Mini projet

Cryptosystème de Vigénère

Consignes générales.

Ce mini-projet porte dans un premier temps sur l'implémentation des fonctions de chiffrement et de déchiffrement de *Vigénère*, puis dans un second temps sur l'attaque de ce cryptosystème.

La production finale attendue est constituée des seuls fichiers contenant les codes sources des fonctions écrites pour répondre à la problématique posée, à savoir :

- le fichier `vigenere.py` pour les fonctions de chiffrement et de déchiffrement ;
- le fichier `attaque.py` pour les fonctions d'attaque du cryptosystème.

I. Cryptosystème de Vigénère

I. 1. Principe de la méthode de chiffrement

Un des principaux défauts du *chiffrement par décalage* est qu'une lettre (par exemple `A`) est toujours chiffrée par la même lettre (par exemple `D`).
Le chiffrement de Vigénère est une version améliorée de la méthode précédente. Il s'agit d'une méthode de *chiffrement symétrique par blocs* pour laquelle une même lettre peut être chiffrée de plusieurs façons différentes.

A titre d'exemple, on considère la phrase à chiffrer `IL ETAIT UNE FOIS`.

1) On regroupe tout d'abord les lettres de la phrase par blocs de longueur n.

   Par exemple, par blocs de longueur $n = 3$, ce qui donne `ILE TAI TUN EFO IS`.

   Remarque. Les espaces sont purement indicatifs, dans la première phrase il séparent les mots, et dans la seconde ils séparent les blocs.

2) Si n est la longueur d'un bloc, alors on choisit une clé de chiffrement k constituée de n nombres entiers compris entre 0 et 25 :

   $$k = (k_0, k_1, ..., k_{n-1})$$

   Par exemple, puisque les blocs sont de longueur $n = 3$, on peut choisir comme clé de chiffrement $k = (k_0, k_1, k_2) = (4, 2, 3)$.

3) Le chiffrement consiste alors à réaliser un chiffrement par décalage dont la valeur dépend du rang de la lettre dans le bloc :
   - un décalage de valeur k_0 pour la première lettre de chaque bloc ;
   - un décalage de valeur k_1 pour la seconde lettre de chaque bloc ;
   - ...
   - un décalage de valeur k_{n-1} pour la dernière lettre de chaque bloc.

Pour notre exemple, avec la clé $k = (k_0, k_1, k_2) = (4, 2, 3)$, on obtient pour le premier bloc `ILE` :

- un décalage de valeur 4 pour la lettre `I` ce qui donne `M` ;
- un décalage de valeur 2 pour la lettre `L` ce qui donne `N` ;
- un décalage de valeur 3 pour la lettre `E` ce qui donne `H`.

Le chiffrement de la séquence de blocs `ILE TAI TUN EFO IS` à partir de la même clé donne `MNH XCL XWQ IHR MU`, soit la phrase chiffrée `MN HXCLX WQI HRMU`.

I. 2. Principe de la méthode de déchiffrement

Pour déchiffrer la phrase il suffit d'inverser la clé, c'est à dire qu'il s'agit de la même procédure que le chiffrement mais avec la clé $k = (-k_0, -k_1, ..., -k_{n-1})$.

II. Travail demandé

II. 1. Fonctions de chiffrement et déchiffrement

L'intégralité des codes demandés est à implémenter dans le fichier `vigenere.py`.

Remarque. Afin de simplifier le chiffrement et le déchiffrement des messages, ces derniers sont écrits en lettres majuscules et sans accents.
D'autre part, lorsque les messages sont constitués de plusieurs mots ou de plusieurs phrases, on conserve les espaces entre les mots et les signes de ponctuation.

On utilise la correspondance suivante pour les lettres de l'alphabet :

lettre	A	B	C	D	E	F	G	H	I	J	K	L	M
nombre	0	1	2	3	4	5	6	7	8	9	10	11	12
lettre	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
nombre	13	14	15	16	17	18	19	20	21	22	23	24	25

1) Écrire le code de la fonction `chiffrer(phrase:str ,k:tuple) -> str` qui prend en paramètres :
   - la `phrase` à chiffrer selon le chiffrement de Vigénère,
   - la clé de chiffrement `k` sous la forme d'un tuple $k = (k_0, k_1, ..., k_{n-1})$,

   et qui renvoie la phrase chiffrée sous la forme d'une chaîne de caractères.

   A titre d'exemple, l'exécution de `chiffrer('IL ETAIT UNE FOIS', (4,2,3))` doit renvoyer en console la chaîne de caractères `'MN HXCLX WQI HRMU'`.

2) Écrire le code de la fonction `dechiffrer(phrase:str, k:tuple) -> str` qui prend en paramètres :
   - la `phrase` chiffrée selon le chiffrement de Vigénère,
   - la clé de chiffrement `k` sous la forme d'un tuple $k = (k_0, k_1, ..., k_{n-1})$,

   et qui renvoie la phrase déchiffrée sous la forme d'une chaîne de caractères.

   A titre d'exemple, l'exécution de `dechiffrer('MN HXCLX WQI HRMU', (4,2,3))` doit renvoyer en console la chaîne de caractères `'IL ETAIT UNE FOIS'`.

II. 2. Attaque du cryptosystème de Vigénère

On cherche maintenant à utiliser les fonctions précédentes pour déchiffrer le message suivant qui a été obtenu à partir d'une *clé de longueur 4* :

```
DL ZHGIVUEL OD UL LQK TYDVL OIL XEU DLC YEIOSASOI KVXJ KBBI WA PYWYC T WBH
QDVBI IBO BWZ QUFZ SDLLVBANODLZ CEFA OCHSSI VL NEMAOI
```

L'attaque envisagée utilise la faille de sécurité suivante : *lorsque les messages sont constitués de plusieurs mots ou de plusieurs phrases, on conserve les espaces entre les mots et les signes de ponctuation.*

Son principe est le suivant :

a) sachant que le message en clair commence par un mot en langue française de deux lettres, établir une liste de mots possibles ;

b) pour chacun des mots de deux lettres envisagés, déterminer la valeur du n-uplet (k_0, k_1) de sorte qu'une fois chiffré ce mot correspond à la séquence `DL` ;

c) dans le texte chiffré, rechercher des mots de trois lettres ayant pour préfixe la séquence `DL` et dont les deux premières lettres `D` et `L` sont chiffrées par la même partie (k_0, k_1) de la clé de chiffrement ;

d) à partir de la liste des mots de deux lettres, établir une liste de mots possibles de trois lettres ;

e) pour chacun des mots de trois lettres envisagés, déterminer les valeurs correspondantes du n-uplet (k_0, k_1, k_2) ;

f) pour chaque valeur du préfixe (k_0, k_1, k_2) de la clé de chiffrement, tester l'ensemble des valeurs de k_3 ;

g) rechercher manuellement dans la liste des résultats l'énoncé probable du message en clair.

L'intégralité des codes demandés est à implémenter dans le fichier `attaque.py`.

8) Définir puis écrire les codes des fonctions et/ou des procédures nécessaires à l'attaque du cryptosystème de Vigènere à partir de la connaissance de la longueur de la clé de chiffrement et du message chiffré :

   ```
   DL ZHGIVUEL OD UL LQK TYDVL OIL XEU DLC YEIOSASOI KVXJ KBBI WA PYWYC T
   WBH QDVBI IBO BWZ QUFZ SDLLVBANODLZ CEFA OCHSSI VL NEMAOI
   ```

   Préciser l'énoncé du message en clair ainsi que la valeur de la clé de chiffrement utilisée.