Tutorat Programmation

L1 Informatique

Introduction

Dans cette séance de tutorat, nous vous proposons d'écrire un programme capable de chiffrer et de déchiffrer des messages selon le code de César. Pour échanger des messages sans qu'ils soient interceptés, César avait pour habitude d'écrire ses lettres avec un alphabet décallé d'un certain rang. De cette manière, le destinataire pouvait donc déchiffrer le message seulement si il avait à sa connaissance la valeur de ce décallage. Qu'entendons-nous par décallage ? Pour vous donner un exemple, la lettre E avec un décallage de 5 donne la lettre J. Aussi, la lettre L avec un décallage de 3 nous donne la lettre O. Plus compliqué, la lettre A avec un décallage de 32 nous donne G.

Nous commencerons par écrire un programme qui chiffre le message contenu dans un fichier. Dans un deuxième temps, nous écrirons une fonction qui déchiffre un fichier encodé par une méthode de force-brute. Nous verrons au moment voulu comment nous y prendre.

Vous avez à votre disposition deux fichiers squellettes *ceasar.py* et *ceasar_reverse.py* pour vous aider. Prenez le temps de lire ces fichiers pour comprendre ce qu'il doit être fait. Vous intègrerez les fonctions nécessaires (section 1, 2 et 4) dans ces fichiers quand il vous le sera indiqué.

1 La fonction d'encodage

Ecrivez une fonction encrypt_string(msg, shift) qui retourne le message encodé (chaine de caractères) avec un décallage d. Vous devrez traiter tous les caractères du message un par un. Pour décaller un caractère, vous utiliserez la fonction encrypt_char(c, shift) que vous devrez coder également (voir section 2).

2 La fonction de décallage

Vous allez devoir écrire une fonction encrypt_char(c, shift) qui retourne le caractère c décallé d'une valeur de d. La fonction ne doit traiter que les caractères entre A et Z et ne pas s'occuper des espaces et autres caractères.

Vous devrez utiliser les deux fonctions du language Python suivantes :
— ord(c) : retourne la valeur ASCII du caractère c
— chr(i) : retourne le caractère correspondant au code ASCII i

Indication : Pour décaller un caractère, il vous faut dans un premier temps connaitre son code ASCII. Vous devez ramener ce code à 0 pour pouvoir appliquer le décallage voulu, le tout modulo 26, car le code ASCII contient bien plus de valeur que notre alphabet. Pour finir il faut rajouter la valeur qu'on à retiré pour ramener notre code à 0 précédement.

3 Intégration des fonctions

Pour créer notre programme d'encodage, nous aurons besoin des fonctions écritent au dessus. Intégrez-les dans le fichier fournit *ceasar.py*.

Pour utiliser ce programme vous devez disposer d'un fichier texte contenant le message à coder et utiliser la commande suivante :

```
./ceasar.py -m message.txt -d 18
```

L'option -m sert à spécifier le fichier contenant le message. L'option -d quant à elle, sert à spécifier une valeur de décallage, ici 18. Vous obtiendrez un fichier enc_message.txt contenant votre message chiffré.

4 Décryptage par force-brute

Cette technique de décryptage est la plus simple à implémenter, mais c'est aussi la moins efficace. Elle peut vite devenir très longue. Néanmoins, dans notre cas, il s'avère que c'est la meilleure sollution. La force-brute consiste bêtement à essayer toutes les sollutions possibles pour résoudre notre problème. Ici nous allons donc essayer toutes les valeurs possibles de décallage pour déchiffrer le message codé.

Ecrivez une fonction decrypt_string(msg, shift) qui retourne le texte (msg) déchiffré avec la clé de décallage (shift). Pour cela, il vous faut vous inspirer des fonctions écritent précédement. L'idée sera d'appeler cette fonction avec toutes les clés possibles et de chercher parmis les résultat celui qui sera le bon message déchiffré.. Vous devrez intégrer cette fonction dans le fichier *ceasar_reverse.py*.

Une fois que vous avez fait tout ça, vous pouvez appeler votre programme de déchiffrage avec la commande suivante :

```
./ceasar_reverse.py -m enc_message.txt
```

Vous pouvez maintenant encoder des messages et déchiffrer des messages dont vous ne connaissez pas la clé.