Cryptographie et Sécurité
Série 2 : TP Python - Cryptographie Classique

25 Septembre 2019

A rendre exclusivement sur **Moodle**, uniquement sous forme de fichier **.py**, implémenté avec **Python 3**, avant le **02 Octobre 2019 à 17h00**.

Votre code doit être **suffisamment commenté**.

Implémentation de diverses méthodes de Cryptographie Classiques

Implémentez des fonctions avec le language de programmation Python 3 (et uniquement Python 3) permettant l'utilisation des trois méthodes vues dans les exercices de la série 1, c'est à dire **le chiffrement de César, la substitution monoalphabétique, et le chiffrement de Vigenère**. Vous devez implémenter **l'encryption et la décryption** pour chacune des trois méthodes. Toutes ces fonctions prendront deux éléments en entrée, un message et un clé, et retourneront un seul élément, le message (encrypté ou décrypté selon le cas).

Vous pouvez facilement tester vos fonctions avec des exemples simples ou ceux des exercices de la série 1, et en vérifiant que lorsque vous décryptez, vous retrouvez bien le même message qu'avant l'encryption.

Format des clés

- Pour César, la clé est un entier.
- Pour la substitution monoalphabétique, la clé est une phrase, à partir de laquelle vous construirez la vraie clé de 26 caractères.
- Pour la substitution polyalphabétique (Vigenère), la clé est un mot.

Format des Messages

Vos messages seront des chaines de caractères. Vous devrez traiter uniquement les caractères en minuscules, c'est à dire avec 26 éléments dans votre alphabet :

- Si des majuscules se trouvent dans le texte, transformez les en minuscules.
- Si d'autres caractères (espaces, virgules, points, caractères avec accents, etc) se trouvent dans le texte, laissez-les en place sans les modifier.

- Si vous avez du mal, commencez par faire fonctionner vos chiffrements pour des messages uniquement constitués de minuscules non accentués. Puis ajoutez les autres éléments (d'abord les majuscules, puis le reste) étape par étape.

Pour ceux qui veulent aller plus loin

Cet exercice ne fait pas partie du rendu attendu.

Si vous voulez aller plus loin, vous pouvez créer une autre copie de votre code (ne modifiez pas celle que vous devez rendre, ce serait dommage), et cette fois, n'utilisez pas les 26 caractères de l'alphabet minuscule, mais essayez avec les 256 de l'ASCII 8 bits. Vous aurez donc 256 clés pour César, une clé de longueur 256 pour la substitution monoalphabétique, et une table avec 256 valeurs pour Vigenère.