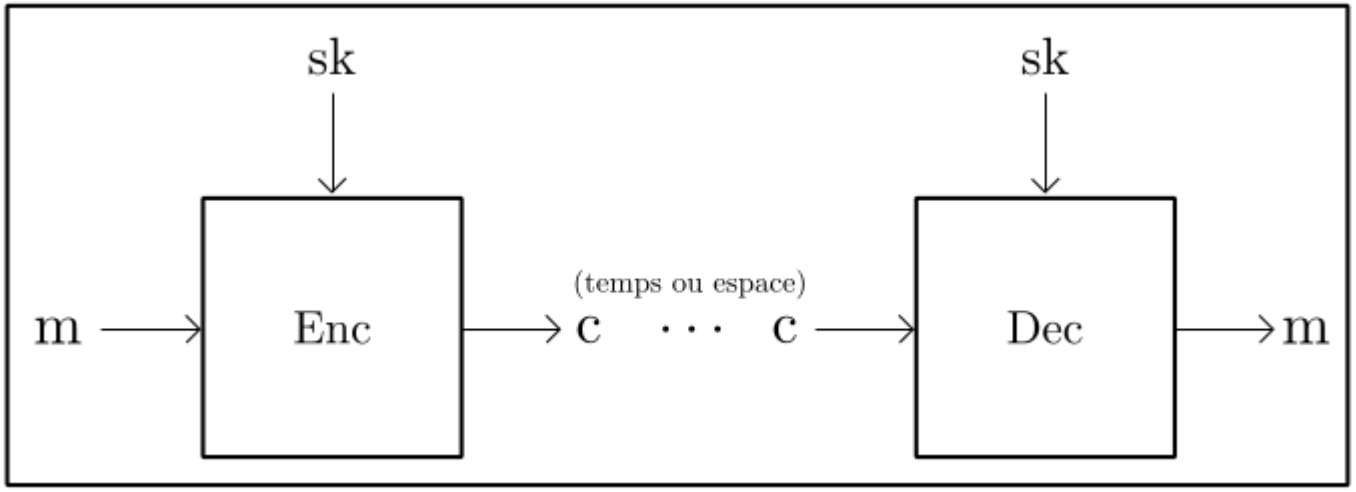
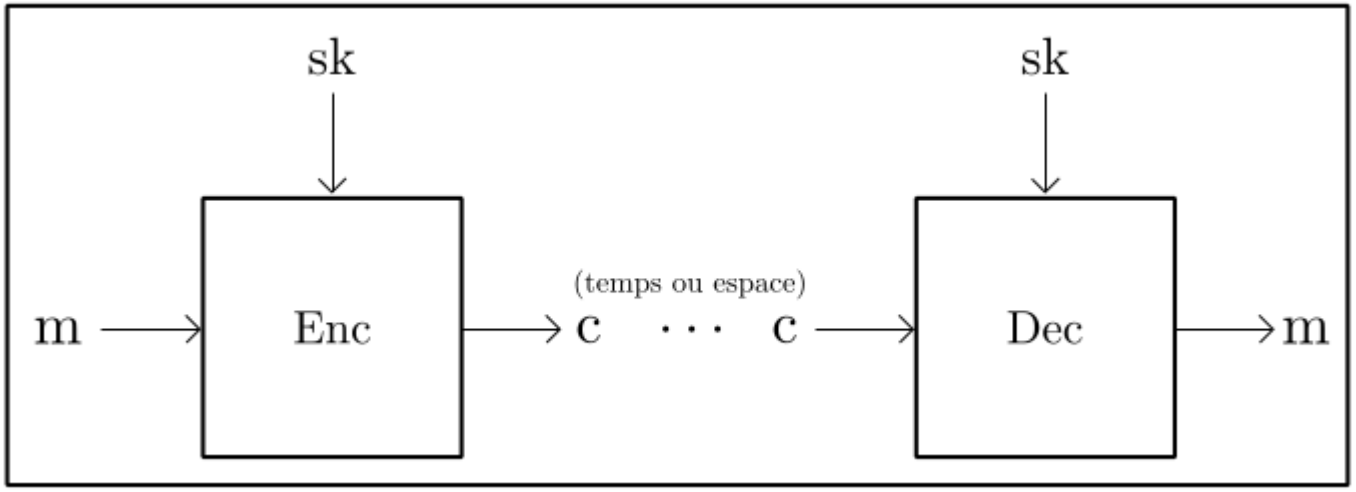
Aléa, pseudo-aléa, chiffrements à flot

Le chiffrement à flot



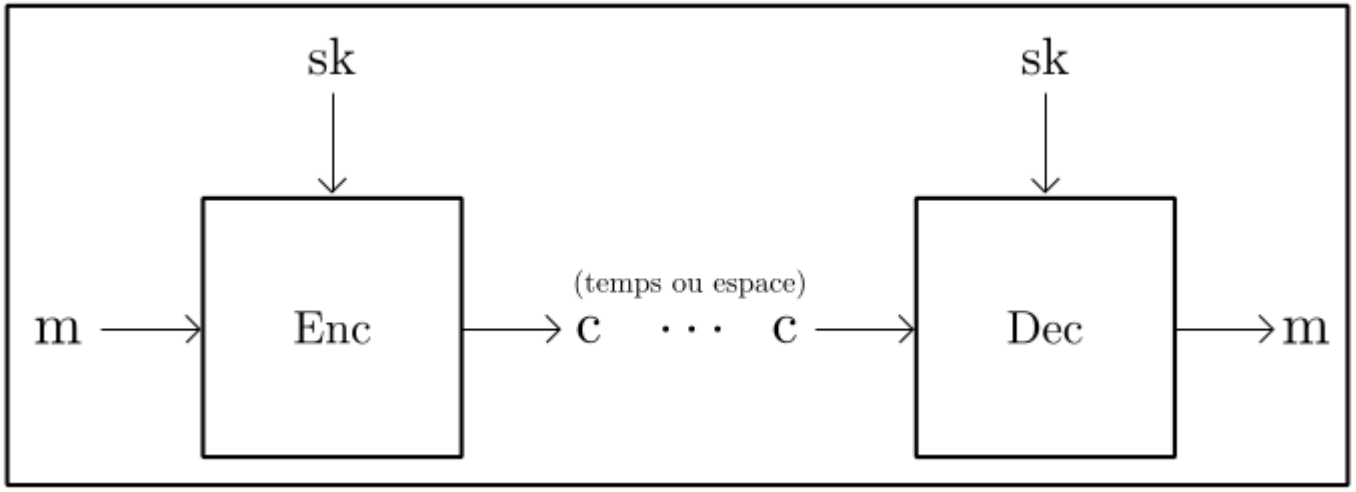
- Le principe :
  - On vient chiffrer un message de taille quelconque.
  - On reprend le principe du OTP :
    - On XOR un message avec un flux aléatoire, de la même taille que le message.

Le chiffrement à flot



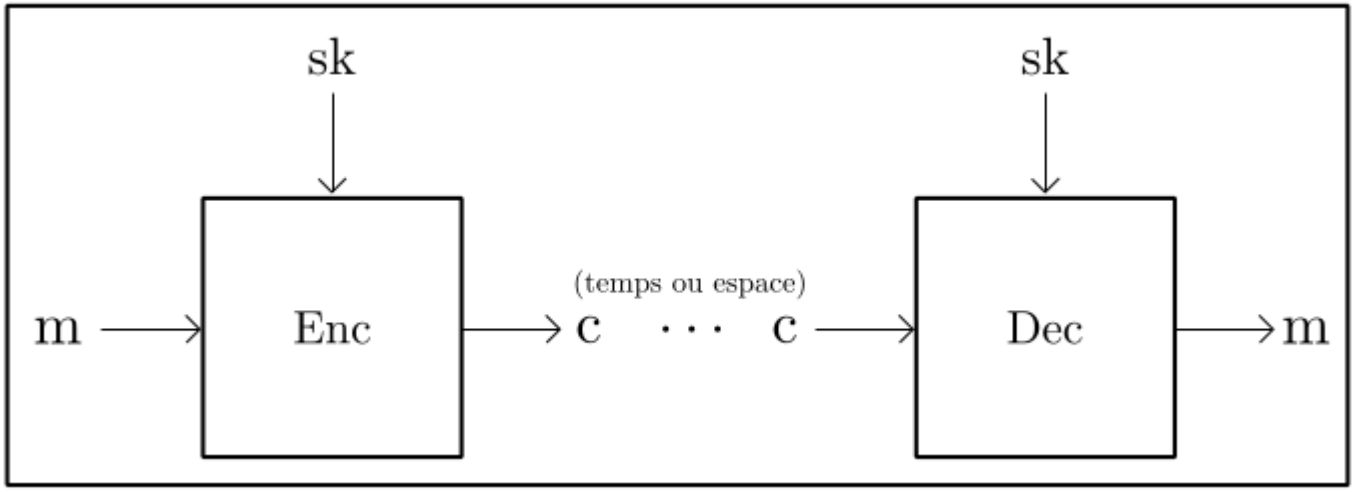
- L'utilisation :
  - Chiffrer un flux de données.
  - Très utile dans les télécommunications (Wifi / Bluetooth / GSM).

Le chiffrement à flot



- Les problèmes :
  - Comment générer une clé de la même taille que le message ?
  - Comment échanger cette clé avec son interlocuteur ?

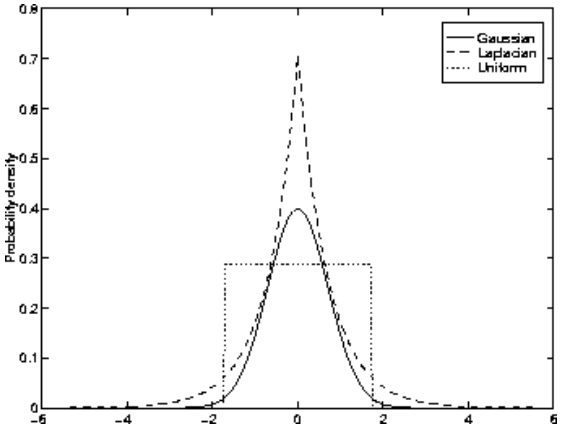
Le chiffrement à flot



- Les problèmes :
  - Comment générer une clé de la même taille que le message ? Utiliser un générateur d'aléa.
  - Comment échanger cette clé avec son interlocuteur ? Utiliser un mécanisme d'échange de clé.

Les générateurs d’aléa

- Deux grandes familles :
  - Les générateurs d’aléa purs :
    - Non reproductibles, pas de biais statistique, ils suivent précisément une distribution de bruit

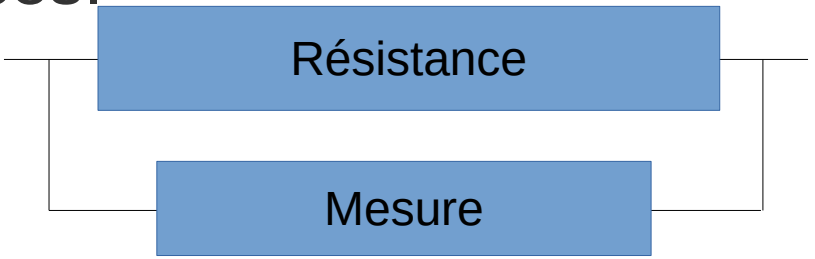


Exemples de distributions :
Bruit uniforme
Bruit Gaussien
Bruit laplacien

  - Les générateurs de pseudo-aléa :
    - Distribution proche d’une distribution de référence.
    - Reproductibles, sont générés à partir d’une graine.

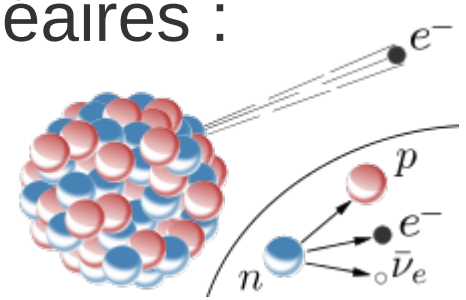
Les générateurs l'aléa purs

- Provient d'un phénomène physique ou quantique, non déterministe.
- Bruit thermique :
  - Agitation thermique des porteurs de charge dans les résistances.



$$v_b^2 = 4 . k_B . T . R . \delta f$$

- Décompositions nucléaires :

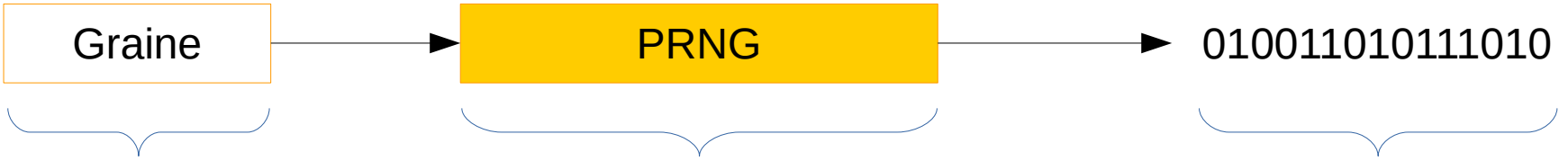


Les générateurs d’aléa purs

- Quelques exemples pratiques :
  - Implémentation directe dans les processeurs (Intel, bruit thermique).
  - Périphériques matériels PCI-E.
    - Exemple : Quantis (phénomènes quantiques).
  - Services web :
    - http://www.fourmilab.ch/hotbits/ (émissions radioactives)
    - https://www.random.org/ (bruit athmosphérique)
  - Cumulation de sources d’entropie (systèmes d’exploitations).

Les générateurs pseudo-aléatoires

- Construction :



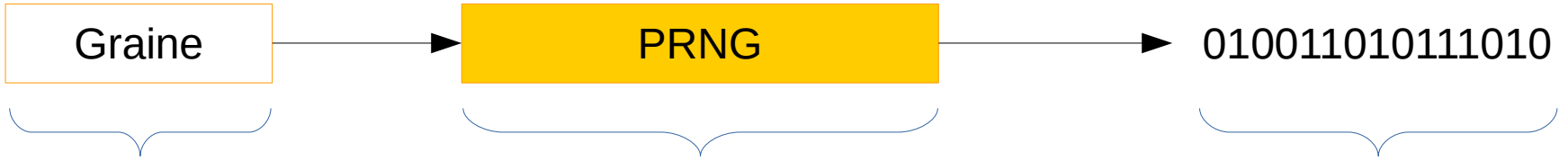
Ne doit pas être devinée simplement. Sa connaissance détermine la suite pseudo-aléatoire

Algorithme déterministe
Mélange la graine pour générer de l'aléa

Suite pseudo-aléatoire, sûre si « pas trop longue »

Les générateurs pseudo-aléatoires

- Construction :



Ne doit pas être devinée simplement. Sa connaissance détermine la suite pseudo-aléatoire

Si provient d'un TRNG, peut faire office de source d'aléa non déterminister

Algorithme déterministe Mélange la graine pour générer de l'aléa

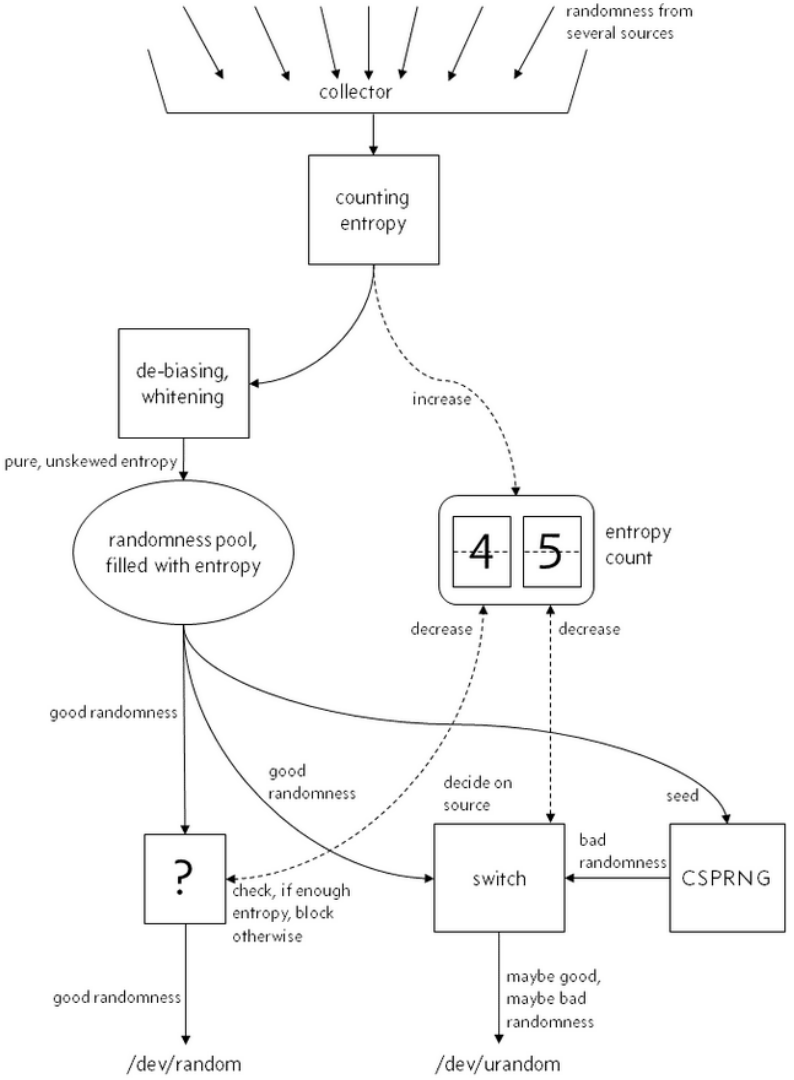
On peut par exemple utiliser un chiffrement par bloc si application crypto, ou des algos plus légers pour le reste

Suite pseudo-aléatoire, sûre si « pas trop longue »

Limitation de la longueur pour éviter des attaques statistiques = renouvellement de la graine

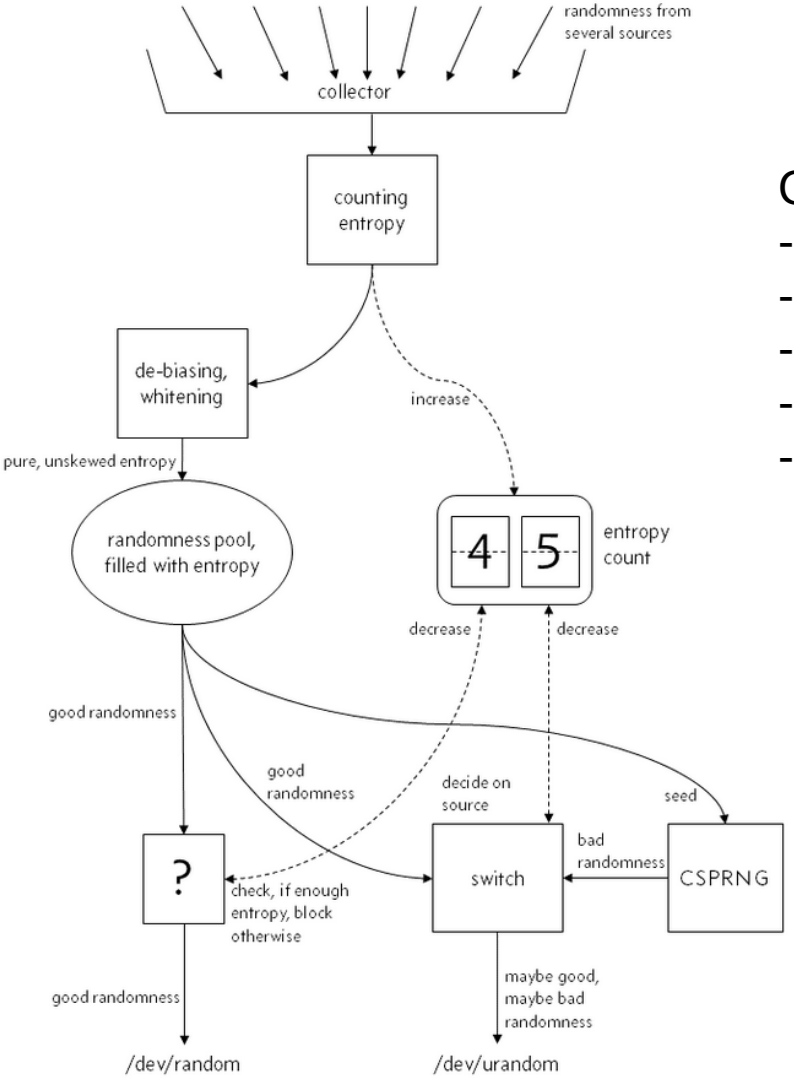
CSPRNG non déterministe dans les systèmes d'exploitation

- Cryptographically Secure PRNG sous Linux :



CSPRNG non déterministe dans les systèmes d'exploitation

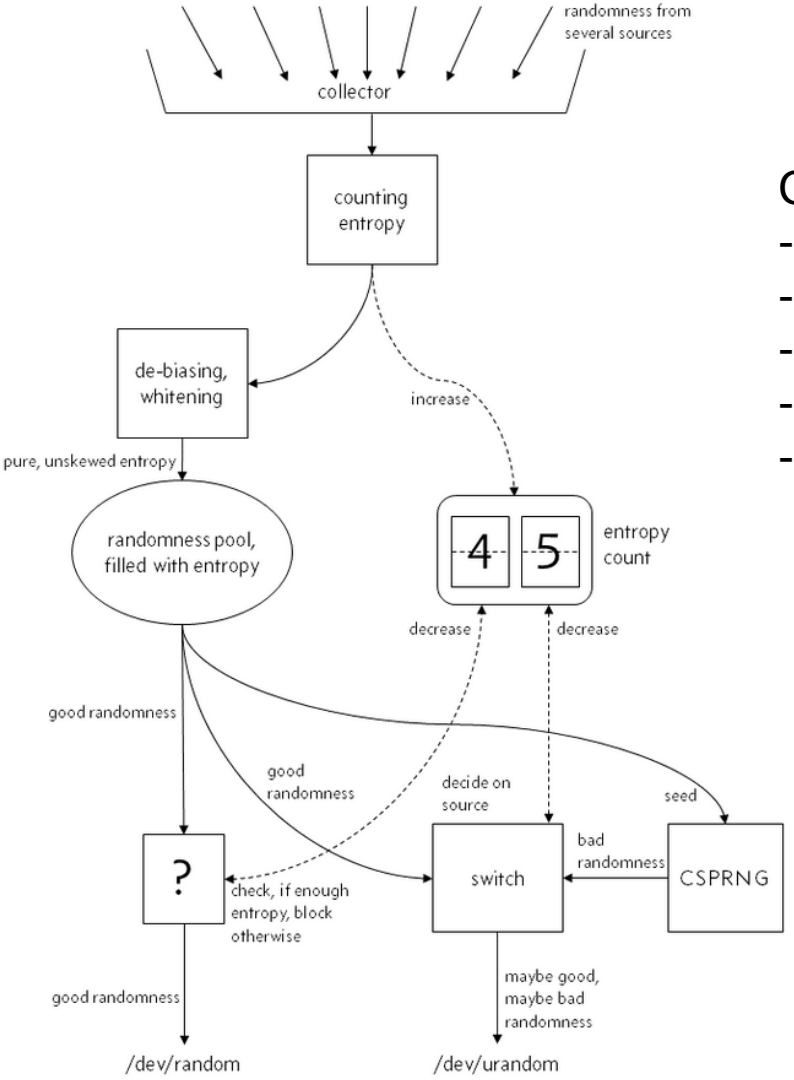
- Cryptographically Secure PRNG sous Linux :



Question : Source d'entropie fiable ?
- L'heure actuelle ;
- Le compteur de cycle processeurs ;
- Le nombre de processus ;
- Le déplacement de la souris ;
- Générateurs d'aléa matériels.

CSPRNG non déterministe dans les systèmes d'exploitation

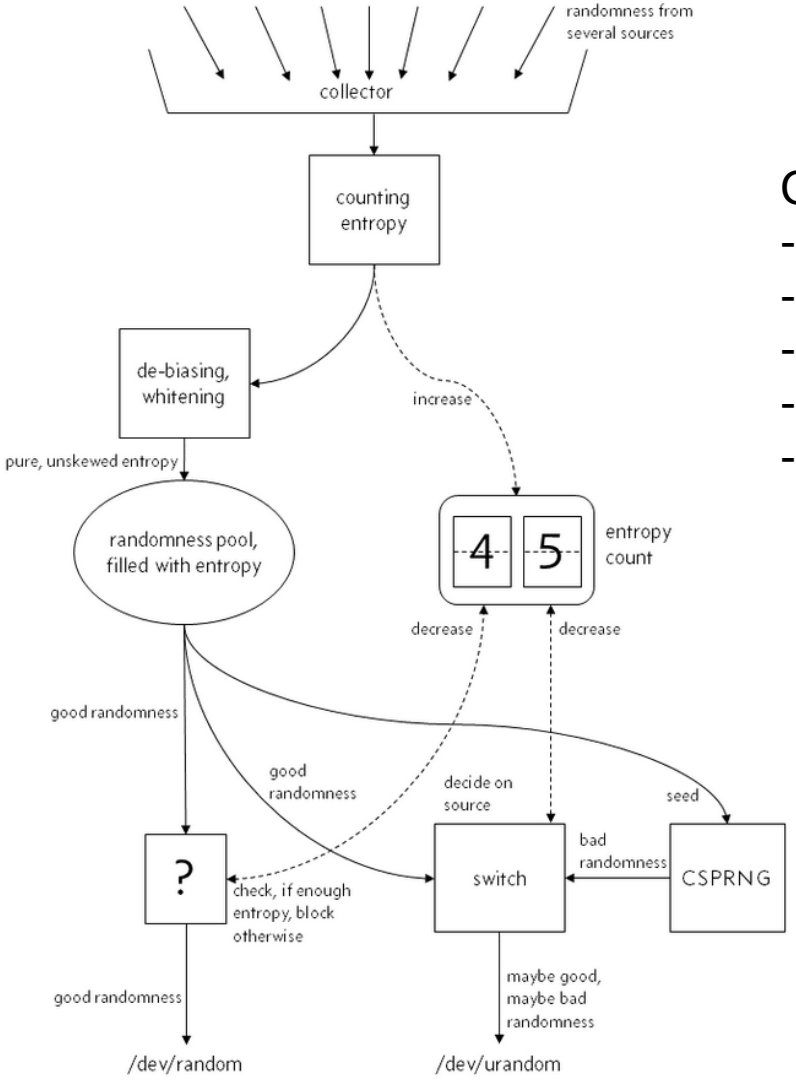
- Cryptographically Secure PRNG sous Linux :



Question : Source d'entropie fiable ?
- L'heure actuelle ; **Non :-)**
- Le compteur de cycle processeurs ; **Possible**
- Le nombre de processus ; **Possible**
- Le déplacement de la souris ; **Possible**
- Générateurs d'aléa matériels. **Possible**

CSPRNG non déterministe dans les systèmes d'exploitation

- Cryptographically Secure PRNG sous Linux :

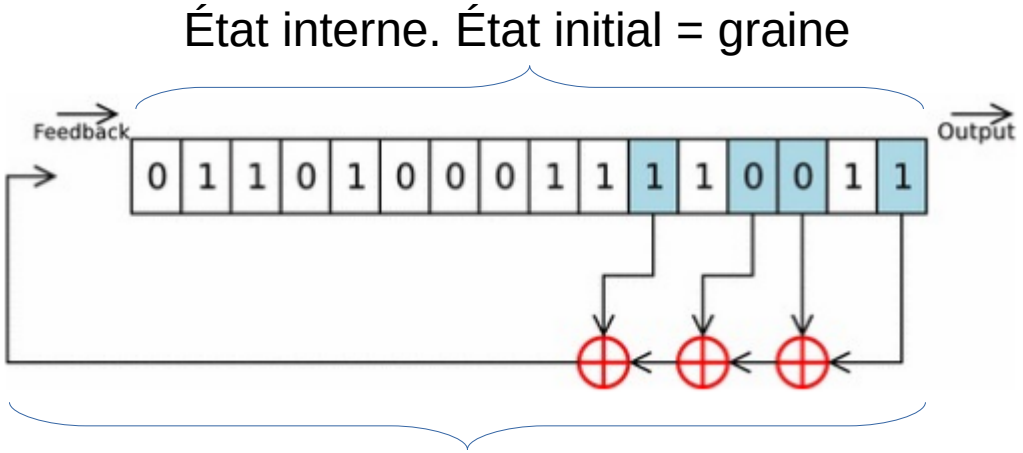


Question : Source d'entropie fiable ?
- L'heure actuelle ; **Non :-)**
- Le compteur de cycle processeurs ; **Possible**
- Le nombre de processus ; **Possible**
- Le déplacement de la souris ; **Possible**
- Générateurs d'aléa matériels. **Possible**

Ces CSPRNG sont considérés sûrs pour une utilisation crypto. Ont été attaqués sans succès (sauf windows, 2007 puis corrigé ensuite), se standardisent contrairement à l'aléa physique.

Exemples classiques de PRNG : Le LFSR

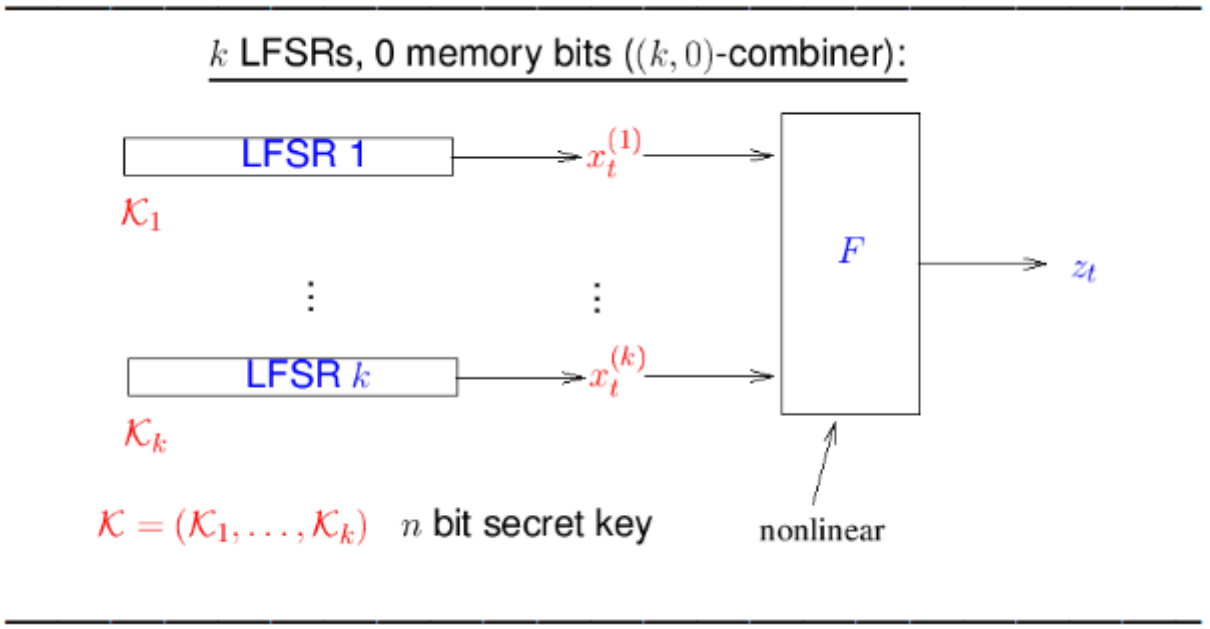
- Le LFSR (Linear Feedback Shift Register)



- Simple à implémenter (3 XOR).
- Simple à casser car linéaire :
  - Algorithme de Berlekamp-Massey : 2 fois la taille de l'état interne suffit à trouver le LFSR.
- Améliorable : Combinaison de plusieurs LFSR.

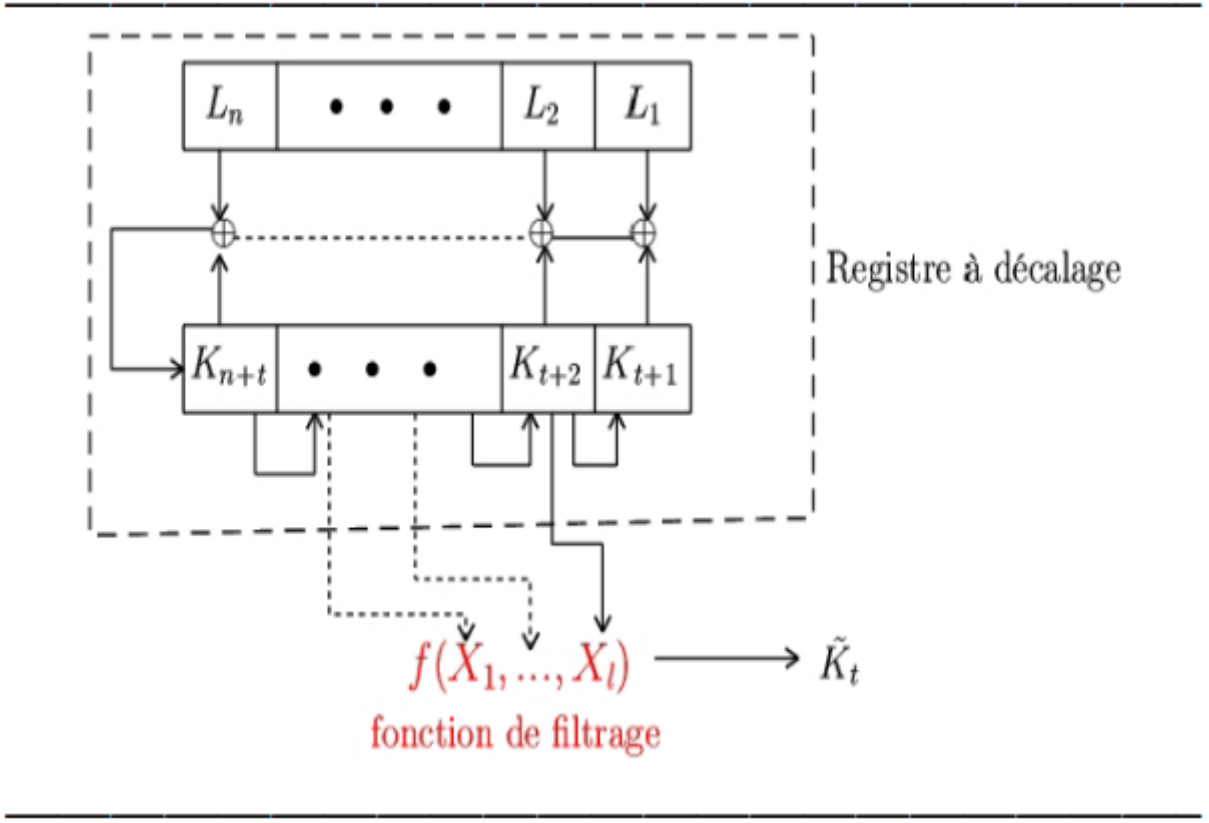
Exemples classiques de PRNG : Le LFSR - Amélioration

- LFSR combinés + étage non-linéaire :



Exemples classiques de PRNG : Le LFSR - Amélioration

- LFSR filtré :



Exemples classiques de PRNG : Le LFSR – Les ratés

- DVD : CSS → 2 LFSR combinés
  - Cassé
- GSM : A5/1 → 3 LFSR combinés
  - Cassé
- Bluetooth : E0 → 4 LFSR combinés
  - Cassé

Autres exemples classiques : PRNG non cryptographiques

- Générateur congruentiel linéaire :

$$X_{n+1} = a \,.\, X_n + b \mod m$$

Définit dans le code — 2^{32}

  - Graine : X_0

  - Utilisation :
    - Fonction rand du langage C / C++ / Java / Delphi / Pascal

Autres exemples classiques : PRNG non cryptographiques

- Mersenne twister (MT19937) :
  - Complexe à expliquer, mais également attaqué par l’algorithme de Berlekamp – Massey.
  - Utilisation répandue :
    - R / Python / PHP2 / Matlab / GMP / boost.

Les générateurs prouvés

- Générateurs capables de produire des nombres pseudo-aléatoires avec une preuve de sécurité associée.
- Blum Blum Shub :
  - On calcule $X_{n+1} = X_n^{\,2} \bmod m$
    - C'est le résidu quadratique
  - On renvoie la parité de X_{n+1}
  - Facile si *m* est un nombre premier (symbole de Legendre), compliqué si c'est le produit de deux nombres premiers.
  - *m* taille 1024 bits minimum, la graine X_0 ne doit pas avoir de facteurs communs avec *m*.

Problème de residuosité quadratique : Savoir si un nombre est le carré d'un autre.

Cas du Dual Elliptic Curve

- Introduit en 2004 dans la bibliothèque cryptographique RSA BSAFE (utilisation dans des produits américains).
- A fait partie du standard du NIST SP800-90A (supprimé en 2014).
- Backdoor de la NSA confirmée par de nombreuses sources !
- Selon Reuters, la NSA aurait payé 10M $ pour inclure ce PRNG dans la bibliothèque BSAFE.
- Bilan :
  - Conforte le fait que les PRNG sont les points sensibles de la crypto.
  - Ne pas se fier les yeux fermés aux standards.

Retour au chiffrement à flot

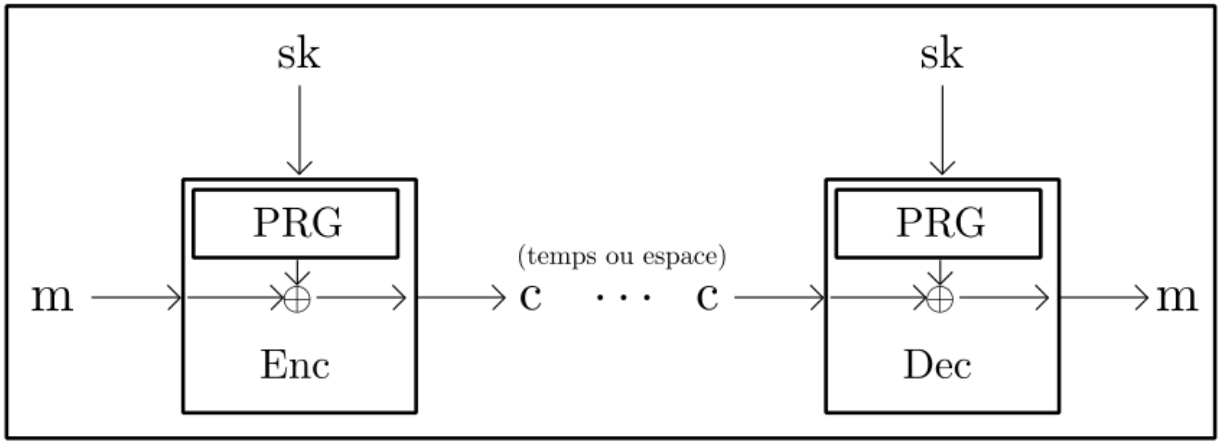
- A votre avis, quelle stratégie peut-on adopter pour concevoir un chiffrement à flot ?

Retour au chiffrement à flot

- A votre avis, quelle stratégie peut-on adopter pour concevoir un chiffrement à flot ?
- Réponse :
  - Utilisation d’un CSPRNG déterministe, avec l’utilisation d’une graine.
  - Seule la graine doit être partagée avec votre interlocuteur.
  - Permet de générer une séquence de bits bien plus grande que la graine.

Chiffrement à flot

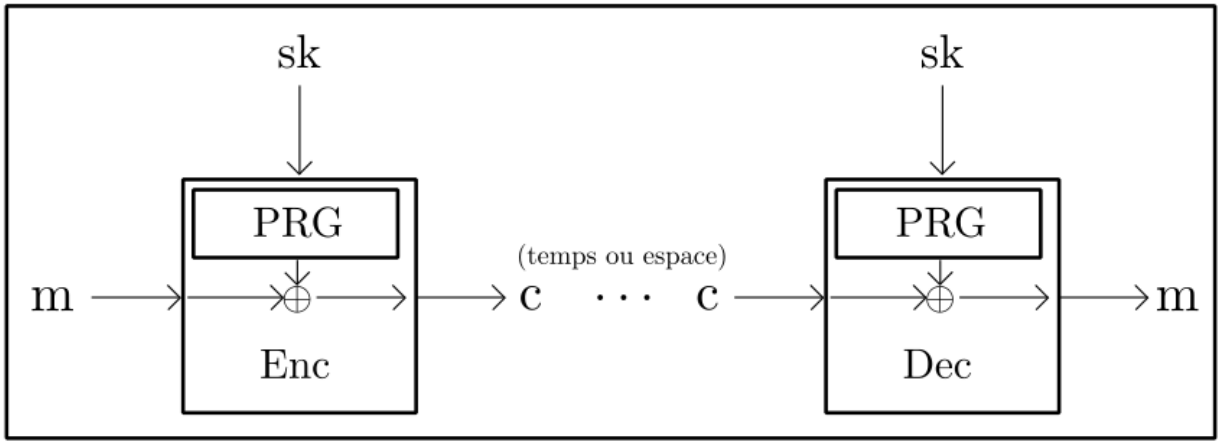
- Construction classique :



- Les avantages de cette construction :
  - Chiffrement/déchiffrement à la volée.
  - Le pseudo-aléa peut être pré-généré pour éviter d'introduire de délai.

Chiffrement à flot

- Construction classique :



- Remarque sur la sécurité :
  - Sous réserve que le PRNG est sûr (équivalent à une source d'aléa pur), alors le chiffrement à flot est sûr (il devient équivalent à l'OTP).

Quelques exemples de Chiffrement à flot

- LFSR :
  - DVD (CSS),
  - GSM/GPRS (A5/1-2),
  - Bluetooth (E0),
  - UMTS/LTE (snow2/snow3g)
- RC4 :
  - WEP
  - TKIP
  - HTTPS
- Standard crypto (projet eStream) :
  - Logiciel : Salsa20 (10 cycles/octet), SOSEMANUK (fr), Rabbit, HC-128.
  - Matériel : Trivium, MICKEY, Grain
- Utilisation classique également :
  - Utiliser le chiffrement symétrique AES en CTR.

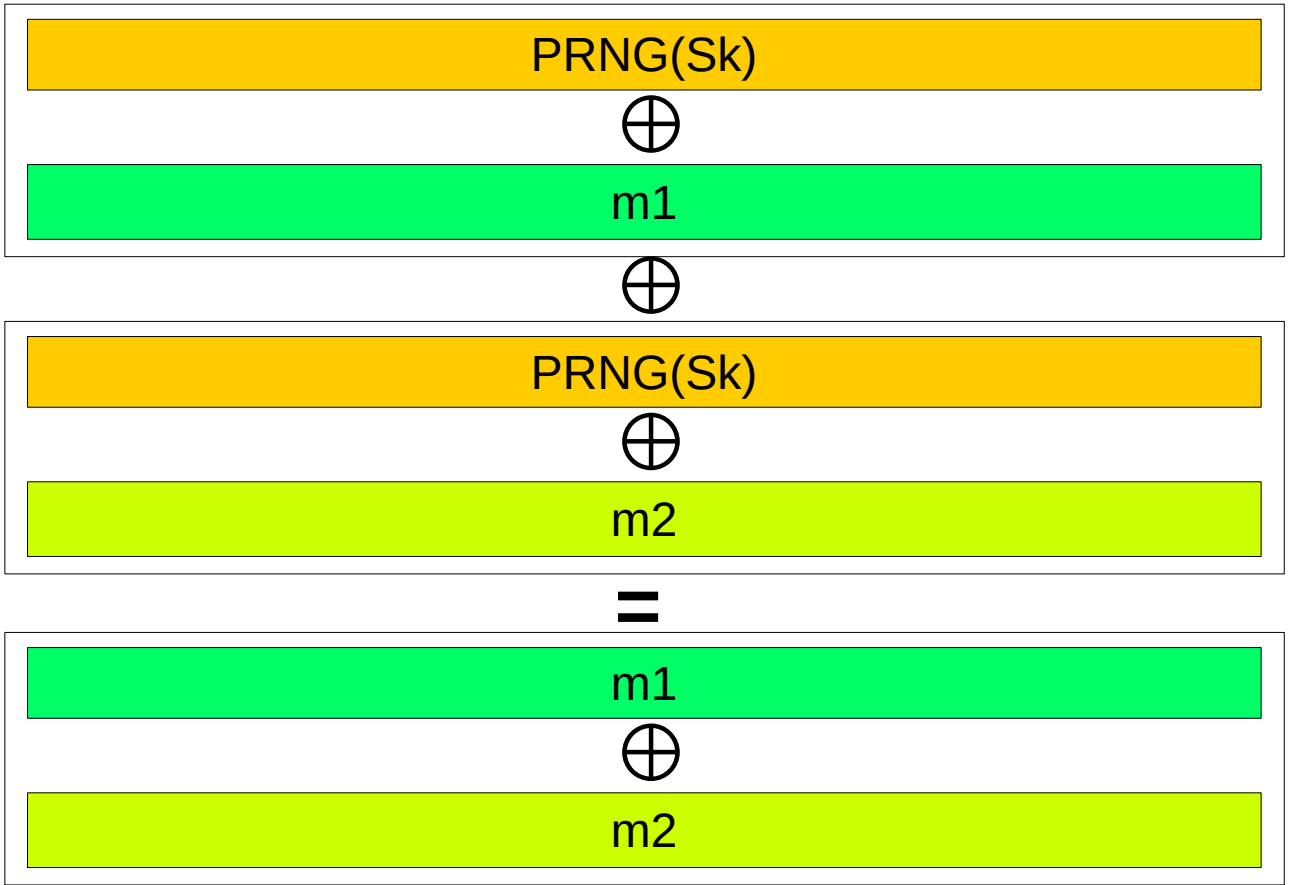
Chiffrement à flot

- Attention au choix de la clé, elle ne doit pas être utilisée plusieurs fois pour une même application.
- Exemple de la première version de PPTP.

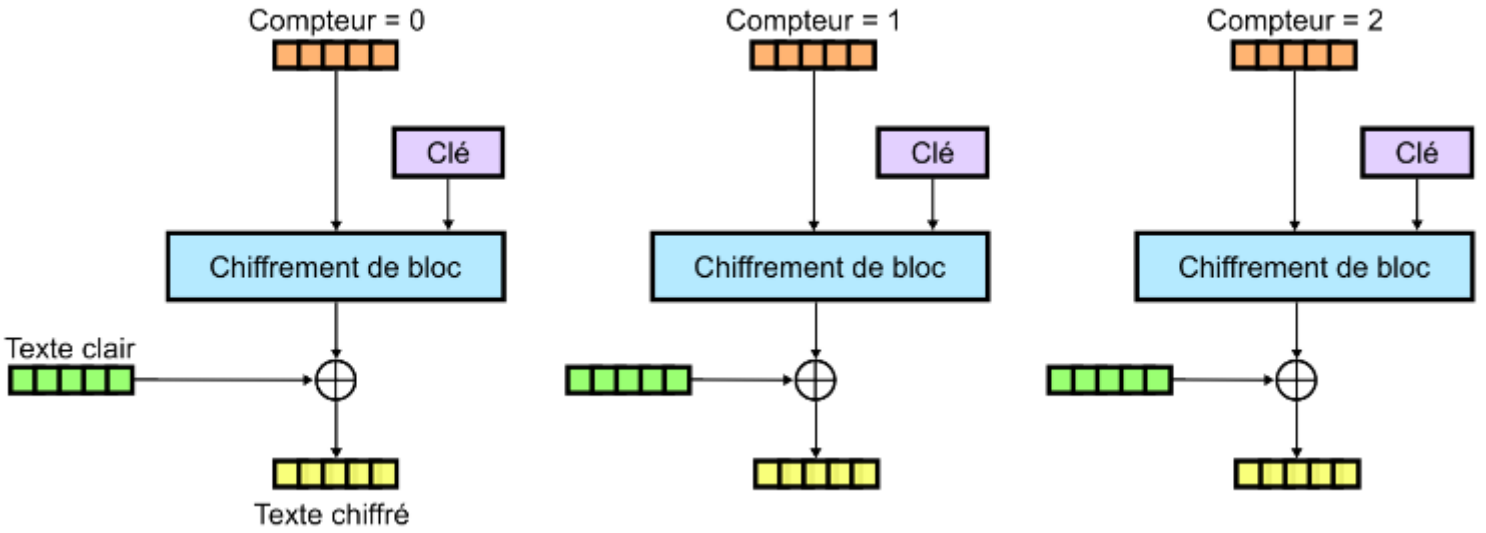
Client → Serveur :
clé Sk

Serveur → Client :
Même clé Sk

Le XOR des deux chiffrés
Révèle de l'info sur m1 et m2



AES-CTR



- Point potentiellement faible : Faible variation de l’entrée du chiffrement par blocs.
- Fonctionne si la permutation ne peux pas être distinguée d’une permutation aléatoire.
- Remarque : Le compteur est associé à un IV.