

# PROTECTION DES DONNÉES VISUELLES

PAULINE PUTEAUX – CNRS, CRISTAL

LE 18/03/22



# BESOIN IMPORTANT EN SÉCURITÉ

- ▶ Besoins grandissants en cybersécurité
- ▶ Vidéo-surveillance, visioconférence, réseaux sociaux, *cloud*...
- ▶ Sécuriser les données elles-mêmes ou leur support physique ?
  - ▶ Protection de la vie privée (sécurité visuelle)
  - ▶ Respect des droits d'auteur
  - ▶ Véracité des données



Protection  
des données

« Au repos »

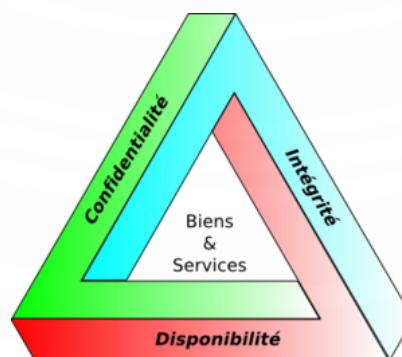
- Contrôle d'accès
  - Identification
  - Autorisation
- Disponibilité

En transit

- Confidentialité
- Intégrité
- Authentification
- Non réputation

# PRINCIPES DE SÉCURITÉ

- **Confidentialité** : L'information n'est accessible qu'à ceux dont l'accès est autorisé
- **Authentification** : Chaque personne est bien celle qu'elle prétend être (légitimité)
- **Intégrité** : Le message envoyé n'a pas été altéré de manière volontaire ou involontaire
- **Non-répudiation** : Aucune des deux parties ne pourra assurer ne pas être l'auteur du message
- **Disponibilité** : L'accès à un service ou à des ressources est garanti



# PROTECTION DES DONNÉES VISUELLES

- D'après CISCO, les données visuelles = **80% du trafic Internet mondial en 2019** (contre 67% en 2014) .
  - Nécessité de proposer des méthodes efficaces pour protéger ces données visuelles !

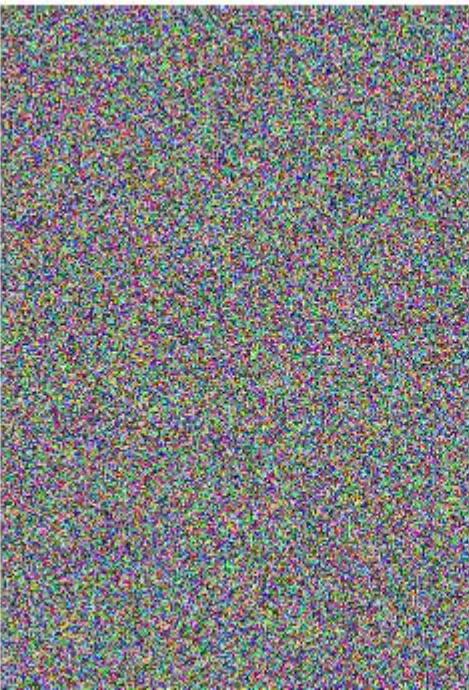
# PROTECTION DES DONNÉES VISUELLES

- D'après CISCO, les données visuelles = **80% du trafic Internet mondial en 2019** (contre 67% en 2014) .
  - Nécessité de proposer des méthodes efficaces pour protéger ces données visuelles !
- De nombreux axes :
  - Chiffrement
  - Tatouage
  - Stéganographie
  - Analyse forensique
  - Biométrie



Attention à ne pas confondre tatouage, stéganographie et cryptographie !

# CHIFFREMENT



# TATOUAGE



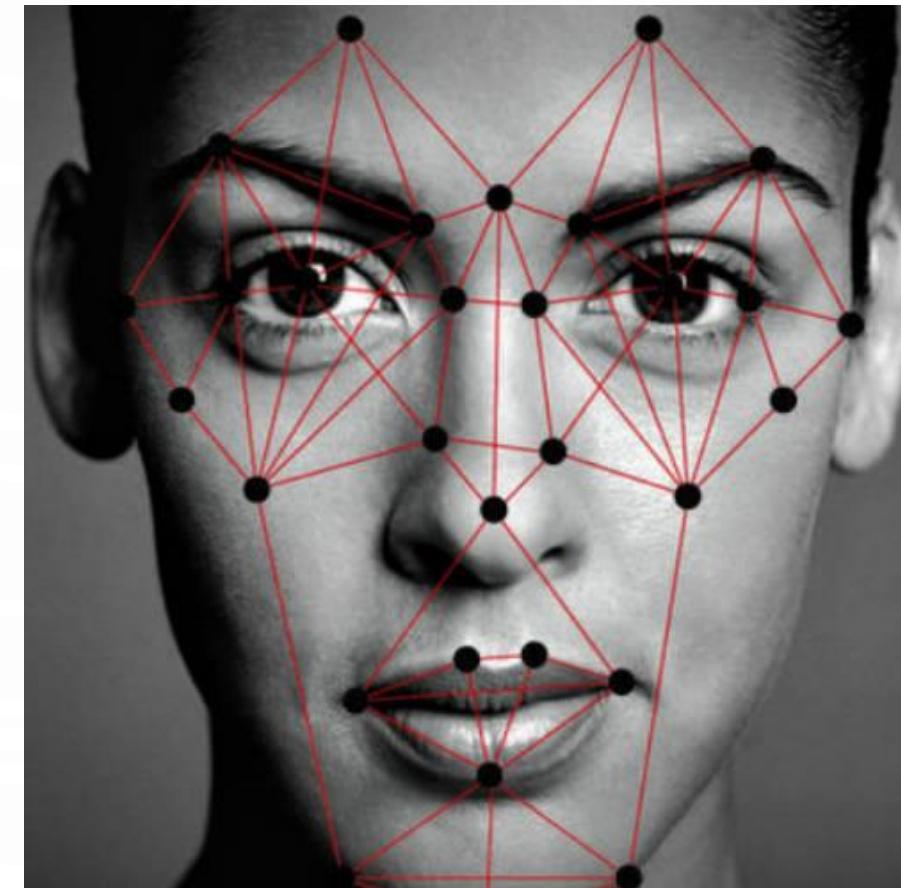
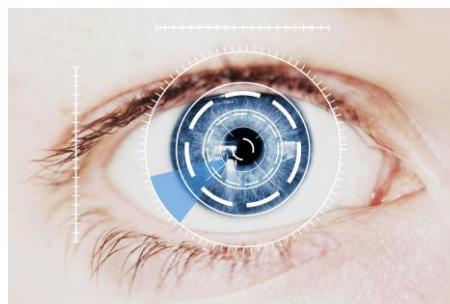
# STÉGANOGRAPHIE



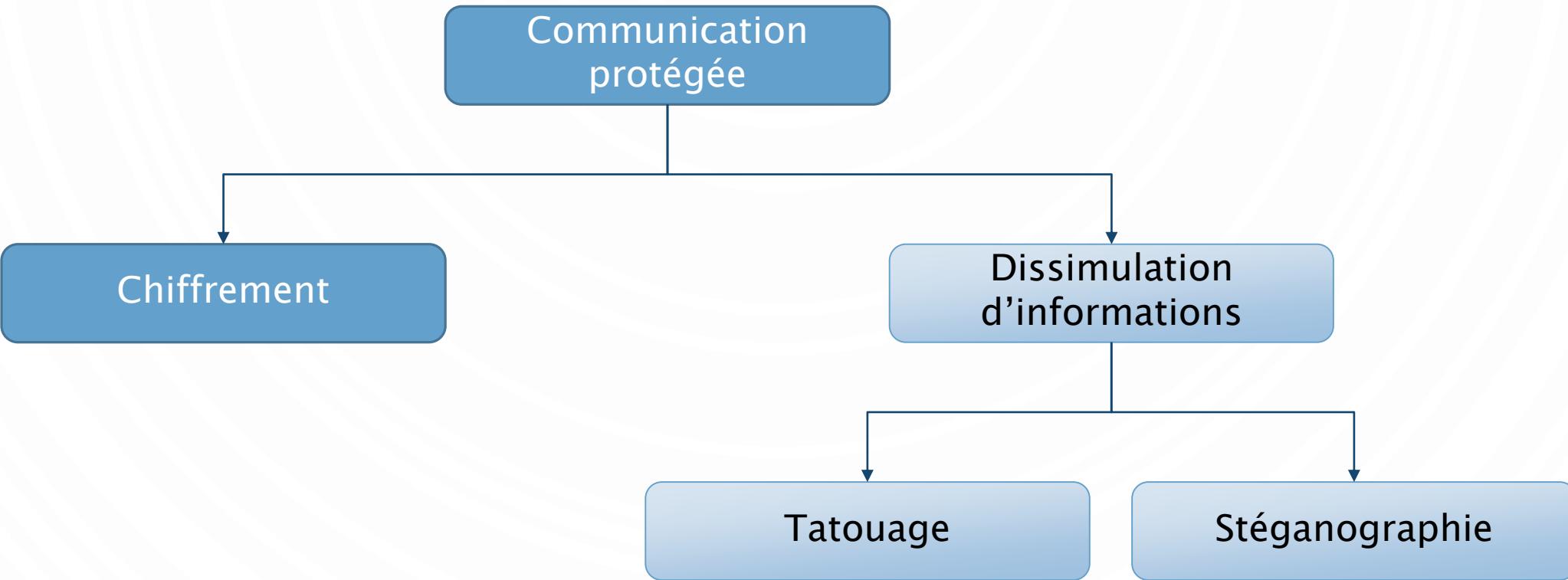
# ANALYSE FORENSIQUE



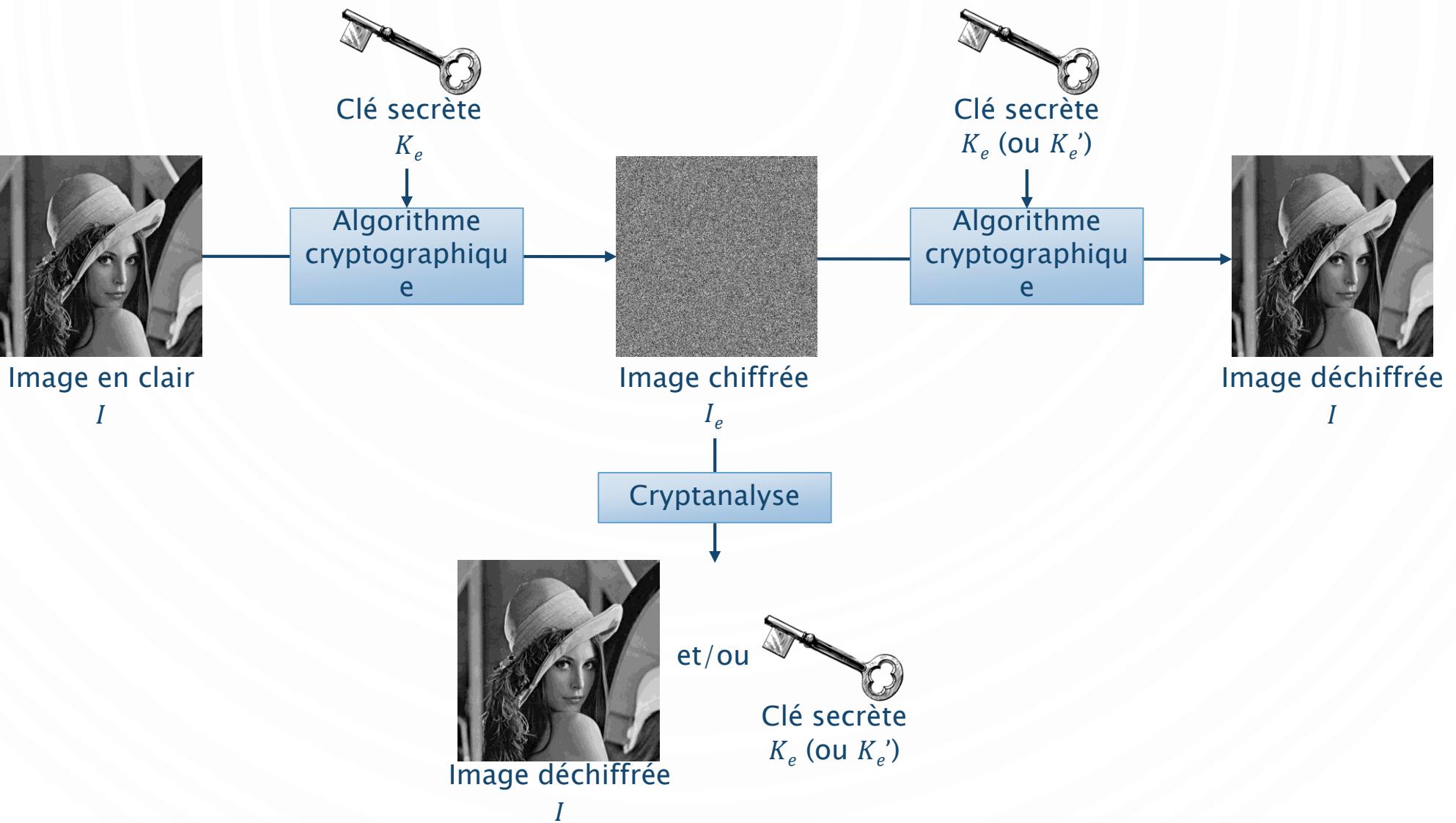
# BIOMÉTRIE



# TECHNIQUES DE PROTECTION DE DONNÉES



# TERMINOLOGIE



# PRINCIPE DE KERCKHOFFS

- Aucun secret ne doit résider dans l'algorithme cryptographique utilisé : **tout réside dans la clé !**
  - « La sécurité ne doit pas dépendre de tout ce qui ne peut pas être facilement changé. »
  - Pour un algorithme : **secret ≠ robustesse**
- Sans la clé, il doit être impossible de retrouver le message clair à partir du chiffré.
- Si on connaît la clé, on doit pouvoir déchiffrer le chiffré sans problème.



Auguste Kerckhoffs, *La cryptographie militaire*, Journal des sciences militaires, vol. IX, pp. 5-38, jan. 1883, pp. 161-191,

# ALGORITHME PUBLIÉ VS SECRET

## ALGORITHME PUBLIÉ

- Possible d'évaluer la sécurité de manière fiable
- Empêche les backdoors cachées par les concepteurs
- Grand nombre d'utilisateurs = Prix réduit + performance élevée
- Pas besoin de protection contre le reverse engineering
- Implémentations logicielles
- Standardisation locale et internationale

## ALGORITHME SECRET

- La cryptanalyse doit inclure la récupération de l'algorithme
- Petit nombre d'utilisateurs = Plus petite motivation à essayer de casser l'algorithme
- Indisponible pour un autre pays

# ENTROPIE

- **Entropie d'ordre zéro :** Si  $x$  est une variable aléatoire discrète, l'entropie de  $x$  est définie par :

$$H(X) = - \sum_x P(X = x) \log(P(X = x))$$

- **Entropie conditionnelle :** Incertitude qui reste sur  $X$  quand on connaît déjà  $Y$  :

$$H(X|Y) = - \sum_{x,y} P(X = x, Y = y) \log(P(X = x|Y = y)) \text{ avec } P(X = x, Y = y) = P(X = x|Y = y)P(y)$$

- **Entropie jointe :**  $H(X, Y) = H(Y) + H(X|Y)$

- Lien entre les entropies :  $H(X, Y) \leq H(X) + H(Y)$
- Si  $X$  et  $Y$  sont indépendantes :  $H(X, Y) = H(X) + H(Y)$

Claude E. Shannon, *A mathematical theory of communication*, Bell System Technical Journal, vol. 27, p. 379–423 and 623



# ENTROPIE ET CRYPTOGRAPHIE

- L'incertitude liée à un système est  $H(M)$  : besoin de  $H(M)$  bits d'information pour retrouver le message
- Pour qu'un système cryptographique soit sûr, il faut et il suffit que :

$$H(M|C) = H(M)$$

Q : Démontrez-le.

# ENTROPIE ET CRYPTOGRAPHIE

- L'incertitude liée à un système est  $H(M)$  : besoin de  $H(M)$  bits d'information pour retrouver le message
- Pour qu'un système cryptographique soit sûr, il faut et il suffit que :

$$H(M|C) = H(M)$$

Q : Démontrez-le.

R : On rappelle qu'un système est sûr si la connaissance de  $C$  (sans la clef) n'apporte aucune information sur  $M$ .

$$H(M|C) = H(M) \Leftrightarrow H(M|C) + H(C) = H(M) + H(C)$$

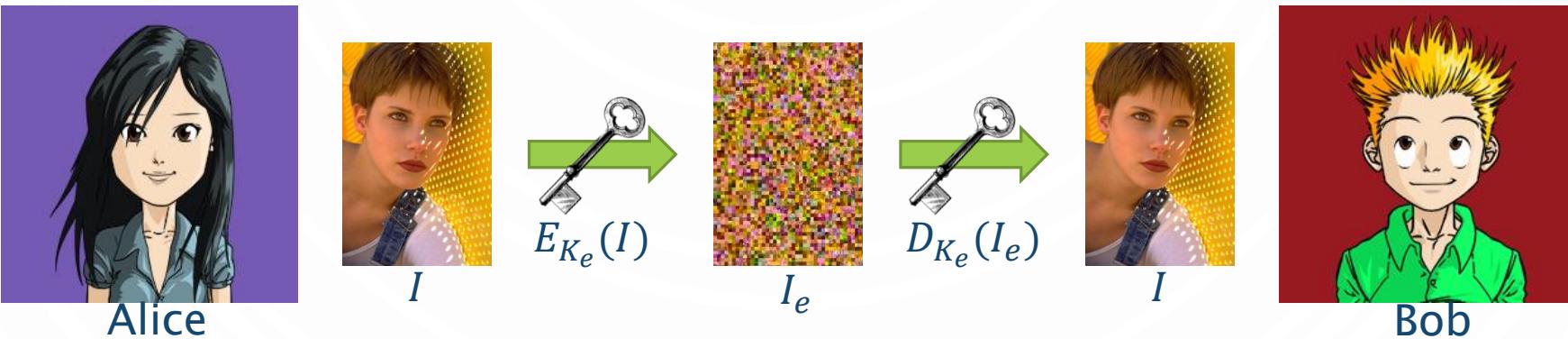
$$\Leftrightarrow H(M, C) = H(M) + H(C)$$

$\Leftrightarrow M$  et  $C$  sont indépendants

$$\Leftrightarrow P(M|C) = P(M)$$

# CRYPTOGRAPHIE SYMÉTRIQUE

- La cryptographie symétrique, également dite à clé secrète permet à la fois de chiffrer et de déchiffrer des messages à l'aide d'une même clé secrète



- Deux catégories de méthodes :
  - Chiffrement par flot : traitement des données de longueur quelconque, sans besoin de les découper
  - Chiffrement par bloc : découpage des données à chiffrer en blocs de taille généralement fixe

# CRYPTOGRAPHIE SYMÉTRIQUE

## ■ Caractéristiques

- Principe : Algorithmes basés sur des opérations en fonction de la clé
- Transposition/permotion
  - Propager l'information relative à chaque bit du message en clair dans le message chiffré (**diffusion**)
- Substitution
  - Supprimer les relations entre le message en clair et le message chiffré (**confusion**)
- Taille des clés : (standard) 128 bits minimum
- Performances : Très rapide
- Distribution des clés :
  - Très critique
  - Doit s'effectuer de manière sécurisée

# CHIFFREMENT PAR MÉLANGE

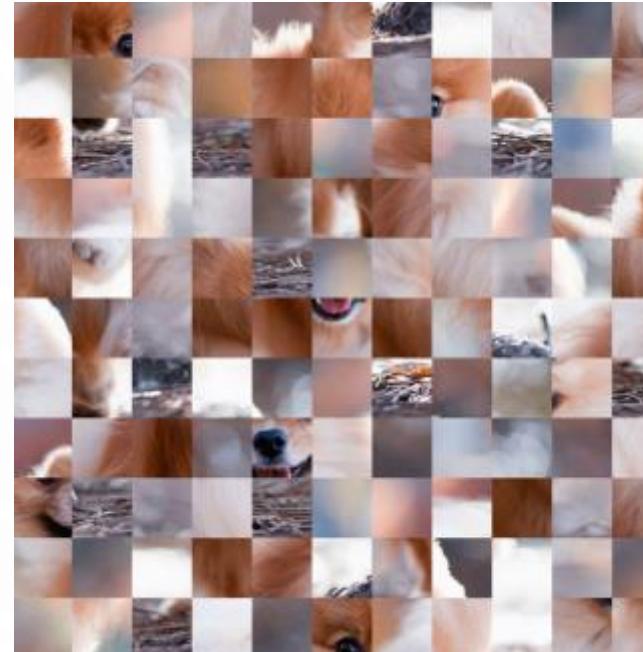
- Génération d'une séquence pseudo-aléatoire à l'aide d'un PRNG
- Clé utilisée comme graine d'initialisation du PRNG
- Utilisation de la séquence pseudo-aléatoire pour permute les pixels (ou les bits) de l'image en clair



$$\sigma = \begin{pmatrix} p(0,0) & p(0,1) & p(0,2) & p(1,0) & p(1,1) & p(1,2) & p(2,0) & p(2,1) & p(2,2) \\ p(2,0) & p(1,2) & p(0,1) & p(2,2) & p(0,2) & p(0,0) & p(1,1) & p(1,0) & p(2,1) \end{pmatrix}$$

Q : Quel est le nombre de permutations possibles ?

# CHIFFREMENT PAR MÉLANGE



# CHIFFREMENT PAR XOR

- Génération d'une **séquence pseudo-aléatoire** à l'aide d'un **PRNG**
- Clé utilisée comme graine d'initialisation du PRNG
- Utilisation de la séquence pseudo-aléatoire pour modifier la valeur des pixels de l'image en clair : ou-exclusif entre l'image en clair et la séquence (**substitution**)

Clair	0	1	1	1	0	0	1	1
Séquence	1	0	1	0	0	1	0	1
Chiffré	1	1	0	1	0	1	1	0

Q : Comment déchiffrer le chiffré ?

# CHIFFREMENT PAR XOR

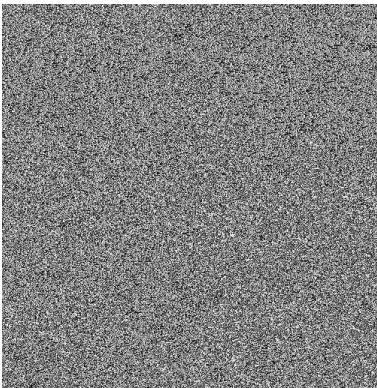
- Utiliser deux fois la même clé ?



$I_1$



$E_{K_e}(I_1)$



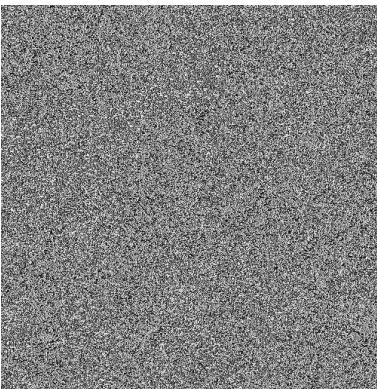
$I_{1e}$



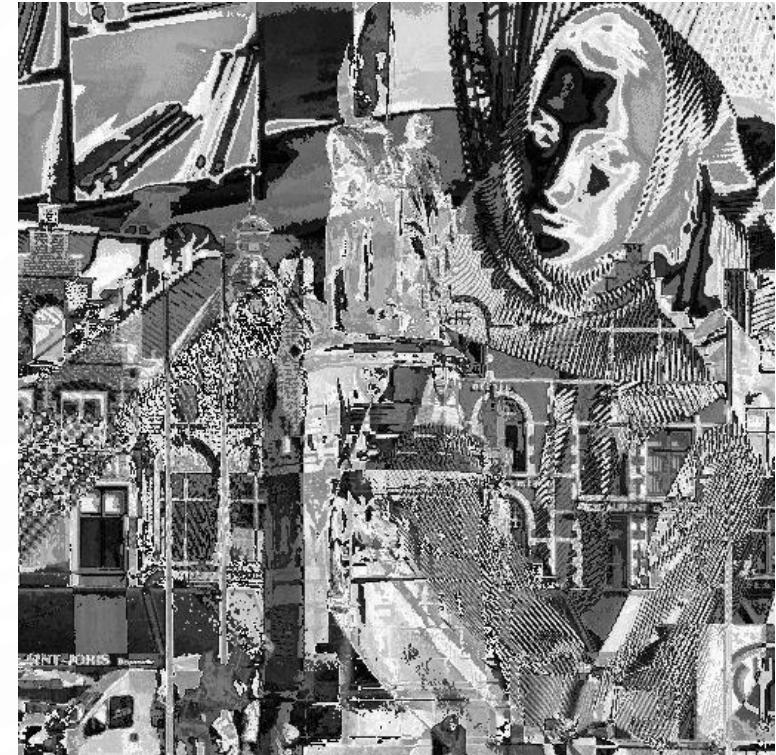
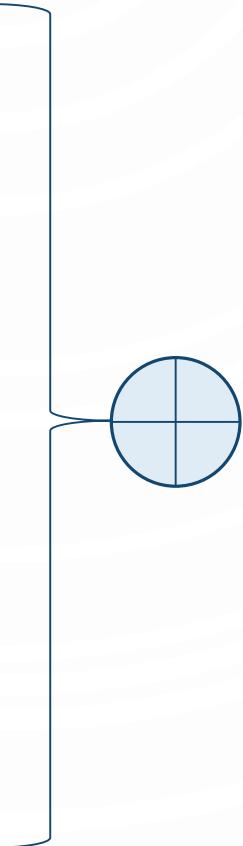
$I_2$



$E_{K_e}(I_2)$

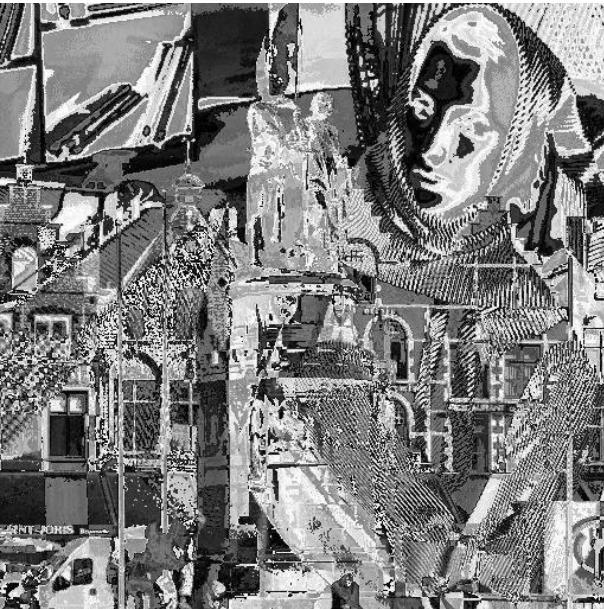


$I_{2e}$



# CHIFFREMENT PAR XOR

- Utiliser deux fois la même clef ?



Négatif →



Q : Comment expliquez-vous cette faiblesse ?

# CHIFFREMENT PARFAIT

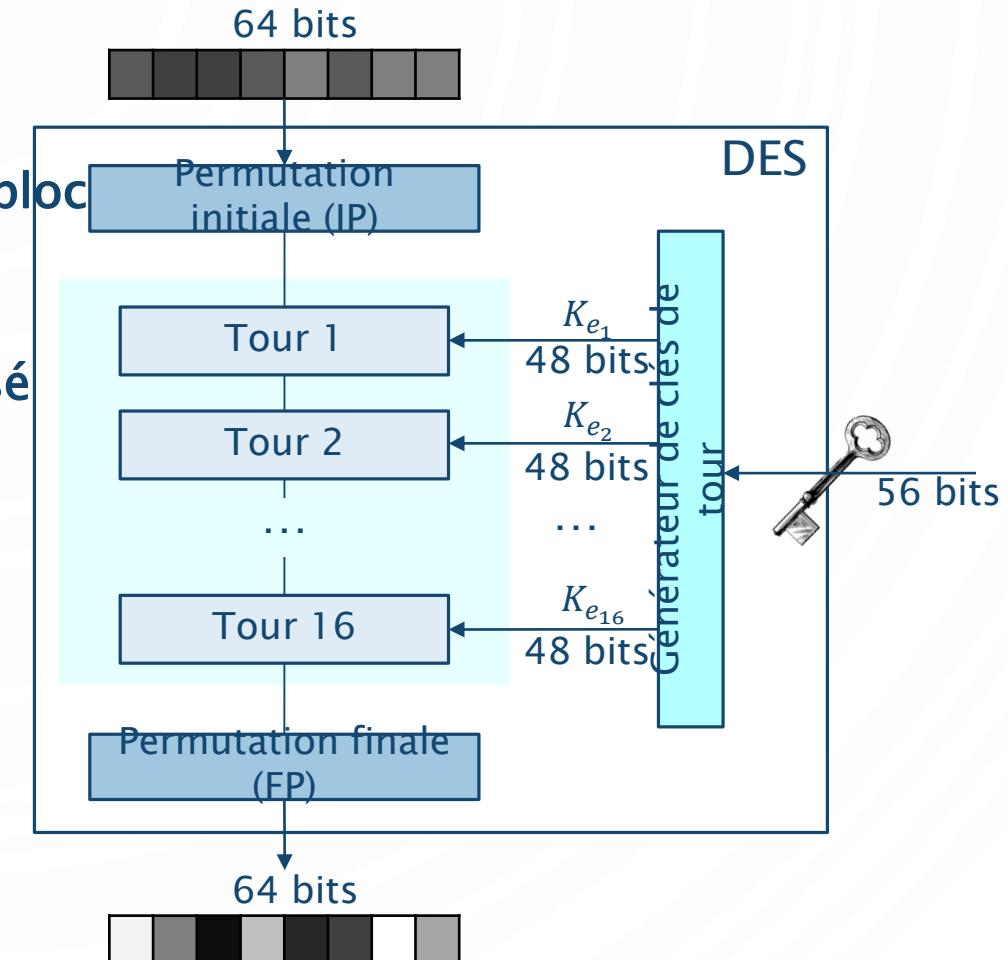
- Chiffre de Vernam (1926) ou « masque jetable »
- Trois impératifs pour la clé :
  - Aussi longue que le message à chiffrer
  - Parfaitement aléatoire
  - Utilisée pour chiffrer un seul message, puis détruite
- Modèle théorique car très difficile à mettre en place...

Q : Quelle est la probabilité d'apparition d'un niveau de gris dans l'image chiffrée ?

Q : Quelle est la valeur de l'entropie mesurée dans l'image chiffrée ?

# DATA ENCRYPTION STANDARD (DES)

- Algorithme de chiffrement symétrique, par bloc
- Publié en 1975, par IBM
- Aujourd'hui : considéré comme non-sécurisé
- Caractéristiques techniques :
  - Taille de la clé : 56 bits (+ 8 bits de parité)
  - Taille des blocs : 64 bits
  - Nombre de tours : 16

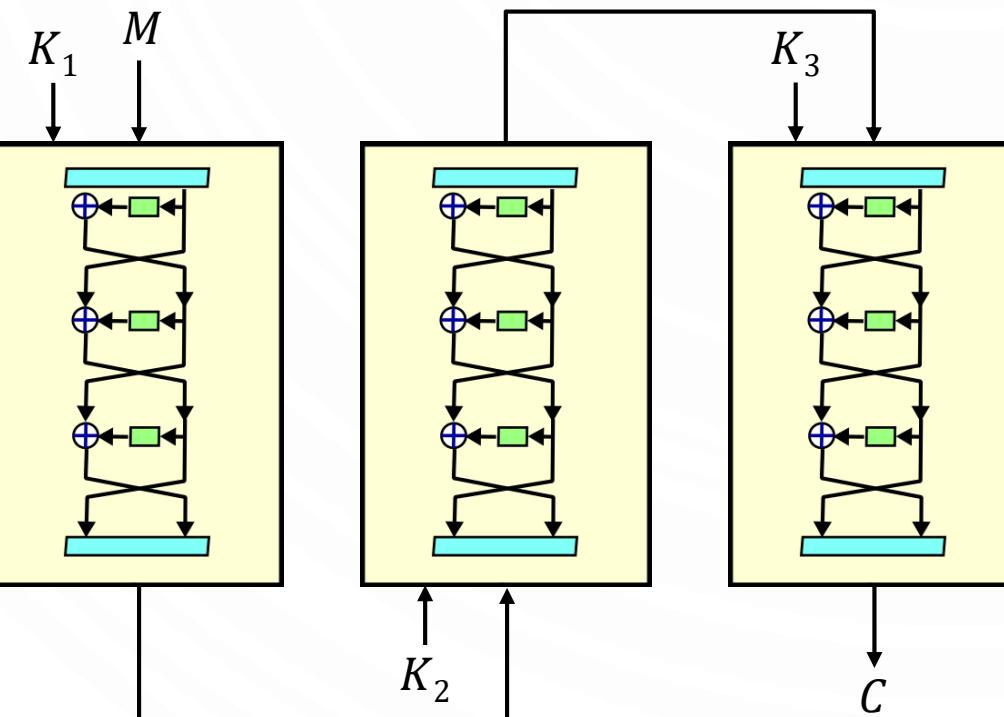


# DATA ENCRYPTION STANDARD (DES)

- Attaques
  - Attaque par force brute possible
    - Diffie–Hellman en 1977 (US\$ 20M), clé retrouvée en 1 jour → théorique
    - Wiener en 1993 (US\$ 1M), clé retrouvée en 7h → théorique
    - Electronic Frontier Foundation en 1998 (US\$ 250k), clé retrouvée en 2 jours → mise en pratique
    - COPACOBANA (Univ. Bochum & Kiel, en Allemagne) en 2006 (US\$ 10k) → mise en pratique
  - Cryptanalyse différentielle
    - Biham–Shamir en 1980 (CPA –Attaque par  $2^{47}$  clairs choisis)
  - Cryptanalyse linéaire
    - Matsui en 1994 (KPA –Attaque par  $2^{43}$  clairs connus)
    - Junod en 2001 (KPA –Attaque par  $2^{40}$  clairs connus)

# TRIPLE DES (3DES)

- Publié en 1998, dérivé de DES
- Plus grande taille des clés pour éviter l'attaque par force brute : **168 bits** possible
- Idée : Utiliser 3 clés de **56 bits** chacune



$$C = E_{K_3}(D_{K_2}(E_{K_1}(M)))$$

$$M = D_{K_1}(E_{K_2}(D_{K_3}(C)))$$

# TRIPLE DES (3DES)

- Attaque « Meet-in-the-middle » sur 2DES (même principe pour 3DES)
  - Attaque KPA :  $M_1, M_2, C_1, C_2$  connus tels que  $C_1 = E_{K_2}(E_{K_1}(M_1))$  et  $C_2 = E_{K_2}(E_{K_1}(M_2))$
  - Chiffrer une fois le message en clair revient à déchiffrer une fois le message chiffré
  - On a donc :  $E_{K_1}(M) = D_{K_2}(C)$
  - Calcul et stockage des  $2^{56}$  couples  $(K, E_K(M_1))$
  - Pour chaque clé  $K'$ , calcul de  $D_{K'}(M_1)$  et recherche de correspondance
  - Si couple de clés candidates  $K_1 = K$  et  $K_2 = K'$ , vérification  $C_2 = E_{K_2}(E_{K_1}(M_2))$
  - Nombre maximal d'essais :  $2 \times 2^{56} = 2^{57} \ll 2^{112}$

# ADVANCED ENCRYPTION STANDARD (AES)

- Algorithme de chiffrement symétrique, par blocs
- Gagnant d'un concours lancé en 1997
- Standard depuis 2001 (NIST)
- Vrai nom : Rijndael → Créé par deux belges Joan Daemen et Vincent Rijmen



# ADVANCED ENCRYPTION STANDARD (AES)

- Pourquoi un nouveau standard ?

- DES est devenu attaquable par force brute
- Développement de systèmes d'évaluation : analyse différentielle et linéaire
- Possible d'avoir une méthode de chiffrement plus rapide en utilisant des instructions processeur

- Pourquoi un concours public ?

- Rassembler la communauté travaillant sur la cryptographie
- Encourager la recherche autour des systèmes sécurisés
- Prévenir les « backdoors »
- Accélérer l'acceptation et l'adoption d'un standard

# ADVANCED ENCRYPTION STANDARD (AES)

## ■ Description générale

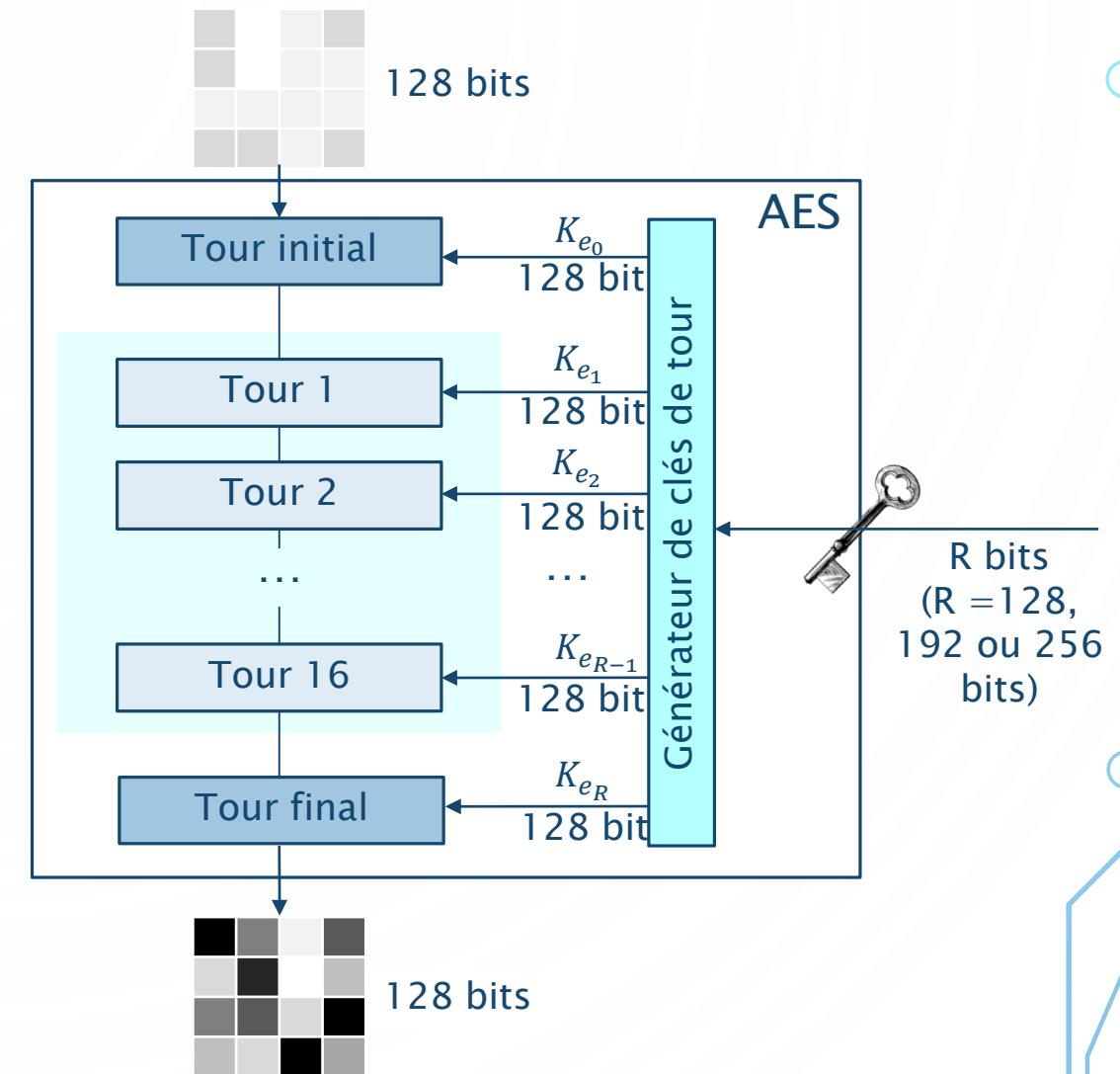
- Nombre de tours : 10, 12 ou 14 (suivant la taille de la clé)
- Chaque tour : 4 opérations
- Taille des blocs du message : 128 bits (4 colonnes de 4 octets)
- Taille de la clé de chiffrement : 128, 192 ou 256 bits

$p(0,0)$	$p(0,1)$	$p(0,2)$	$p(0,3)$
$p(1,0)$	$p(1,1)$	$p(1,2)$	$p(1,3)$
$p(2,0)$	$p(2,1)$	$p(2,2)$	$p(2,3)$
$p(3,0)$	$p(3,1)$	$p(3,2)$	$p(3,3)$

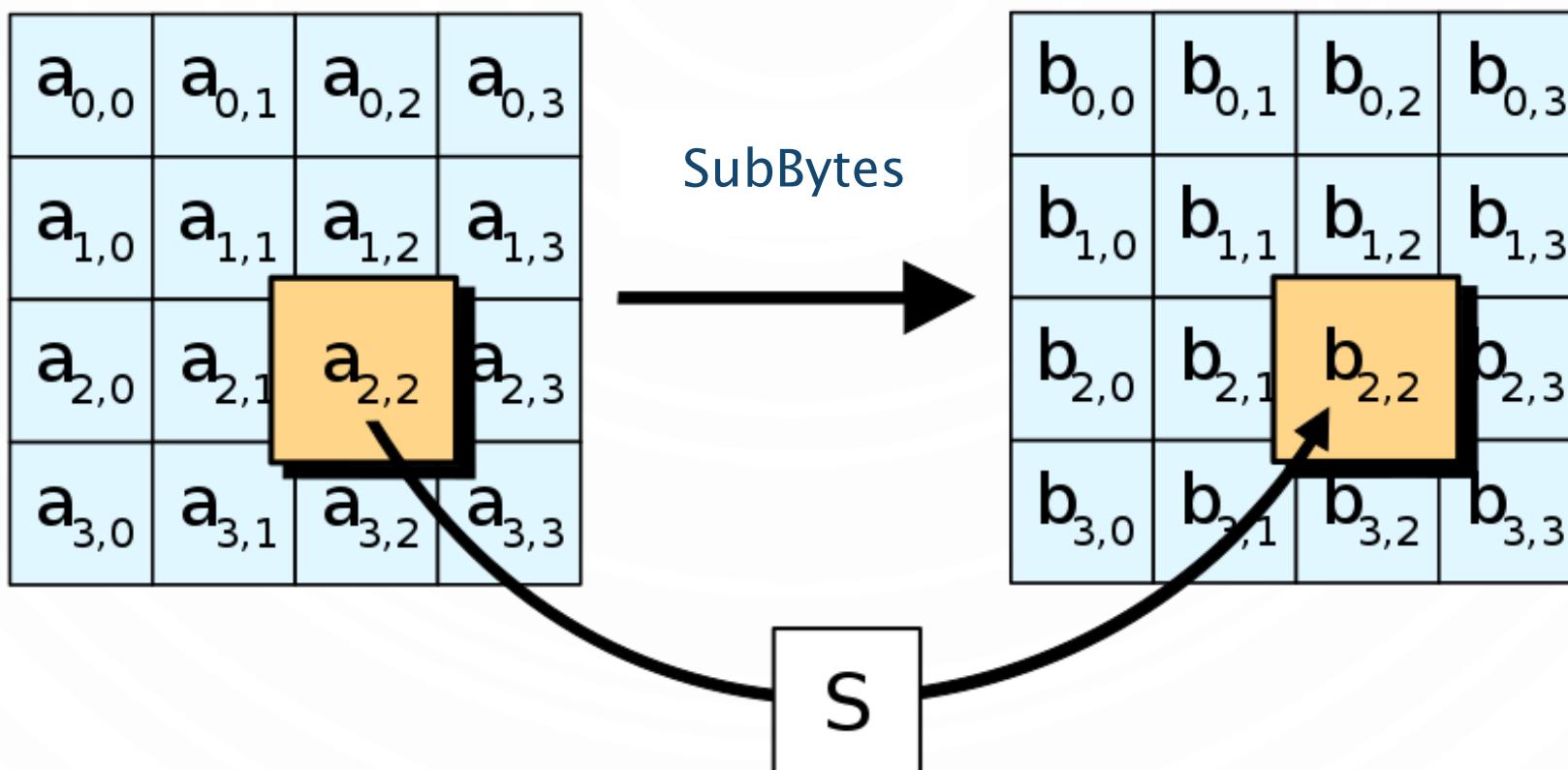
# ADVANCED ENCRYPTION STANDARD (AES)

## ■ Description générale

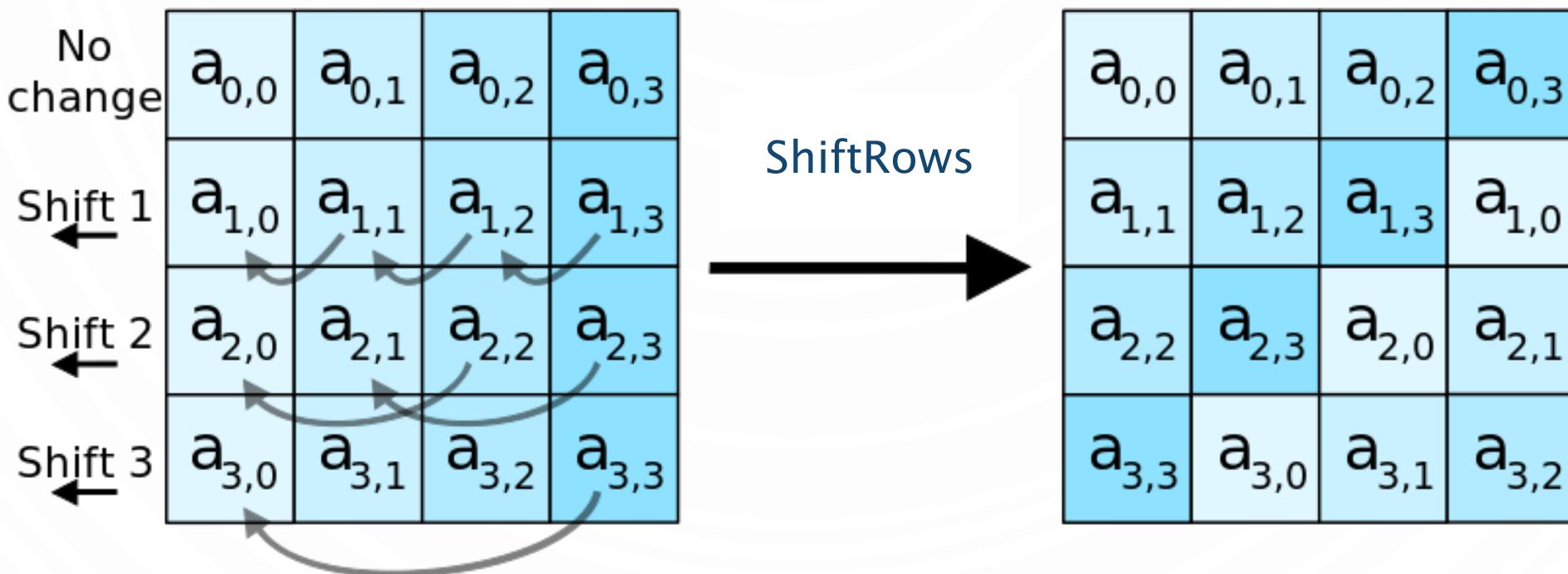
1. KeyExpansion
2. Tour initial
  1. AddRoundKey
3. Pour chaque tour suivant
  1. SubBytes
  2. ShiftRows
  3. MixColumns
  4. AddRoundKey
4. Tour final (pas de MixColumns)
  1. SubBytes
  2. ShiftRows
  3. AddRoundKey



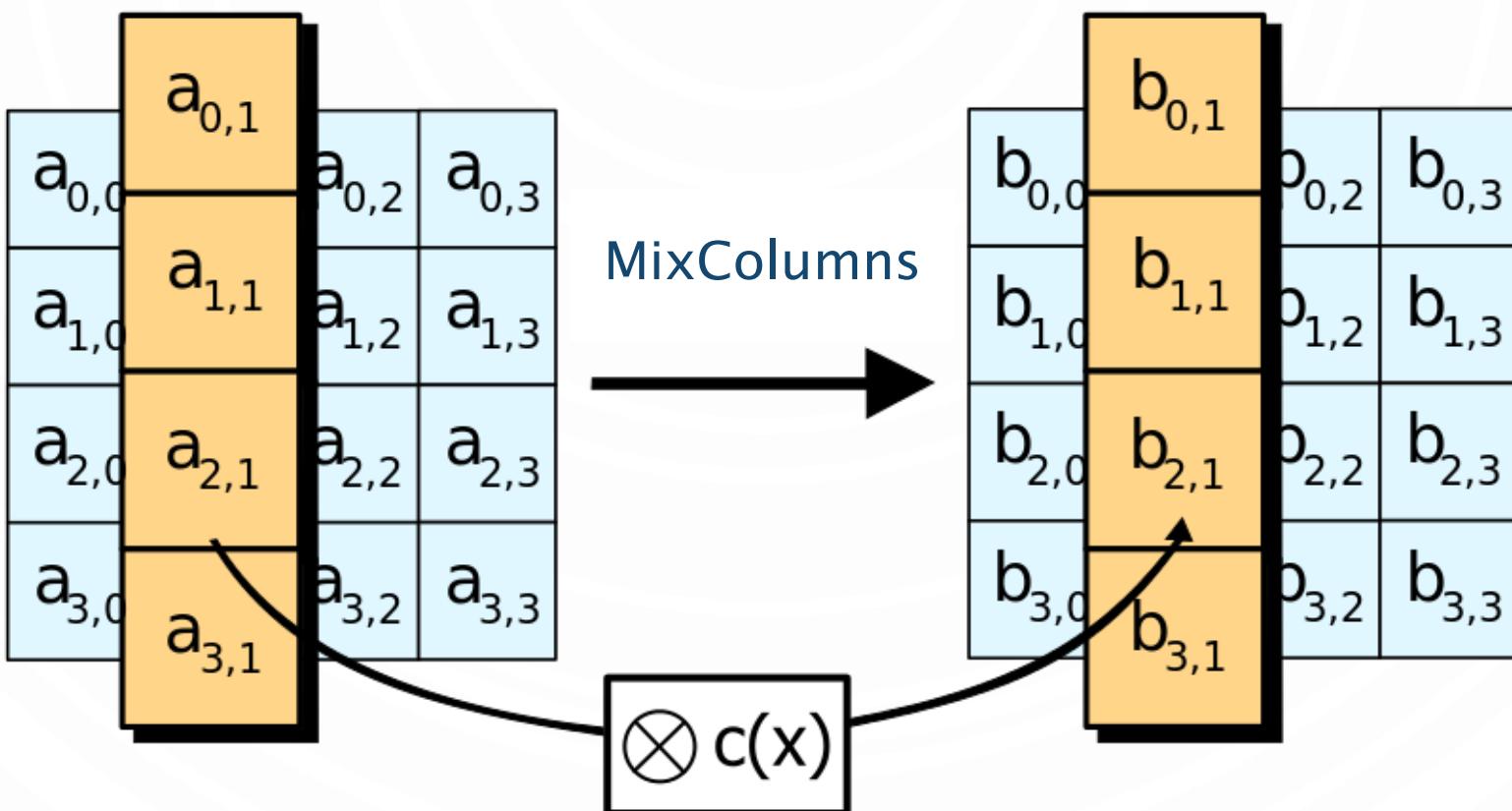
# ADVANCED ENCRYPTION STANDARD (AES)



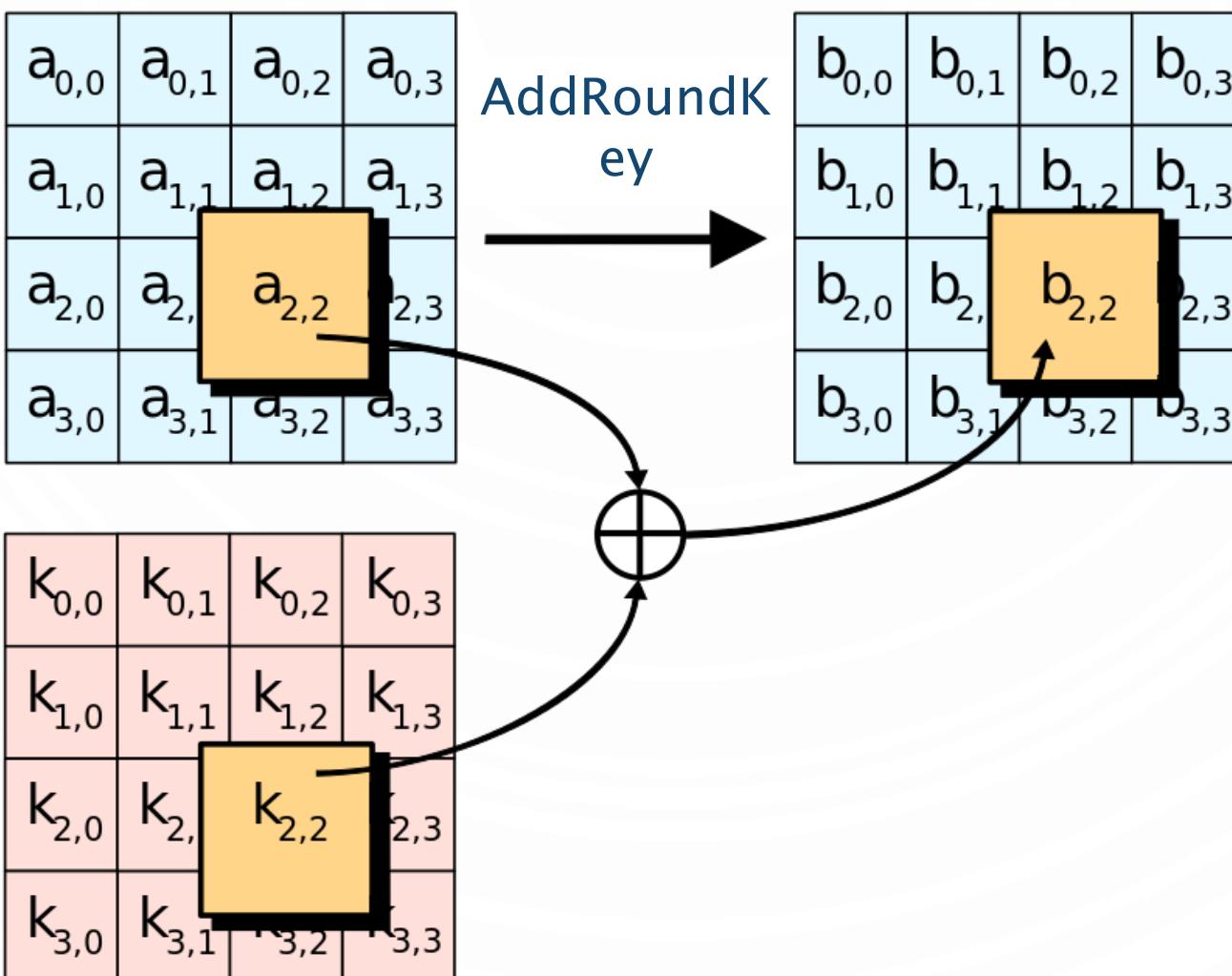
# ADVANCED ENCRYPTION STANDARD (AES)



# ADVANCED ENCRYPTION STANDARD (AES)

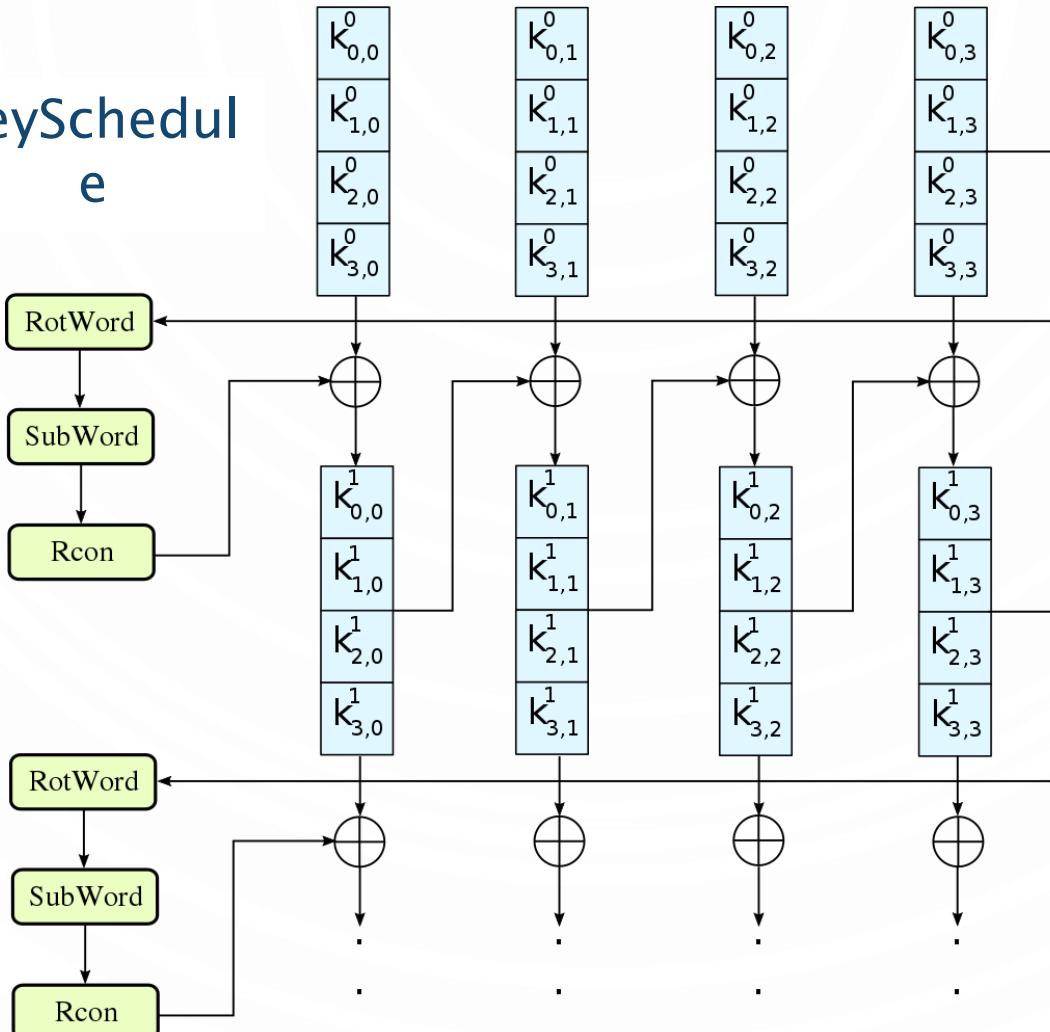


# ADVANCED ENCRYPTION STANDARD (AES)



# ADVANCED ENCRYPTION STANDARD (AES)

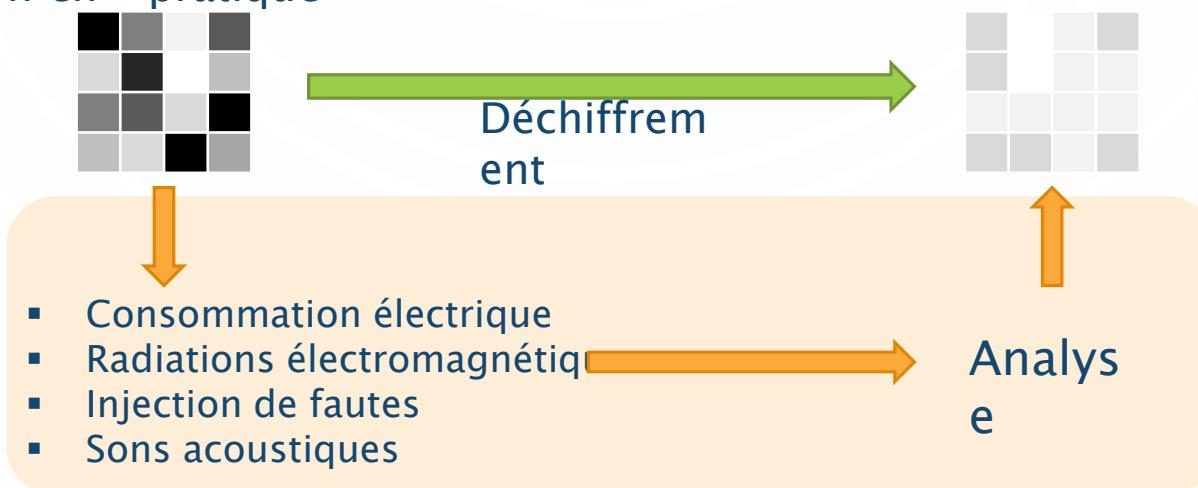
KeySchedule



# ADVANCED ENCRYPTION STANDARD (AES)

## ■ Attaque par canal auxiliaire

- Recherche et exploitation des failles dans l'implémentation, logicielle ou matérielle
- Ne remet pas en cause la robustesse théorique des méthodes et procédures de sécurité
- Une sécurité « mathématique » ne garantit pas forcément une sécurité lors de l'utilisation en « pratique »

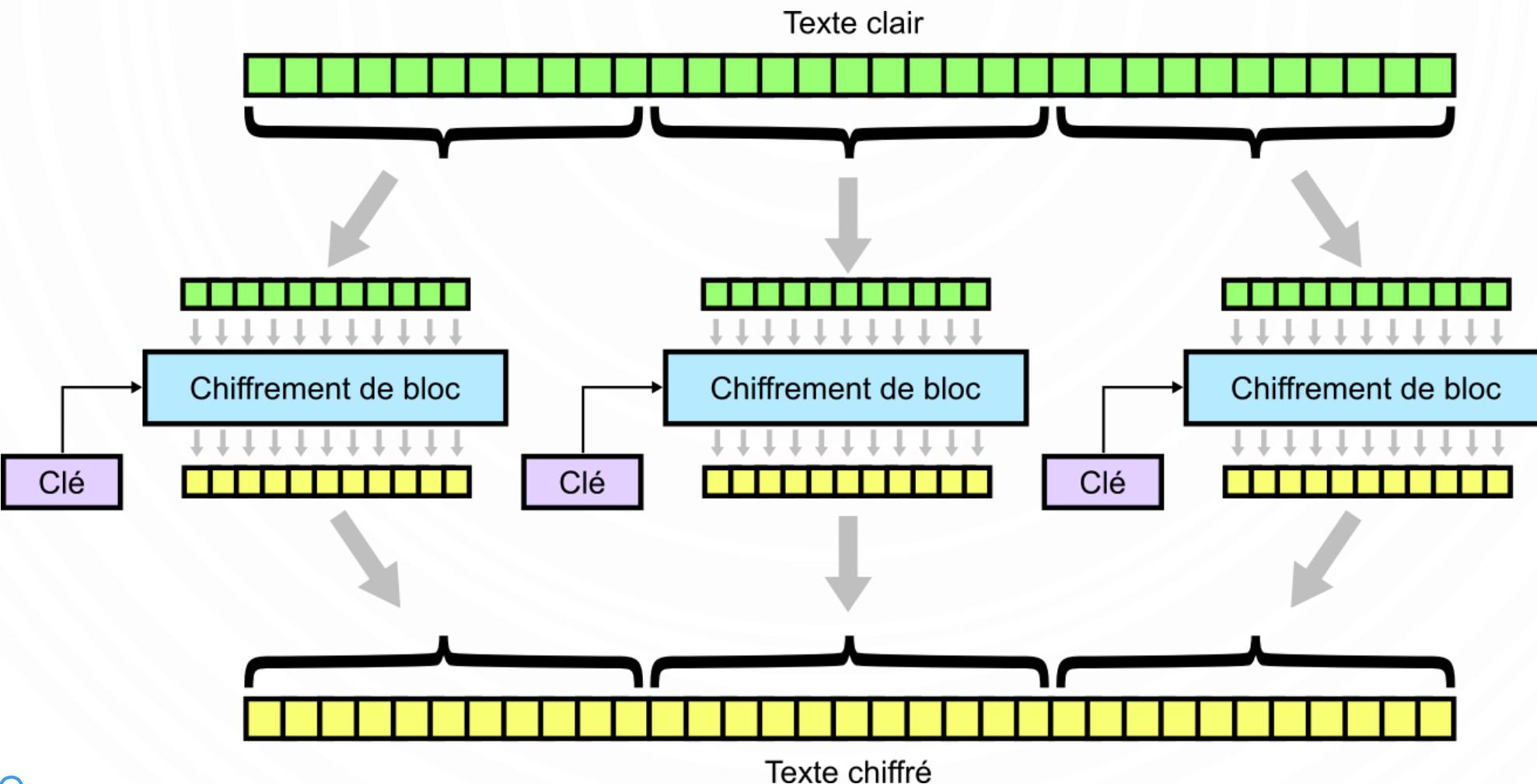


- Beaucoup d'attaques publiées non faisables en pratique (2013)
- Considéré sûr à ce jour

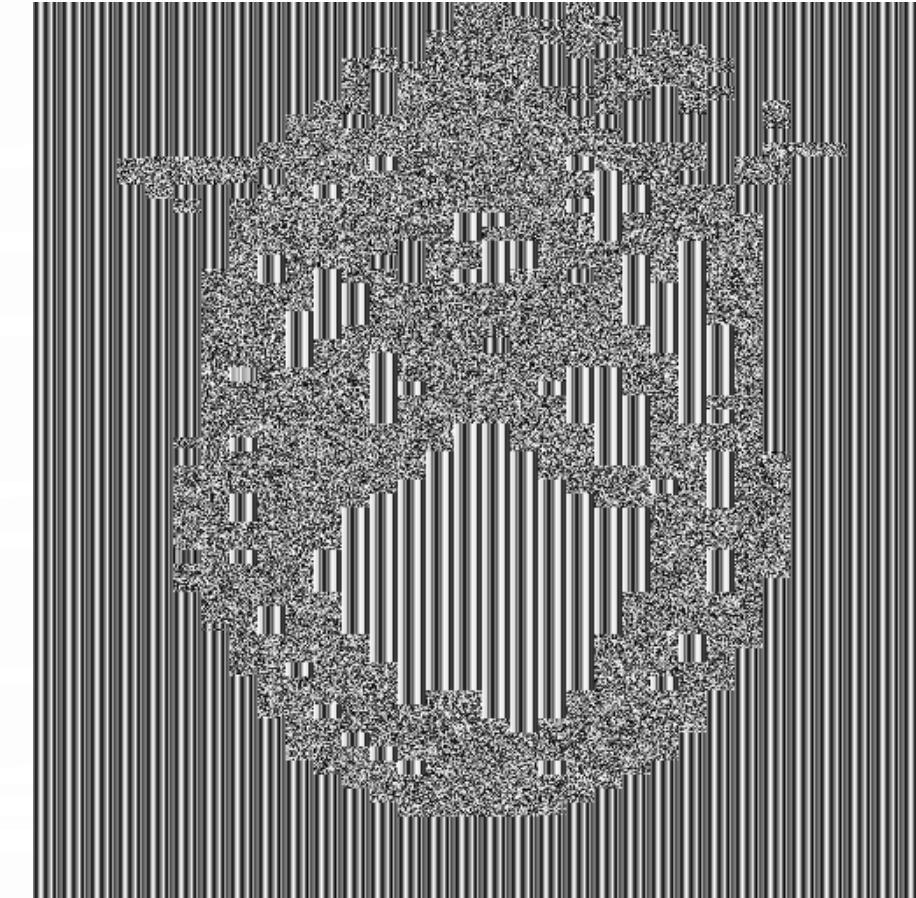
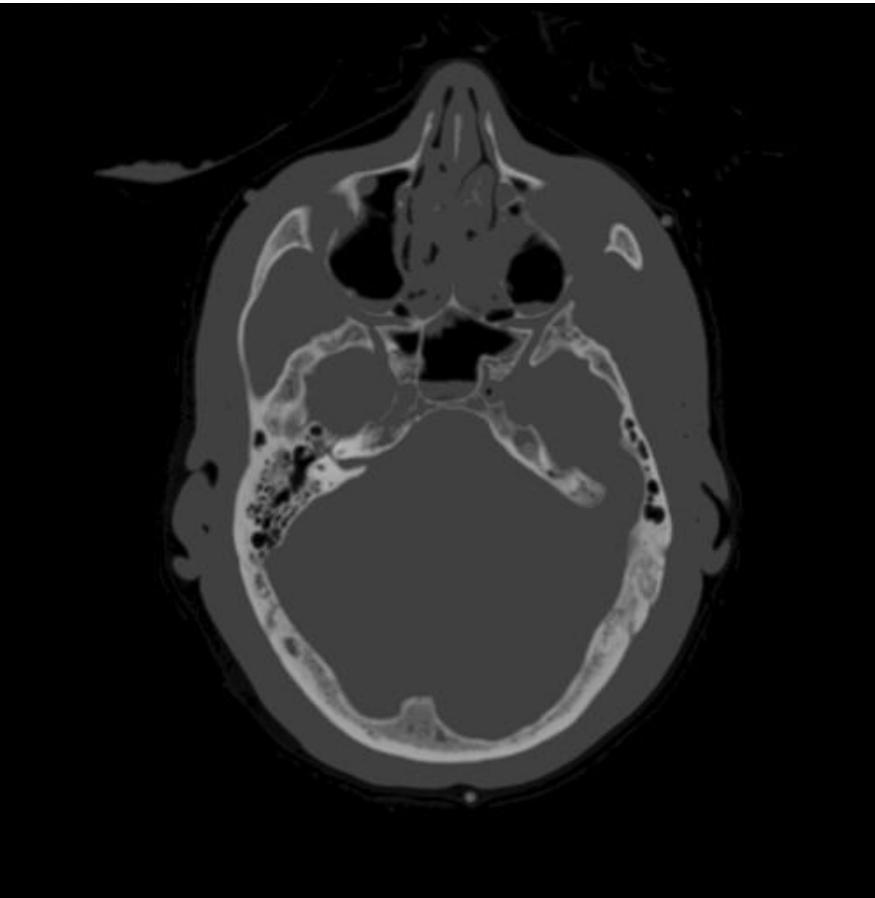
# MODES DE CHIFFREMENT

- Généralement défini pour le chiffrement par bloc (peut-être étendu au chiffrement de pixels)
- 5 modes de chiffrement principaux :
  - ECB (« Electronic CodeBook » – Dictionnaire de codes)
  - CBC (« Cipher Block Chaining » – Enchaînement de blocs)
  - CFB (« Cipher FeedBack » – Chiffrement à rétroaction)
  - OFB (« Output FeedBack » – Chiffrement à rétroaction de sortie)
  - CTR (« CounTeR » – Chiffrement basé sur un compteur)

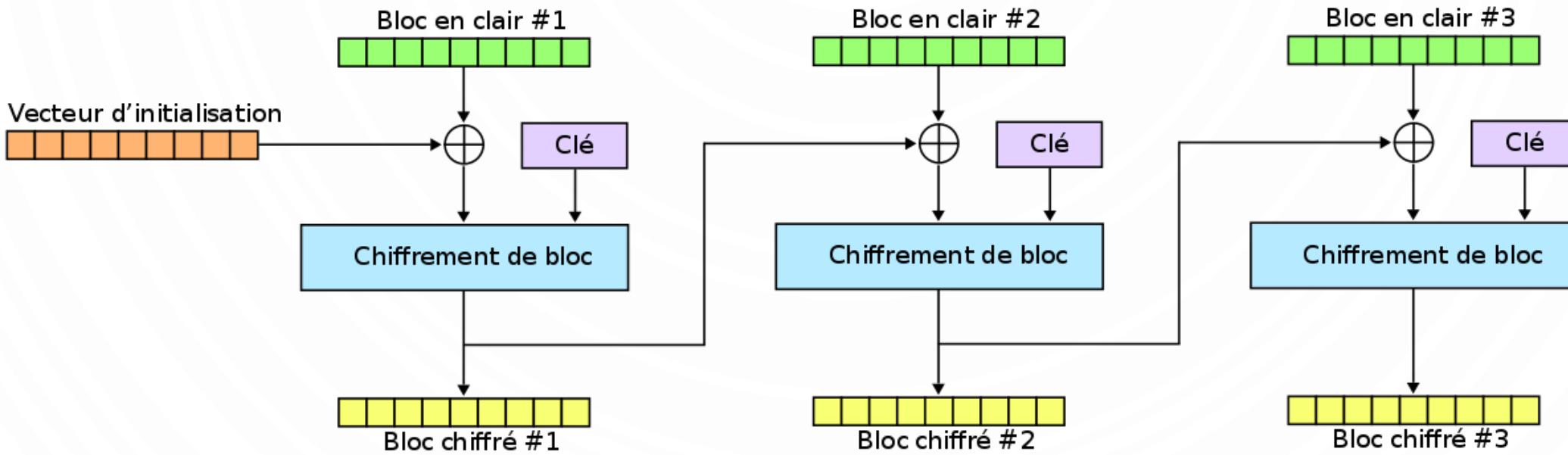
# ECB (« ELECTRONIC CODEBOOK »)



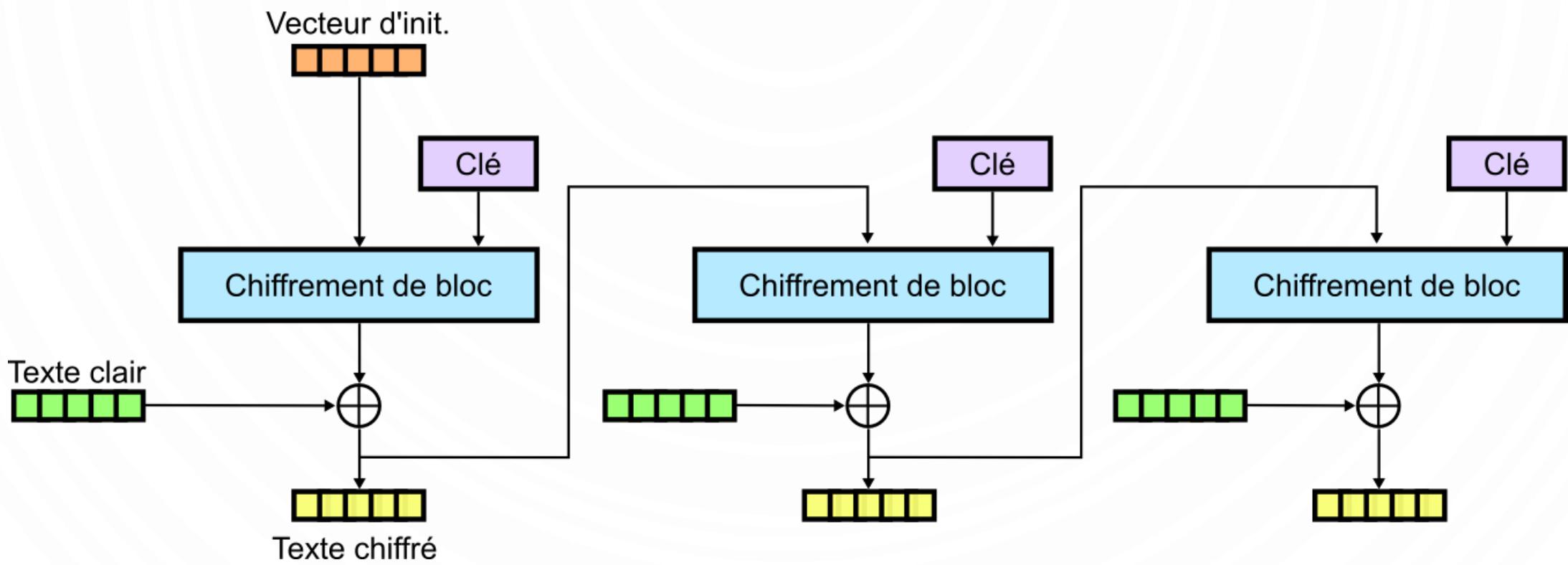
# PROBLÈME DU MODE ECB



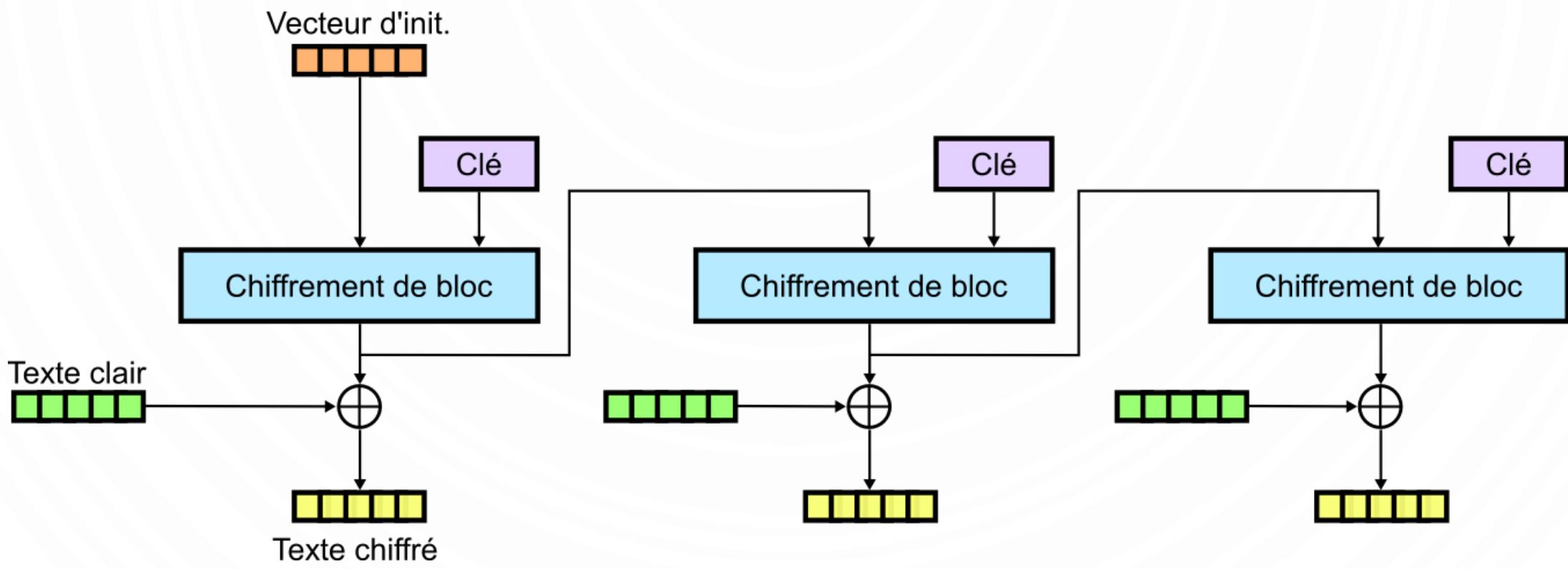
# CBC (« CIPHER BLOCK CHAINING »)



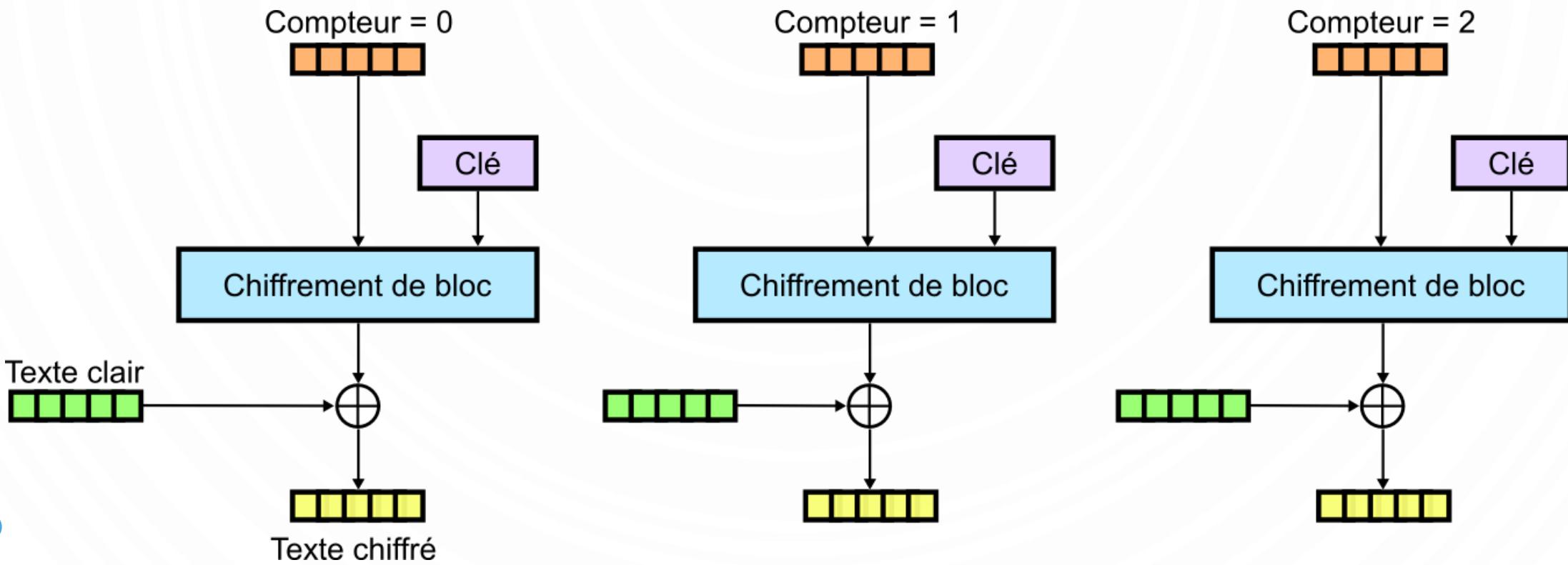
# CFB (« CIPHER FEEDBACK »)



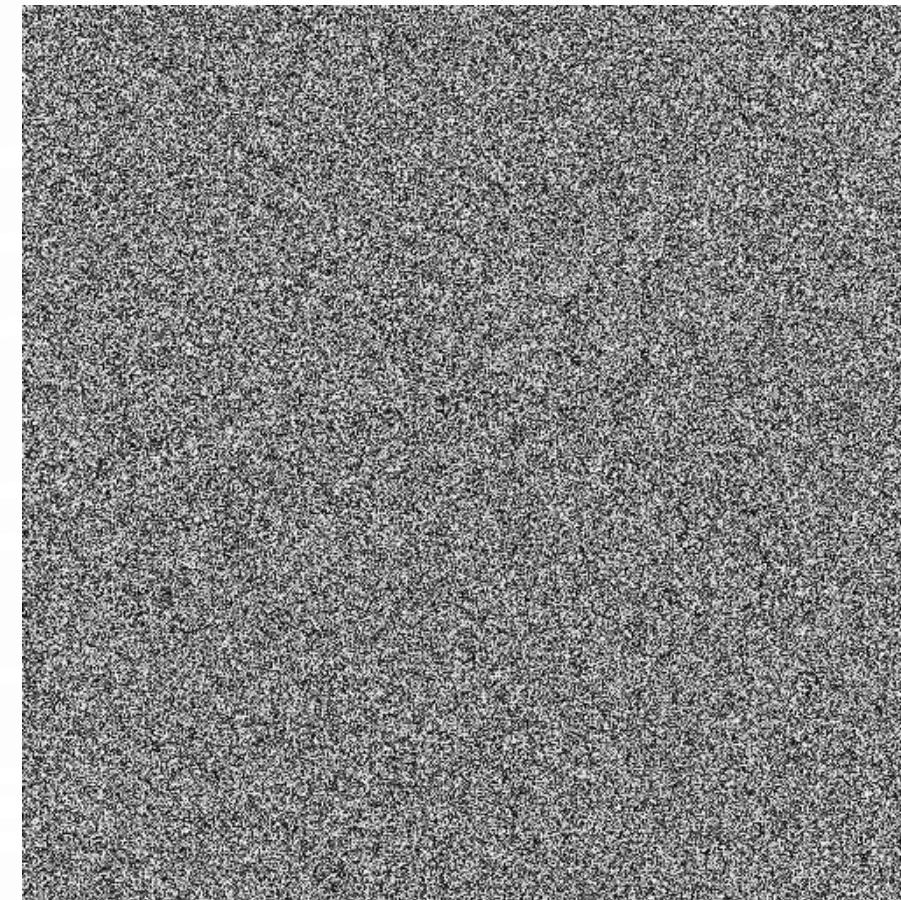
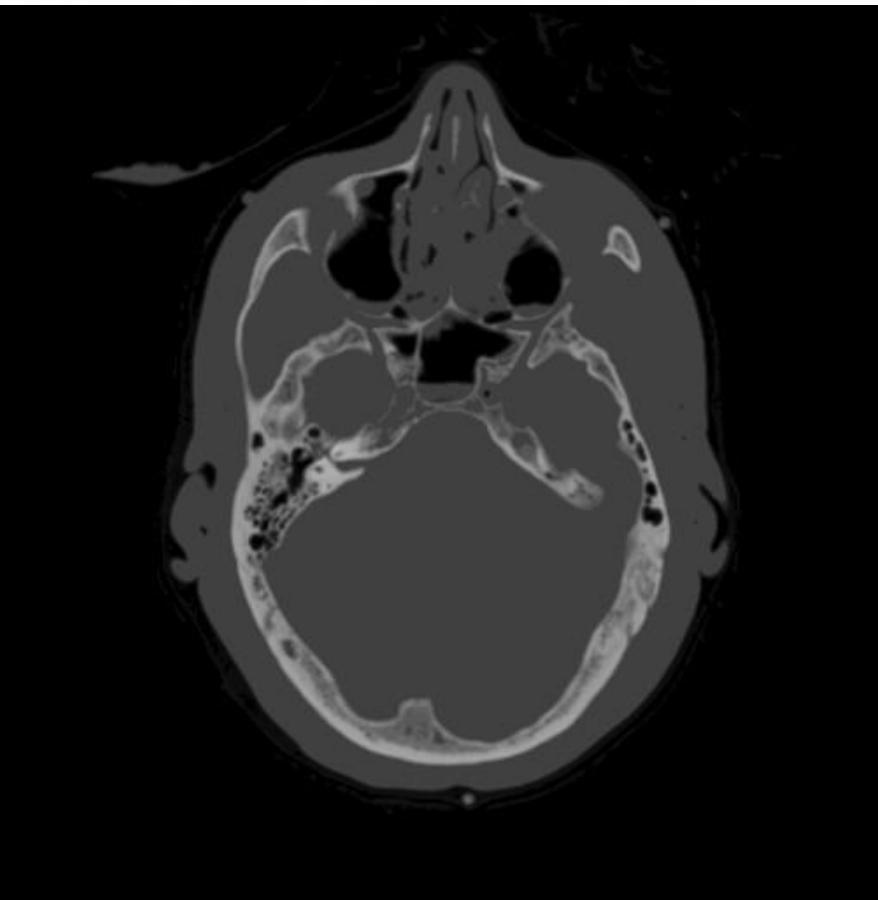
# OFB (« OUTPUT FEEDBACK »)



# CTR (« COUNTER »)



# MODE AVEC CLÉ DYNAMIQUE



# CRYPTOGRAPHIE ASYMÉTRIQUE

- Caractéristiques

- Une clé privée  $K_{priv}$  et une clé publique  $K_{pub}$
- Propriétés :
  - La connaissance de la clé publique  $K_{pub}$  ne permet pas de déduire la clé privée  $K_{priv}$
  - $D_{K_{priv}}(E_{K_{pub}}(M)) = M$

- Principe : Fonction unidirectionnelle à trappe

- « Facile » à calculer dans un sens, « difficile » à inverser
- Sauf si on connaît une information secrète (la trappe)
- Algorithmes basés sur des opérations d'exponentiation en algèbre modulaire

# CRYPTOGRAPHIE ASYMÉTRIQUE

- Caractéristiques

- Génération des clés :

- A partir de grands nombres premiers  $K_{pub} = f(K_{priv})$
    - Calcul de  $K_{priv} = f^{-1}(K_{pub})$  impossible
    - Taille des clés : 512 bits ou 1024 bits
    - Performances : 1000 fois plus lents que les algorithmes symétriques !
    - Nombre de clés : autant de paires que d'entités
    - Distribution des clés : Facilitée car pas d'échange de clés secrètes
      - Clé secrète conservée par les entités
      - Clé publique échangée

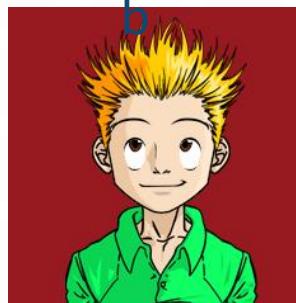
# PROTOCOLE DE DIFFIE-HELLMAN (1)



Alic



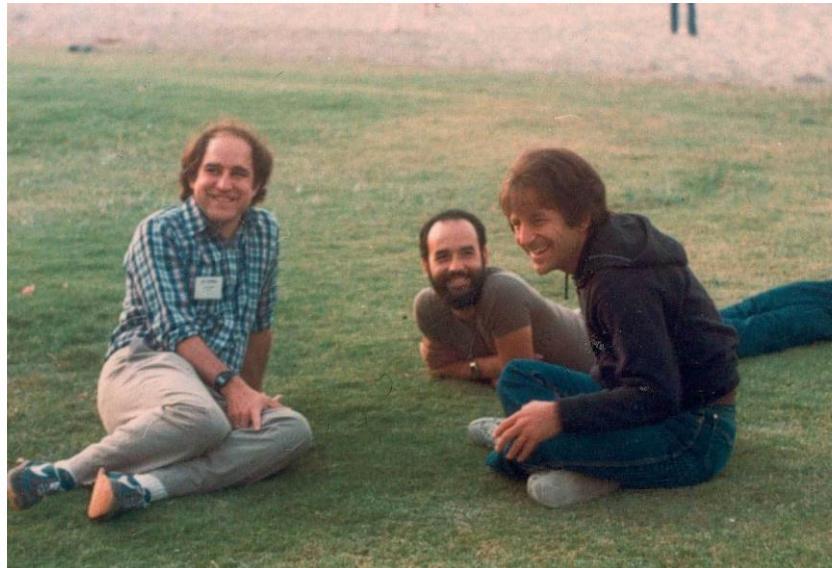
Bo



- 1) Alice et Bob choisissent un grand nombre premier  $p$  et d'un entier  $1 \leq a < p$
- 2) Alice choisit secrètement  $x_A$   
secrètement  $x_B$
- 3) Alice calcule  $y_A = a^{x_A} \pmod{p}$   
 $a^{x_B} \pmod{p}$
- 4) Alice et Bob s'échangent les valeurs de  $y_A$  et  $y_B$
- 5) Alice calcule  $y_B^{x_A} = (a^{x_B})^{x_A}$
- 3) Bob calcule  $y_A^{x_B} = (a^{x_A})^{x_B}$
- Q (sur quoi) repose la sécurité de l'échange des clés K

# RIVEST-SHAMIR-ADLEMAN (RSA)

- Crée en 1977 par Ron Rivest, Adi Shamir et Leonard Adleman
- Breveté par le MIT en 1983
- Basé sur le problème de la factorisation des grands nombres entiers



# RIVEST-SHAMIR-ADLEMAN (RSA)

- Génération du couple de clés ( $K_{pub}, K_{priv}$ )

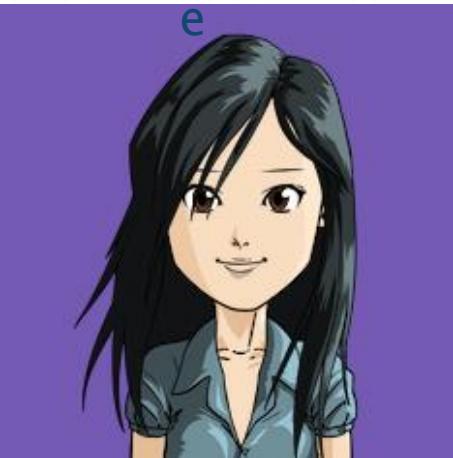
- Choisir  $p$  et  $q$ , deux nombres premiers distincts
- Calculer leur produit  $n = pq$  (**module de chiffrement**)
- Calculer  $\varphi(n) = (p - 1)(q - 1)$
- Choisir un nombre  $e$  premier avec  $\varphi(n)$  et strictement inférieur à ce nombre (**exposant de chiffrement**)
- Calculer l'entier naturel  $d$ , inverse de  $e$  modulo  $\varphi(n)$  (**exposant de déchiffrement**)
- On a  $K_{pub} = (e, n)$  et  $K_{priv} = d$

Q : Quel algorithme utilise t-on pour calculer  $d$ , l'inverse de  $e$  modulo  $\varphi(n)$  ?

# RIVEST-SHAMIR-ADLEMAN (RSA)

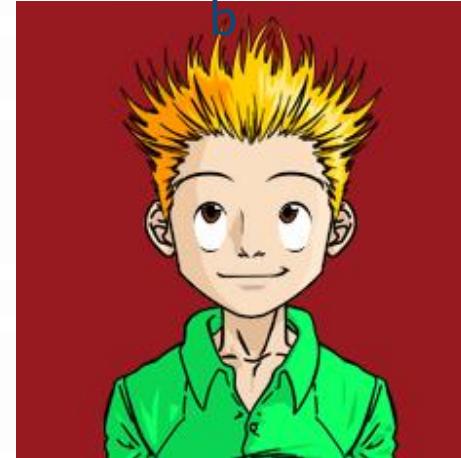
- L'algorithme de chiffrement/déchiffrement

Alic



$$C = M^{e_B} \pmod{n_B}$$

Bo



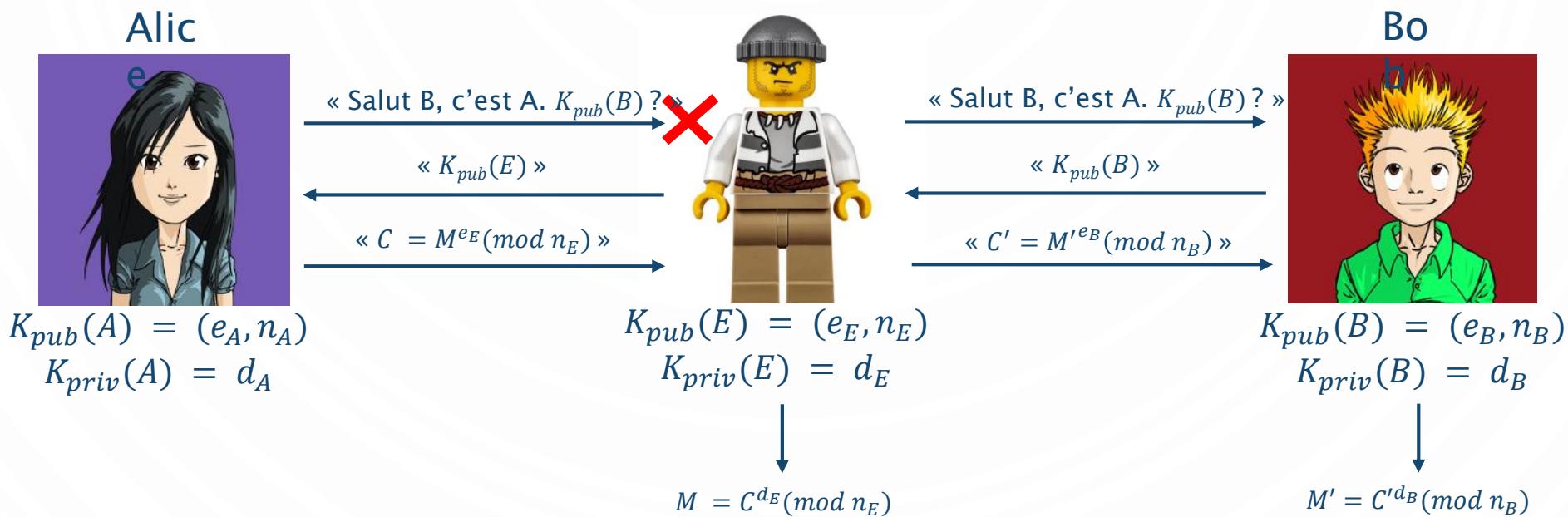
$$M = C^{d_B} \pmod{n_B}$$

$$M = (M^{e_B})^{d_B} \pmod{n_B}$$

Q : Comment calculer efficacement l'exponentiation modulaire ?

# RIVEST–SHAMIR–ADLEMAN (RSA)

#### ■ Attaque de « l'homme du milieu »



Par exemple,  $M$  = « Viens me chercher à la gare » et  $M'$  = « Viens me chercher au stade »

## Q : Comment remédier à ce problème ?

# CRYSTOSYSTÈME DE PAILLIER

- Alice choisit 2 nombres premiers distincts  $p$  et  $q$ , tels que  $pq$  et  $(p - 1)(q - 1)$  premiers entre eux.
- Elle calcule le module de chiffrement  $n = pq$  et  $\lambda = \text{ppcm}((p - 1), (q - 1))$ .
- Elle sélectionne un nombre  $g \in (\mathbb{Z}/n^2\mathbb{Z})^*$ , tel qu'il existe un nombre  $\mu$  égal à  $\left(L(g^\lambda \text{ mod } (n^2))\right)^{-1} \text{ mod } (n)$ , où  $L(x) = \frac{x-1}{n}$ , avec  $x \in \mathbb{N}^*$ .
- La clé publique est alors  $(n, g)$  et la clé privée est  $(\lambda, \mu)$ .
- Bob choisit un message  $m < n$  à envoyer à Alice. Il génère  $r \in (\mathbb{Z}/n\mathbb{Z})^*$  (non déterminisme).
- Il calcule  $c$  le chiffré de  $m$ , en utilisant la clé publique d'Alice  $(n, g)$  :  $c = g^m \cdot r^n \text{ mod } (n^2)$ .
- Alice déchiffre  $c$  avec sa clé privée  $(\lambda, \mu)$  et retrouve la valeur de  $m = L(c^\lambda \text{ mod } (n^2)) \cdot \mu \text{ mod } (n)$ .

# CHIFFREMENT HOMOMORPHE

- Un algorithme de chiffrement  $\mathcal{E}(\cdot)$  est dit homomorphe si lorsque les versions chiffrées de 2 messages en clair  $m_1$  et  $m_2$  sont connues, il est possible d'obtenir le chiffré d'une opération entre ces 2 messages :
$$\mathcal{E}(m_1 \Delta m_2) = \mathcal{E}(m_1) \boxdot \mathcal{E}(m_2)$$
- $\Delta$  et  $\boxdot$  peuvent désigner une addition, une soustraction ou une multiplication.
- Ils ne sont pas nécessairement les mêmes entre les messages en clair et leurs versions chiffrées.

# CHIFFREMENT HOMOMORPHE

- RSA est partiellement homomorphe vis-à-vis de la multiplication :

$$\begin{aligned}\mathcal{E}(m_1 \cdot m_2) &= (m_1 \cdot m_2)^e \bmod (n) \\ &= m_1^e \cdot m_2^e \bmod (n) \\ &= \mathcal{E}(m_1) \cdot \mathcal{E}(m_2)\end{aligned}$$

- Le cryptosystème de Paillier est un homomorphisme additif :

$$\begin{aligned}\mathcal{E}(m_1 + m_2) &= g^{m_1+m_2} \cdot (r_1 \cdot r_2)^n \bmod (n^2) \\ &= (g^{m_1} + r_1^n) \cdot (g^{m_2} + r_2^n) \bmod (n^2) \\ &= \mathcal{E}(m_1) \cdot \mathcal{E}(m_2)\end{aligned}$$

# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

## ■ Niveaux de sécurité visuelle

1. **Niveau transparent** : la haute résolution de l'image originale est préservée mais son contenu peut être pré-visualisé grâce à une version dégradée en clair,
2. **Niveau suffisant** : le contenu de l'image originale est protégé mais certaines formes et contours peuvent être distingués,
3. **Niveau confidentiel** : aucune information relative au contenu de l'image en clair ne peut être extraite de l'image chiffrée.



D. Engel, T. Stütz, et A. Uhl. *A survey on JPEG2000 encryption*. Multimedia Systems, 15(4) :243-270, 2009.

# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

- PSNR (Rapport Signal-Bruit, Peak Signal to Noise Ratio)

$$PSNR = 10 \cdot \log_{10} \frac{(2^l - 1)^2}{\frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (p(i,j) - p'(i,j))^2}$$

où  $p(i,j)$  est un pixel de l'image originale et  $p'(i,j)$  le pixel associé dans l'image chiffrée, toutes deux de même taille et dont les pixels sont codés sur  $2^l$  niveaux de gris. Le PSNR se mesure en décibels (dB).

# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

## ■ SSIM (Similarité structurelle, Structural Similarity)

$$SSIM(x, y) = \frac{(2E(x)E(y) + \gamma_1)(2Cov(x, y) + \gamma_2)}{(E(x)^2 + E(y)^2 + \gamma_1)(V(x)^2 + V(y)^2 + \gamma_2)}$$

avec  $x$  et  $y$  des fenêtres des 2 images,  $E(x)$  la moyenne de l'ensemble  $x$ ,  $V(x)$  sa variance,  $Cov(x, y)$  la covariance entre  $x$  et  $y$ ,  $\gamma_1 = (0,01 \times (2^l - 1))^2$  et  $\gamma_2 = (0,03 \times (2^l - 1))^2$ .



La fonction SSIM est appliquée à différentes fenêtres, puis on calcule la moyenne.  
*Image quality assessment: from error visibility to structural similarity.*  
IEEE Transactions on Image Processing, 13(4) :600-612, 2004.

# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

## ■ Test du $\chi^2$

Evaluation du caractère uniforme d'une distribution :

$$\chi^2 = 2^l \sum_{k=0}^{2^l-1} \left( P(\alpha_k) - \frac{1}{2^l} \right)^2$$

où les pixels de l'image sont codés sur  $2^l$  valeurs  $\alpha_k$  et  $P(\alpha_k)$  est la probabilité associée à  $\alpha_k$ .

# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

## ■ Coefficient de corrélation

Sélection de  $M$  paires de pixels voisins  $(x_i, y_i)$  dans les 3 directions, avec  $x_i \in x$  et  $y_i \in y$

$$corr_{x,y} = \frac{\frac{1}{M} \sum_{i=1}^M (x_i - E(x))(y_i - E(y))}{\sqrt{\frac{1}{M} \sum_{i=1}^M (x_i - E(x))^2} \sqrt{\frac{1}{M} \sum_{i=1}^M (y_i - E(y))^2}}$$

avec  $E(x)$  la moyenne de  $x$ .

# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

- NPCR (Taux de pixels modifiés, Number of Changing Pixel Rate)

$$NPCR = \frac{\sum_{i=0}^{m-1} \sum_{j=0}^{n-1} d(i,j)}{mn} \times 100,$$

où  $d(i,j)$  est défini par :

$$d(i,j) = \begin{cases} 1, & \text{si } p(i,j) \neq p'(i,j), \\ 0, & \text{sinon.} \end{cases}$$

- UACI (Moyenne unifiée des changements d'intensité, Unified Averaged Changed Intensity)

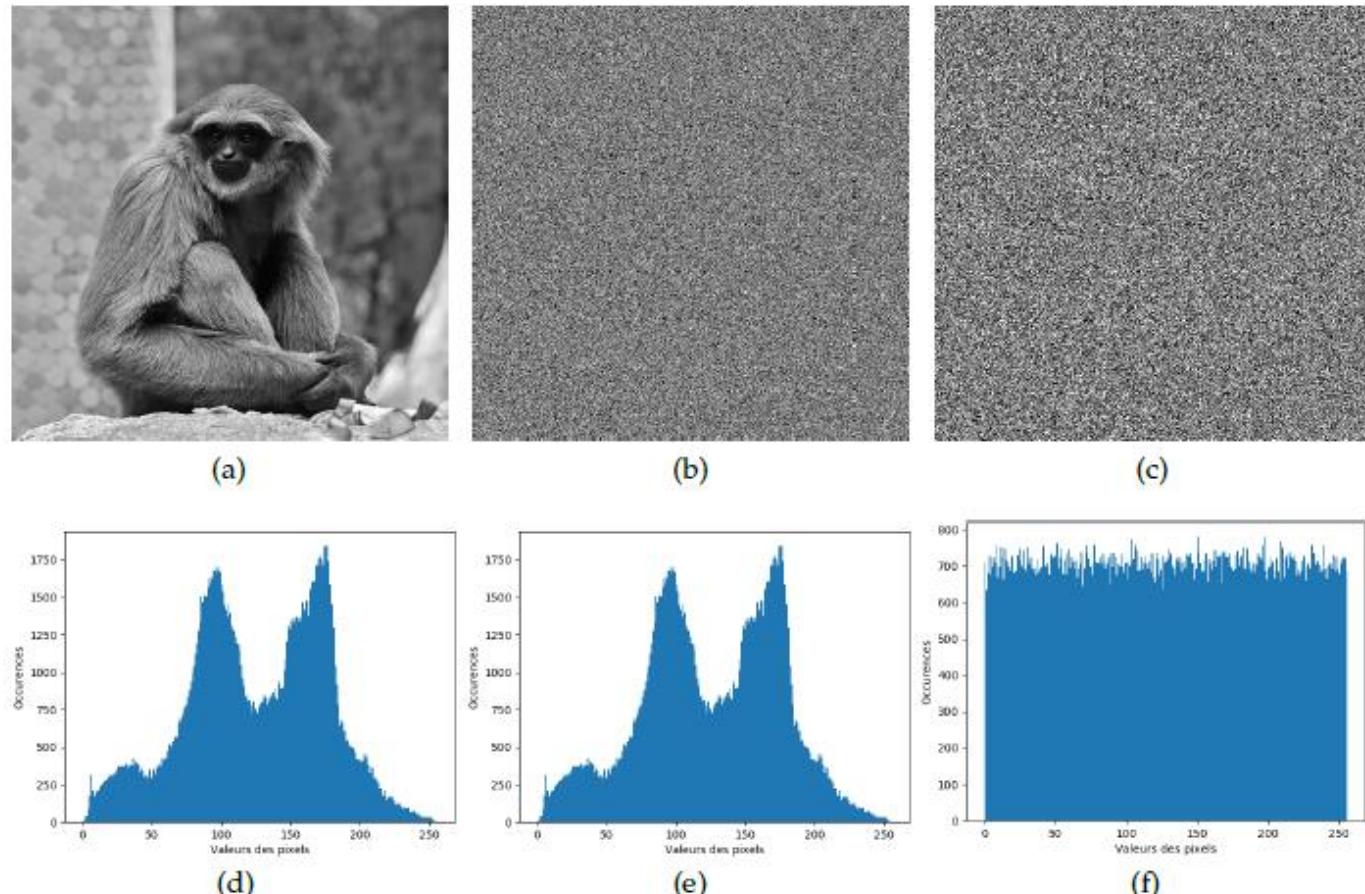
$$UACI = \frac{100}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \frac{|p(i,j) - p'(i,j)|}{2^l - 1}$$

Y. Wu, J. P. Noonan, et S. Agaian. *NPCR and UACI randomness tests for image encryption.*  
Cyber journals : Multidisciplinary Journals in Science and Technology,  
Journal of Selected Areas in Telecommunications (JSAT), 1(2) :31–38, 2011.

# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

## ■ Histogrammes

- a) Image originale
- b) Image chiffrée par mélange
- c) Image chiffrée par substitution
- d) Histogramme associé à (a)
- e) Histogramme associé à (b)
- f) Histogramme associé à (c)

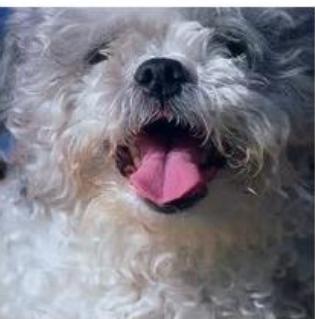


# ÉVALUATION DU NIVEAU DE SÉCURITÉ VISUELLE

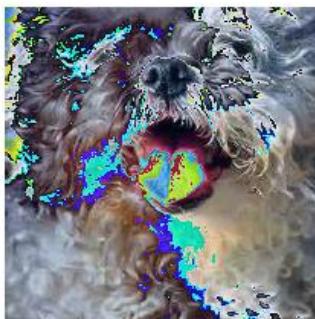
## ■ Reconnaissabilité

Si un modèle peut prédire la classe d'une image, son contenu est supposé reconnaissable :

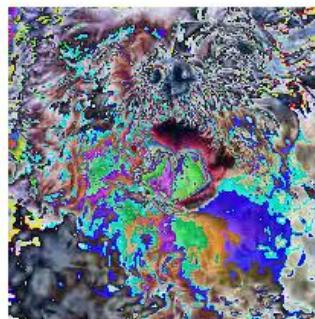
► **Indice de sécurité visuelle:**  $1 - \text{precision}_{\text{reconnaissabilite}}$



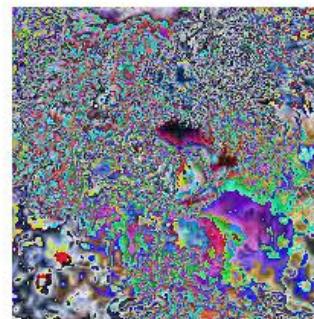
$s = 0$   
 $p_{ani} = 4,07$   
prédiction :  
animal



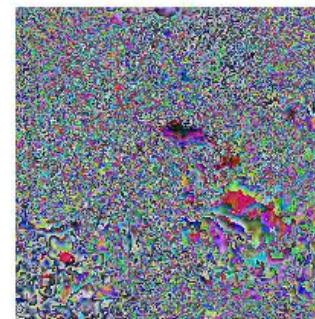
$s = 1$   
 $p_{ani} = 6,01$   
prédiction :  
animal



$s = 2$   
 $p_{ani} = 4,17$   
prédiction :  
animal



$s = 3$   
 $p_{ani} = 2,4$   
prédiction :  
animal



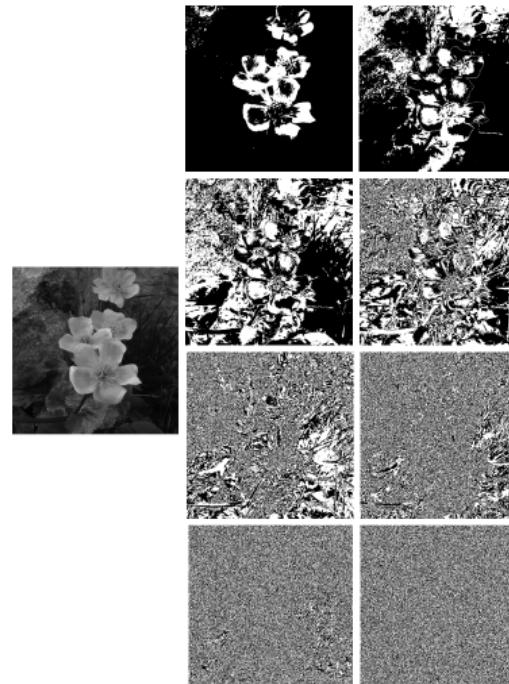
$s = 4$   
 $p_{ani} = 0,7$   
prédiction :  
plante

# CHIFFREMENT SÉLECTIF (IMAGES NON COMPRESSÉES)

## ■ Représentation d'une image

### Représentation classique

- ▶ Matrice deux dimensions
- ▶ Coefficients = pixels
- ▶ Codage sur 8 bits pour une image en niveaux de gris (valeurs de 0 à 255)



### Plan binaire

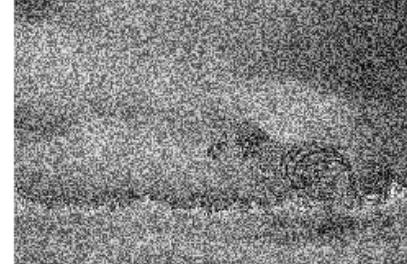
- ▶ Ensemble de bits à une position donnée dans chacun des pixels
- ▶ Bit le plus significatif : MSB
- ▶ Bit le moins significatif : LSB

# CHIFFREMENT SÉLECTIF (IMAGES NON COMPRESSÉES)

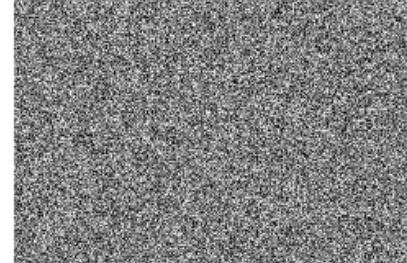
- ▶ Nombre  $s$  de plans binaires chiffrés
- ▶ Séquence de bits pseudo-aléatoires  $b^k(i, j)$ , avec  $0 \leq k < s$
- ▶ Bit chiffré  
 $p_c^k(i, j) = p^k(i, j) \oplus b^k(i, j)$ , avec  $p^k(i, j)$  le bit en clair



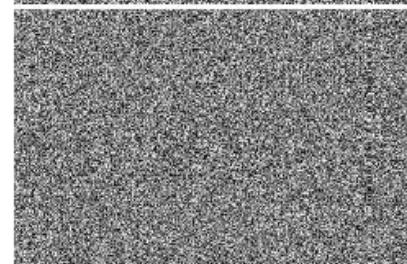
$s = 0$



$s = 1$



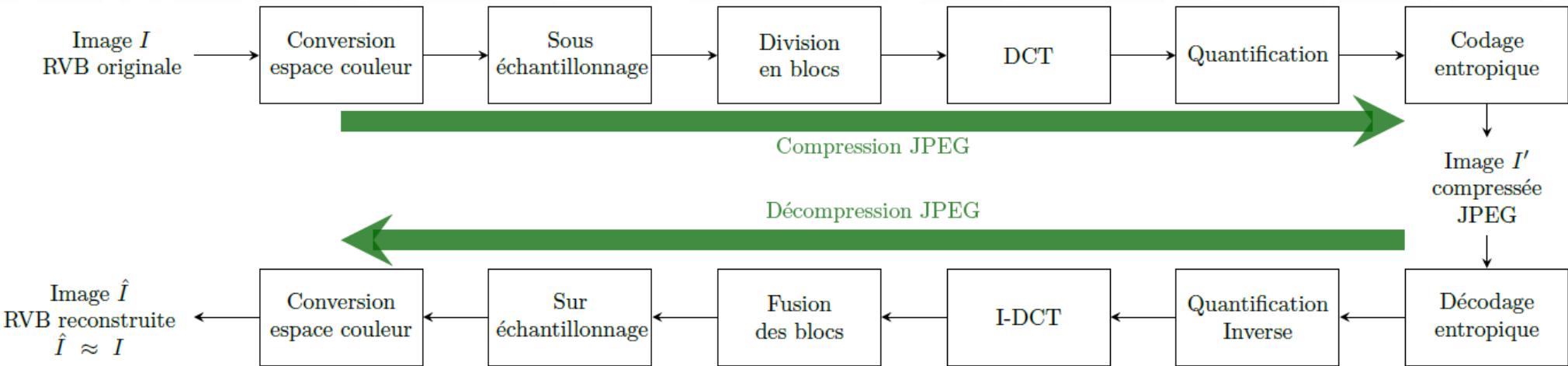
$s = 4$



$s = 6$

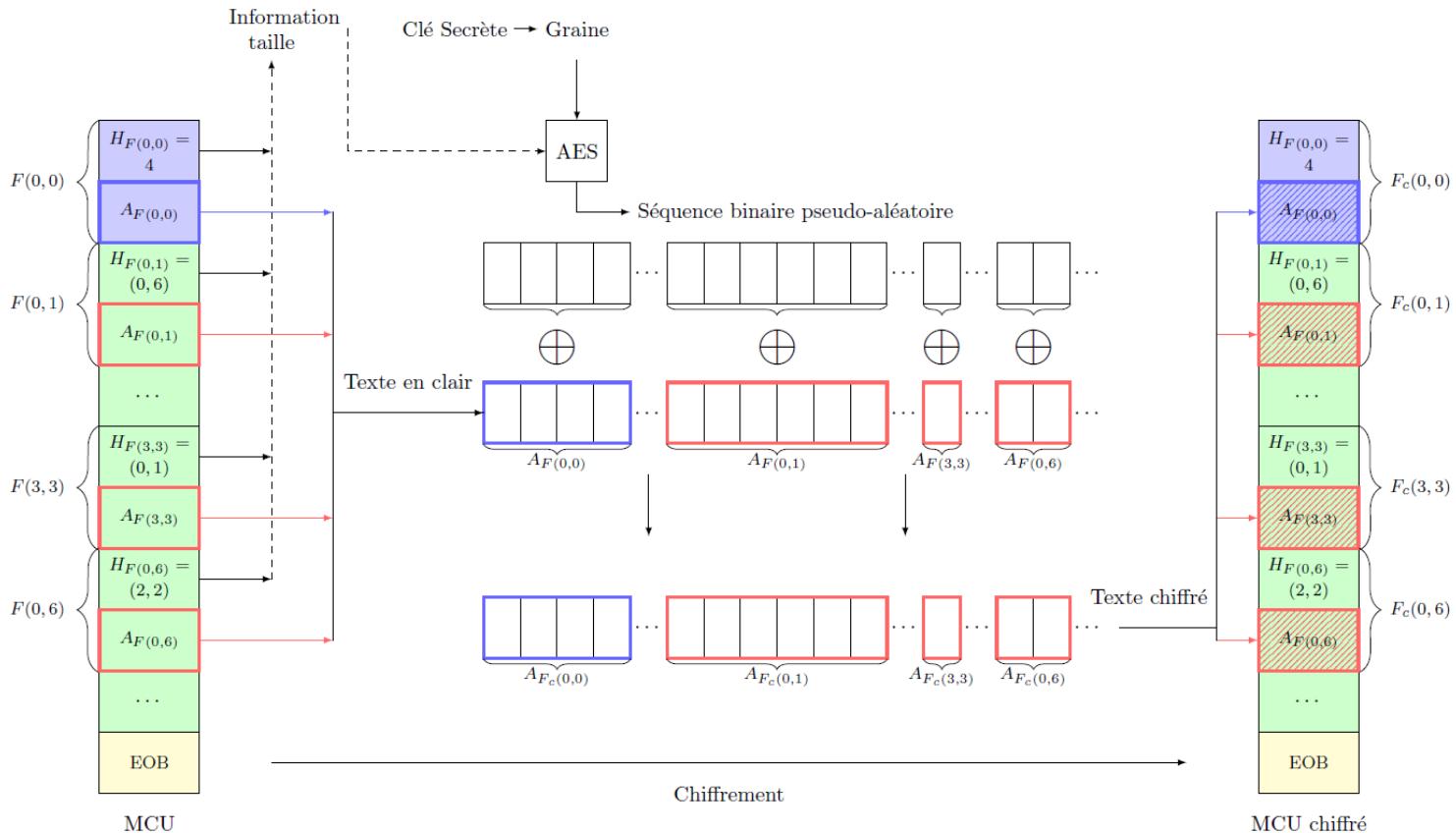
# CHIFFREMENT SÉLECTIF (IMAGES COMPRESSÉES)

## ■ Compression JPEG



# CHIFFREMENT SÉLECTIF (IMAGES COMPRESSÉES)

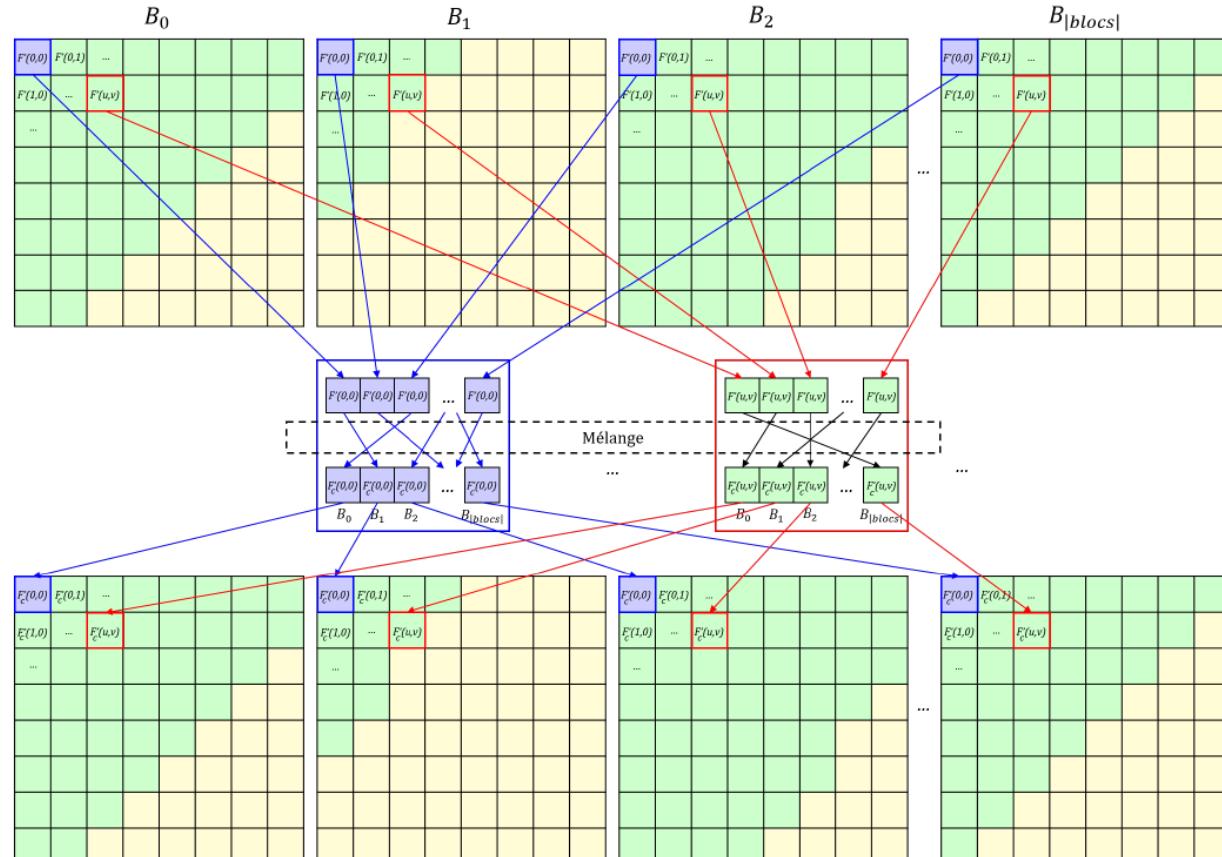
- Chiffrement par substitution



W. Puech et J. M. Rodrigues. *Crypto-compression of medical images by selective encryption of DCT*. European Signal Processing Conference (EUSIPCO), pages 1–4, 2005.

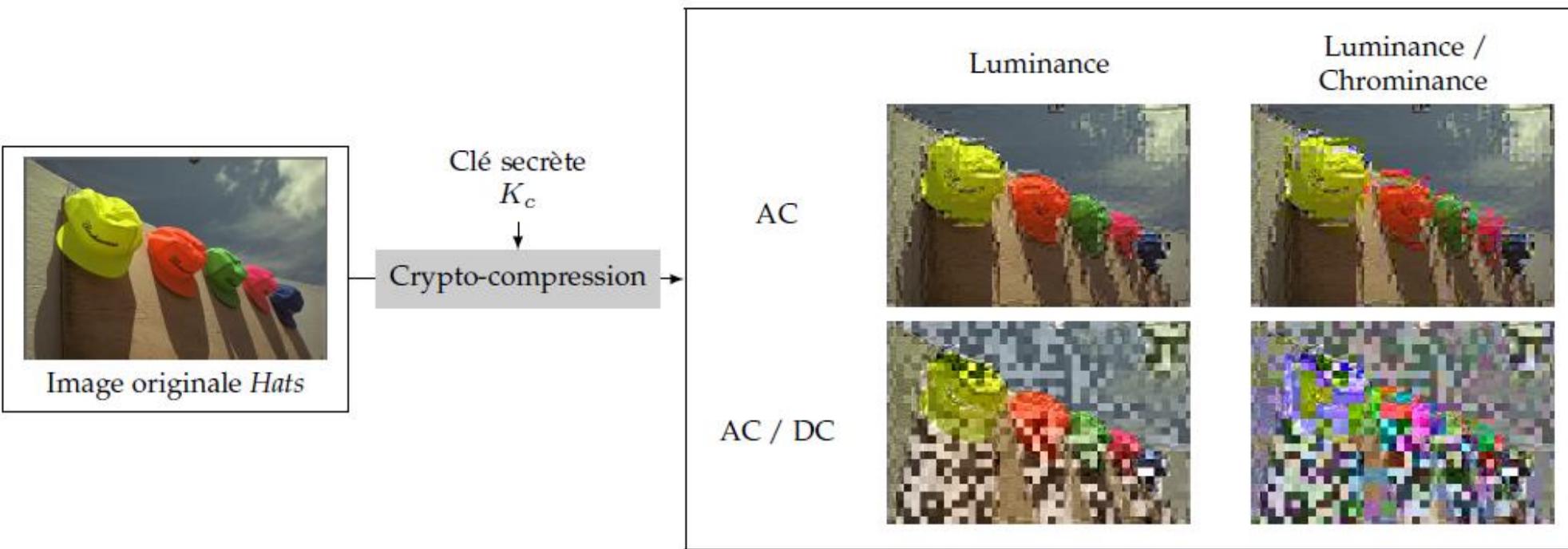
# CHIFFREMENT SÉLECTIF (IMAGES COMPRESSÉES)

- Chiffrement par permutation



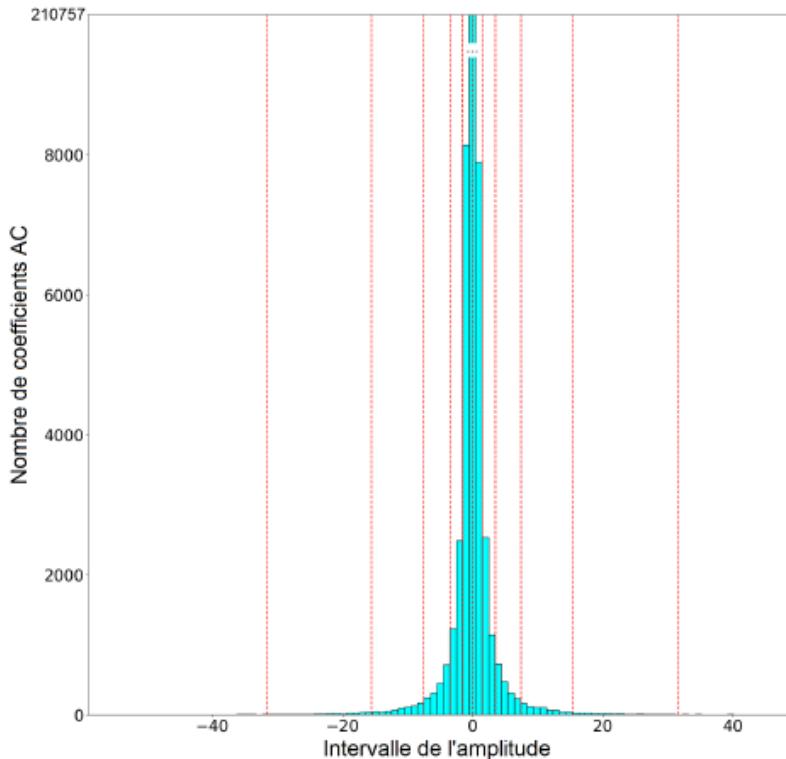
W. Li et Y. Yuan. *A leak and its remedy in JPEG image encryption.*  
International Journal of Computer Mathematics, 84(9) :1367–1378, 2007.

# CHIFFREMENT SÉLECTIF (IMAGES COMPRESSÉES)

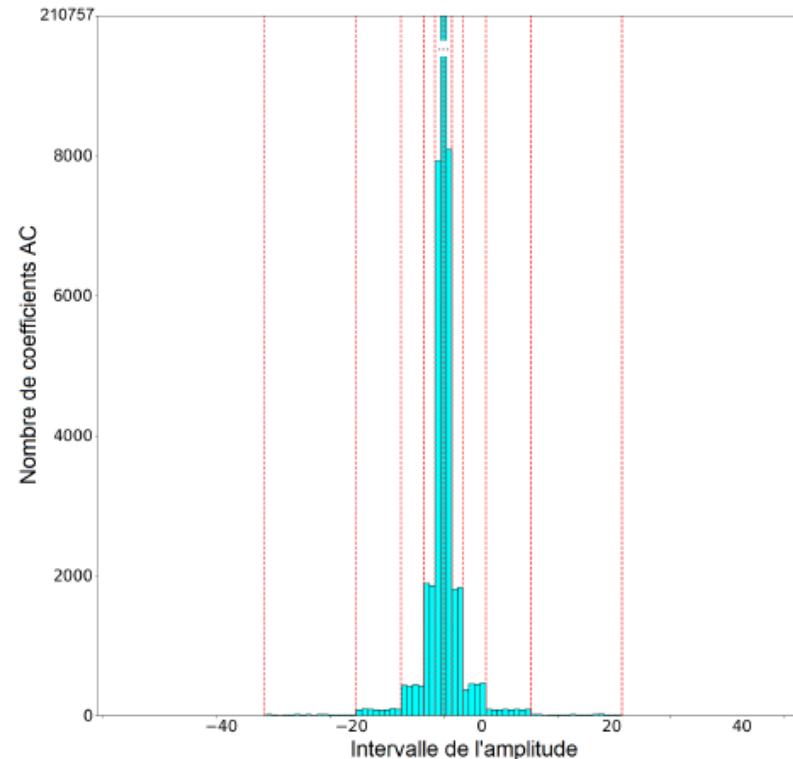


# CHIFFREMENT SÉLECTIF (IMAGES COMPRESSÉES)

## Distributions des coefficients AC



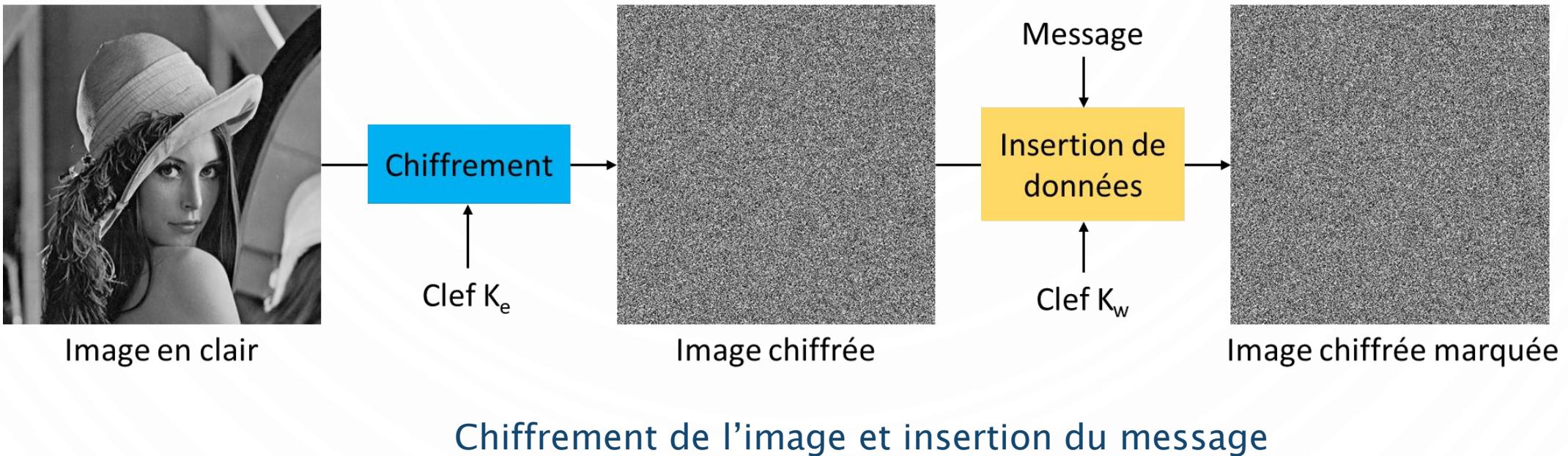
(a) Avant le chiffrement.



(b) Après le chiffrement.

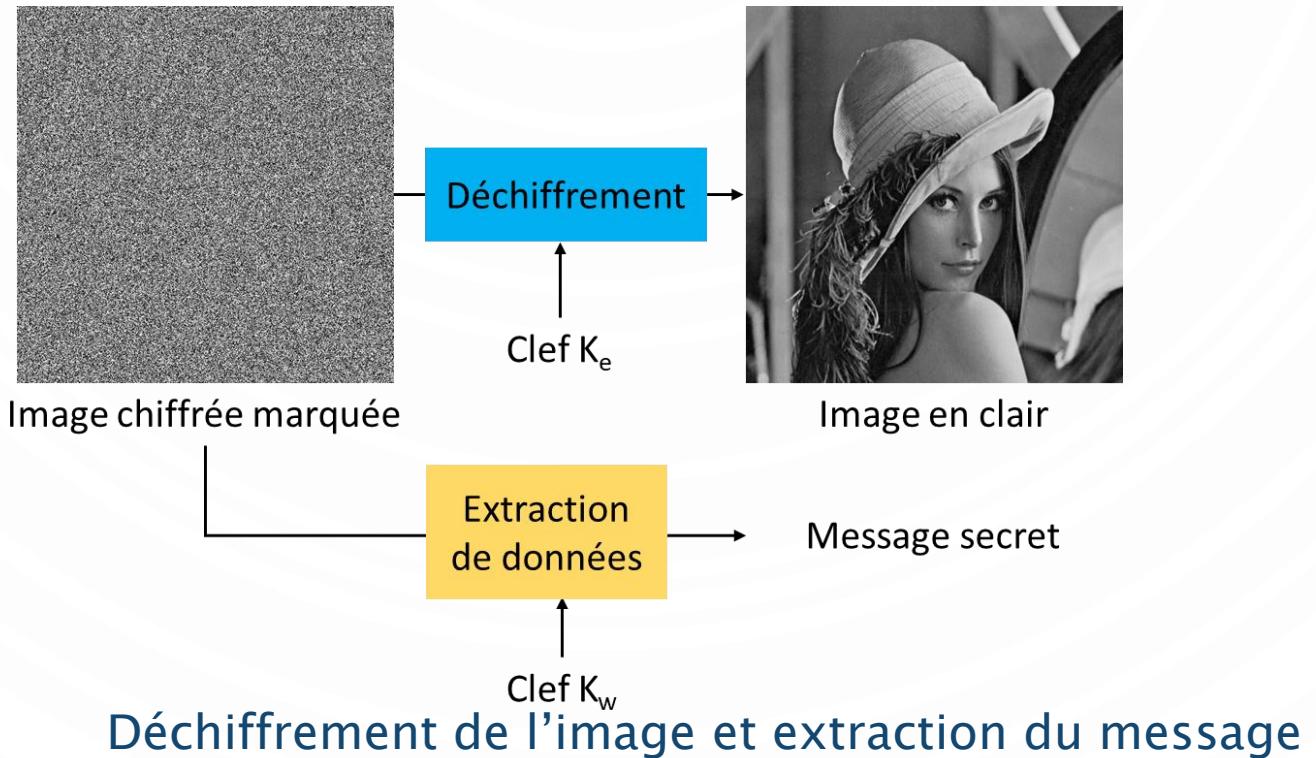
# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

## ■ Insertion de données cachées dans les images chiffrées



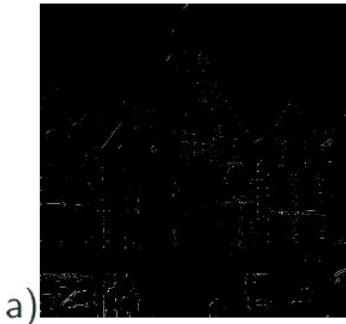
# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

- Insertion de données cachées dans les images chiffrées

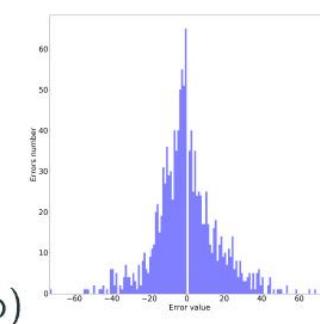


# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

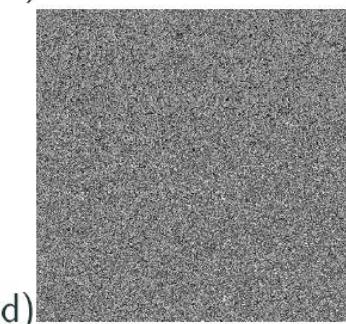
## ■ Insertion de données cachées dans les images chiffrées



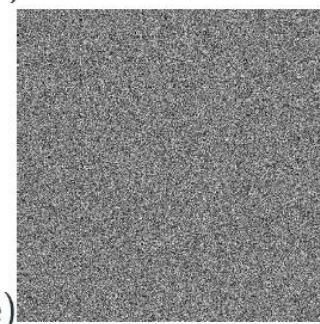
a)



b)



d)



e)



c)

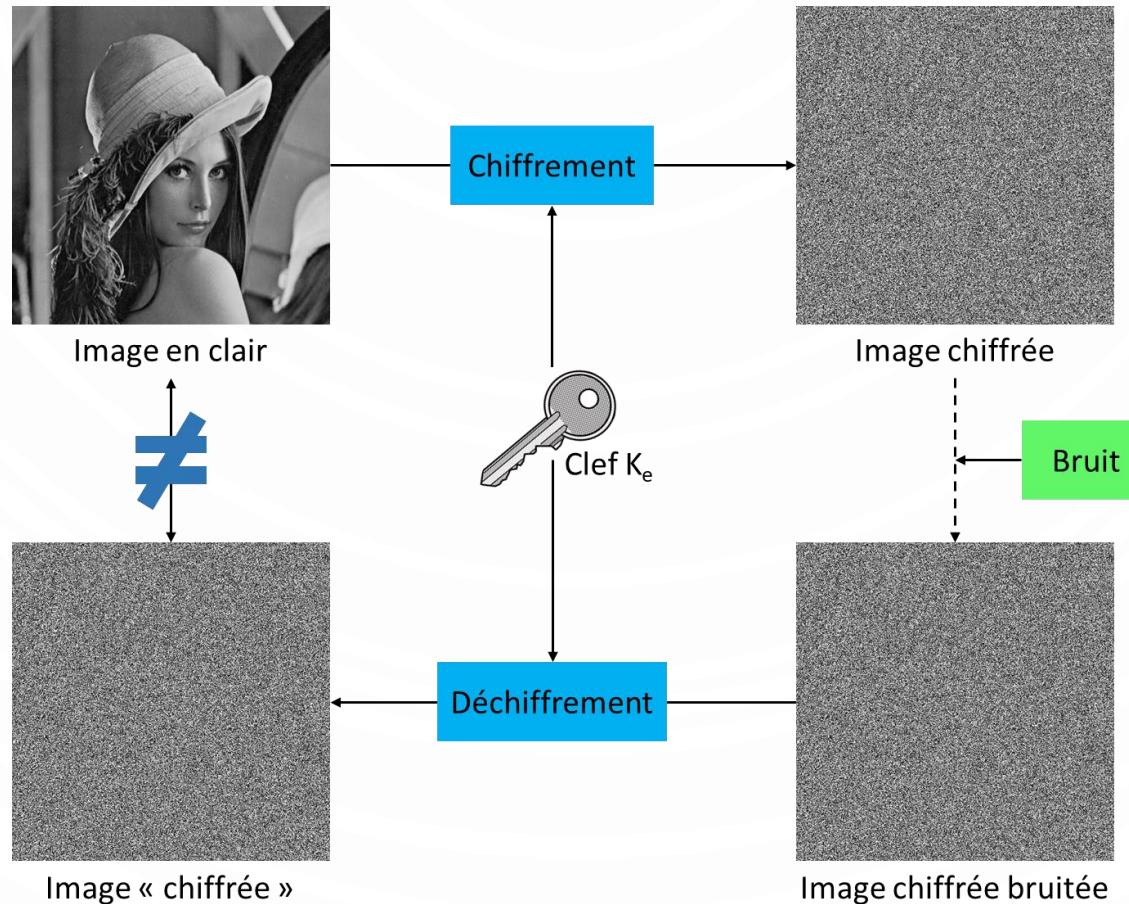


f)

Exemple détaillé pour l'approche  $1 \text{ bpp}$  : a) Emplacement des erreurs de prédiction, nombre d'erreurs = 1242 (0,47%), b) Histogramme des erreurs de prédiction, c) Image pré-traitée, PSNR = 46,87 dB, d) Image chiffrée, e) Image chiffrée marquée, capacité d'insertion = 1 bpp, f) Image reconstruite, PSNR = 46,87 dB, SSIM = 0,99.

# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

## ■ Correction d'images chiffrées bruitées



# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

- Partage de secret visuel



Image  
originale

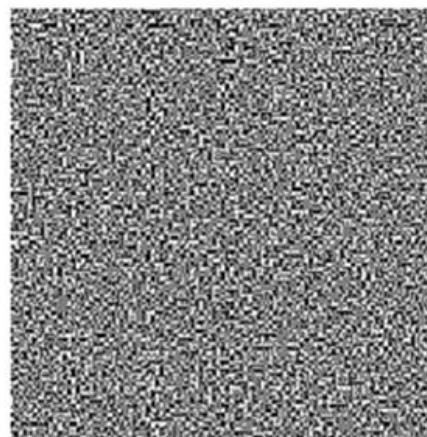


Image share  
1

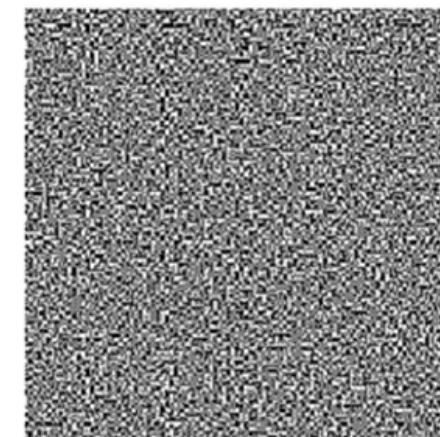


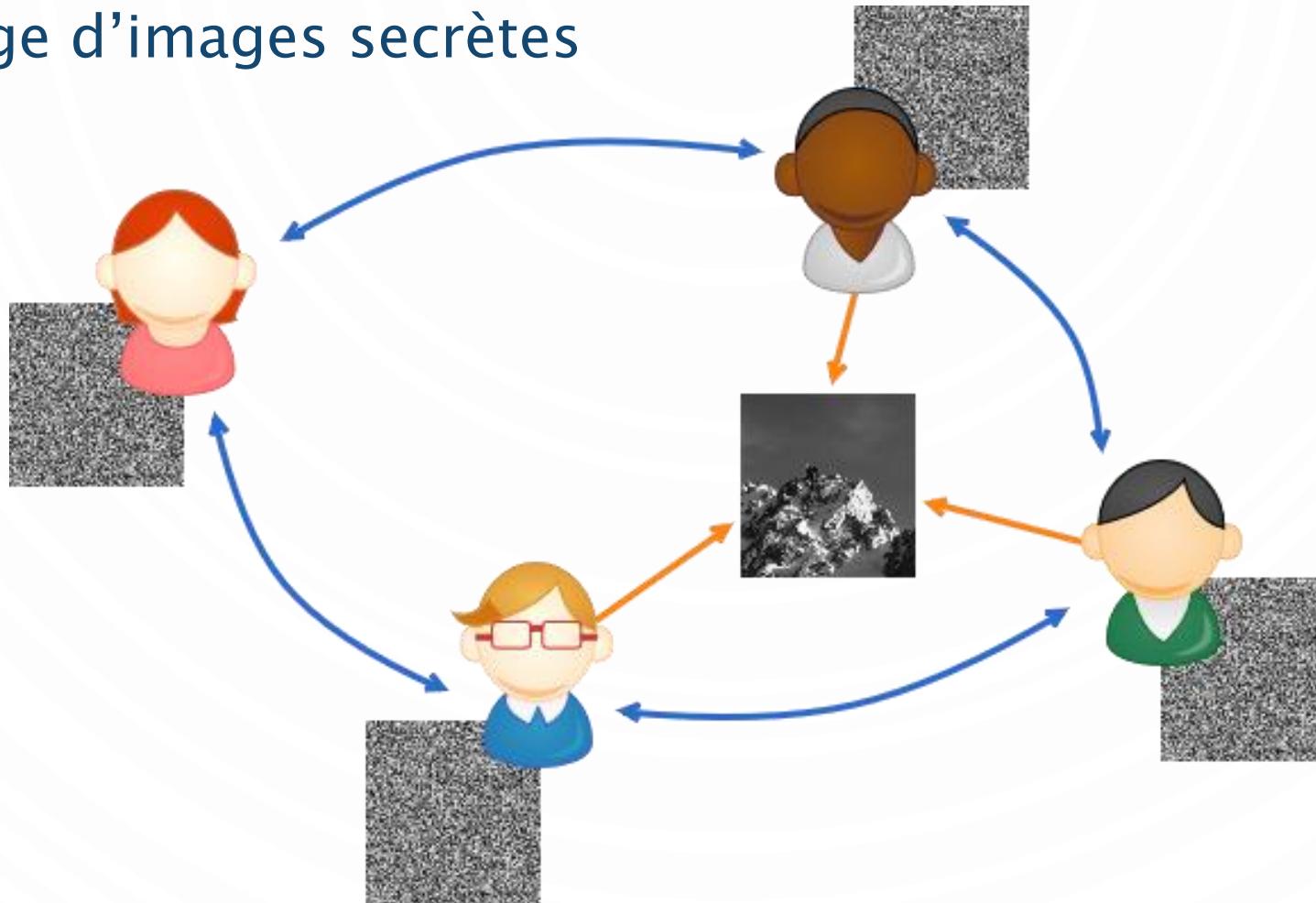
Image  
share 2



Image  
reconstruite

# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

- Partage d'images secrètes



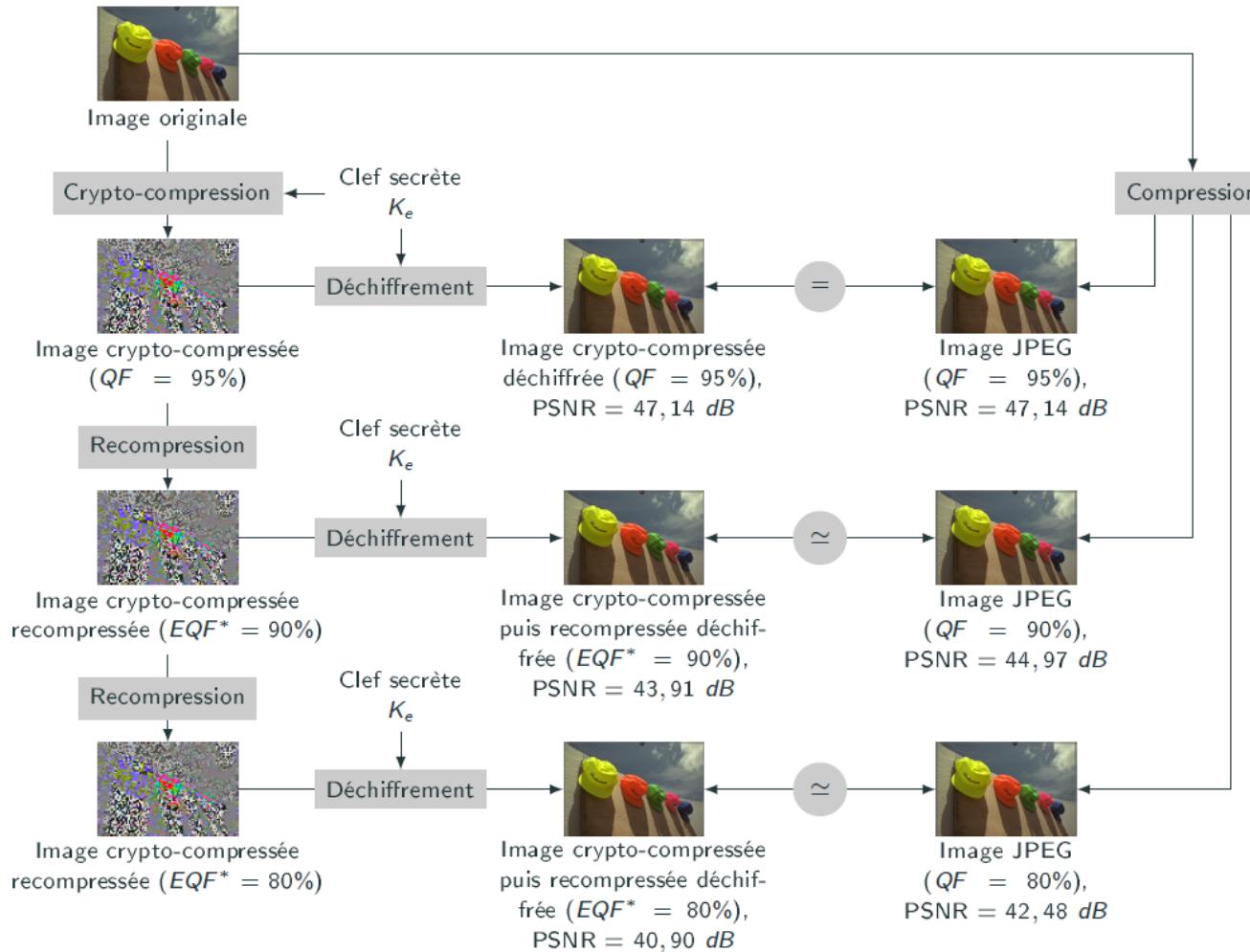
# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

## ■ Partage d'images secrètes



# TRAITEMENTS DANS LES IMAGES CHIFFRÉES

## ■ Recompression d'images JPEG crypto-compressées



MERCI POUR VOTRE ATTENTION !