Le chiffrement de César et variantes

Louis Granboulan

23 février 2000

1 Motivations

Il s'agit d'apprendre à manipuler les structures de données de `perl`. On fera un programme de chiffrement par substitution alphabétique, puis on utilisera un dictionnaire pour faire un programme qui déchiffre un texte chiffré avec une clef inconnue.

Tous les programmes fonctionneront en *filtres*, c'est-à-dire qu'ils lisent leurs données sur l'entrée standard, que leur résultat est sur la sortie standard, et qu'ils peuvent admettre des options en ligne de commande.

2 Exercices

2.1 Chiffrement par rotation circulaire

Dans le texte, toute lettre dont le numéro dans l'alphabet est n est remplacée par la lettre $n+3$ (modulo la taille de l'alphabet, bien évidemment). Une variante du chiffrement de César est d'utiliser un entier quelconque au lieu de 3, ce qu'on appelle chiffrement par rotation circulaire.

1. Faire un filtre à partir de la commande `tr` qui effectue le chiffrement de César. Faire aussi le filtre qui effectue le déchiffrement de César.
2. Faire en perl un petit filtre `rot` qui a comme paramètre l'entier qui sert de décalage pour un chiffrement par rotation circulaire.
3. Faire la même chose en shell, en utilisant `tr` et `expr`.

NB : vous pouvez trouver une correction sur `http ://www.di.ens.fr/~vaudenay/mmfai/tp1.html`.

2.2 Substitution alphabétique

Une généralisation est de chiffer en utilisant une bijection de l'alphabet. En pratique, on se rappelle de cette bijection par une astuce mnémotechnique, qui sert de clef.

Par exemple, la clef est un mot quelconque. On en élimine les lettres doubles, puis on rajoute à sa suite les lettres de l'alphabet[1] qui ne sont pas dans ce mot, dans l'ordre alphabétique. On écrit ceci sur cinq ligne decinq caractères. On lit par colonnes pour obtenir la valeur de la permutation.

Exemple : la clef `CRYPTOLOGIE` donne le tableau ci-contre et donc le chiffrement correspond à la commande `tr A-VXYZ COAJSRLBKUYGDMVPIFNXTEHQZ`.

C	R	Y	P	T
O	L	G	I	E
A	B	D	F	H
J	K	M	N	Q
S	U	V	X	Z

1. Faire en perl un filtre qui a comme paramètre la clef et qui fit un tel chiffrement.
2. Faire la même chose en shell, en fabriquant les bonnes options pour `tr`.

2.3 Cryptanalyse

Ce genre de système se casse dès qu'on connait la langue utilisée. On utilisera par exemple les dictionnaires `/usr/dict/words` et `/usr/local/util/lib/words.french` pour connaître la fréquence des lettres et savoir quels sont les mots.

1. Faire un premier programme qui calcule la fréquence des lettres dans le dictionnaire. NB : ceci n'est bien évidemment pas la même chose que la fréquence d'apparition des lettres dans un texte, mais cela servira de première approximation.
2. Modifier ce programme pour en faire un programme de cryptanalyse qui calcule les fréquences des lettres dans le texte chiffré, les classe par fréquence décroissante et leur associe les lettres de l'alphabet par fréquence décroissante dans le dictionnaire. NB : on remarque bien évidemment que les fréquences dans le texte ne respectent quasi jamais les fréquences du dictionnaire.
3. Modifier le programme ci-dessus pour choisir parmi les petites variations de l'ordre des fréquences celle qui fait apparaître dans le texte déchiffré le plus de mots du dictionnaire.

[1] On suppose ici que l'alphabet a 25 lettres, le W reste inchangé et on écrit tout en majuscules.