuve: $x \rightsquigarrow \text{polynôme } A(X) \quad I(X) = X^8 + X^4 + X^3 + X + 1$

$y \rightsquigarrow \text{polynôme } B(X) \quad \hookrightarrow \text{polynôme irréductible}$

$$\text{pgcd}(A(X), I(X)) = 1$$

D'après Bézout $\Rightarrow \exists U(X), V(X) \text{ tq } \underbrace{A(X)U(X)}_{d^o U} + I(X) + V(X) = 1$

avec $d^o U < d^o I$ et $d^o V < d^o A^2$

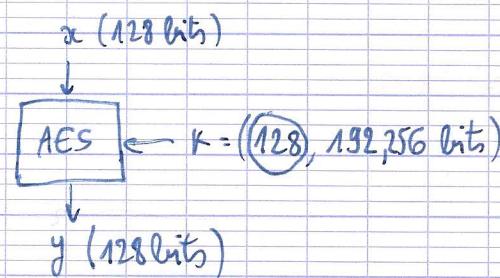
$$x = \underbrace{A(X)U(X)}_{d^o U} + I(X) + V(X)$$

On définit l'octet y qui a comme représentation polynomiale $\frac{U(X)}{d^o \leq 8}$

$$\text{et on a } x \times y = y \times x = 1 \pmod{I(X)}$$

$$A(X) \quad U(X)$$

Description AES.



CMINF15

Suite chapitre 2

17/10/18

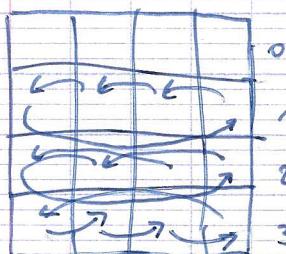
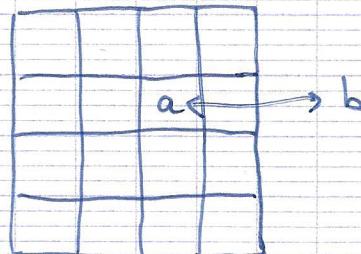
$$x = (x_1, \dots, x_{16})$$

avec \downarrow

x_1	x_5	x_9	x_{13}
x_2	x_6	x_{10}	x_{14}
x_3	x_7	x_{11}	x_{15}
x_4	x_8	x_{12}	x_{16}

 $S =$ $K \rightarrow K_0 \text{ (init)}$ $\rightarrow K_1$ K_{10} $K_i =$

sous clé

 $(0 \leq i \leq 10)$ $\approx 128 \text{ bits}$ $S = \text{Shift Rows}(S)$  $S := \text{ByteSub}(S)$ 

a	ByteSub(a)
0	0
1	1
2	2
3	3
...	...
255	255

$$1) c = \text{INV}(a) = \begin{cases} \frac{1}{a} & \text{si } a \neq 0 \\ 0 & \text{si } a = 0 \end{cases}$$

$$2) b = (b_7, b_6, b_5, \dots, b_0)$$

$$x = (c_7, c_6, c_5, \dots, c_0)$$

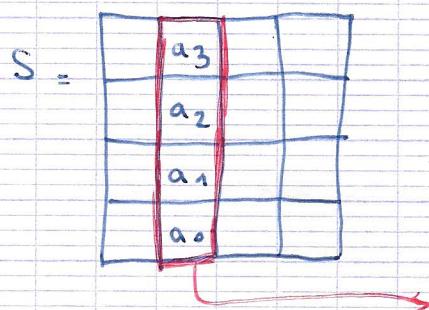
$$\begin{pmatrix} b_7 \\ b_6 \\ \vdots \\ b_0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} c_7 \\ c_6 \\ \vdots \\ c_0 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

matrice circulaire

constante

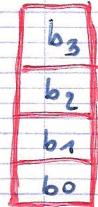
8x8 constante

$S = \text{MixColumn}(S)$



constante hex

$$\begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \times \begin{pmatrix} a_3 \\ a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} b_3 \\ b_2 \\ b_1 \\ b_0 \end{pmatrix}$$



$$\underline{\text{ex:}} \quad b_1 = 01 \times a_3 + 01 \times a_2 + 02 \times a_1 + 03 \times a_0$$

Il faut vérifier que toutes les transformations effectuées sont bijectives et que la bijection réciproque est calculable "en temps raisonnables".

- $S = S \oplus K_i \rightarrow$ bijective
- $S = \text{ByteSub}(S) \rightarrow$ bijective
- $\boxed{a} \quad x = \text{INV}(a) \rightarrow$ bijective
- $\boxed{(b)} = (M)(x) + (v) \rightarrow$ bijective car $(x) = M^{-1}((b) - (v))$
- $S = \text{ShiftRows}(S) \rightarrow$ bijective
- $S = \text{MixColumn}(S) \rightarrow$ bijective avec le pivot de gauß.

IV. Modes opératoires.

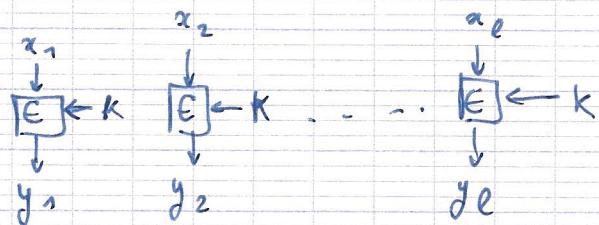
$x = x_1 \parallel x_2 \parallel x_3 \parallel \dots \parallel x_l$

$\xleftarrow[m]{\text{bits}}$ $\xleftarrow[m]{\text{bits}}$ $\xleftarrow[m]{\text{bits}}$ $\xleftarrow[m]{\text{bits}}$

message clair

1) Mode ECB (Electronic Code Book)

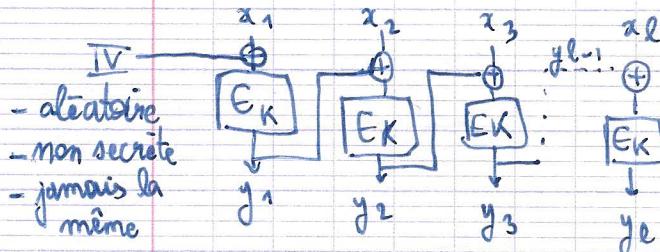
$$y = y_1 \parallel \dots \parallel y_e$$



Inconvénient:

- si $x_i = x_j$ alors $y_i = y_j$

2) Mode CBC (Cipher Block Chaining)



$$y_1 = E_K(x_1 \oplus IV)$$

$$\forall i \geq 2 \quad y_i = E_K(x_i \oplus y_{i-1})$$

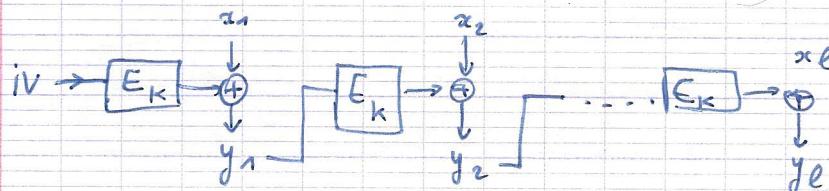
$$\text{Déchiffrement: } x_i = E_K^{-1}(y_i) \oplus IV$$

$$\forall i \geq 2 \quad x_i = E_K^{-1}(y_i) \oplus y_{i-1}$$

Alice $\xrightarrow{IV, y_1, y_2, \dots, y_e}$ Bob
 $C = (IV, y_1, \dots, y_e)$

Mode le plus utilisé actuellement

3) Mode CFB (Cipher Feed Back)



$$\begin{cases} y_1 = E_K(IV) \oplus x_1 \\ \forall i \geq 2, y_i = E_K(y_{i-1}) \oplus x_i \end{cases}$$

Déchiffrement:

$$x_1 = E_K(IV) \oplus y_1$$

$$\forall i \geq 2, x_i = E_K(y_{i-1}) \oplus y_i$$

Même si E n'est pas bijective \rightarrow ça marche

4) Mode OFB (Output Feed Back)



Message clair $x = x_1 \| x_2 \| \dots \| x_l$

Message chiffré $y = y_1 \| y_2 \| \dots \| y_l$ où $\forall i \quad y_i = x_i \oplus z_i$

Avantage : on peut calculer à l'avance z_1, \dots, z_l avant de chiffrer

5) Mode CTR (compteur)

