ESIEE Paris – Département Santé Energie Environnement

OUAP-4222 - « Sécurité des données »

TP N° 1 : conception et programmation d'un algorithme de chiffrement basé sur le « chiffre de Vigenère »

Alain Lacombe / 25 novembre 2021

Introduction

Le but de ce TP est de vous apprendre à concevoir et programmer un algorithme de chiffrement/déchiffrement de fichiers textes basé sur une méthode poly-alphabétique. L'implémentation choisie pour illustrer cette méthode est celle du chiffre de Vigenère. Cette technique a été conçue à l'origine pour chiffrer des textes composés à partir d'un alphabet à jeu de caractères limité (26 lettres capitales). Votre travail consistera donc à réaliser une première version adaptée au traitement de chaînes de caractère de ce type. Vous élaborerez dans un deuxième temps une version du programme permettent de chiffrer des contenus numériques de tous types (sons, images, vidéos, fichiers texte, fichiers pdf,…).

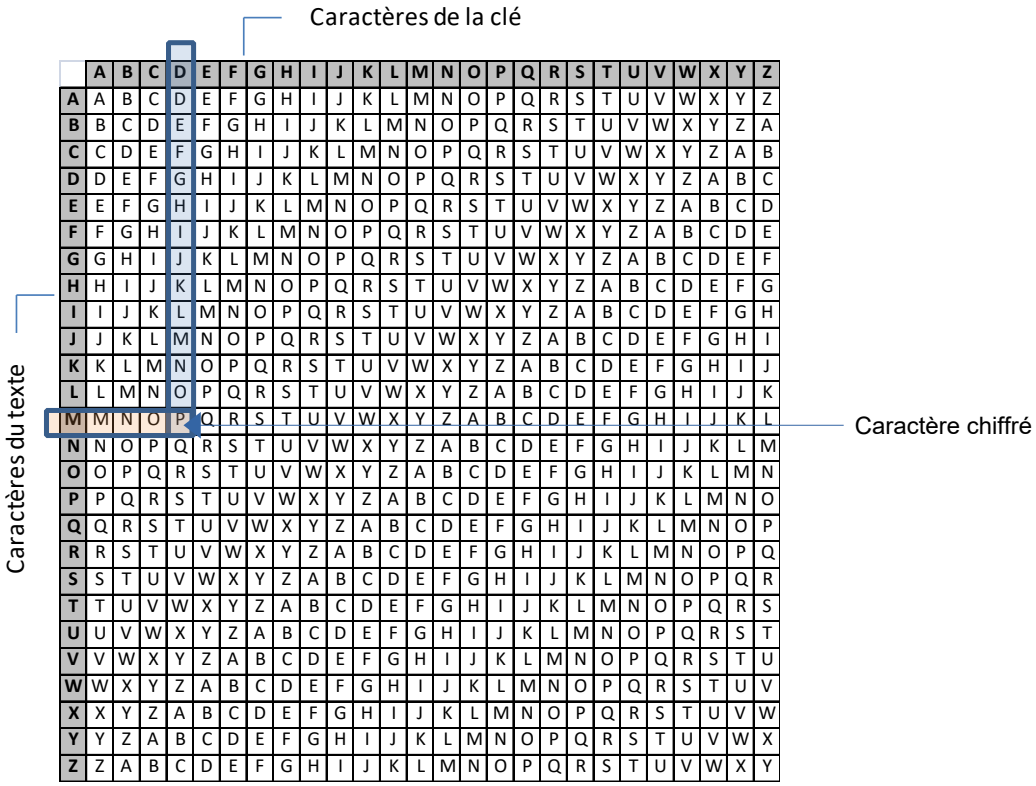
Nota bene : Il est important de soigner l 'étape de conception de l'algorithme qui constitue un préalable essentiel à l'écriture du code. La programmation sera réalisée en langage Python (syntaxe python 3).

Rappels du principe de chiffrement mis en œuvre dans le chiffre de Vigenère

Le chiffre de Vigenère est une technique qui a été élaborée pour chiffrer/déchiffrer des contenus textuels à structure alphabétique. Le principe chiffrant est basé sur l'utilisation d'un matrice de transcodage dont la structure est rappelée ci-après et d'une clé constituée à partir des caractères de l'alphabet cible. La taille de la clé (Nc) est variable et comprise entre 1 et Nt caractères (Nt étant la longueur du texte à chiffrer ou à déchiffrer).

Partie 1 : ***chiffrement / déchiffrement de Vigenère "original"***

Dans cette partie, vous concevrez l'algorithme et écrirez le programme de chiffrement/déchiffrement correspondant à la version originale conçue par Blaise Vigenère et destinée à des textes contenant seulement les 26 lettres capitales. Cette méthode utilise une matrice de transcodage dont la forme est rappelée ci-dessous :



Q1 – Exprimez en langage naturel le principe du chiffrement et du déchiffrement d'un texte constitué à partir de l'alphabet à 26 lettres capitales.

Q2 – Ecrivez et testez le programme correspondant en Python.

NOTA BENE : il est fortement conseillé de passer par une étape de conception de l'algorithme avant d'écrire le programme en langage Python. Pour cela, vous pouvez vous inspirer du formalisme d'écriture en pseudo-code proposé sur le site :

https://info.blaisepascal.fr/pseudo-code

Partie 2 : *chiffrement / déchiffrement de Vigenère "étendu"*

L'alphabet cible du chiffre de Vigenère original étant constitué des 26 lettres capitales de la langue française, il ne permet de convertir que des textes au contenu très limité. Il est cependant possible d'étendre cette technique de chiffrement à un ensemble de caractères plus large incluant les caractères accentués ainsi que certains caractères de contrôles insérés dans les textes pour en organiser la structure (tabulations, sauts de ligne, etc.), mais également à tous types de contenus numériques (fichiers textes, sons, images, vidéos, programmes exécutables, etc...).

Cette possibilité découle de l'observation suivante : tout contenu numérique (donc codé en binaire) peut être représenté sous sa forme hexadécimale qui utilise un « alphabet » à 16 caractères :

["0","1","2","3","4","5","6","7","8","9","a","b","c","d","e","f"]

Chaque caractère hexadécimal correspondant à une suite de 4 bits (**0**→0000, **1**→0001, **2**→0010,…, **9**→1001, **a**→1010,... **f**→1111), chacun des octets contenus dans un fichier numérique est représenté par 2 caractères hexadécimaux successifs (par exemple **3f** → **00111111**). Il est donc possible d'opérer le chiffrement et le déchiffrement de n'importe quel fichier à l'aide d'une matrice de transcodage

modifiée basée sur l'alphabet hexadécimal après avoir transcodé son contenu de binaire à hexadécimal. L'opération de chiffrement/déchiffrement produit alors une chaîne hexadécimale qu'il suffit de reconvertir en binaire pour obtenir le contenu du cryptogramme ou du fichier original, selon que l'on procède à un chiffrement ou à un déchiffrement.

Q6 - Créez un copie du programme précédent, réalisez les ajouts et modifications nécessaires et testez le chiffrement et le déchiffrement étendus sur les différents exemples de contenus numériques présents dans l'archive .zip disponible dans le dossier du TP N° 1 (BlackBoard).

Pour élaborer cette deuxième version, vous pouvez vous servir des fonctions fournies dans le script python nommé f*onctionsConversionFichierHexa.py* disponible dans le dossier du TP N° 1 :

- la fonction ***fichierVersChaineHexa(fichier)*** renvoie la chaîne composée de la suite de caractères hexadécimaux correspondant au contenu d'un fichier existant dont le nom est passé en paramètre.
- la fonction ***chaineHexaVersFichier(chaineHexa, fichier)*** crée un fichier dont le nom est fourni en deuxième paramètre et y enregistre le contenu binaire correspondant à la chaîne contenant une suite de caractères hexadécimaux fournie en premier paramètre.

Q7 – Observez le temps de calcul nécessaire pour chiffrer/déchiffrer les différents fichiers. Le chiffre de Vigenère est-il adapté au chiffrement de contenus volumineux ?