Sécurité Appliquée : chiffrement symétrique, TP

Jean-François COUCHOT
couchot[arobase]femto-st[point]fr

15 février 2022

Les deux premiers exercices sont inspirés de [1].

1 Substitution mono alphabétique

Exercice 1.1 (Décryptage par analyse fréquentielle, essentiellement). *Le texte ci-dessous est le le résultat du chiffrement par substitution mono-alphabétique d'un texte en français.*

```
bfj onflj gxs, rfixsj rf zykuefxjfj czzffj, jyzh c bc hfhf rfj cekffj leczocsjfj, yzh lyekf xz wyxdfezfkfzh.
of wyxdfezfkfzh, cbbfwxczh bc rflcshf rf zyj cekffj, j'fjh ksj fz eciiyeh cdfo b'fzzfks iyxe ofjjfe bf oykuch.
ofehfj, zyxj cdyzj fhf, zyxj jykkfj jxukfewfj ice bc lyeof kfoczsgxf, hfeefjhef fh cfesfzzf rf b'fzzfks.
szlszskfzh ibxj gxf bfxe zykuef, of jyzh bfj oncej, bfj cdsyzj, bc hcohsgxf rfj cbbfkczrj gxs zyxj lyzh efoxbfe.
of jyzh bfj oncej, bfj cdsyzj, bc hcohsgxf rfj cbbfkczrj gxs yzh jxeiesj zyj onflj cx iyszh rf bfj
ckfzfe bc yx sbj fz jyzh cxtyxer'nxs.
kcsj bf rfezsfe kyh fjh-sb rsh ? b'fjifeczof rysh-fbbf rsjicecshef ? bc rflcshf fjh-fbbf rflszshsdf ? zyz !
oeypfm-kys, kys gxs dyxj icebf fz oyzzcsjjczof rf ocxjf fh dyxj rsj gxf esfz z'fjh iferx iyxe bc leczof.
bfj kfkfj kypfzj gxs zyxj yzh dcszoxj ifxdfzh lcsef dfzse xz tyxe bc dsohysef.
oce bc leczof z'fjh icj jfxbf ! fbbf z'fjh icj jfxbf ! fbbf z'fjh icj jfxbf !
fbbf c xz dcjhf fkisef rfeesfef fbbf.
fbbf ifxh lcsef ubyo cdfo b'fkisef ueshczzsgxf gxs hsfzh bc kfe fh oyzhszxf bc bxhhf.
fbbf ifxh, oykkf b'czwbfhfeef, xhsbsjfe jczj bskshfj b'skkfzjf szrxjhesf rfj fhchj-xzsj.
ofhhf wxfeef z'fjh icj bskshff cx hfeeshysef rf zyhef kcbnfxefxv icpj.
ofhhf wxfeef z'fjh icj heczonff ice bc uchcsbbf rf leczof.
ofhhf wxfeef fjh xzf wxfeef kyzrscbf.
hyxhfj bfj lcxhfj, hyxj bfj efhcerj, hyxhfj bfj jyxlleczofj z'fkifonfzh icj gx'sb p c, rczj b'xzsdfej,
hyxj bfj kypfzj iyxe foecjfe xz tyxe zyj fzzfksj.
lyxreypfj cxtyxer'nxs ice bc lyeof kfoczsgxf,
zyxj iyxeeyzj dcszoef rczj b'cdfzse ice xzf lyeof kfoczsgxf jxifesfxef.
bf rfjhsz rx kyzrf fjh bc.
kys, wfzfecb rf wcxbbf, cohxfbbfkfzh c byzrefj,
t'szdshf bfj yllsosfej fh bfj jybrchj leczocsj gxs jf heyxdfzh fz hfeeshysef
ueshczzsgxf yx gxs dsfzrecsfzh c j'p heyxdfe, cdfo bfxej cekfj yx jczj bfxej cekfj,
t'szdshf bfj szwfzsfxej fh bfj yxdesfej
jifoscbsjfj rfj szrxjhesfj r'cekfkfzh gxs jf heyxdfzh fz hfeeshysef ueshczzsgxf yx gxs dsfzrecsfzh c j'p heyxdfe,
c jf kfhhef fz eciiyeh cdfo kys.
gxys gx'sb ceesdf, bc lbckkf rf bc efjsjhczof leczocsjf zf rysh icj j'fhfszref fh zf j'fhfszrec icj.
rfkcsz, oykkf cxtyxer'nxs, tf icebfecs c bc ecrsy rf byzrefj.
```

Décrypter progressivement ce texte sachant que :

— *les lettres les plus fréquentes en français sont E, A, S, I, N, . . . par ordre décroissant* [2]

— *les digrammes les plus fréquents en français sont ES, LE, EN, DE, RE, . . . par ordre décroissant* [3].

2 Substitution poly-alphabétique

Exercice 2.1 (Décryptage de chiffré selon Vigenère (non corrigé en TP)). *Le texte ci-dessous est le le résultat du chiffrement de Vigenère d'un texte en français.*

```
gmyxzoocxziancxktanmyolupjrztgxwshctzluibuic
yzwxyqtvqxzukibkotuxkagbknmimmzzyajvjzampqyz
loinoiqknaumbknknvkaiakgwtnilvvzvqydmvjcximr
vzkilxzqtomrgqmdjrzyazvzmmyjgkoaknkuiaivknvvy
```

1. Vergnaud, Damien. Exercices et problèmes de cryptographie-3e éd. Dunod, 2018.
2. https://fr.wikipedia.org/wiki/Fr%C3%A9quence_d%27apparition_des_lettres_en_fran%C3%A7ais
3. https://fr.wikipedia.org/wiki/Analyse_fr%C3%A9quentielle#Analyse_fr%C3%A9quentielle_des_digrammes

1. *Utilisez l'indice de coïncidence pour déterminer la longueur de la clef.*
2. *Utilisez une analyse fréquentielle pour déterminer celle-ci.*
3. *Exploitez un carré de Vigenère pour déchiffrer alors le message.*

3 AES par la pratique

On va utiliser une bibliothèque de cryptographie pour chiffrer avec AES.

Exercice 3.1 (Exploitation d'un code pour chiffrer une image selon AES).

```
from Crypto.Cipher import AES

message = "The answer is 1.The answer is 2.The answer is 1.".encode("utf-8")
#Part 1
aes_ecb_e = AES.new("This is a key123".encode('utf-8'), AES.MODE_ECB)
ciphertext_aes_ecb = aes_ecb_e.encrypt(message)
print("ciphertext with ECB: " + str([x for x in ciphertext_aes_ecb]))

aes_ecb_d = AES.new("This is a key123".encode("utf-8"), AES.MODE_ECB)
res=aes_ecb_d.decrypt(ciphertext_aes_ecb)
print(res.decode("utf-8"))

#Part 2
aes_cbc_e = AES.new("This is a key123".encode('utf-8'), AES.MODE_CBC, "This is an IV456".encode('utf-8'))
ciphertext_aes_cbc = aes_cbc_e.encrypt(message)
print("ciphertext with ECB: " + str([x for x in ciphertext_aes_cbc]))

aes_cbc_d = AES.new("This is a key123".encode('utf-8'), AES.MODE_CBC, "This is an IV456".encode('utf-8'))
res=aes_cbc_d.decrypt(ciphertext_aes_cbc)
print(res.decode("utf-8"))
```

1. *Expliquez le programme précédent, notamment les différences entre la partie 1 et la 2*
2. *Exécutez le programme.*
3. *Chiffrez et déchiffrez une chaîne de caractère de votre choix avec AES et CBC.*
4. *Chiffrez une image avec AES et ECB. Affichez l'image chiffrée. Que remarquez-vous ? Vous pouvez utiliser le programme suivant pour manipuler des images.*
5. *Chiffrez une image avec AES et CBC. Affichez l'image chiffrée. Que remarquez-vous ?*

```
from Crypto.Cipher import AES
from PIL import Image

im = Image.open("imgISIFC.png")
#im.show()
message = im.tobytes()
aes_ecb_e = AES.new('This is a key123'.encode("utf-8"), AES.MODE_ECB)
ciphertext_aes_ecb = aes_ecb_e.encrypt(message)

#output the encrypted image
imb= Image.frombytes(im.mode,im.size,ciphertext_aes_ecb)
imb.show()
```