Introduction à la Cryptographie
Cryptographie symétrique

Claire Delaplace & Léo Robert

Organisation du cours

Contenu:

- 2 séances de cours, 4 TD/TP
- Tout matériel sur MesCoursJV

Modalités de contrôle:

- Examen à la dernière séance : 1H30 avec document A4 recto/verso autorisé.

⚠ Les notions vues en TD/TP seront aussi évaluées.

Qu'est-ce que la cryptographie?

Qu'est-ce que la cryptographie?

CRYPTOGRAPHIE

Kruptos: caché

Graphein: écrire

Qu'est-ce que la cryptographie?

CRYPTOGRAPHIE

Kruptos: caché

Graphein: écrire

Cryptographie: écrire des messages secret.

Cryptologie vs Cryptographie

CRYPTOLOGIE

Science des « **messages secrets** »

Cryptologie vs Cryptographie

CRYPTOLOGIE

Science des « **messages secrets** »

CRYPTOGRAPHIE

Construction de mécanismes pour protéger les communications

Cryptologie vs Cryptographie

CRYPTOLOGIE

Science des « **messages secrets** »

CRYPTOGRAPHIE

Construction de mécanismes pour protéger les communications

CRYPTANALYSE

Attaquer les schémas cryptographiques

Cryptologie vs Cryptographie

CRYPTOLOGIE

Science des « **messages secrets** »

CRYPTOGRAPHIE

Construction de mécanismes pour protéger les communications

CRYPTANALYSE

Attaquer les schémas cryptographiques

Cryptologie vs Cryptographie

CRYPTOLOGIE

Science des « **messages secrets** »

CRYPTOGRAPHIE

Construction de mécanismes pour protéger les communications

CRYPTANALYSE

Attaquer les schémas cryptographiques

La cryptologie est une branche de la **Théorie de l'Information**

Théorie de l'Information

- Formalisée par Claude Shannon en 1948.

Théorie de l'Information

- Formalisée par Claude Shannon en 1948.
- Permet de quantifier l'information.

Théorie de l'Information

- Formalisée par Claude Shannon en 1948.
- Permet de quantifier l'information.
- Divisée en plusieurs branches:
  - Représentation de l'information (Codage)
  - Stockage de l'information (compression de données)
  - Transfère d'information (correction d'erreur)
  - Sécurisation de l'information (cryptologie)

La cryptographie, ça sert à quoi?

Assurer...

La cryptographie, ça sert à quoi?

Assurer...

- la confidentialité : l'information n'est accessible qu'à ceux dont l'accès est autorisé.
- l'authenticité: vérifier l'identité de la personne/machine lors de la communication.
- L'integrité: les données ne subissent pas d'altération.

Un peu de vocabulaire...

Dans le contexte de la cryptographie, on ne dit pas...

- coder un message.
  Coder où encoder un message est une façon de représenter l'information, pas de la sécuriser. Un code n'est généralement pas secret.

Un peu de vocabulaire...

Dans le contexte de la cryptographie, on ne dit pas...

- coder un message.
  Coder où encoder un message est une façon de représenter l'information, pas de la sécuriser. Un code n'est généralement pas secret.
- crypter un message.
  Ce terme ne veut rien dire.

Un peu de vocabulaire...

Dans le contexte de la cryptographie, on ne dit pas...

- coder un message.
  Coder où encoder un message est une façon de représenter l'information, pas de la sécuriser. Un code n'est généralement pas secret.
- crypter un message.
  Ce terme ne veut rien dire.

On dit...

- chiffrer un message.

Un peu de vocabulaire...

Chiffrer:

- Qui ? l'émetteur d'un message
- Quoi ? Transformer un message clair en un « message secret »

Un peu de vocabulaire...

Chiffrer:

- Qui ? l'émetteur d'un message
- Quoi ? Transformer un message clair en un « message secret »

Déchiffrer:

- Qui ? le destinataire du message
- Quoi ? Retrouver le clair à partir du chiffré et d'un secret appelé « clé ».

Un peu de vocabulaire...

Chiffrer:

- Qui ? l'émetteur d'un message
- Quoi ? Transformer un message clair en un « message secret »

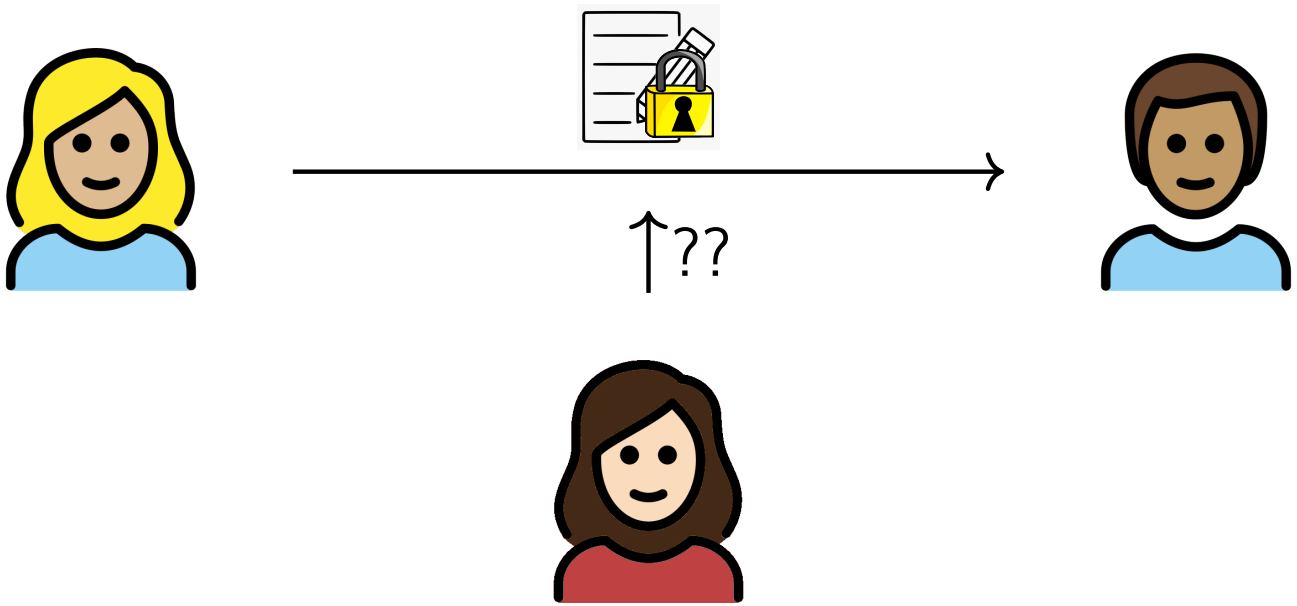
Déchiffrer:

- Qui ? le destinataire du message
- Quoi ? Retrouver le clair à partir du chiffré et d'un secret appelé « clé ».

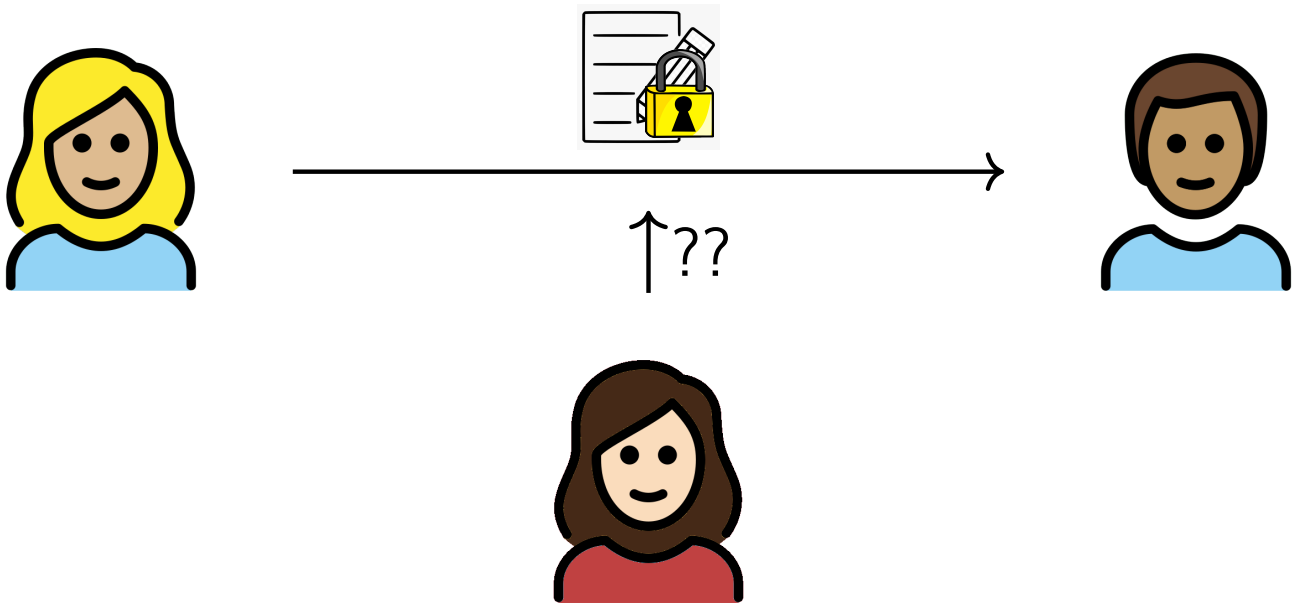
Décrypter:

- Qui ? adversaire
- Quoi ? Retrouver le clair par des moyens détourné: « casser »

Confidentialité

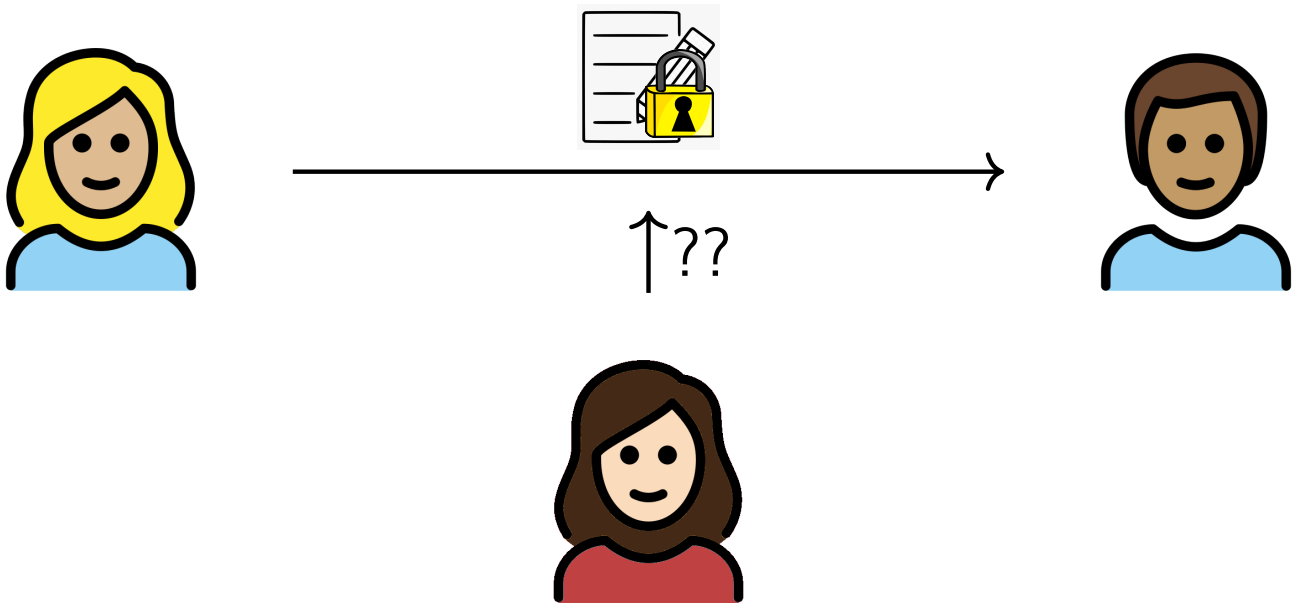


Confidentialité



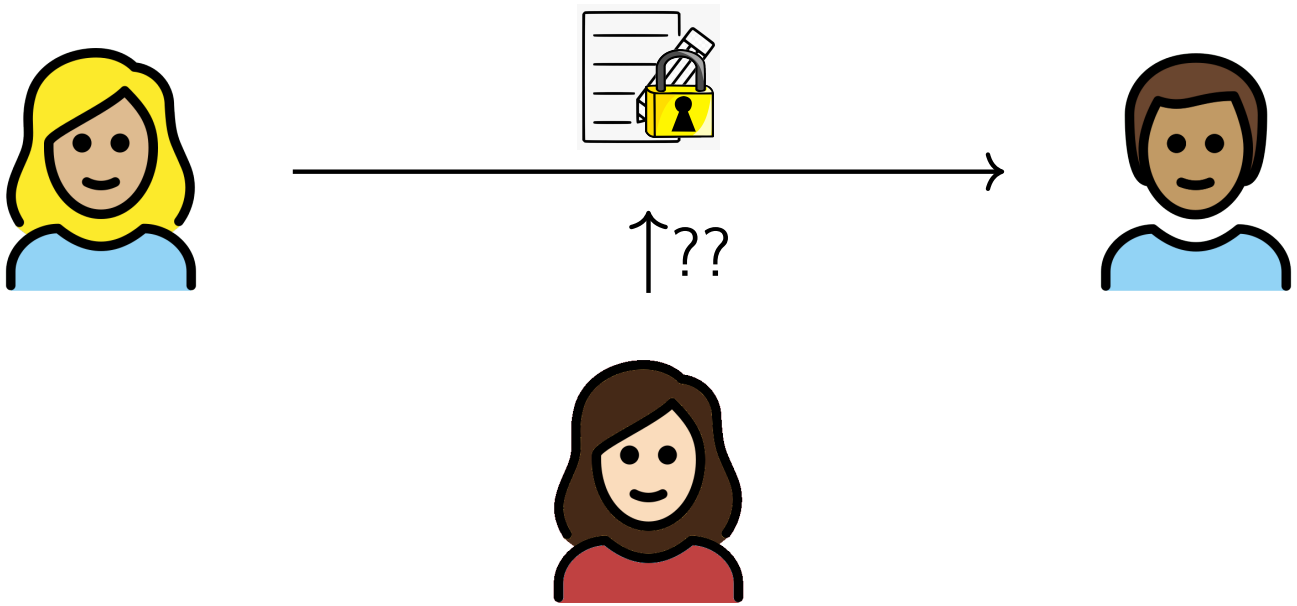
- Alice chiffre un message qu'elle envoie à Bob

Confidentialité



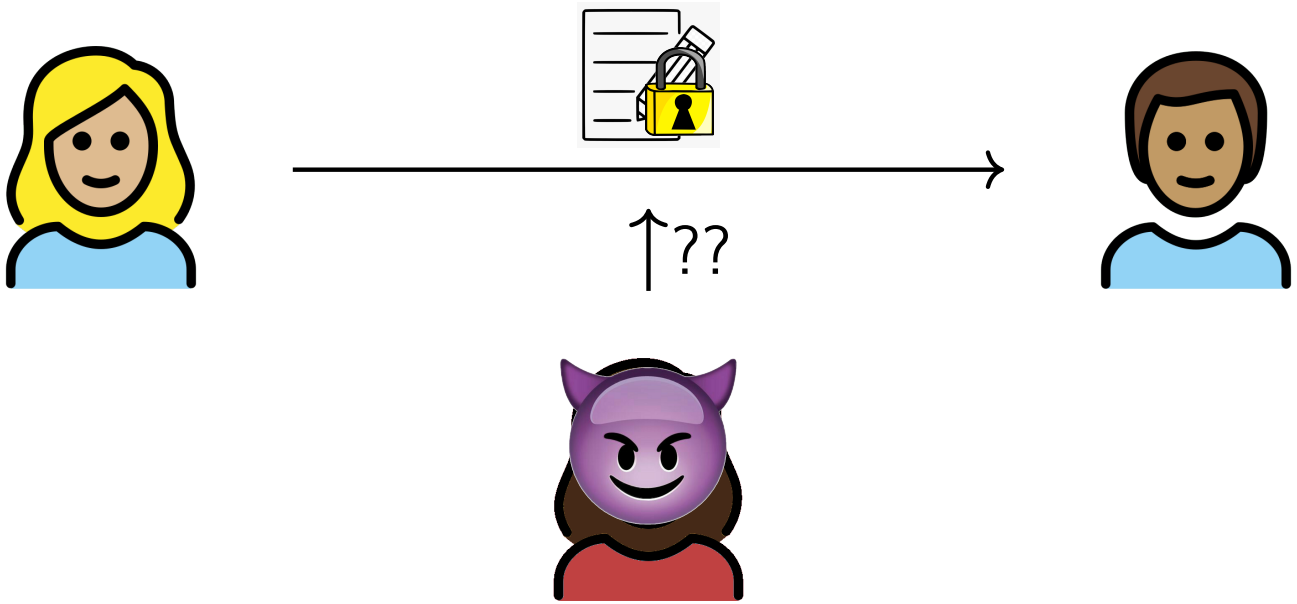
- Alice chiffre un message qu'elle envoie à Bob
- Bob déchiffre le message

Confidentialité



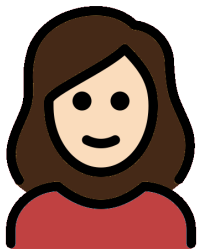
- Alice chiffre un message qu'elle envoie à Bob
- Bob déchiffre le message
- Eve ne peut pas déchiffrer le message

Confidentialité



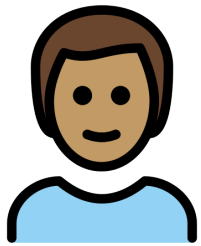
- Alice chiffre un message qu'elle envoie à Bob
- Bob déchiffre le message
- Eve ne peut pas déchiffrer le message
- Eve ne peut pas décrypter le message (confidentialité)

Authenticité

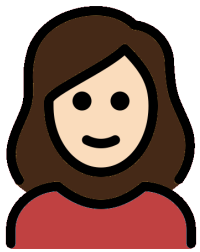


Je suis Alice!

Non, c'est faux!

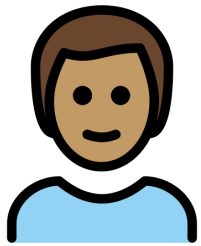


Authenticité



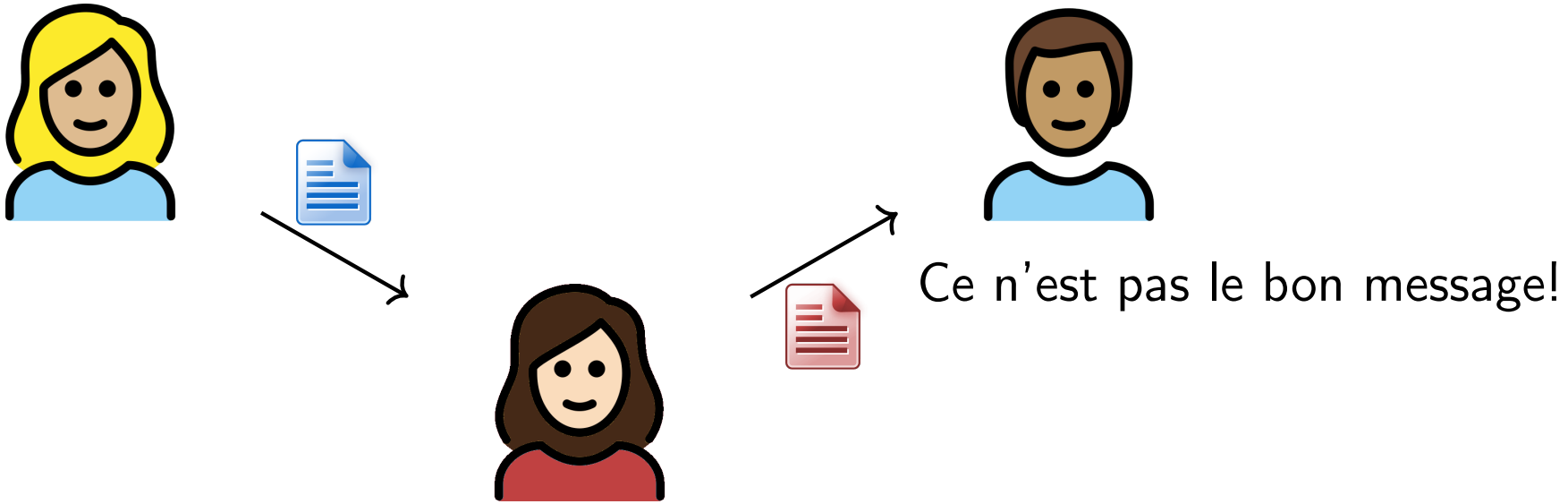
Je suis Alice!

Non, c'est faux!

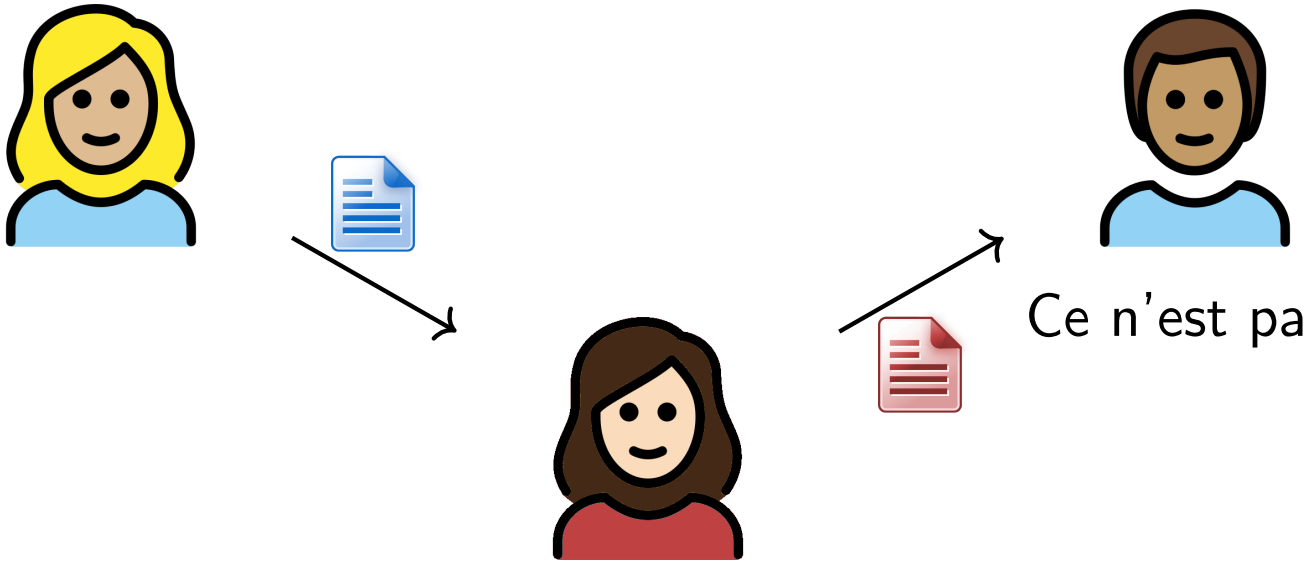


Eve ne peut pas usurper l'identité d'Alice.

Intégrité



Intégrité



Eve ne peut pas modifier les messages échangés.

Attention! Cryptographie $\neq$ Stéganographie

- Les deux: rendre une information illisible à une tierce personne

Attention! Cryptographie $\neq$ Stéganographie

- Les deux: rendre une information illisible à une tierce personne
- Stéganographie: Cacher un message avec une méthode secrète.

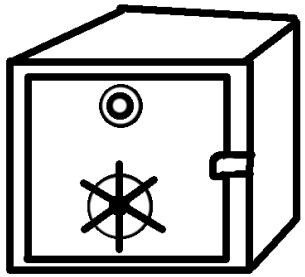
Attention! Cryptographie $\neq$ Stéganographie

- Les deux: rendre une information illisible à une tierce personne
- Stéganographie: Cacher un message avec une méthode secrète.
- Cryptographie: Chiffrer un message avec une méthode connue et un élément secret appelé clé.

Attention! Cryptographie ≠ Stéganographie

- Les deux: rendre une information illisible à une tierce personne
- Stéganographie: Cacher un message avec une méthode secrète.
- Cryptographie: Chiffrer un message avec une méthode connue et un élément secret appelé clé.

Stéganographie



Cryptographie

Principe d'Auguste Kerchoff von Nieuwenhof 1883

"*La sécurité d'un chiffrement doit totalement dépendre du secret de la clé et non du secret de l'algorithme utilisé.*"

Principe d'Auguste Kerchoff von Nieuwenhof 1883

"*La sécurité d'un chiffrement doit totalement dépendre du secret de la clé et non du secret de l'algorithme utilisé.*"

1. Plus facile de garder secret une petite clé que tout un algorithme.

Principe d'Auguste Kerchoff von Nieuwenhof 1883

"*La sécurité d'un chiffrement doit totalement dépendre du secret de la clé et non du secret de l'algorithme utilisé.*"

1. Plus facile de garder secret une petite clé que tout un algorithme.
2. Si k n'est plus secret, il est plus simple de changer la clé que tout un algorithme.

Principe d'Auguste Kerchoff von Nieuwenhof 1883

"*La sécurité d'un chiffrement doit totalement dépendre du secret de la clé et non du secret de l'algorithme utilisé.*"

1. Plus facile de garder secret une petite clé que tout un algorithme.
2. Si k n'est plus secret, il est plus simple de changer la clé que tout un algorithme.

Aujourd'hui, les algorithmes sont *publiques*:

- Beaucoup de personnes vont regarder/améliorer les algos.
- Il vaut mieux révéler les failles de sécurité par des *"ethical hackers"* que par des entités malicieuses.
- Le code d'un algo peut être cassé (reverse engineering). La clé (nombre aléatoire) ne fait pas partie du code.
- Un design publique rend possible l'établissement de *standard*.

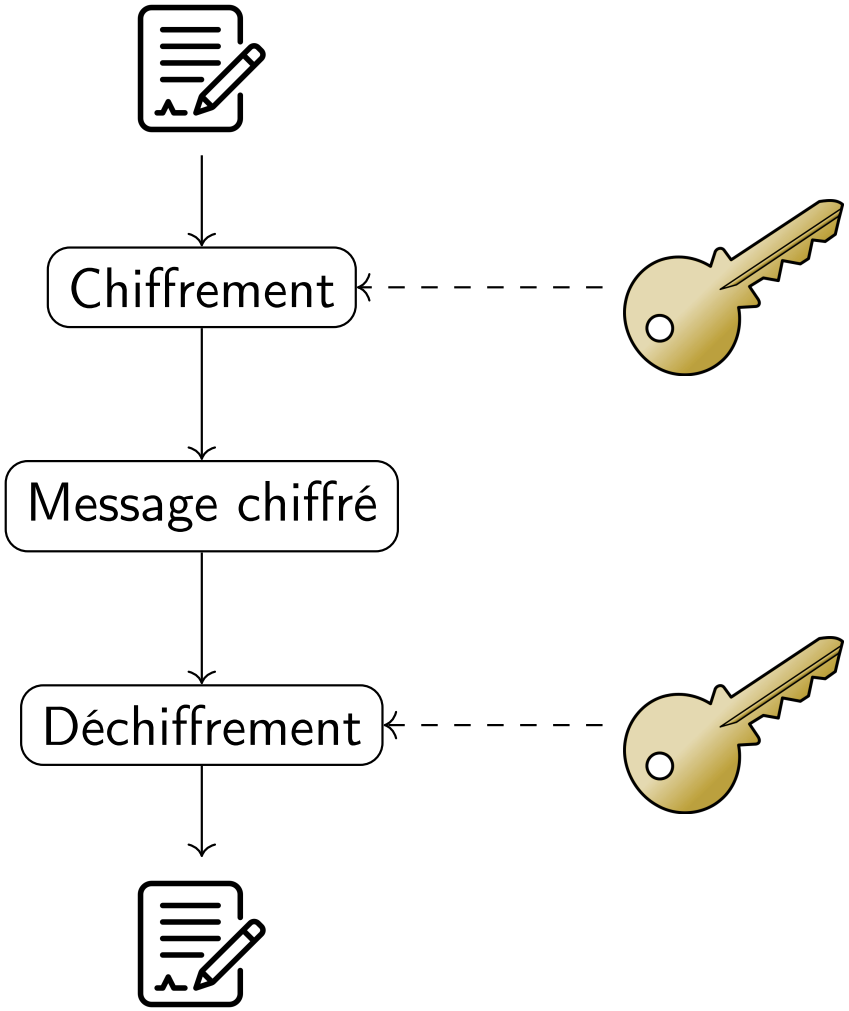
Exemples : A5/0 et A5/1 pour GSM, DVD CSS (1996), I-message.

Le chiffrement symétrique

Principe :

- Une clé secrète unique est partagée entre l'émetteur et le récepteur.
- La même clé sert à :
  - chiffrer le message
  - déchiffrer le message
- Exemples : AES, DES, ChaCha20.

Avantage : rapide et efficace.
Problème : échange sécurisé de la clé.



One-time pad (Vernam 1917) ou masque jetable

Propriétés du XOR ($\oplus$)

- $x \oplus y = y \oplus x$ — Commutativité
- $x \oplus (y \oplus z) = (x \oplus y) \oplus z$ — Associativité
- $x \oplus x = 0$ — Nilpotence
- $x \oplus 0 = x$ — Élément neutre

- La clé k doit être de la même taille que le message m à chiffrer
- Le chiffré c de m par k vaut $c = m \oplus k$
- Le déchiffrément $m = c \oplus k$

Preuve :

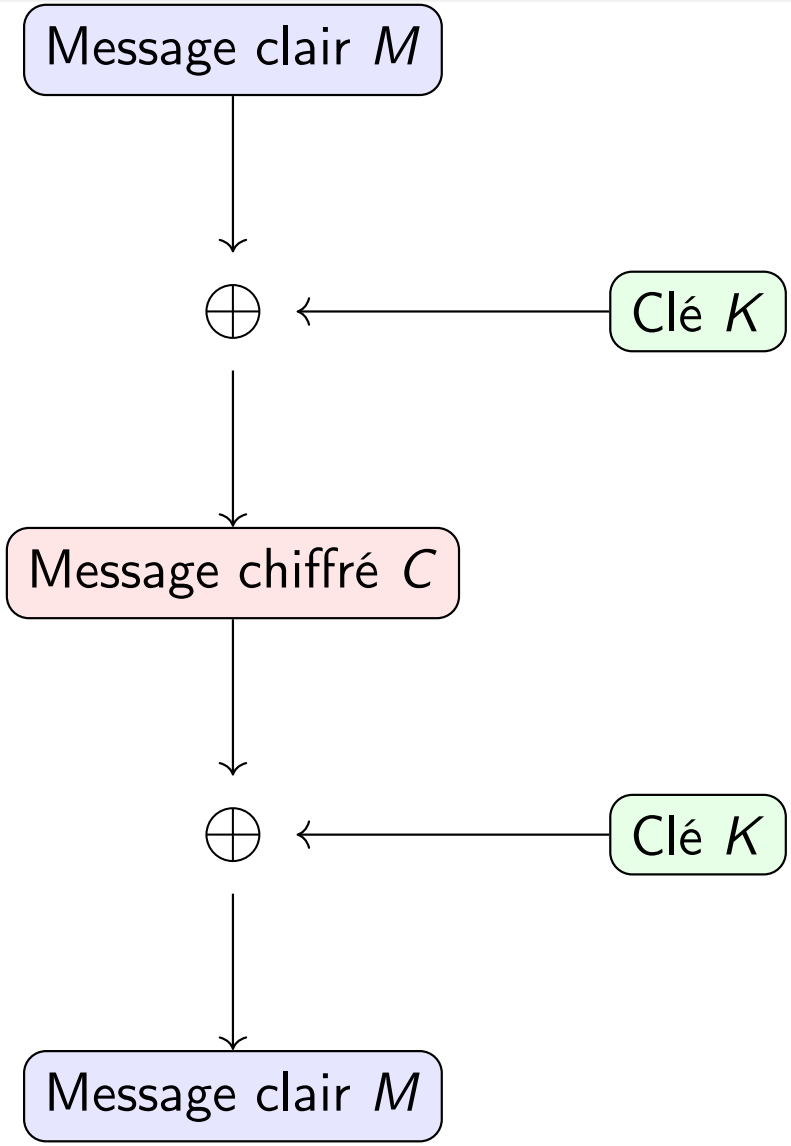
$c \oplus k = (m \oplus k) \oplus k = m \oplus (k \oplus k) = m \oplus 0 = m$

Chiffrement avec une sécurité inconditionelle (C. Shannon)

One-Time Pad (Vernam 1917)

Exemple:

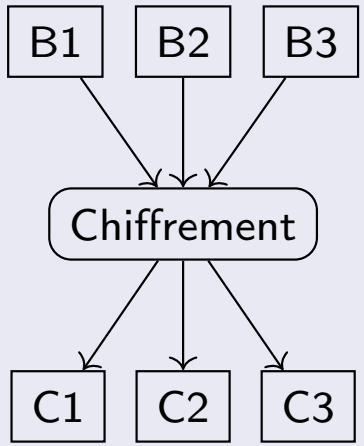
$$\begin{array}{rcl} m & = & 010111 \\ \oplus k & = & 110010 \\ \hline c & = & 100101 \end{array}$$



Deux types de chiffrement symétrique

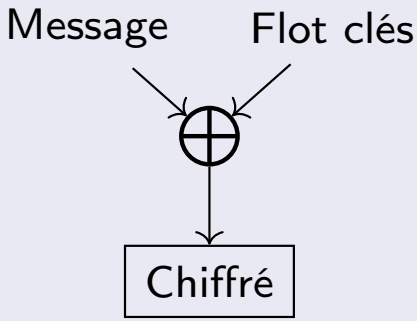
Block ciphers (chiffrement par blocs)

- Le message est découpé en blocs de taille fixe (ex : 128 bits).
- Chaque bloc est chiffré avec la clé.
- Nécessite un mode d'opération (CBC, CTR, GCM...).
- **Exemples :** AES, DES.



Stream ciphers (chiffrement par flot)

- Génèrent un flot de clés pseudo-aléatoire.
- Chiffrement bit par bit ou octet par octet.
- Fonctionnent comme un OTP pratique.
- **Exemples :** ChaCha20, RC4.



Advanced Encryption Standard (AES)

- Conçu par deux cryptographes belges Joan Daemen et Vincent Rijmen
- Block de 128 bits, clé de 128, 192, ou 256 bits.

Taille de la clé	Nombre de tours
128	10
192	12
256	14

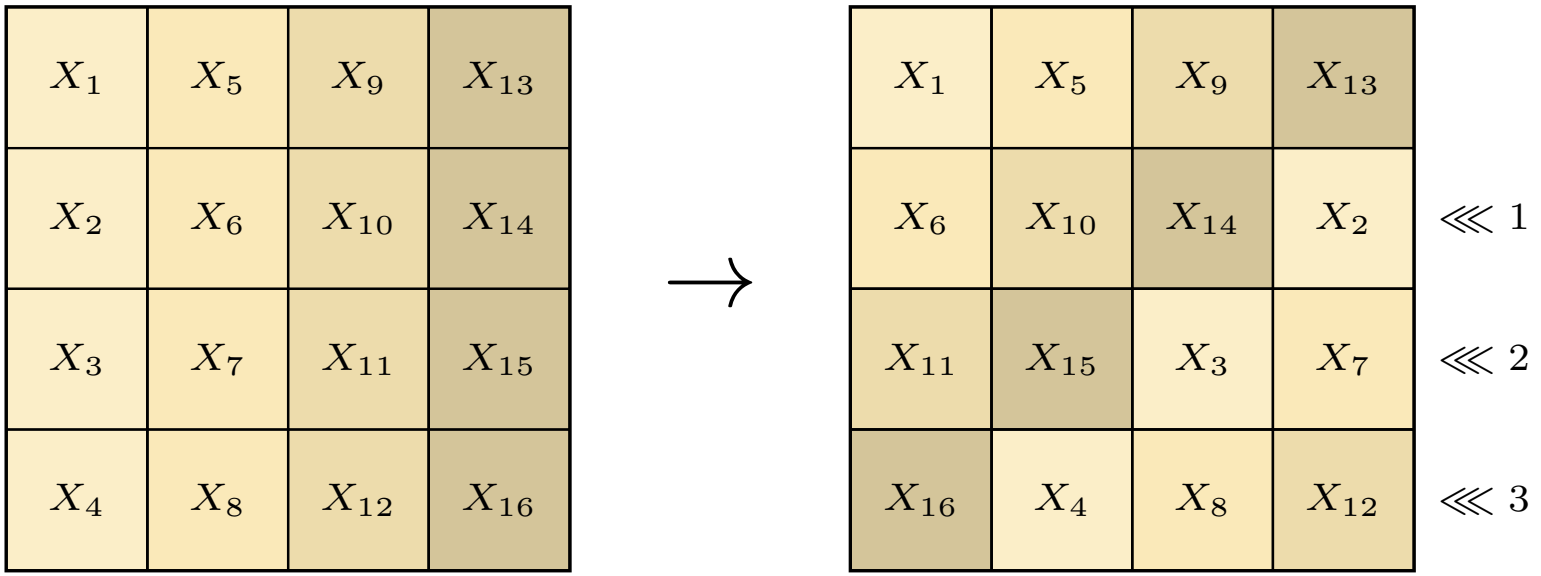
AES:Chiffrement

- Dérivation de la clé
- Tour initial : AddRoundKey
- Tours :
  1. SubBytes
  2. ShiftRows
  3. MixColumns
  4. AddRoundKey
- Tour final :
  1. SubBytes
  2. ShiftRows
  3. AddRoundKey

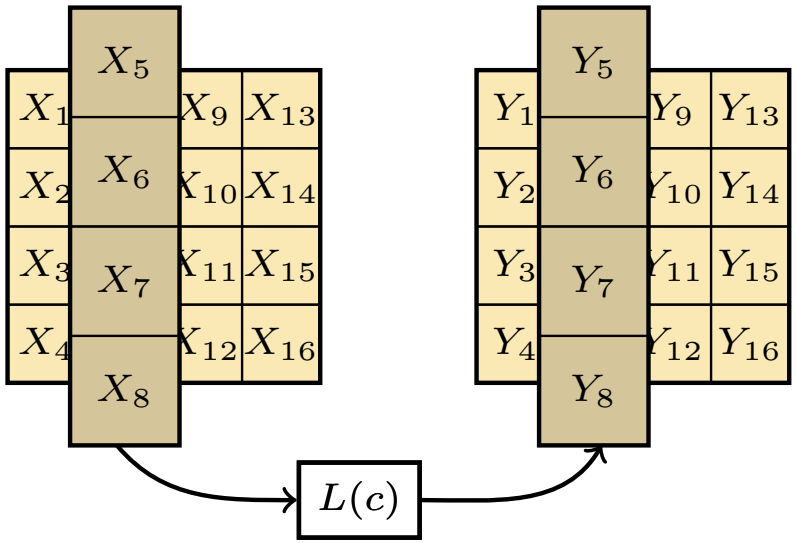
AES: SubBytes

$S(X_1)$	$S(X_5)$	$S(X_9)$	$S(X_{13})$
$S(X_2)$	$S(X_6)$	$S(X_{10})$	$S(X_{14})$
$S(X_3)$	$S(X_7)$	$S(X_{11})$	$S(X_{15})$
$S(X_4)$	$S(X_8)$	$S(X_{12})$	$S(X_{16})$

AES: ShiftRows



AES: MixColumns



L une fonctinon linéaire.

AES: AddRoundKey

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}

$K_1^{(i)}$	$K_5^{(i)}$	$K_9^{(i)}$	$K_{13}^{(i)}$
$K_2^{(i)}$	$K_6^{(i)}$	$K_{10}^{(i)}$	$K_{14}^{(i)}$
$K_3^{(i)}$	$K_7^{(i)}$	$K_{11}^{(i)}$	$K_{15}^{(i)}$
$K_4^{(i)}$	$K_8^{(i)}$	$K_{12}^{(i)}$	$K_{16}^{(i)}$

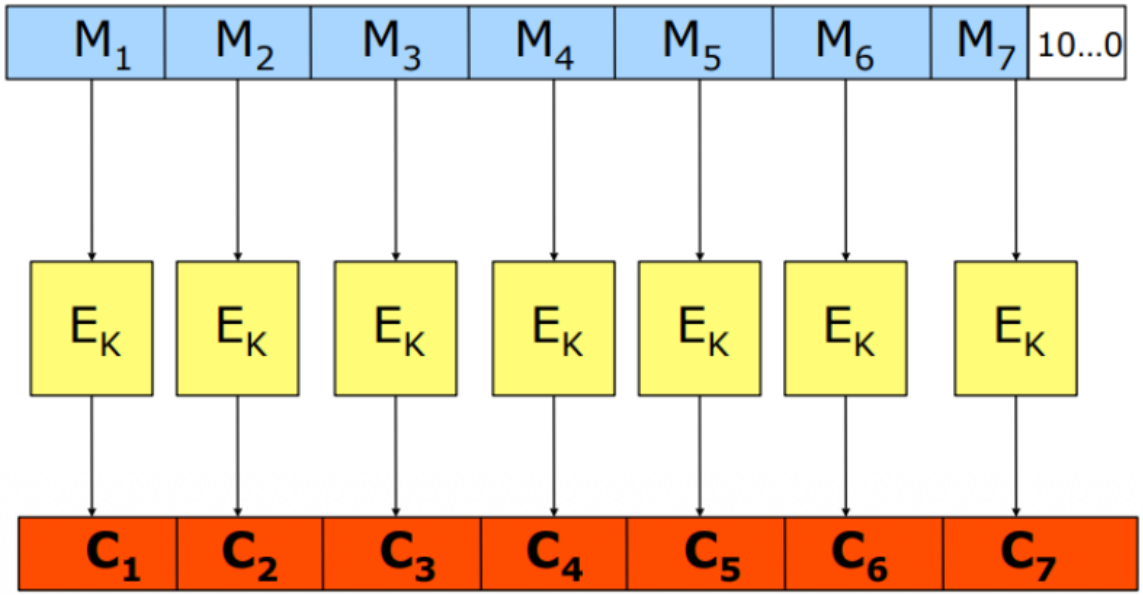
D'autres chiffrements symétriques

Blowfish, Serpent, Twofish, 3-Way, ABC, Akelarre, Anubis, ARIA, BaseKing, BassOmatic, BATON, BEAR and LION, C2, Camellia, CAST-128, CAST-256, CIKS-1, CIPHERUNICORN-A, CIPHERUNICORN-E, CLEFIA, CMEA, Cobra, COCONUT98, Crab, CRYPTON, CS-Cipher, DEAL, DES-X, DFC, E2, FEAL, FEA-M, FROG, G-DES, GOST, Grand Cru, Hasty Pudding Cipher, Hierocrypt, ICE, IDEA, IDEA NXT, Intel Cascade Cipher, Iraqi, KASUMI, KeeLoq, KHAZAD, Khufu and Khafre, KN-Cipher, Ladder-DES, Libelle, LOKI97, LOKI89/91, Lucifer, M6, M8, MacGuffin, Madryga, MAGENTA, MARS, Mercy, MESH, MISTY1, MMB, MULTI2, MultiSwap, New Data Seal, NewDES, Nimbus, NOEKEON, NUSH, Q, RC2, RC5, RC6, REDOC, Red Pike, S-1, SAFER, SAVILLE, SC2000, SEED, SHACAL, SHARK, Skipjack, SMS4, Spectr-H64, Square, SXAL/MBAL, TEA, Treyfer, UES, Xenon, xmx, XTEA, XXTEA, Zodiac.

Modes

- Pour chiffrer un message de taille arbitraire, nous devons le découper en blocs de taille fixe
- Les blocs sont chiffrés un par un
- Le *mode opératoire* est l'algorithme qui utilise le chiffrement par bloc pour chiffrer un message de taille quelconque.

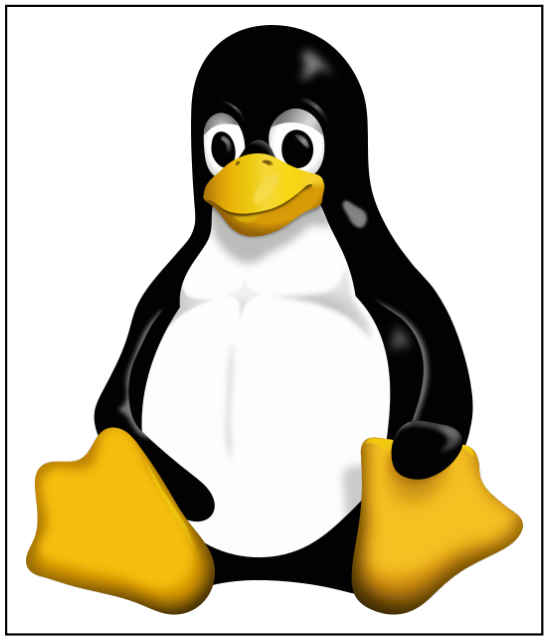
Electronic Code Book (ECB)

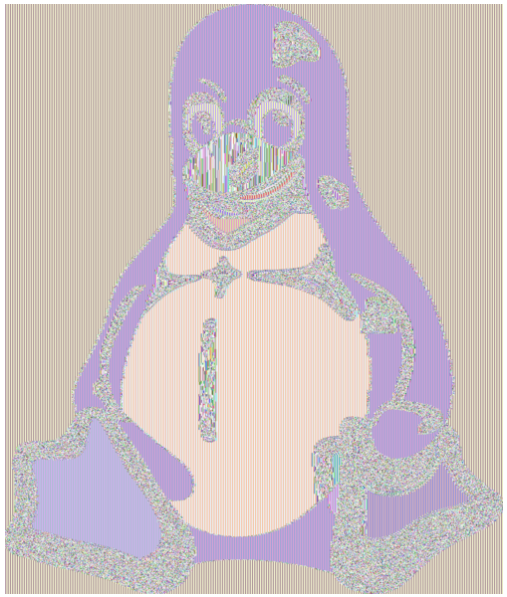


Bad ECB, bad!

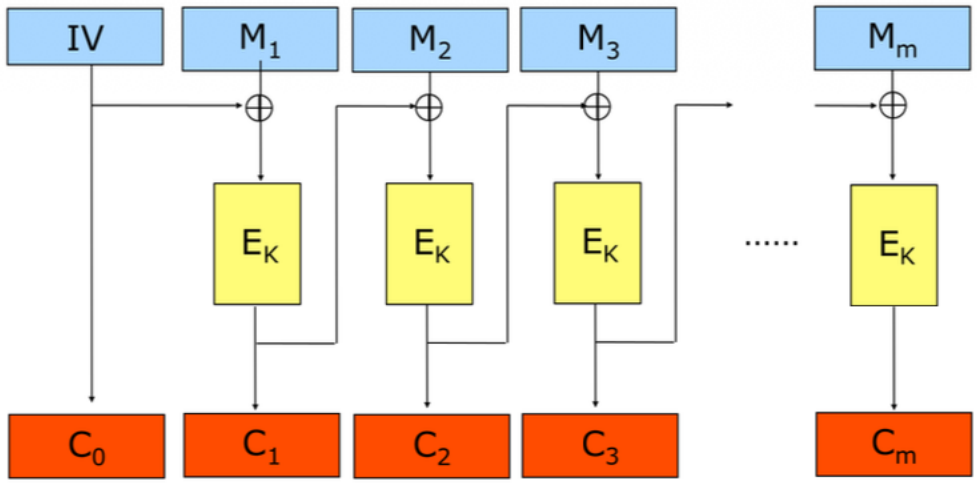
Chaque bloc est chiffre indépendamment des autres

- Avantage : on peut facilement rajouter ou supprimer des blocs
- Faille: attaque par rejeu en faisant des correlations.





Le mode CBC



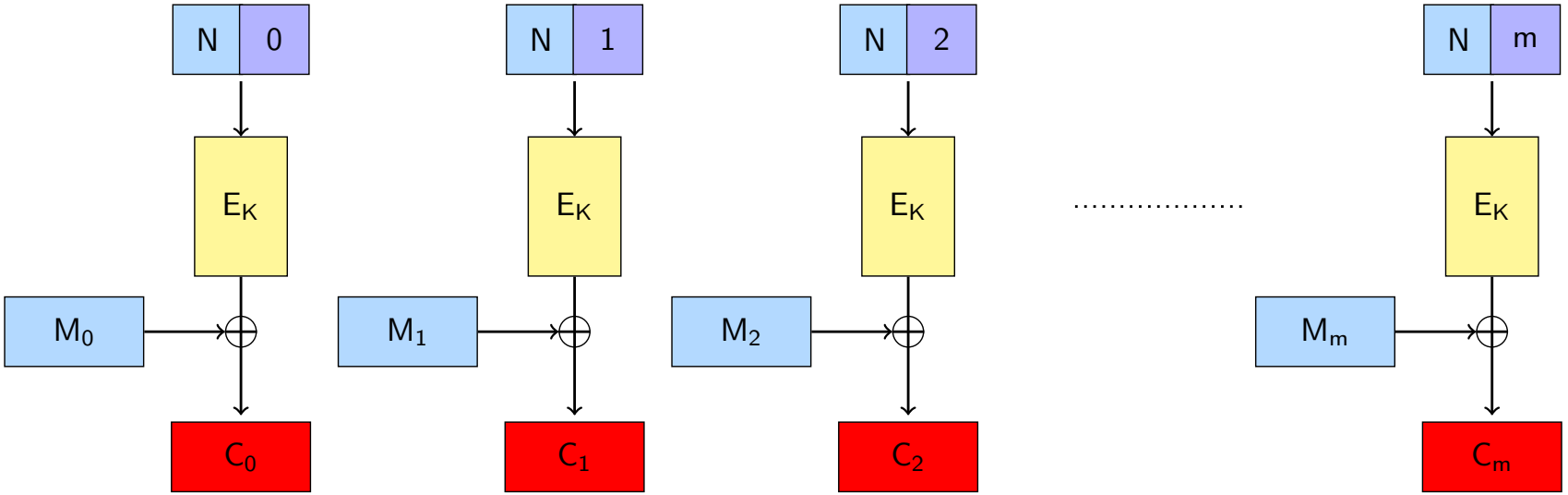
Le bloc de chiffré précédent est utilisé pour randomiser le bloc de clair :

$$C_0 = IV, \qquad C_{i+1} = E_K(C_i \oplus M_{i+1})$$

Le mode CBC

- Les attaques par rejeu ne sont plus possibles.
- Gestion de IV : transmis en clair avec le chiffré, valeur non prédictible
- Non-parallélisable
- Propagation d'erreur d'un bloc à l'autre.

Le mode CTR

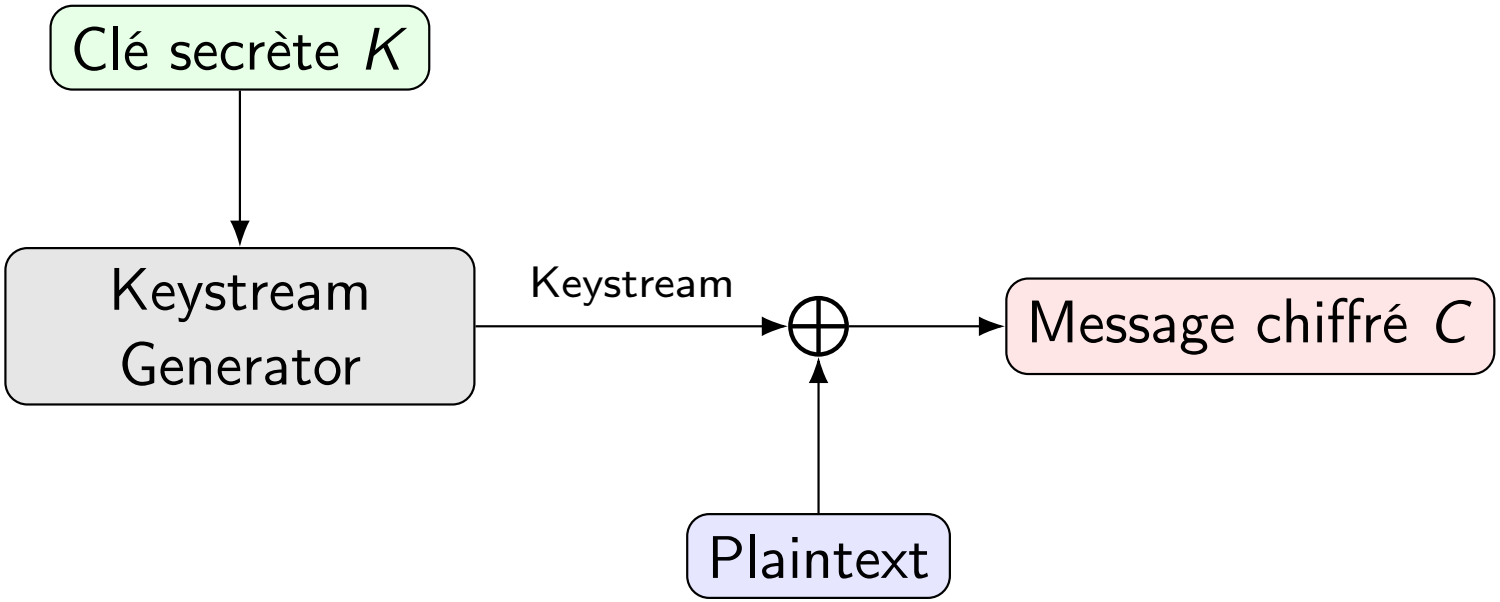


Transforme un chiffrement par bloc en chiffrement par flot:

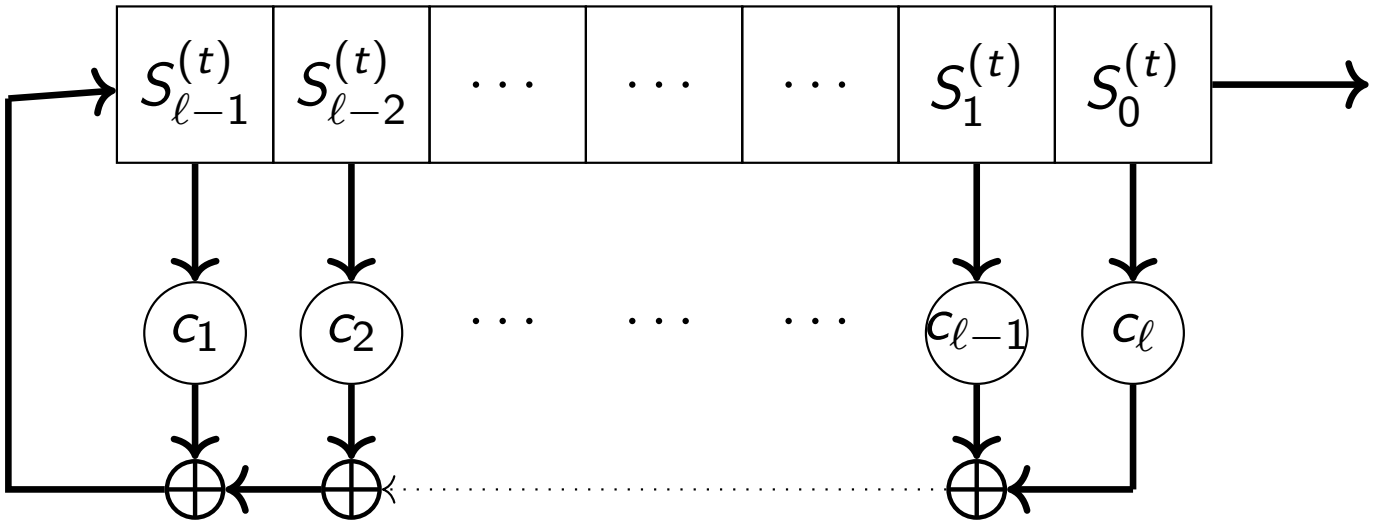
$$C_i = E_K(\text{Nonce}|i) \oplus M_i$$

Chiffrement par flots

- Les messages sont chiffrés par un XOR avec une clé.
- Une graine (clé partagée) permet de générer un flot de bits pseudo-aléatoires servant de clé OTP.



Linear Feedback Shift Register (LFSR)



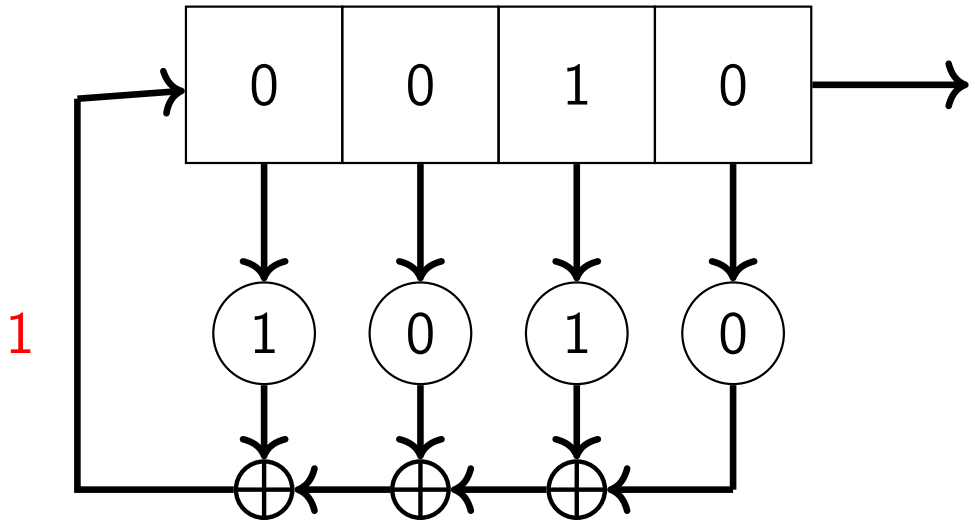
- Longueur du registre est ℓ, $s^{(0)}$ est la graine (seed)
- $\forall c_i \in \{0, 1\}$

$$\forall t \geq 0, s_{\ell-1}^{(t+1)} = \sum_{i=1}^{\ell} c_i s_{\ell-i}^{(t)}$$

$$Shift : s_i^{(t+1)} = s_{i+1}^{(t)}, \forall i, 0 \leq i \leq \ell - 2$$

Exemple

Graine $s^{(0)} = 0010$ et $c_1 = 1$ $c_2 = 0$ $c_3 = 1$ and $c_4 = 0$



$$\begin{aligned} s_3^{(1)} &= (s_3^{(0)} \cdot c_1) \oplus (s_2^{(0)} \cdot c_2) \oplus (s_1^{(0)} \cdot c_3) \oplus (s_0^{(0)} \cdot c_4) \\ &= (0 \cdot 1) \oplus (0 \cdot 0) \oplus (1 \cdot 1) \oplus (0 \cdot 0) \\ &= 1 \end{aligned}$$

Ron Rivest

Merci pour votre attention.

Questions ?

“‘Once you have something on the Internet, you are telling the world, please come hack me.”