

# HAI709I — TDs

Ivan Lejeune

15 septembre 2025

## Table des matières

TD1 — Introduction . . . . .	2
TD2 . . . . .	4

## TD1 — Introduction

**Exercice 1.1.** Répondez aux questions suivantes :

- Supposons qu'un adversaire sache qu'un message, chiffré avec le chiffrement par décalage, est soit **aaa**, **aab**, **aac**, **abc**, **acd** soit **abd**. En examinant le texte chiffré obtenu, est-il (toujours) en mesure de déterminer le message d'origine ? Expliquez votre réponse : fournissez une stratégie si la réponse est correcte ou un contre-exemple (une clé) si elle ne l'est pas.
- Considérez le point précédent, mais cette fois-ci, le message est chiffré avec le chiffrement de Vigenère. Supposons d'abord que la période soit 2, puis que la période soit 3. Est-il (toujours) en mesure de déterminer le message d'origine ?
- Cette fois-ci, les messages possibles, chiffrés avec le chiffrement de Vigenère de période 2, sont soit **abcd**, **aabc** soit **bccd**. L'adversaire est-il capable de déterminer le texte en clair à partir du texte chiffré ? Pourquoi ?

**Solution.**

- Oui, il est toujours en mesure de déterminer le message d'origine. En effet, il suffit de décaler toutes les lettres du texte chiffré jusqu'à obtenir un des messages possibles. Comme les messages possibles ne peuvent pas être obtenus par décalage les uns des autres, il y a une et une seule possibilité.
- Non, il n'est pas toujours en mesure de déterminer le message d'origine. Si 2 messages partagent les mêmes lettres aux positions 1 et 3 (pour la période 2) alors ce n'est pas possible.
- Oui, facile, regarder les écarts entre les paires.

**Exercice 1.2.** Nous chiffrons le message **abacbc** avec le chiffrement de Vigenère et une clé de période 2. Trouvez une clé telle que le texte chiffré contienne exactement 4 fois le caractère **c**. Existe-t-il une clé pour laquelle le caractère **c** apparaît plus de 4 fois dans le texte chiffré ? Et pour les autres caractères ?

**Solution.** Oui, pour obtenir exactement 4 fois un caractère donné, on peut utiliser le chiffrement suivant :

- Pour **c** : la clé est (2,0)
- Pour toute autre lettre, la clé est celle du **c** décalée de la différence entre **c** et la lettre voulue.

Ce n'est pas possible d'avoir plus de 4 fois un caractère dans le texte chiffré.

**Exercice 1.3.** Montrer que les chiffrements par décalage et par substitution mono-alphabétique ne sont pas inconditionnellement sûrs.

**Solution.** Le mot **aaa** a une probabilité non nulle d'être encodée en **bbb** avec le chiffrement par décalage ou par substitution mono-alphabétique. Ce n'est en revanche pas le cas du mot **aab**. Donc les deux chiffrements ne sont pas inconditionnellement sûrs. □


**Exercice 1.4.** A partir de la définition de la sécurité inconditionnelle que nous avons donnée, prouver la proposition vue pendant le cours :


**Proposition.** Un schéma de chiffrement (**gen**, **enc**, **dec**) avec un espace de messages  $M$  est inconditionnellement sûr si et seulement si, pour chaque  $m, m' \in M$  et chaque  $c \in C$ , on a


$$\mathbb{P}(\text{enc}_K(m) = c) = \mathbb{P}(\text{enc}_K(m') = c).$$

 **Solution.** Exercice solution

 **Exercice 1.5.**

 **Solution.** Exercice solution

 **Exercice 1.6 Optional title 2.** Exercice 2 content

 **Solution.** Exercice solution

 **Exercice 1.7 Optional title 2.** Exercice 2 content

 **Solution.** Exercice solution

## TD2

**Exercice 2.1.** Dire si les fonctions suivantes sont :

- bornées par un polynôme :

$$f_1(n) = n, \quad f_2(n) = n^3 - 4n^2 + 2n + 9, \quad f_3(n) = n \log_2(n), \\ f_4(n) = 1.1^n, \quad f_5(n) = 2^{\frac{1}{2}n}, \quad f_6(n) = 2^{\sqrt{n}}, \quad f_7(n) = n^{\log_2(n)}$$

- négligeables :

$$g_1(n) = \frac{1}{n^2 + n + 1}, \quad g_2(n) = \frac{3}{\log_2(n)}, \quad g_3(n) = 2^{-n}, \\ g_4(n) = 2^{-\frac{1}{2}n}, \quad g_5(n) = \frac{n^2 + n + 1}{e^n}, \quad g_6(n) = 2^{-\sqrt{n}}, \quad g_7(n) = n^{-\log_2(n)}$$

**Solution.**

- On a

- $f_1 = n$  est bornée par  $n^2$ ,
- $f_2 = n^3 - 4n^2 + 2n + 9$  est bornée par  $n^4$ ,
- $f_3 = n \log_2(n)$  est bornée par  $n^2$ ,
- $f_4 = 1.1^n$  n'est pas bornée par un polynôme,
- $f_5 = 2^{\frac{1}{2}n}$  n'est pas bornée par un polynôme,
- $f_6 = 2^{\sqrt{n}}$  n'est pas bornée par un polynôme,
- $f_7 = n^{\log_2(n)}$  n'est pas bornée par un polynôme.

- On a

- $g_1(n) = \frac{1}{n^2+n+1}$  n'est pas négligeable,
- $g_2(n) = \frac{3}{\log_2(n)}$  n'est pas négligeable,
- $g_3(n) = 2^{-n}$  est négligeable par rapport à  $\frac{1}{n}$ ,
- $g_4(n) = 2^{-\frac{1}{2}n}$  est négligeable par rapport à  $\frac{1}{n}$ ,
- $g_5(n) = \frac{n^2+n+1}{e^n}$  est négligeable par rapport à  $\frac{1}{n}$ ,
- $g_6(n) = 2^{-\sqrt{n}}$  est négligeable par rapport à  $\frac{1}{n}$ ,
- $g_7(n) = n^{-\log_2(n)}$  est négligeable par rapport à  $\frac{1}{n}$ ,

**Exercice 2.2.** Considérons un chiffrement de Vigenère  $\pi$  où l'espace de messages est constitué de toutes les chaînes de 3 caractères et  $\text{gen}$  fonctionne de la manière suivante : d'abord, une période  $t$  est tirée uniformément au hasard dans  $1, 2, 3$ , puis une clé  $k$  est tirée uniformément au hasard dans  $0, \dots, 25^t$ .

- Définissez un adversaire  $\mathcal{A}$  tel que  $\mathbb{P}(\text{PrivK}_{\mathcal{A}, \pi}^{\text{eav}} = 1) > \frac{1}{2}$ .
- Définissez un adversaire  $\mathcal{A}$  tel que  $\mathbb{P}(\text{PrivK}_{\mathcal{A}, \pi}^{\text{eav}} = 1) \geq \frac{3}{4}$ .

**Solution.**

**Exercice 2.3.** Soit  $G$  un PRG avec un facteur d'expansion  $l(n) > 2n$ . Dans chacun des cas suivants, prouver que  $G'$  est aussi un PRG, si tel est le cas. Sinon, trouver un contre-exemple.

- Définissez  $G' \stackrel{\text{def}}{=} G(s_1, \dots, s_{\lceil \frac{n}{4} \rceil})$ , où  $s = s_1, \dots, s_n$ .
- Définissez  $G' \stackrel{\text{def}}{=} G(s_1, \dots, s_{\lceil \frac{n}{2} \rceil})$ , où  $s = s_1, \dots, s_n$ .
- Définissez  $G' \stackrel{\text{def}}{=} G(s \parallel 0^{|s|})$ .

■ **Exercise 2.4.** TO FILL

■ **Solution.** TO FILL

■ **Exercise 2.5.** TO FILL

■ **Solution.** TO FILL