HAI709I - Fondements cryptographiques pour la sécurité

Cours 2 - Cryptographie à clé secrète et notions de sécurité

Rocco Mora

15 September, 2025

Université de Montpellier – Faculté des Sciences
M1 informatique, parcours Algo, IASD, Imagine

Cryptographie à clé secrète et securité

Cryptographie à clé secrète/symetrique [Symmetric/Private-key cryptography]
(crypto classique/1ère partie de ce cours)

- **à clé secrète :** 2 parties communicantes partagent à l'avance une clé qui est inconnue de l'espion
- **symétrique :** la même clé est utilisée pour convertir le texte en clair en texte chiffré et vice versa

---

La sécurité inconditionnelle exige trop :

- L'espion dispose d'une puissance de calcul illimitée
- aucune information n'est divulguée

→ **Sécurité calculatoire :**

- l'espion dispose d'un certain temps raisonnable
- l'adversaire peut potentiellement réussir avec une très faible probabilité

Les définitions et les preuves restent essentielles

Sécurité calculatoire : l'approche concrète

Définition concrète de la sécurité

Un schéma est (t, ϵ)-sûr si tout adversaire disposant d'un temps maximal t réussit à le casser avec une probabilité maximale ϵ.

$\rightarrow$ il faut définir ce que signifie "casser"

Exemple : longueur de clé $n = 64$ ($\mathcal{K} = \{0,1\}^{64}$)

L'adversaire dispose d'un processeur de 4 GHz avec 16 cœurs, chaque cœur exécutant $4 \cdot 10^9$ cycles/seconde

$\Rightarrow$ la force brute nécessite $2^{64}/(16 \cdot (4 \cdot 10^9))$ secondes ≈ 9 ans.

$\Rightarrow$ La force brute réussit avec une probabilité de $1/2^{28}$ en $2^{64}/(16 \cdot (4 \cdot 10^9) \cdot 2^{28}) \approx 1$ seconde.

+ approche importante en pratique
− dépend de la puissance de calcul utilisée et actuellement disponible

Sécurité calculatoire : l'approche asymptotique

- Une fonction $f : \mathbb{N} \to \mathbb{R}_+$ est bornée par un polynôme s'il existe une constante c et $N \in \mathbb{N}$ tels que, pour tout $n > N$, $f(n) < n^c$. On désigne une fonction bornée par un polynôme arbitraire par $\texttt{poly}$.
- Une fonction $f : \mathbb{N} \to \mathbb{R}_+$ est négligeable si, pour toutes les constantes c, il existe $N \in \mathbb{N}$ tel que, pour tout $n > N$, on a $f(n) < \frac{1}{n^c}$. On désigne une fonction négligeable arbitraire par $\texttt{negl}$.
- Un algorithme est probabiliste (ou aléatoire) s'il a accès à une séquence de bits aléatoires indépendants et non biaisés.

Propriétés :

- $\texttt{poly}_1(n) + \texttt{poly}_2(n), \texttt{poly}_1(n) \cdot \texttt{poly}_2(n)$ et $\texttt{poly}_1(\texttt{poly}_2(n))$ sont des fonctions bornée par un polynôme.
- $\texttt{negl}_1(n) + \texttt{negl}_2(n)$ et $\texttt{poly}(n) \cdot \texttt{negl}(n)$ sont des fonctions négligeables.

Un adversaire $\mathcal{A}$ probabiliste en temps polynomial PPT est un adversaire qui exécute un algorithme probabiliste se terminant en un temps donné par une fonction bornée de manière polynomiale.

Soit n le paramètre de sécurité [security parameter] (p. ex., la longueur de la clé).

Définition asymptotique de la sécurité

Un schéma est sûr si, pour chaque adversaire PPT $\mathcal{A}$ menant une attaque, la probabilité que $\mathcal{A}$ réussisse son attaque est donnée par une fonction négligeable.

Cela permet d'augmenter la sécurité en augmentant le paramètre de sécurité

La définition est significative par rapport aux attaques triviales. Soit $\mathcal{K} = \{0,1\}^n$:

- Une attaque par force brute réussit avec une probabilité $\mathcal{O}(1)$ mais s'exécute en temps exponentiel $\mathcal{O}(|\mathcal{K}|) = \mathcal{O}(2^n)$, c'est-à-dire qu'elle n'est pas limitée de manière polynomiale.
- Deviner la clé k et vérifier $\mathrm{Dec}_k(c) = m$. L'attaquant est PPT mais réussit avec une probabilité négligeable $1/|\mathcal{K}| = 1/2^n$.

Revenons à la définition du schéma de chiffrement à clé secrète.

Schéma de chiffrement à clé secrète [🇬🇧 Private-key encryption scheme]

Un schéma de chiffrement à clé secrète Π se compose de trois algorithmes en temps polynomial $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ tels que :

- L'algorithme aléatoire/randomisé de génération de clé Gen prend en entrée 1^n (le paramètre de sécurité écrit en unaire) et produit une clé $k \leftarrow \mathsf{Gen}(1^n)$. Nous supposons sans perte de généralité que toute valeur possible de k satisfait $|k| \geq n$.
- L'algorithme de chiffrement Enc prend en entrée une clé k et un texte en clair $m \in \{0,1\}^*$ et produit un texte chiffré $c \leftarrow \mathsf{Enc}_k(m)$.
- L'algorithme de déchiffrement Dec prend en entrée une clé k et un texte chiffré c et produit un message $m := \mathsf{Dec}_k(c) \in \{0,1\}^*$ ou une erreur $\perp$.

Pour chaque n, $k \leftarrow \mathsf{Gen}(1^n)$ et $m \in \{0,1\}^*$, on a $\boxed{\mathsf{Dec}_k(\mathsf{Enc}_k(m)) = m}$.

- Enc peut être aléatoire, mais on suppose toujours que Dec est déterministe.
- Si Enc_k n'est défini que pour $m \in \{0,1\}^{\ell(n)}$, alors Π a une longueur fixe $\ell(n)$.

Le modèle de menace

Il reste à définir le type de sécurité que nous souhaitons. Une définition de la sécurité est donnée par :

- un modèle de menace [threat model]
- et un objectif de sécurité [security goal]

Soit $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ un schéma de chiffrement à clé secrète, $\mathcal{A}$ un adversaire et n le paramètre de sécurité.

L'expérience d'EAV-sécurité $\mathtt{PrivK}^{eav}_{\mathcal{A},\Pi}(n)$

1. L'adversaire $\mathcal{A}$ reçoit l'entrée 1^n et produit deux messages m_0, m_1 t.q. $|m_0| = |m_1|$.
2. Une clé k est générée avec $\mathsf{Gen}(1^n)$ et un bit uniforme $b \in \{0,1\}$ est choisi. Le texte chiffré $c \leftarrow \mathsf{Enc}_k(m_b)$ est donné à $\mathcal{A}$ et est appelé texte chiffré défi [challenge ciphertext].
3. $\mathcal{A}$ produit un bit b'.
4. Le résultat de l'expérience est 1 (c-à-d que $\mathcal{A}$ réussit) si $b' = b$, sinon 0.

L'objectif de sécurité

EAV-sécurité [🇬🇧 EAV-security]

Un schéma de chiffrement à clé secrète $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ est EAV-sûr [🇬🇧 EAV-secure], si pour tout PPT adversaire $\mathcal{A}$, il existe une fonction négligeable $\mathtt{negl}$ telle que, pour tout n,

$$\Pr(\mathtt{PrivK}^{eav}_{\mathcal{A},\Pi}(n) = 1) \leq \frac{1}{2} + \mathtt{negl}(n).$$

L'objectif de sécurité

EAV-sécurité [EAV-security]

Un schéma de chiffrement à clé secrète $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ est EAV-sûr [EAV-secure], si pour tout PPT adversaire $\mathcal{A}$, il existe une fonction négligeable $\mathtt{negl}$ telle que, pour tout n,

$$\Pr(\mathtt{PrivK}^{eav}_{\mathcal{A},\Pi}(n) = 1) \leq \frac{1}{2} + \mathtt{negl}(n).$$

- L'EAV-sécurité est plus faible que la sécurité inconditionnelle en raison de deux relaxations, à savoir

Sécurité inconditionnelle (définition équivalente)

Un schéma de chiffrement à clé secrète $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ est inconditionnellement sûr si, pour tout adversaire $\mathcal{A}$

$$\Pr(\mathtt{PrivK}^{eav}_{\mathcal{A},\Pi}(n) = 1) = \frac{1}{2}.$$

Générateurs pseudo-aléatoires

Avant de construire des schémas de chiffrement sûrs, nous avons besoin de

Générateur pseudo-aléatoire [Pseudorandom generators] (PRG)

Soit G un algorithme déterministe en temps polynomial tel que, pour tout n et toute entrée $s \in \{0,1\}^n$, appelée **graine** [seed], la chaîne $G(s)$ ait une longueur $\ell(n)$. G est un générateur pseudo-aléatoire (PRG) si

1. **Expansion :** pour tout n, $\ell(n) > n$.
2. **pseudo-aléatoire :** pour tous les distingueurs [distinguishers] PPT D, il existe une fonction négligeable $\texttt{negl}$ telle que

$$|\Pr(D(G(s)) = 1) - \Pr(D(r) = 1)| \leq \texttt{negl}(n),$$

où $s \in \{0,1\}^n$ et $r \in \{0,1\}^{\ell(n)}$ sont choisis uniformément au hasard.

On appelle $\ell(n)$ le **facteur d'expansion** de G.

EAV-sécurité à partir d'un PRG

Construction

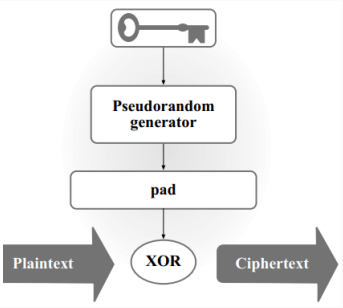
Soit G un PRG avec un facteur d'expansion $\ell(n)$.
Définissons le schéma de chiffrement à clé secrète suivant :

- Gen(1^n) choisit un $k \in \{0,1\}^n$ uniforme.
- Étant donné $m \in \{0,1\}^{\ell(n)}$,

$$\mathrm{Enc}_k(m) = m \oplus G(k).$$

- Étant donné $c \in \{0,1\}^{\ell(n)}$,

$$\mathrm{Dec}_k(c) = G(k) \oplus c.$$



Théorème

Si G est un PRG, alors la construction ci-dessus est un schéma de chiffrement à clé secrète de longueur fixe et EAV-sûr pour les messages de longueur $\ell(n)$.

Que se passe-t-il si plusieurs messages sont envoyés ?

L'expérience d'EAV-sécurité pour chiffrements multiples $\texttt{PrivK}^{mult}_{\mathcal{A},\Pi}(n)$

1. L'adversaire $\mathcal{A}$ reçoit l'entrée 1^n et produit une paire de listes de messages $M_0 = (m_{0,1}, \ldots, m_{0,t}), M_1 = (m_{1,1}, \ldots, m_{1,t})$ avec $|m_{0,i}| = |m_{1,i}|$.
2. Une clé k est générée en exécutant $\text{Gen}(1^n)$ et un bit uniforme $b \in \{0,1\}$ est choisi. Les textes chiffrés $c_i \leftarrow \text{Enc}_k(m_{b,i})$ sont donnés à $\mathcal{A}$.
3. $\mathcal{A}$ produit un bit b'.
4. Le résultat de l'expérience est 1 si $b' = b$ et 0 dans le cas contraire.

EAV-sécurité pour chiffrements multiples

Un schéma de chiffrement à clé secrète $\Pi = (\text{Gen}, \text{Enc}, \text{Dec})$ est EAV-sûr pour chiffrements multiples si, pour tout adversaire PPT $\mathcal{A}$, il existe une fonction négligeable $\texttt{negl}$ telle que, pour tout n,

$$\Pr(\texttt{PrivK}^{mult}_{\mathcal{A},\Pi}(n) = 1) \leq \frac{1}{2} + \texttt{negl}(n).$$

Cette notion est-elle plus forte ?

Pouvez-vous trouver un schéma qui soit EAV-sûr mais pas EAV-sûr pour chiffrements multiples ?

Cette notion est-elle plus forte ?

Pouvez-vous trouver un schéma qui soit EAV-sûr mais pas EAV-sûr pour chiffrements multiples ?

Réponse : le masque jetable !

Preuve :

- Le OTP est inconditionnellement sûr, il est donc également EAV-sûr.
- $\rightarrow$ l'adversaire $\mathcal{A}$ envoie les listes de messages $M_0 = (0^\ell, 0^\ell)$ et $M_1 = (0^\ell, 1^\ell)$.
  $\rightarrow$ $\mathcal{A}$ reçoit (c_1, c_2)
  $\rightarrow$ si $c_1 = c_2$, $\mathcal{A}$ produit $b' = 0$, sinon $b' = 1$.
  $\rightarrow$ si $b = 0$, alors $c_1 = c_2$ (car OTP est déterministe)
  $\rightarrow$ si $b = 1$, alors $c_1 \neq c_2$
  $\Rightarrow$ $\mathcal{A}$ réussit avec une probabilité de 1.
  $\Rightarrow$ L'OTP n'est pas EAV-sûr pour chiffrements multiples.

Une condition nécessaire est que Enc doit être aléatoire !

Attaques par texte en clair choisi et sécurité CPA

Intuitivement : l'adversaire a un contrôle (partiel) sur les textes en clair qui sont chiffrés.

Ce modèle de menace est-il réaliste ? Oui, par exemple

- **Scénario réel :** un attaquant tape sur un terminal qui chiffre tout à l'aide d'une clé partagée avec un serveur distant. L'attaquant décide donc de ce qui est chiffré et ne devrait rien apprendre lorsque le même schéma de chiffrement est utilisé par quelqu'un d'autre.
- **Scénario historique :** pendant la Seconde Guerre mondiale, les Britanniques ont placé des mines à certains endroits. Lorsqu'ils trouvaient des mines, les Allemands chiffraient leur emplacement et l'envoyaient au quartier général. Ces informations ont aidé les cryptanalystes britanniques à déchiffrer le système de chiffrement allemand.

L'expérience de CPA-sécurité $\mathtt{PrivK}^{cpa}_{\mathcal{A},\Pi}(n)$

Enc_K est traité comme une boîte noire

1. Une clé k est générée en exécutant $\mathrm{Gen}(1^n)$.
2. L'adversaire $\mathcal{A}$ reçoit l'entrée 1^n et l'accès à l'oracle Enc_k et produit une paire de messages m_0, m_1 avec $|m_0| = |m_1|$.
3. Un bit uniforme $b \in \{0,1\}$ est choisi. Le chiffré $c \leftarrow \mathrm{Enc}_k(m_b)$ est donné à $\mathcal{A}$.
4. L'adversaire a toujours accès à l'oracle Enc_k et produit un bit b'.
5. Le résultat de l'expérience est 1 (c-à-d que $\mathcal{A}$ réussit) si $b' = b$, sinon 0.

CPA-sécurité [🇬🇧 Chosen plaintext attack (CPA) security]

Un schéma de chiffrement à clé secrète $\Pi = (\mathrm{Gen}, \mathrm{Enc}, \mathrm{Dec})$ est CPA-sûr [🇬🇧 CPA-secure], si pour tout adversaire PPT $\mathcal{A}$, il existe une fonction négligeable $\mathtt{negl}$ telle que, pour tout n,

$$\Pr(\mathtt{PrivK}^{cpa}_{\mathcal{A},\Pi}(n) = 1) \leq \frac{1}{2} + \mathtt{negl}(n).$$

⚠ Comme pour la EAV-sécurité, on peut définir la CPA-sécurité pour chiffrement multiples. Cependant, il s'avère que cela est équivalent à la CPA-sécurité .

Fonction pseudo-aléatoire

Avant de construire des schémas CPA-sûr, nous avons besoin d'une fonction pseudo-aléatoire [🇬🇧 Pseudorandom function] (PRF), qui généralise un générateur pseudo-aléatoires :

- un PRG est une chaîne "à l'apparence aléatoire",
- une PRF est une fonction "à l'apparence aléatoire".

Une fonction à clé et à longueur préservée est une fonction à deux entrées, la première étant appelée clé :

$$F\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n,$$
$$F(k,x) \in \{0,1\}^n.$$

Si k est fixe, alors nous définissons une fonction à entrée unique

$$F_k\colon \{0,1\}^n \to \{0,1\}^n,$$
$$F_k(x) \stackrel{\text{def}}{=} F(k,x).$$

Soit $\mathtt{Func}_n$ l'ensemble de toutes les fonctions $f : \{0,1\}^n \to \{0,1\}^n$.

Fonction pseudo-aléatoire (PRF)

Une fonction à clé et préservant la longueur efficace $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ est une fonction pseudo-aléatoire si, pour tous les distingueurs PPT D, il existe une fonction négligeable $\mathtt{negl}$ telle que

$$|\Pr(D(F_k(\cdot)) = 1) - \Pr(D(f(\cdot)) = 1)| \leq \mathtt{negl}(n)$$

où la première probabilité est prise sur $k \in \{0,1\}^n$ uniforme et sur le caractère aléatoire de D, tandis que la deuxième probabilité est prise sur un choix uniforme de $f \in \mathtt{Func}_n$ et sur le caractère aléatoire de D.

⚠ D ne connaît pas la clé k.

⚠ On a $|\mathtt{Func}_n| = 2^{n \cdot 2^n} \gg 2^n = |\mathcal{K}|$

Relation étroite entre les fonctions pseudo-aléatoires et les générateurs pseudo-aléatoires

- Soit F une fonction pseudo-aléatoire. On peut alors construire un générateur pseudo-aléatoire G tel que

$$G(s) = F_s(1)||F_s(2)||\ldots||F_s(\ell).$$

- Soit G un générateur pseudo-aléatoire. On peut alors construire une fonction pseudo-aléatoire F. Il faut un problème mathématique difficile pour obtenir une bonne longueur d'entrée.

Permutation pseudo-aléatoire : identique à une fonction pseudo-aléatoire, mais

- pour toutes les clés k, F_k est une permutation.
- doit être indistinguable des permutations uniformes, c'est-à-dire des éléments pris dans $\mathtt{Perm}_n$. $|\mathtt{Perm}_n| = (2^n)!$.

Une tentative infructueuse

On pourrait être tenté de définir un schéma de chiffrement à partir d'une permutation pseudo-aléatoire de la manière suivante :

$$\mathrm{Enc}_k(m) = F_k(m).$$

- Apparemment, aucune information n'est révélée car l'image d'une permutation pseudo-aléatoire est une chaîne pseudo-aléatoire.
- Cependant, ce chiffrement est déterministe.

CPA-sûr
$\iff$ CPA-sûr pour chiffrement multiples
$\Rightarrow$ EAV-sûr pour chiffrement multiples
$\Rightarrow$ Chiffrement randomisé

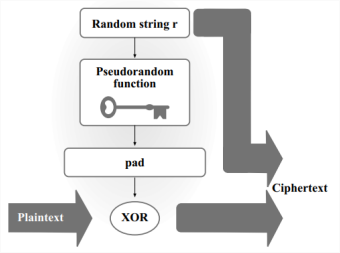
(La même contrainte s'appliquera aux notions de sécurité plus fortes que CPA)

CPA-Sécurité à partir d'une PRF

Construction

Soit F une PRF. Définissons le schéma de chiffrement à clé secrète pour messages de longueur n suivant :

- $\mathrm{Gen}(1^n)$ choisit un $k \in \{0,1\}^n$ uniforme.
- Étant donné $m \in \{0,1\}^n$, choisit un $r \in \{0,1\}^n$ uniforme et soit
$$\mathrm{Enc}_k(m) = (r, m \oplus F_k(r)).$$
- Étant donné $c = (r, s)$,
$$\mathrm{Dec}_k(c) = F_k(r) \oplus s.$$



Théorème

Si F est une PRF, alors la construction ci-dessus est un schéma de chiffrement à clé secrète de longueur fixe et CPA-sûr pour les messages de longueur n.