Exercice 1 (Chiffrement de Vigenère par substitution polyalphabétique)

Soit $\Gamma = (\mathsf{Dec}, \mathsf{Enc}, \mathcal{M}, \mathcal{C})$ le cryptosystème suivant :

— L'ensemble $\mathcal{M}$ des messages en clair est l'ensemble des mots sur l'alphabet $\mathcal{A} = [\mathtt{a}, \mathtt{z}]$

— Une clé k est la donnée de n permutations $\langle \sigma_0, \ldots, \sigma_{n-1} \rangle$, où chaque σ_i est une permutation de $[\mathtt{a}, \mathtt{z}]$. L'ensemble des clés $\mathcal{K}$ est l'ensemble des séquences de permutations de $[\mathtt{a}, \mathtt{z}]$. La clé est choisie aléatoirement de façon uniforme parmi toutes les permutations.

— Soient $m = a_0 \ldots a_{\ell-1} \in \mathcal{M}$ et $k = \langle \sigma_0, \ldots, \sigma_{n-1} \rangle \in \mathcal{K}$.

$$\mathsf{Enc}_k(m) = \sigma_0(a_0)\sigma_1(a_1)\ldots\sigma_{i \mod n}(a_i)\ldots\sigma_{\ell-1 \mod n}(a_{\ell-1})$$

1. Définir l'algorithme de déchiffrement Dec.
2. Est-il facile de concevoir une attaque à texte clair choisi ?
3. Est-il facile de concevoir une attaque à texte clair connu ?
4. Est-il facile de concevoir une attaque à texte chiffré ? Pour chaque attaque, indiquer la longueur des textes nécessaires pour retrouver la clé.

Exercice 2 (Cryptanalyse du chiffrement par substitution monoalphabétique)

Le but de cet exercice est de se rendre compte par la pratique de la faiblesse du chiffrement par substitution monoalphabétique. Le texte en clair est entièrement écrit en minuscules, sans espaces ni accents (pourquoi ?).

Retrouver le texte en clair, à partir des tables des fréquences des caractères, digrammes et trigrammes du texte chiffré.

qbjo raep fbjp bnka qjpb qrwb fkna rekm kabe nbnb likt gbxr gffb wfrp
....
ganj wkfq bjxr gfek gctg bqre pbmr fkan pger baqk mapb bnwk jjar eebo
....
kaja qbnk ankl fkae pfbt gbpb jrfo kajl bpam apbe pbpa enbf bnpa jnfa
....
hgbk gwgh qale bygn yrfn paoa egbb nlik lgej byyf kdka npbe kmra fwqg
....
jknr glib fjkw kfnp born arej tgrn apab eebj aqeb eygn fabe qael apbe
....
nqbw qgja eknn bepg qbwq gjbc nfkr fpae kafb qbwq gjae lfrd khqb qbwq
....
gjae mfka jboh qkhq bmae nyke knaj bfke rgmb kgqb jbjw fanj ikqb nken
....
jbnf bxbn bfqb orep bben abfj rgjq blrg wpge bwra zeke nbjg fbcl ankn
....
areg exrg fpbj bwnb ohfb knfr ajib gfbj tgkf kenb jbwn oaeg nbjp gjra
....
fgen bqbz fkoo bnfk ejoa jwkf qblg hqba oobf zbbe nfbm kqbe naka fqke
....
pbnb ffbe bgmb bnqk lrnb kobf alka ebkf famk kqkp fbjj bpgw fbja pben
....
hkfh alke bqbw fbja pben hkfh alke bfro wanq bemb qrww bqgn qkpb wbli
....
bbnt gbqt gbyg njre wrgm rafj gfqg aobo bjbj qbmf bjwg qafb enjb jdbg
....
cjbn frgh qbfb enkq kqbl ngfb pbjm aezn ornj pblb nbqb zfko obmr alaq
....
bnbc nbpb lbnn bpbw blib tgay azgf boka enbe kenk gckf liam bjpg zgel
....
qghf bowq klbu rhgj jwib fatg bwkf wfrx blna qbld qaep frlr eatg bwkf
....
nafk apbp kejk ffam bfka wkfj nbko bfkn qken kjal bnnb yrgp frdk enbe
....
rgmb qqbk gqab gpbm rqbf jgfq bjya qjbq blnf atgb jygn kffa mbbj aowq
....
bobe nwkf qkwr jnbb njrg jbem bqrw wblk libn bbja qbjb owqr dbjy fkel
....
kaja fqke pkaj nbff bebg mabe jkob falk aeje bgjj benw kjbn bebl bjjk
....

Lettres				Trigrammes				Digrammes					
b	191	o	28	ben	11	wqg	5	qb	30	fk	15	bo	10
a	92	m	23	tgb	8	kaj	5	bj	27	kf	14	na	9
k	90	h	14	bnb	7	qgj	5	be	25	bl	13	bb	9
f	87	y	12	enb	7	qbl	5	nb	24	bq	13	ob	9
n	84	t	11	wkf	7	bbn	5	pb	22	bw	13	gf	9
e	77	i	10	apb	6	qbw	5	fb	21	ja	12	nf	9
j	75	z	7	bqb	6	gfb	5	bn	21	eb	12	aj	9
g	69	c	6	fbj	6	lib	4	en	21	rg	12	nk	9
q	65	d	6	ken	6	bpb	4	ka	18	tg	11	wk	9
r	49	x	5	bja	5	epb	4	jb	17	qk	11	ep	9
p	42	u	1	fka	5	aqb	4	bf	17	fa	10	ra	8
w	39	s	0	bjb	5	fqb	4	ae	16	gb	10	mb	8
l	35	v	0	pbe	5	hqb	4	ke	15	gj	10	re	8

lettre	fréq.	lettre	fréq.
A	8.40 %	N	7.13 %
B	1.06 %	O	5.26 %
C	3.03 %	P	3.01 %
D	4.18 %	Q	0.99 %
E	17.26 %	R	6.55 %
F	1.12 %	S	8.08 %
G	1.27 %	T	7.07 %
H	0.92 %	U	5.74 %
I	7.34 %	V	1.32 %
J	0.31 %	W	0.04 %
K	0.05 %	X	0.45 %
L	6.01 %	Y	0.30 %
M	2.96 %	Z	0.12 %

TABLE 1 – Fréquence d'apparition des lettres en français

lettre	fréq.	lettre	fréq.
W	0.04 %	P	3.01 %
K	0.05 %	C	3.03 %
Z	0.12 %	D	4.18 %
Y	0.30 %	O	5.26 %
J	0.31 %	U	5.74 %
X	0.45 %	L	6.01 %
H	0.92 %	R	6.55 %
Q	0.99 %	T	7.07 %
B	1.06 %	N	7.13 %
F	1.12 %	I	7.34 %
G	1.27 %	S	8.08 %
V	1.32 %	A	8.40 %
M	2.96 %	E	17.26 %

TABLE 2 – Les lettres ordonnées par leur fréquence d'apparition

Digramme	#occ.	trigramme	#occ.
ES	3318	ENT	900
DE	2409	LES	801
LE	2366	EDE	630
EN	2121	DES	609
RE	1885	QUE	607
NT	1694	AIT	542
ON	1646	LLE	509
ER	1514	SDE	508
TE	1484	ION	477
EL	1382	EME	472
AN	1378	ELA	437
SE	1377	RES	432
ET	1307	MEN	425
LA	1270	ESE	416
AI	1255	DEL	404
IT	1243	ANT	397
ME	1099	TIO	383
OU	1086	PAR	360
EM	1056	ESD	351
IE	1030	TDE	350

TABLE 3 – Digrammes et trigrammes les plus fréquents et leur nombre d'occurrences dans un texte français de 100000 signes

Exercice 3 (Modes d'utilisation des chiffrements par blocs)

Soit $\Gamma = (\mathsf{Enc}, \mathsf{Dec}, \mathcal{M}, \mathcal{C})$ un cryptosystème sur $\mathcal{M} = \mathcal{C} = \{0,1\}^n$. On souhaite utiliser ce cryptosystème pour chiffrer/déchiffrer des messages m (mots binaires) de taille quelconque $|m| > n$.

Pour cela le mot m est découpé en blocs $m_1, \ldots, m_\ell$ tels que $m = m_1 m_2 \ldots m_\ell$. Le chiffrement c_j de chaque bloc m_j est une fonction des blocs $m_1, \ldots, m_j$. Cette fonction peut évidemment appliquer l'algorithme de chiffrement Enc sur l'un ou plusieurs de ces blocs.

1. Écrire les fonctions de déchiffrement des modes ECB (Electronic Codebook), CBC (Cipher-block Chaining), CFB (Cipher Feedback) et OFB (Output Feedback). Ces modes sont associées aux fonctions $c_j = \mathsf{Enc}_k(m_j)$, $c_j = \mathsf{Enc}_k(c_{j-1} \oplus m_j)$, $c_j = m_j \oplus \mathsf{Enc}_k(c_{j-1})$ et $c_j = m_j \oplus z_j$ avec $z_j = \mathsf{Enc}_k(z_{j-1})$.
2. Si une erreur de transmission venait à altérer l'un des bits de l'un des blocs du texte chiffré, pouvez-vous estimer pour chacun des 4 modes la différence entre le texte clair initial et le texte déchiffré ?
3. Définir un nouveau mode de chiffrement par blocs utilisant une et une seule fois les opérateurs Enc_k et $\oplus$ (XOR). Présente-t-il un intérêt ?

Exercice 4 (Sécurité inconditionnelle)

Le but de cet exercice est d'étudier les conditions dans lesquelles les chiffrements par décalage, par substitution monoalphabétique ainsi que le chiffrement de Vigenère sont inconditionnellement sûrs.

1. Montrer que si un seul caractère est chiffré, le chiffrement par décalage est inconditionnellement sûr.
2. Quel est la plus grande taille de l'espace des messages en clair $\mathcal{M}$ pour laquelle le chiffrement par substitution monoalphabétique est inconditionnellement sûr ?
3. Montrer comment utiliser le chiffrement de Vigenère pour chiffrer tout mot de longueur t de façon inconditionnellement sûr.

Exercice 5 (Chiffrement de Vernam)

Lorsqu'on utilise le chiffrement de Vernam (Masque jetable) avec la clé $k = 0^\ell$, on obtient $\mathsf{Enc}_k(m) = 0^\ell \oplus m = m$ et le message est donc envoyé en clair ! Il est suggéré pour améliorer le cryptosystème d'éliminer 0^ℓ de l'ensemble des clés.(C'est-à-dire que l'algorithme Gen choisit la clé aléatoirement et uniformément parmi l'ensemble des chaînes binaires de longueur ℓ différentes de 0^ℓ). Ce nouveau système est-il inconditionnellement sûr ? Donner une démonstration qui justifie votre réponse. En particulier, si votre réponse est positive, expliquer pourquoi le chiffrement de Vernam n'est pas décrit de cette façon. Si votre réponse est négative, réconcilier celle-ci avec le fait que pour la clé 0^ℓ, le chiffrement ne change pas le texte en clair.

Exercice 6 (Attaque exhaustive de cryptosystèmes à clé secrète)

Soit $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ un cryptosystème à clé secrète sur l'ensemble de messages en clair $\mathcal{M}$. Soient $\mathcal{K}$ et $\mathcal{C}$ les ensembles des clés et des messages chiffrés de ce cryptosystème. Supposons les égalités $\mathcal{M} = \mathcal{C} = \mathcal{K}$ et supposons que l'on sache efficacement calculer pour un message fixé $m_0 \in \mathcal{M}$ toute image inverse de la fonction $f : \mathcal{K} \to \mathcal{C}$ définie par $f(k) = \mathsf{Enc}_k(m_0)$. Quelle attaque peut-on

mener ? Préciser notamment sa nature et sa complexité en temps et en espace en fonction de celles du calcul de f^{-1}.

Exercice 7 (Générateur pseudo aléatoire matériel)

Un *registre à décalage à rétroaction linéaire (RDRL)* est un moyen matériel de générer efficacement des suites de bits pseudo aléatoires. Un RDRL consiste en un registre R à décalage à p positions. Initialement, le bit à la position i est le $i$ème bit de la clé k ($R[i] = k[i]$). A chaque cycle d'horloge, un nouveau bit de la suite pseudo aléatoire est généré et les opérations suivantes sont exécutées :

— Le bit $R[1]$ est renvoyé (c'est le bit généré), comme dernier bit de la suite ;
— Les bits $R[2], \ldots, R[p]$ sont décalés d'un pas vers la gauche ;
— La nouvelle valeur de $R[p]$ est donnée par

$$\sum_{j=1}^{p-1} \alpha_j R[j] \mod 2$$

où $\alpha_j \in \{0, 1\}$, pour chaque $j, 1 \leq j \leq p$ (rétroaction linéaire).

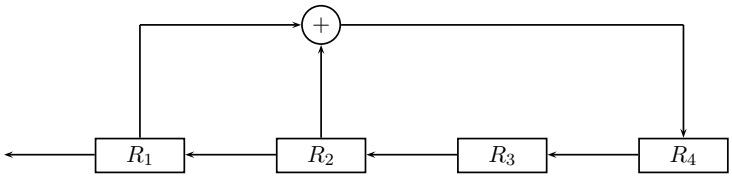

FIGURE 1 – Un registre à décalage et à rétroaction linéaire

1. Donner l'équation qui correspond au registre de la Figure 1. Montrez que la suite générée est ultimement périodique. Quelle est la période maximale ? Étudiez les variations de la période en fonction de la valeur de la clé.
2. *Attaque à texte clair connu.* Soit le RDRL défini par l'équation

$$z_{p+i} = \sum_{j=1}^{p-1} \alpha_j z_{i+j} \mod 2 \tag{1}$$

où $\alpha_j, 0 \leq j \leq p-1$ sont des constantes $\in \{0, 1\}$ inconnues de l'attaquant. Un message m est chiffré par Alice en $c = m \oplus s$ où s est la suite de bits générée par le RDRL décrit à l'équation 1 initialisé avec la clé k.

L'attaquant obtient le texte chiffré c et connaît également les ℓ premiers bits $m|_{1..\ell}$ du message en clair. En supposant que p est connu, expliquez comment l'attaquant peut calculer la suite s et ainsi retrouver la totalité du texte en clair.
3. Que pensez vous de l'utilisation de RDRL dans le cadre d'applications cryptographiques ? Argumentez votre réponse.

Exercice 8 (Chiffrement double)

Il a été suggéré d'augmenter la sécurité du DES en appliquant deux fois la fonction de chiffrement du DES avec deux clés différentes. Soit $DES_k : \{0,1\}^\ell \to \{0,1\}^\ell$ la fonction de chiffrement par bloc du DES. On note n la taille des clés k. La nouvelle fonction de chiffrement est alors $F_{k_1,k_2} : x \longrightarrow DES_{k_2}(DES_{k_1}(x))$.

Supposons que l'on dispose d'un ensemble de paires textes clairs/textes chiffrés $\{(x_1, y_1), \ldots, (x_a, y_a)\}$ obtenus avec la même clé (k_1, k_2), c'est-à-dire $y_i = F_{k_1,k_2}(x_i), \forall i, 1 \leq i \leq a$.

1. Montrer que $\forall i, 1 \leq i \leq a, DES_{k_1}(x_i) = DES^{-1}_{k_2}(y_i)$. Expliquez pourquoi le nombre de clés (k, k') telles que $\forall i, 1 \leq i \leq a, F_{k,k'}(x_i) = y_i$ est approximativement $2^{2n-a\ell}$.
2. En supposant que $a \geq \frac{2n}{\ell}$, concevoir une attaque de complexité en temps $O(2^n)$ qui permet de retrouver la clé (k_1, k_2). On pourra pré-calculer deux listes triées. Chaque élément de chacune des liste est un couple (z, k) où k est une clé et z est obtenu par application de DES_k ou DES^{-1}_k. Évaluez la complexité en temps et en espace de votre attaque.
3. Que peut-on dire sur la sécurité de double DES ?

Exercice 9 (Inversion d'une fonction à sens unique)

Soit $f : \mathcal{K} \to \mathcal{K}$ une fonction pour lequel le calcul de $f(x)$ pour un quelconque $x \in \mathcal{K}$ a pour complexités en temps et en espace $O(1)$. L'objet de cet exercice est de fournir différents algorithmes de calcul de f^{-1} et d'en estimer les complexités. Dans chacun des cas suivants, calculer la complexité en temps et en espace des attaques suivantes :

1. Attaque exhaustive sans précalcul
2. Attaque exhaustive avec précalcul

   Une attaque exhaustive peut se dérouler en deux étapes :
   — un précalcul calculant une table contenant tout couple $(f(k), k)$ avec $k \in \mathcal{K}$.
   — l'attaque proprement dite.
3. Compromis espace-temps

   L'intérêt du compromis espace-temps est d'obtenir une complexité en temps de l'attaque en $O(|\mathcal{K}|^c)$ et une complexité en espace du précalcul en $O(|\mathcal{K}|^d)$ avec $0 < c < 1, 0 < d < 1$ (la complexité en temps du précalcul est $O(|\mathcal{K}|)$. Hellman implémente cette idée (voir l'ouvrage Cryptography Theory and Pratice de D. Stinson) et obtient une attaque du D.E.S avec $c = d = \frac{2}{3}$.

   Indication : lors du précalcul, on calcule une table contenant $k_0, f^T(k_0), f^{2\cdot T}(k_0), \ldots$, où k_0 est fixé et $1 < T < |\mathcal{K}|$ est un entier fixé que vous choisirez. Afin de simplifier, nous supposerons que f est bijective et possède un unique cycle c'est-à-dire qu'il existe $k_0 \in \mathcal{K}$ tel que tout élément $k \in \mathcal{K}$ est de la forme $f^n(k_0)$ où n est un entier.

On pourra donner une valeur précise des complexités en temps et en espace des différentes attaques en supposant :
— un espace $\mathcal{K}$ de cardinalité 2^{56} (cas du D.E.S) ou 2^{128} (cas de A.E.S)
— qu'une instruction élémentaire s'exécute en $10^{-9}s$.
— que 10 Gigaoctets coûtent 1 euro.

D'autres hypothèses peuvent être faites : en 1993, Wiener proposa de construire une machine d'un million de dollars contenant 57600 puces, réalisant chacune 50000 chiffrements D.E.S par seconde.

Exercice 10 (Fonction à sens unique)

Une fonction à sens unique est une fonction facile à calculer mais difficile à inverser.

1. Soit $f : \{0,1\}^* \to \{0,1\}^*$. Montrer que l'inverse de f est calculable.
2. Démontrer que l'existence de fonctions à sens unique implique que P $\neq$ NP.

Voici une définition formelle de la notion de fonction à sens unique. Une expérience dans laquelle un « challenger » cherche à vérifier qu'un « adversaire » $\mathcal{A}$ est capable d'inverser une fonction f est définie comme suit : Expérience $\textsc{Invert}_{\mathcal{A},f}(n)$

— Le challenger choisit aléatoirement $x \in \{0,1\}^n$ et calcule $y = f(x)$
— $\mathcal{A}$ reçoit y en entrée et produit x'
— Le résultat de l'expérience est 1 si $f(x') = y$ et 0 sinon.

Définition Une fonction $f : \{0,1\}^* \to \{0,1\}^*$ est *à sens unique* si et seulement si :

— Il existe un algorithme de complexité polynomiale en temps qui sur chaque entrée x produit $f(x)$;
— Pour tout algorithme probabiliste de complexité polynomiale en temps $\mathcal{A}$, il existe une fonction négligeable negl [1] telle que

$$Pr[\textsc{Invert}_{\mathcal{A},f}(n) = 1] \leq \mathsf{negl}(n)$$

3. Soit $f : \mathbb{N} \times N \to \mathbb{N}$ la fonction $p, q \longrightarrow p \cdot q$. f est-elle à sens unique ?

1. Une fonction negl est *négligeable* si pour tout entier d, il existe un entier n_d tel que $\forall n \geq n_0, \mathsf{negl}(n) \leq \frac{1}{n^d}$.