TD 1: Fondements de la sécurité

Exercice 1

Soit le cryptogramme suivant : **H A W U D R U G L Q D L U H**

1. En utilisant l'algorithme de césar, le cryptanalyste teste l'ensemble des clés possibles pour essayer de déchiffrer le cryptogramme.
   Au bout de combien d'essai, le cryptanalyste parvient à identifier la bonne clé ? Justifier votre réponse ?
   26 clés si l'espace est comptabilisé dans la clé
   25 clés si l'espace est n'est pas comptabilisé dans la clé
2. Utiliser l'algorithme de César (clé = 3) pour déchiffrer le cryptogramme ci-dessus.
   EXTRAORDINAIRE
3. L'algorithme de César est un crypto-système mono alphabétique ou poly-alphabétique ? Justifier votre réponse ?

mono alphabétique car chaque lettre du message est remplacé par une autre lettre de l'alphabet de manière unique

4. Quels sont les inconvénients du crypto-système de César ?

Le langage du message clair est connu et facilement identifiable
Maximum 26 possibilités de clé à essayer
La distribution fréquentielle des symboles est préservée dans le *ciphertext.*
Vulnérabilité aux attaques de cryptanalyse statistique : il suffit de calculer la fréquence d'apparition de chaque symbole dans le *ciphertext* et de le comparer aux fréquences d'apparition des lettres de l'alphabet dans une langue particulière.
→ cryptanalyse possible facilement : n'est pas sécuritaire :

5. Soit **F** la fonction de cryptage suivante :

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

   a. Trouver la fonction $\mathbf{F^{-1}}$ de décryptage ?

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

b. Crypter le texte « ***resource reservation***» avec la fonction **F**

njzwfnaj njzjngimkwe

c. Quel est l'avantage de cet algorithme (**F**) par rapport à celui de césar ?

Il est plus sécurisé car le nombre de clés est beaucoup plus élevé(26 !)→donc plus résistant à la cryptanalyse

d. L'algorithme **F** est un crypto-système mono alphabétique ou poly-alphabétique ? Justifier votre réponse ?

mono alphabétique car chaque lettre du message est remplacé par une autre lettre de l'alphabet de manière unique

6. Utiliser la clé « BCDE » (1234) pour déchiffrer avec le crypto système de Vignère le cryptogramme suivant : **D T B T U C Q E M A V I**

D	T	B	T	U	C	Q	E	M	A	V	I
B	C	D	E	B	C	D	E	B	C	D	E
C	R	Y	P	T	A	N	A	L	Y	S	E

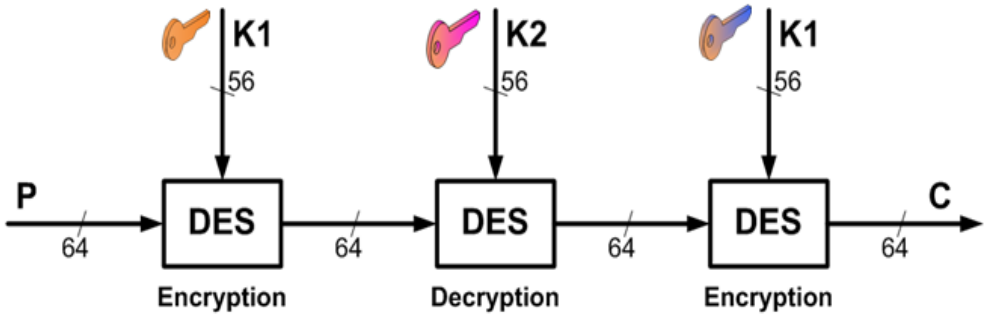
7. Déduire à partir de la question précédente si le crypto-système de Vignère est poly-alphabétique ou non ? Justifier votre réponse ?

Oui le crypto-système de Vignère est poly-alphabétique, car chaque lettre de l'alphabet peut être remplacé par plusieurs possibilités et non pas de manière unique exemple la lettre A est remplacée la première fois par C et la deuxième fois par E

Exercice 2

Soit le texte clair suivant : M : **Examen de cryptographie**

1. Soit f une fonction de substitution (code césar). La notation M'=f(M,k) consiste à chiffrer le message M en utilisant le code césar avec la clé k.
   a) Calculer M1=f1(M,4)
      Ibeqir hi gvctxskvetlmi
   b) Calculer M2=f2(M1,3)
      Lehtlu kl jyfwavnyhwopl
2. Expliquer la réalisation de 3-DES avec une clé de taille 112 bits.



3. Quel est l'avantage de l'algorithme 3-DES par rapport à l'algorithme DES.
   Le 3DES permet d'augmenter significativement la sécurité du DES
   Résiste plus aux attaques brute force en augmentant l'espace des clés possibles
4. Décrire les étapes d'un tour de l'algorithme AES-128.
   a) **SubBytes() :** Il s'agit d'une étape de substitutions appliquée indépendamment à chacun des octets de l'état en utilisant une table de substitution (Sbox) prédéfinie
   b) **ShiftRows()** : Il s'agit d'une étape de Permutation cyclique des octets sur les lignes de l'état. Le décalage des octets correspond à l'indice de la ligne considérée
   c) **MixColumns() :** Transformation appliquée à un état colonne après colonne: un produit matriciel. Chaque octet de la colonne est remplacé par une valeur qui dépend des 4 octets de la colonne: Cette valeur est obtenue en multipliant la colonne par une matrice prédéfinie.
   d) **AddRoundKey: Addition de la sous-clé** : ajout de la clé (initiale ou de la clef lors de la ronde) à l'état considéré (l'addition étant prise au sens ou exclusif).
   Un XOR (au niveau bit) est appliqué entre chacun des octets de l'état et de la clef de ronde.
5. L'opération SubByte de l'algorithme AES consiste à appliquer à chaque $m_{i,j}$ la fonction de substitution S.

		y															
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
x	**0**	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
	1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
	2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
	3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
	4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
	5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
	6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
	7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
	8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
	9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
	A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
	B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
	C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
	D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
	E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
	F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Appliquez SubBytes à l'octet (00001001)

0000 : 0x0

1001 :0x9

→01

6. Expliquez la procédure inverse de chacune des procédures SubBytes, et AddRoundKey.

La procédure de la procédure SubBytes est :

InvSubBytes() : Inverse de la transformation SubBytes()

x \ y	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	73
9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
a	47	f1	1a	71	1d	29	c5	89	6f	b7	62	0e	aa	18	be	1b
b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
d	60	51	7f	a9	19	b5	4a	0d	2d	e5	7a	9f	93	c9	9c	ef
e	a0	e0	3b	4d	ae	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
f	17	2b	04	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d

Pour $s_{i,j} = \{ed\}$

$s'_{i,j} = \text{InvSubBytes}(s_{i,j}) = \{53\}$

La procédure inverse de chacune de la procédure AddRoundKey est la meme que AddRoundKey sauf que les clefs de ronde sont utilisées dans l'ordre inverse de celui du chiffrement.

Exercice 3

On souhaite réaliser un système de messagerie sécurisée à l'intérieur d'une société (employant N personnes). Pour cela, on utilise des « messages électroniques ».

A) Supposons que les N personnes souhaitent communiquer avec chacune des N-1 autres en utilisant un système à clés publiques (chiffrement asymétrique).

On note KA_{Pub} la clé publique *de A et* kA_{Pri} la clé privée de A.

On note aussi KB_{Pub} la clé publique *de B et* kB_{Pri} la clé privée de B.

1. Avec le chiffrement asymétrique, comment assurer l'authentification de l'expéditeur d'un message ?

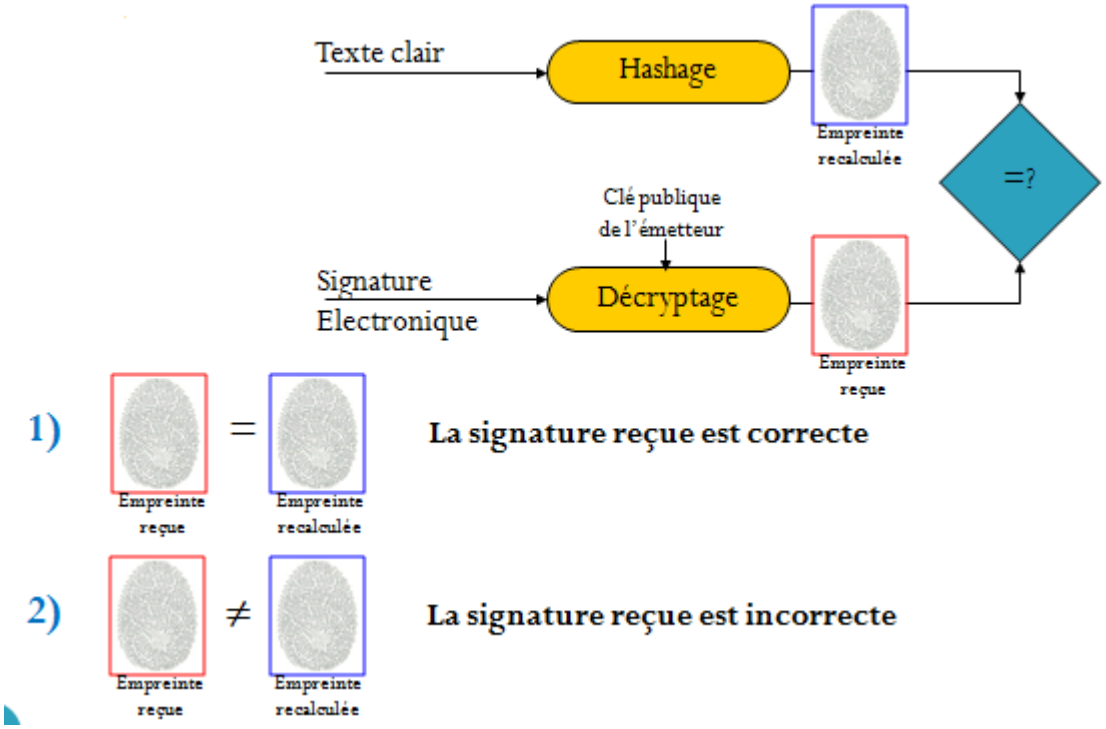
Par une signature numérique associée au message envoyé

2. Comment assurer la confidentialité d'une information M envoyée par l'expéditeur A au destinataire B ?

   Par la clé publique de B

3. Comment assurer l'intégrité d'une information M envoyée par l'expéditeur A ?

   Par la vérification de la signature numérique de M envoyé avec M.



B) Supposons que les N personnes souhaitent communiquer avec chacune des N-1 autres en utilisant un chiffrement à clé symétrique. Toute communication entre deux personnes *i* et *j* est invisible de toutes les autres. De combien de clés a-t-on besoin en tout ?

N (N-1)/2