MATHÉMATIQUES DISCRÈTES (4)
CRYPTOGRAPHIE "CLASSIQUE"

Michel Rigo

http://www.discmath.ulg.ac.be/

Année 2007–2008

CRYPTOGRAPHIE. N. F.

Art d'écrire en chiffres ou d'une façon secrète quelconque.

Ensemble des principes, méthodes et techniques dont l'application assure le chiffrement et le déchiffrement des données, afin d'en préserver la confidentialité et l'authenticité.

EXEMPLE

"Mon numéro de carte VISA est le 1234-3552-1209-7633"

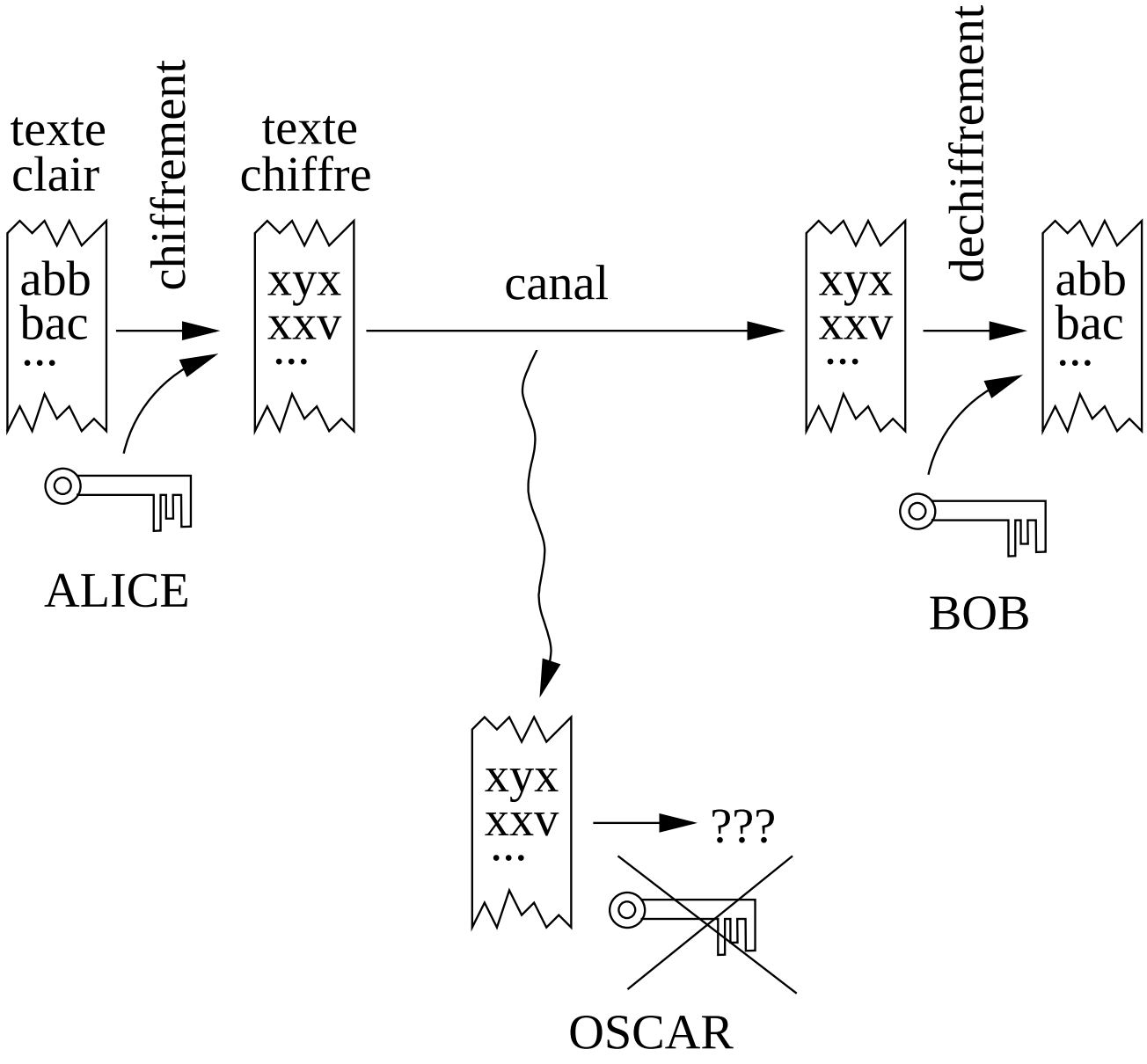
↓

"XFHEBBCASKOIUUSBCKKQHDGGDDSJQQIEUEU"

Applications

- Armée, gouvernement
- Banques, transactions bancaires, bancontact, . . .
- Internet, paiement en ligne par carte de crédit, . . .
- Vote électronique
- GSM (identification, code PIN)
- Télévision payante (à la carte)
- Signatures électroniques, recommandés électroniques, . . .
- Mots de passe informatiques, . . .

Les protagonistes...


texte
clair
chiffrement
texte
chiffre
canal
dechiffrement
abb
bac
...
xyx
xxv
...
xyx
xxv
...
abb
bac
...
ALICE
BOB
xyx
xxv
...
???
OSCAR

Système cryptographique

Définition

cryptosystème $(\mathcal{P}, \mathcal{C}, \mathcal{K})$

- $\mathcal{P}$ est l'ensemble fini des textes clairs possibles (plaintexts),
- $\mathcal{C}$ est l'ensemble fini des textes chiffrés possibles (ciphertexts),
- $\mathcal{K}$ est l'ensemble fini des clés possibles, appelé parfois espace des clés (keys),
- Pour tout $k \in \mathcal{K}$, il existe une fonction de chiffrement (encryption rule) e_k t.q.

$$e_k : \mathcal{P} \to \mathcal{C} : t \mapsto e_k(t)$$

et $\exists$ une fonction de déchiffrement (decryption rule) d_k t.q.

$$d_k : \mathcal{C} \to \mathcal{P} : t \mapsto d_k(t) \quad \text{et} \quad \forall t \in \mathcal{P},\ d_k(e_k(t)) = t.$$

REMARQUE

Pour permettre le déchiffrage, la fonction e_k doit bien évidemment être injective pour tout $k \in \mathcal{K}$.

Exemple : chiffrement par décalage

Chiffrement par décalage

$\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}_{26}$. $\forall k \in \{0, \ldots, 25\}$,

$$e_k(x) = (x + k) \mod 26 \quad \text{et} \quad d_k(y) = (y - k) \mod 26.$$

suite $\mathbf{x} = x_1 x_2 \cdots x_\ell$, $x_i \in \mathcal{P}$ chiffrée avec e_k ($k \in \mathcal{K}$ choisi de commun accord entre A et B), A transmet

$$\mathbf{y} = e_k(x_1) e_k(x_2) \cdots e_k(x_\ell).$$

Etape souvent préalable au chiffrement

Codage = ensemble des conventions pour modifier le texte clair en un texte équivalent plus simple à traiter du point de vue cryptographique.

EXEMPLE : CHIFFREMENT PAR DÉCALAGE

CODAGE "STANDARD"

A veut transmettre le message "`bonsoir`", chaque lettre est remplacée par sa position dans l'alphabet $\Sigma = \{a, \ldots, z\}$

a	b	c	d	e	f	g	h	i	j	k	l	m	n
0	1	2	3	4	5	6	7	8	9	10	11	12	13

o	p	q	r	s	t	u	v	w	x	y	z
14	15	16	17	18	19	20	21	22	23	24	25

"`bonsoir`" est *codé* en "$\mathbf{x} = 1,14,13,18,14,8,17$".

codage : bijection entre l'alphabet Σ et $\{0, \ldots, 25\}$.

décodage : application inverse

Ne pas confondre : codage $\neq$ chiffrement.

Exemple : chiffrement par décalage

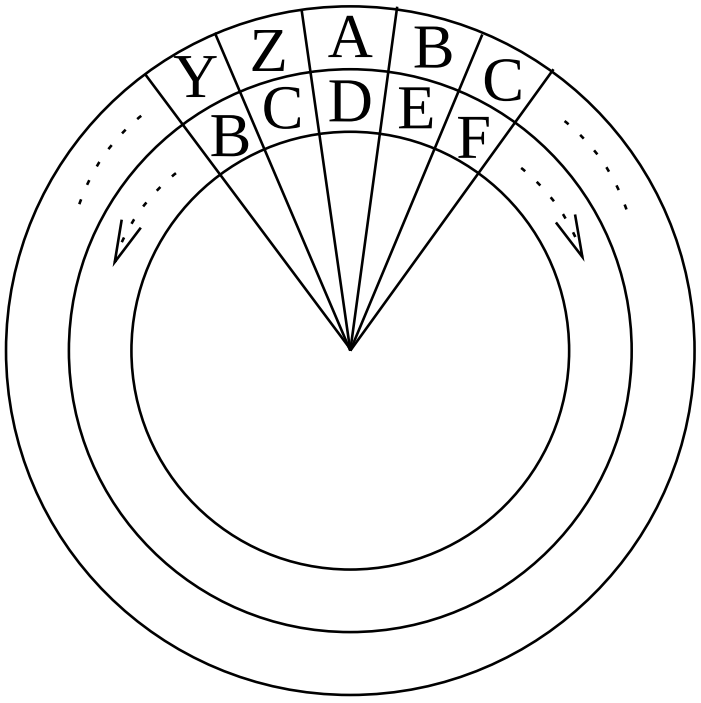
Fin de l'exemple...

Si la clé choisie est $k = 3$ (Jules César), alors

$$\begin{aligned}\mathbf{y} &= e_3(1)e_3(14)e_3(13)e_3(18)e_3(14)e_3(8)e_3(17)\\ &= 4, 17, 16, 21, 17, 11, 20.\end{aligned}$$

“`bonsoir`” $\longrightarrow$ “`erqvrlu`”.

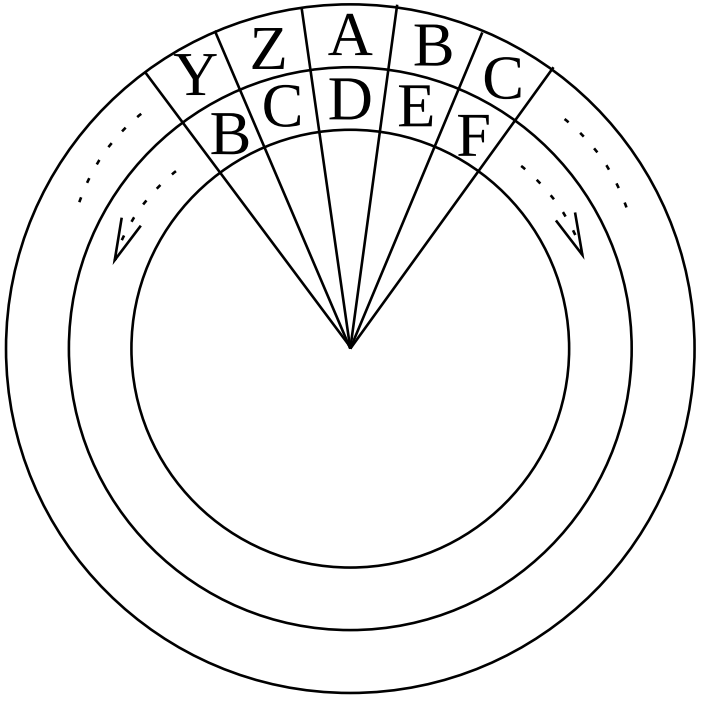
Exemple : chiffrement par décalage



Remarque

Cryptosystème peu sûr. . .

EXEMPLE : CHIFFREMENT PAR DÉCALAGE



REMARQUE

Cryptosystème peu sûr. . .

EXEMPLE : CHIFFREMENT PAR DÉCALAGE

2001 : l'odysée de l'espace, HAL

Principe de Kerckhoff

Définition

cryptanalyse= ensemble des moyens et techniques mis en oeuvre pour retrouver le texte clair ou au moins une certaine information contenue dans celui-ci.

Hypothèses de travail

Le principe de Kerckhoff suppose qu'Oscar connaît le cryptosystème utilisé.

La sécurité du système réside alors dans la protection de la clé k choisie par Alice et Bob.

Attaques

Types d'attaque dont dispose Oscar (par difficulté ↓)

- Texte chiffré connu : Oscar connaît uniquement un fragment de texte chiffré **y**.
- Texte clair connu : Oscar connaît un texte clair **x** et le texte chiffré **y** correspondant.
- Texte clair choisi : Oscar a accès à une machine chiffrante. Il peut choisir un texte clair **x** et obtenir le texte chiffré **y** correspondant.
- Fonction de chiffrement connue : Oscar connaît précisément la fonction utilisée pour le chiffrement. Son but est alors de découvrir la fonction de déchiffrement. Situation typique des cryptosystèmes à clé publique.
- Texte chiffré choisi : Oscar a temporairement accès à une machine déchiffrante. Il peut choisir un texte chiffré **y** et obtenir le texte clair **x** correspondant.

Implémentation

```
In[1]:= Characters["bonjour"]

Out[1]= {b, o, n, j, o, u, r}

In[2]:= StringJoin[{"a", "b", "c"}]

Out[2]= abc

In[3]:= Map[Sqrt[#] &, {4, 2, 9}]

Out[3]= {2, √2, 3}
```

IMPLÉMENTATION

code / decode

```
alphabet = {" ", "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", ",", ".", "'", ":", "?"};

code[mot_] := Map[Position[alphabet, #][[1, 1]] - 1 &, Characters[mot]]

texteclair = "le jour de ses onze ans, harry potter, un orphelin eleve par un
    oncle et une tante qui le detestent, voit son existence bouleversee.";

codetxclair = code[texteclair]

{12, 5, 0, 10, 15, 21, 18, 0, 4, 5, 0, 19, 5, 19, 0, 15, 14, 26, 5, 0, 1, 14, 19, 27, 0, 8, 1,
 18, 18, 25, 0, 16, 15, 20, 20, 5, 18, 27, 0, 21, 14, 0, 15, 18, 16, 8, 5, 12, 9, 14, 0, 5,
 12, 5, 22, 5, 0, 16, 1, 18, 0, 21, 14, 0, 15, 14, 3, 12, 5, 0, 5, 20, 0, 21, 14, 5, 0, 20, 1,
 14, 20, 5, 0, 17, 21, 9, 0, 12, 5, 0, 4, 5, 20, 5, 19, 20, 5, 14, 20, 27, 0, 22, 15, 9, 20, 0,
 19, 15, 14, 0, 5, 24, 9, 19, 20, 5, 14, 3, 5, 0, 2, 15, 21, 12, 5, 22, 5, 18, 19, 5, 5, 28}

decode[liste_] := StringJoin[Table[alphabet[[liste[[i]] + 1]], {i, 1, Length[liste]}]]

decode[codetxclair]

le jour de ses onze ans, harry potter, un orphelin eleve par un
  oncle et une tante qui le detestent, voit son existence bouleversee.
```

IMPLÉMENTATION

```
cesar[n_] := Mod[n + 3, 32]

codetxchiffre = cesar[codetxclair]

{15, 8, 3, 13, 18, 24, 21, 3, 7, 8, 3, 22, 8, 22, 3, 18, 17, 29, 8, 3, 4, 17,
 22, 30, 3, 11, 4, 21, 21, 28, 3, 19, 18, 23, 23, 8, 21, 30, 3, 24, 17, 3, 18,
 21, 19, 11, 8, 15, 12, 17, 3, 8, 15, 8, 25, 8, 3, 19, 4, 21, 3, 24, 17, 3, 18,
 17, 6, 15, 8, 3, 8, 23, 3, 24, 17, 8, 3, 23, 4, 17, 23, 8, 3, 20, 24, 12, 3, 15,
 8, 3, 7, 8, 23, 8, 22, 23, 8, 17, 23, 30, 3, 25, 18, 12, 23, 3, 22, 18, 17, 3,
 8, 27, 12, 22, 23, 8, 17, 6, 8, 3, 5, 18, 24, 15, 8, 25, 8, 21, 22, 8, 8, 31}

txchiffre = decode[codetxchiffre]

ohcmrxucghcvhvcrq'hcdqv:ckduu.csrwwhu:
  cxqcruskholqchohyhcsducxqcrqfohchwcxqhcwdqwhctxlcohcghwhvwhqw:
  cyrlwcvrqch,lvwhqfhcerxohyhuvhh?
```

IMPLÉMENTATION

Cryptanalyse ?

```
In[17]:= Count[{1, 2, 1, 1, 2}, 1]

Out[17]= 3

In[22]:= Count[Characters[texteclair], "e"]

Out[22]= 24

In[19]:= Map[Count[codetxclair, #] &, Table[i, {i, 0, 31}]]

Out[19]= {23, 4, 1, 2, 2, 24, 0, 0, 2, 4, 1, 0, 6, 0,
          11, 8, 3, 1, 7, 7, 10, 6, 3, 0, 1, 1, 1, 3, 1, 0, 0, 0}

In[18]:= Map[Count[codetxchiffre, #] &, Table[i, {i, 0, 31}]]

Out[18]= {0, 0, 0, 23, 4, 1, 2, 2, 24, 0, 0, 2, 4, 1,
          0, 6, 0, 11, 8, 3, 1, 7, 7, 10, 6, 3, 0, 1, 1, 1, 3, 1}
```

Fréquence d'apparition

fréquences d'apparition des différentes lettres apparaissant dans les textes écrits en français.

lettre	%
e	15,87
a	9,42
i	8,41
s	7,90
t	7,26
n	7,15
r	6,46
u	6,24
l	5,34

Codage par blocs

Définition

Unité fondamentale pour réaliser le codage d'une chaîne, non pas une lettre, mais un bloc de m lettres consécutives.

Un bloc $t_1 \cdots t_m$ de $m \geq 1$ entiers consécutifs < 32, représente un nombre n écrit en base 32 :

$$n = \sum_{j=1}^{m} t_j \, 32^{m-j}.$$

Bloc de longueur m représente un nombre $0 \leq n \leq 32^m - 1$.

$\mathcal{P} = \mathcal{C} = \mathbb{Z}_{32^m}$ et non plus $\mathbb{Z}_{32}$.

Codage par blocs

Remarque

Utiliser un codage dans lequel on considère des blocs de m éléments augmente la sécurité du cryptosystème.

Sur notre exemple, $\mathcal{K} = \mathbb{Z}_{32}$ passe à $\mathcal{K} = \mathbb{Z}_{32^m}$.

Puisque le nombre de clés augmente, une recherche exhaustive menée par Oscar prend beaucoup plus de temps

si $m = 5$, $32^5 \simeq 33 \times 10^6$ et en testant 1000 clés/seconde, un peu plus de 9 heures pour parcourir l'espace des clés.

Codage par blocs

Remarque

Si nous découpons le texte clair en tronçons de longueur m, il faudrait que ce texte soit de longueur divisible par m.

S'il ne l'était pas, on ajouterait au préalable des symboles à la fin du texte pour que sa longueur soit divisible par m.

Attention

Faille possible de sécurité !

IMPLÉMENTATION

```
ajout[liste_, m_] := PadRight[liste, Length[liste] + Mod[-Length[liste], m]]

codetxclair = ajout[codetxclair, 5]

{12, 5, 0, 10, 15, 21, 18, 0, 4, 5, 0, 19, 5, 19, 0, 15, 14, 26, 5, 0, 1, 14,
 19, 27, 0, 8, 1, 18, 18, 25, 0, 16, 15, 20, 20, 5, 18, 27, 0, 21, 14, 0, 15, 18,
 16, 8, 5, 12, 9, 14, 0, 5, 12, 5, 22, 5, 0, 16, 1, 18, 0, 21, 14, 0, 15, 14, 3,
 12, 5, 0, 5, 20, 0, 21, 14, 5, 0, 20, 1, 14, 20, 5, 0, 17, 21, 9, 0, 12, 5, 0,
 4, 5, 20, 5, 19, 20, 5, 14, 20, 27, 0, 22, 15, 9, 20, 0, 19, 15, 14, 0, 5, 24,
 9, 19, 20, 5, 14, 3, 5, 0, 2, 15, 21, 12, 5, 22, 5, 18, 19, 5, 5, 28, 0, 0, 0}

codebloc[liste_, m_] := Map[FromDigits[#, 32] &, Partition[liste, m]]

codetxclair = codebloc[codetxclair, 5]

{12747087, 22610053, 628320, 16214176, 1527648, 8440409, 540308, 5860373, 1469601
 8565038, 176310, 5259314, 702479, 14790816, 5898926, 5263406, 21135925, 9449632,
 4378803, 21150363, 736564, 638400, 6039156, 5704864, 2610565, 23251557, 6160384}

decodebloc[liste_, m_] := decode[Flatten[Map[IntegerDigits[#, 32, m] &, liste]]]

decodebloc[codetxclair, 5]

le jour de ses onze ans, harry potter, un orphelin eleve par un
  oncle et une tante qui le detestent, voit son existence bouleversee.
```

IMPLÉMENTATION

```
codetxchiffre = Mod[codetxclair + 12345678, 32^5]

{25092765, 1401299, 12973998, 28559854, 13873326, 20786087,
 12885986, 18206051, 27041694, 20910716, 12521988, 17604992, 13048157,
 27136494, 18244604, 17609084, 33481603, 21795310, 16724481, 33496041,
 13082242, 12984078, 18384834, 18050542, 14956243, 2042803, 18506062}

decodebloc[codetxchiffre, 5]

w'xt'ajxnslk''n,gronmglenszj'glig?bqkskcyyg.:s:ds.k:dpdpyhl lnfj'
  y.donqlx?.pylk.?'x.ctydono:lpa?:f?ilogtbllgxnqqa:bqf,onnhmvsa:j'sqtxjn

decodebloc[Mod[codetxchiffre - 12345678, 32^5], 5]

le jour de ses onze ans, harry potter, un orphelin eleve par un
  oncle et une tante qui le detestent, voit son existence bouleversee.
```

Implémentation

`code[mot]`		chaîne de longueur n
	$\mapsto$	liste de n éléments de $\mathbb{Z}_{32}$
`decode[liste]`		liste de n éléments de $\mathbb{Z}_{32}$
	$\mapsto$	chaîne de longueur n
`ajout[liste,m]`		liste quelconque
	$\mapsto$	liste de longueur divisible par m
`codebloc[liste,m]`		liste de $n.m$ éléments de $\mathbb{Z}_{32}$
	$\mapsto$	liste de n éléments de $\mathbb{Z}_{32^m}$
`decodebloc[liste,m]`		liste de n éléments de $\mathbb{Z}_{32^m}$
	$\mapsto$	chaîne de longueur $n.m$.

CHIFFREMENT PAR SUBSTITUTION

Si $\mathcal{P} = \mathcal{C} = \mathbb{Z}_{32}$, alors $\mathcal{K}$ est l'ensemble des permutations de $\{0, \ldots, 31\}$, i.e.,

$$\mathcal{K} = \{\nu \in \mathcal{S}_{32} \mid \nu : \mathbb{Z}_{32} \to \mathbb{Z}_{32} \text{ bijection}\}.$$

pour la clé $k = \nu$, on a

$$e_k(x) = \nu(x) \quad \text{et} \quad d_k(y) = \nu^{-1}(y).$$

REMARQUE

Par rapport au chiffrement par décalage, $\#\mathcal{K}$ est grand : $32! \simeq 2,6 \times 10^{35}$.

Quelques cryptosystèmes classiques

Chiffrement par substitution

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
e	:	d	k	,	l	p	n		q	x	f	s	z	.	u

p	q	r	s	t	u	v	w	x	y	z	,	.	’	:	?
’	a	c	i	g	?	t	v	o	y	w	r	b	m	j	h

le jour de ses onze ans, harry potter, un orphelin eleve par un oncle et une tante qui le detestent, voit son existence bouleversee.

slexu ?ce,leilieu.wle :.i,e :ccye’ugglc,e ?.euc’ lsq.elsltle’ :c e ?.eu.kslelge ?.leg :.glea ?qesle,lgligl.g,etuqgeiu.eloqigl.kl edu ?sltlcilllb

CHIFFREMENT PAR SUBSTITUTION

Nombre de clés important, mais ce cryptosystème est cassé en effectuant une analyse des fréquences : rechercher les lettres apparaissant le plus souvent, mais aussi les couples ou les triplets.

Ce cryptosystème NE peut être considéré comme sûr.

Seul intérêt : cryptogrammes dans les livres de jeux.

CHIFFREMENT AFFIN

$\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$. Soient $a, b \in \mathbb{Z}_n$ avec $pgcd(a, n) = 1$.

$$\mathcal{K} = \{(a, b) \mid a \in \mathbb{Z}_n^*, b \in \mathbb{Z}_n\}$$

et pour une clé $k = (a, b)$ choisie dans $\mathcal{K}$, on a

$$e_k(x) = ax + b \mod n \quad \text{et} \quad d_k(y) = a^{-1}(y - b) \mod n.$$

a inversible pour que e_k soit injective.

Le nombre de clés est $n\,\varphi(n)$.

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

```
texteclair = "ving minutes plus tard, harry sortit
du magasin de hiboux avec une grande cage a
l'interieur de laquelle une magnifique chouette aux
plumes blanches comme la neige dormait
paisiblement.";
```

$k = (13, 8)$

decode[Mod[13 code[texteclair]+8, 32]]

```
"f' :chq' :yli?hxdy?hlur.gpurrmh?krl'lh.yhqucu?'
:h.ihp'bky hufiohy :ihcru :.ihoucihuhda' :lir'iyr
h.ihdueyiddihy :ihquc :'v'eyihopkyillihuy hxdyqi
?hbdu :opi?hokqqihduh :i'cih.krqu'lhxu'?'bdiqi :lt"
```

REMARQUE

Cas particulier de chiffrement par substitution.
cryptanalyse : analyse statistique des fréquences d'apparition des lettres.

MONOALPHABÉTIQUE VS. POLYALPHABÉTIQUE

Les chiffrements par décalage, par substitution et affin sont monoalphabétiques.

e_k s'applique à un seul élément de $\mathcal{P}$ à la fois (une lettre, un élément de $\mathbb{Z}_{32}$)

à chaque élément de $\mathcal{P}$ est appliquée la même fonction de chiffrement.

Pour un bloc de m symboles, aussi un système monoalphabétique car $\mathcal{P} = \mathbb{Z}_{32^m}$

Le chiffrement de Vigenère (Blaise de Vigenère, XVIè siècle) est polyalphabétique car il traite m symboles simultanément.

CHIFFREMENT DE VIGENÈRE

Il s'agit en quelque sorte de m chiffrements par décalage, $\mathcal{P} = \mathcal{C} = \mathcal{K} = (\mathbb{Z}_{32})^m$, si $k = (k_1, \ldots, k_m) \in (\mathbb{Z}_{32})^m$, alors

$$e_k(x_1, \ldots, x_m) = (x_1 + k_1, \ldots, x_m + k_m) \mod 32$$

$$d_k(y_1, \ldots, y_m) = (y_1 - k_1, \ldots, y_m - k_m) \mod 32.$$

Pour chiffrer une suite de longueur $rm + s$ de $\mathbb{Z}_{32}$, $0 \leq s < m$,

$$\mathbf{x} = x_1 \cdots x_m \,|\, \cdots \,|\, x_{(r-1)m+1} \cdots x_{rm} \,|\, x_{rm+1} \cdots x_{rm+s},$$

on calculera

$$\begin{aligned} \mathbf{y} \;=\; & (x_1 + k_1) \cdots (x_m + k_m) \,|\, \cdots \,|\, (x_{(r-1)m+1} + k_1) \cdots (x_{rm} + k_m) \,| \\ & (x_{rm+1} + k_1) \cdots (x_{rm+s} + k_s) \mod 32. \end{aligned}$$

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

L'implémentation est très simple

```
In[2]:= PadRight[{1, 2}, 10, {5, 3, 7}]

Out[2]= {1, 2, 7, 5, 3, 7, 5, 3, 7, 5}
```

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

```
texteclair = "ving minutes plus tard,harry sortit du magasin de
    hiboux avec une grande cage a l'interieur de laquelle une magnifique
    chouette aux plumes blanches comme la neige dormait paisiblement.";

codetxclair = code[texteclair];

Shallow[codetxclair]

{22, 9, 14, 7, 0, 13, 9, 14, 21, 20, <<174>>}

vigenere[liste_, cle_] := Mod[liste + PadRight[cle, Length[liste], cle], 32]

codetxchiffre = vigenere[codetxclair, {10, 8, 20}]

{0, 17, 2, 17, 8, 1, 19, 22, 9, 30, 13, 7, 10, 24, 0, 31, 27, 20, 30, 9, 6, 14, 3, 28,
 26, 6, 3, 8, 7, 25, 26, 8, 19, 28, 20, 14, 29, 20, 23, 9, 27, 11, 27, 29, 24, 8, 24,
 15, 8, 28, 19, 10, 3, 31, 0, 20, 11, 30, 25, 13, 8, 9, 24, 13, 20, 17, 26, 21, 24, 12,
 25, 10, 11, 21, 17, 13, 20, 11, 8, 0, 7, 17, 2, 30, 13, 6, 19, 13, 9, 28, 8, 24, 15,
 8, 0, 11, 25, 9, 15, 20, 0, 15, 8, 9, 24, 13, 20, 23, 9, 27, 24, 17, 26, 19, 25, 9,
 15, 8, 23, 18, 23, 9, 15, 28, 8, 15, 8, 21, 31, 0, 20, 26, 20, 9, 23, 13, 7, 10, 10,
 0, 11, 22, 23, 18, 13, 7, 10, 11, 3, 23, 21, 25, 10, 20, 21, 10, 22, 25, 19, 15, 25,
 10, 12, 3, 28, 21, 21, 19, 28, 20, 26, 9, 29, 29, 17, 22, 22, 13, 1, 15, 22, 8, 6}

decode[codetxchiffre]

 qbqhasvi:mgjx ?,t:ifnc.kzfchgyzhs.tn'twi,k,'xhxoh.sjc? tk:
  ymhixmtqzuxlyjkuqmtkh gqb:mfsmi.hxoh kyiot ohixmtwi,xqzsyiohwrwio.
  hohu? tztiwmgjj kvwrmgjkcwuyjtujvysoyjlc.uus.tzi''qvvmaovhf
```

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

Pour le déchiffrement...

```
vigenere[codetxchiffre, -{10, 8, 20}]
```

```
{22, 9, 14, 7, 0, 13, 9, 14, 21, 20, 5, 19, 0, 16, 12, 21, 19, 0, 20, 1, 18, 4, 27, 8,
 25, 0, 19, 15, 18, 20, 9, 20, 0, 4, 21, 0, 13, 1, 7, 1, 19, 9, 14, 0, 4, 5, 0, 8, 9, 2,
 21, 24, 0, 1, 22, 5, 3, 0, 21, 14, 5, 0, 7, 18, 1, 14, 4, 5, 0, 3, 1, 7, 5, 0, 1, 0, 12
 29, 9, 14, 20, 5, 18, 9, 5, 21, 18, 0, 4, 5, 0, 12, 1, 17, 21, 5, 12, 12, 5, 0, 21, 14
 5, 0, 13, 1, 7, 14, 9, 6, 9, 17, 21, 5, 0, 3, 8, 15, 21, 5, 20, 20, 5, 0, 1, 21, 24, 0,
 16, 12, 21, 13, 5, 19, 0, 2, 12, 1, 14, 3, 8, 5, 19, 0, 3, 15, 13, 13, 5, 0, 12, 1, 0,
 5, 9, 7, 5, 0, 4, 15, 18, 13, 1, 9, 20, 0, 16, 1, 9, 19, 9, 2, 12, 5, 13, 5, 14, 20, 28
```

REMARQUES

Dans le chiffrement de Vigenère, un même symbole peut être transformé en m symboles distincts suivant la position qu'occupe ce symbole dans un m-uple donné.

Cryptanalyse : test de Kasiski / indice de coïncidence

CHIFFREMENT DE HILL, LESTER S. HILL (1929)

Cryptosystème polyalphabétique. $\mathcal{P} = \mathcal{C} = (\mathbb{Z}_{32})^m$ et

$$\mathcal{K} = GL_m(\mathbb{Z}_{32}),$$

l'ensemble des matrices inversibles à coefficients dans $\mathbb{Z}_{32}$.

PROPOSITION

Une matrice carrée A à coefficients dans $\mathbb{Z}_n$ est inversible, i.e., il existe B tel $AB = BA = I$, SSI $\det(A)$ est inversible dans $\mathbb{Z}_n$.

règle des mineurs :

$$A\,\widetilde{\mathrm{cof}(A)} = \det A\, I = \widetilde{\mathrm{cof}(A)}\,A$$

$\mathrm{cof}(A)$: matrice des cofacteurs (ou mineurs algébriques) de A.

$\Leftarrow$. Si $\det A$ est inversible, l'inverse de A est $\widetilde{\mathrm{cof}(A)}$.

$\Rightarrow$. Si A est inversible et d'inverse A^{-1}, alors

$$1 = \det(AA^{-1}) = \det A \det A^{-1}$$

ce qui montre que $\det A$ est inversible.

CHIFFREMENT DE HILL

Si $k = (a_{ij}) \in GL_m(\mathbb{Z}_{32})$,
$e_k : \mathbb{Z}_{32}^m \to \mathbb{Z}_{32}^m : (x_1, \ldots, x_m) \mapsto (y_1, \ldots, y_m)$ est donné par

$$\begin{pmatrix} y_1 \\ \vdots \\ y_m \end{pmatrix} = \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & & \vdots \\ a_{m1} & \cdots & a_{mm} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix}.$$

CHIFFREMENT/DÉCHIFFREMENT

La matrice A détermine une bijection $x \mapsto Ax$ de $(\mathbb{Z}_n)^m$ dans lui-même SSI A inversible.

$\Leftarrow$ Si $Ax = Ay$, en multipliant par A^{-1} : $x = y$ (injectif).
(surjectif) $\forall y \in (\mathbb{Z}_n)^m$, $x = A^{-1}y$ est tel que $Ax = y$.

$\Rightarrow$ $x \mapsto Ax$ est surjectif, pour tout e_i, $i = A, \ldots, m$, il existe un vecteur colonne C_i tel que $AC_i = e_i$. De là, la matrice dont les colonnes sont $C_1, \ldots, C_m$ est inverse de A.

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

```
texteclair = "le phenix sur lequel a ete preleve la plume qui se trouve dans
    votre baguette a egalement fourni une autre plume a une autre baguette.";

codetxclair = ajout[code[texteclair], 2];

Shallow[codetxclair]
```

```
{12, 5, 0, 16, 8, 5, 14, 9, 24, 0, <<124>>}
```

```
a = {{5, 17}, {4, 21}};

MatrixForm[a]
```

$$\begin{pmatrix} 5 & 17 \\ 4 & 21 \end{pmatrix}$$

```
hill[a_, liste_] := Flatten[Map[Mod[a.# , 32] &, Partition[liste, Length[a]]]]

codetxchiffre = hill[a, codetxclair]
```

```
{17, 25, 16, 16, 29, 9, 31, 21, 24, 0, 4, 5, 26, 8, 17, 25, 26, 29, 5, 16, 17, 21,
 21, 9, 25, 25, 16, 16, 15, 17, 17, 25, 3, 1, 12, 28, 5, 4, 28, 28, 6, 5, 25, 20,
 26, 29, 13, 4, 20, 21, 20, 4, 25, 3, 31, 2, 25, 20, 5, 5, 9, 7, 22, 14, 31, 0, 15,
 17, 2, 10, 28, 23, 30, 29, 24, 20, 25, 20, 5, 4, 16, 7, 17, 0, 22, 5, 7, 26, 4, 16,
 29, 19, 27, 14, 31, 21, 5, 25, 27, 1, 17, 21, 29, 24, 15, 17, 16, 16, 1, 9, 22, 29,
 17, 21, 5, 25, 27, 1, 17, 21, 29, 24, 15, 17, 2, 10, 28, 23, 30, 29, 24, 20, 21, 0}
```

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

```
Inverse[a, Modulus → 32]
```

```
{{17, 3}, {12, 1}}
```

```
hill[Inverse[a, Modulus → 32], codetxchiffre]
```

```
{12, 5, 0, 16, 8, 5, 14, 9, 24, 0, 19, 21, 18, 0, 12, 5, 17, 21, 5, 12, 0, 1, 0, 5, 20, 5, 0,
 18, 5, 12, 5, 22, 5, 0, 12, 1, 0, 16, 12, 21, 13, 5, 0, 17, 21, 9, 0, 19, 5, 0, 20, 18, 15,
 21, 22, 5, 0, 4, 1, 14, 19, 0, 22, 15, 20, 18, 5, 0, 2, 1, 7, 21, 5, 20, 20, 5, 0, 1, 0, 5,
 7, 1, 12, 5, 13, 5, 14, 20, 0, 6, 15, 21, 18, 14, 9, 0, 21, 14, 5, 0, 1, 21, 20, 18, 5, 0,
 16, 12, 21, 13, 5, 0, 1, 0, 21, 14, 5, 0, 1, 21, 20, 18, 5, 0, 2, 1, 7, 21, 5, 20, 20, 5, 28
```

Pour trouver cet inverse manuellement, on calcule $\det A = 5$, son inverse modulo 32 grâce à l'algorithme d'Euclide étendu, $(\det A)^{-1} = 13$ et enfin, la matrice des cofacteurs transposée,

$$\widetilde{\operatorname{cof} A} = \begin{pmatrix} 21 & -17 \\ -4 & 5 \end{pmatrix}.$$

On trouve,

- $13.21 = 273 \equiv 17 \mod 32$,
- $13.(-17) = -221 \equiv 3 \mod 32$,
- $13.(-4) = -52 \equiv 12 \mod 32$,
- $13.5 = 65 \equiv 1 \mod 32$.

CRYPTANALYSE DU CHIFFREMENT

cas où un texte clair de longueur m^2 est connu. Si Oscar connaît la valeur $m = 2$, le texte clair $12, 5, 0, 16$ et le texte chiffré correspondant $17, 25, 16, 16$, alors il en déduit que

$$A \underbrace{\begin{pmatrix} 12 & 0 \\ 5 & 16 \end{pmatrix}}_{B} = \begin{pmatrix} 17 & 16 \\ 25 & 16 \end{pmatrix}.$$

Pas de chance B n'est pas inversible, $\det(B) = 0$.

Oscar doit construire une matrice inversible dont les colonnes correspondent à des couples de texte clair lorsque ce dernier est décomposé en m-uples consécutifs.

CRYPTANALYSE DU CHIFFREMENT

On peut par exemple prendre pour premier couple $(12, 5)$ correspondant au texte chiffré $(17, 25)$ et comme second couple $(19, 21)$ qui correspond à $(4, 5)$

$$A \begin{pmatrix} 12 & 19 \\ 5 & 21 \end{pmatrix} = \begin{pmatrix} 17 & 4 \\ 25 & 5 \end{pmatrix}.$$

Puisque

$$\det \begin{pmatrix} 12 & 19 \\ 5 & 21 \end{pmatrix} \equiv 29 \mod 32, \quad A = \begin{pmatrix} 17 & 4 \\ 25 & 5 \end{pmatrix} \begin{pmatrix} 12 & 19 \\ 5 & 21 \end{pmatrix}^{-1}$$

et Oscar retrouve A.

CHIFFREMENT PAR PERMUTATION

Cas particulier du chiffrement de Hill, matrice pour le chiffrement : une matrice de permutation P.

Le chiffrement de Hill revient alors à permuter les lettres d'un même m-uple au moyen de la permutation définie par P.

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

```
p = {{0, 1, 0}, {0, 0, 1}, {1, 0, 0}}
```

```
{{0, 1, 0}, {0, 0, 1}, {1, 0, 0}}
```

```
decode[hill[p, codetxclair]]
```

```
e lhepixnsu  lrquel e eae trepevl le pauml qei ue srotveuda s
  notve ragbetue t eaalgmeet noufnirun  aetru peuml aeun  aetru beguatte
```

CHIFFREMENT PAR FLOT, UN EXEMPLE

on génère une suite de clés $\mathbf{z} = z_1 z_2 \cdots$ utilisées successivement pour chiffrer une suite $\mathbf{x} = x_1 x_2 \cdots$.

Soient $\mathcal{P} = \mathcal{C} = \mathcal{K} = \mathbb{Z}_{32}$. Pour chiffrer

$$\mathbf{x} = x_1 x_2 x_3 \cdots, \quad \forall i \geq 1,\ x_i \in \mathcal{P}$$

avec une clé k fixée initialement, on procède comme suit :

$$\mathbf{y} = (x_1 + k)(x_2 + x_1)(x_3 + x_2) \cdots \mod 32.$$

Revient à ajouter à la suite $\mathbf{x}$, la suite décalée d'une unité vers la droite.

QUELQUES CRYPTOSYSTÈMES CLASSIQUES

```
texteclair = "essayer un chapeau valait beaucoup mieux que d'etre oblige de jeter un
    sort,mais il aurait prefere ne pas avoir a le faire devant tout le monde.";

codetxclair = code[texteclair]

{5, 19, 19, 1, 25, 5, 18, 0, 21, 14, 0, 3, 8, 1, 16, 5, 1, 21, 0, 22, 1, 12, 1, 9,
 20, 0, 2, 5, 1, 21, 3, 15, 21, 16, 0, 13, 9, 5, 21, 24, 0, 17, 21, 5, 0, 4, 29, 5,
 20, 18, 5, 0, 15, 2, 12, 9, 7, 5, 0, 4, 5, 0, 10, 5, 20, 5, 18, 0, 21, 14, 0, 19,
 15, 18, 20, 27, 13, 1, 9, 19, 0, 9, 12, 0, 1, 21, 18, 1, 9, 20, 0, 16, 18, 5, 6, 5,
 18, 5, 0, 14, 5, 0, 16, 1, 19, 0, 1, 22, 15, 9, 18, 0, 1, 0, 12, 5, 0, 6, 1, 9, 18,
 5, 0, 4, 5, 22, 1, 14, 20, 0, 20, 15, 21, 20, 0, 12, 5, 0, 13, 15, 14, 4, 5, 28}

flot[liste_, k_] := Mod[Drop[Prepend[liste, k], -1] + liste, 32]

flot[codetxclair, 5]

{10, 24, 6, 20, 26, 30, 23, 18, 21, 3, 14, 3, 11, 9, 17, 21, 6, 22, 21, 22, 23, 13, 13, 10,
 29, 20, 2, 7, 6, 22, 24, 18, 4, 5, 16, 13, 22, 14, 26, 13, 24, 17, 6, 26, 5, 4, 1, 2,
 25, 6, 23, 5, 15, 17, 14, 21, 16, 12, 5, 4, 9, 5, 10, 15, 25, 25, 23, 18, 21, 3, 14, 19,
 2, 1, 6, 15, 8, 14, 10, 28, 19, 9, 21, 12, 1, 22, 7, 19, 10, 29, 20, 16, 2, 23, 11, 11,
 23, 23, 5, 14, 19, 5, 16, 17, 20, 19, 1, 23, 5, 24, 27, 18, 1, 1, 12, 17, 5, 6, 7, 10,
 27, 23, 5, 4, 9, 27, 23, 15, 2, 20, 20, 3, 4, 9, 20, 12, 17, 5, 13, 28, 29, 18, 9, 1}

decode[%]

jxftz:wrucnckiqufvuvwmmj'tbgfvxrdepmvnzmxqfzedabyfweoqnupledie joyywrucnsbafohnj.
  siulavgsj'tpbwkkwwensepqtsawex,raalqefgj,wedi,wobttcditlqem.'ria
```

DÉFINITION

chiffrement par flot : $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{L}, \mathcal{F})$

- $\mathcal{L}$ ensemble fini : alphabet du flot de clés,
- $\mathcal{F}$ suite de fonctions $(f_n)_{n\geq 1}$: générateur du flot de clés t.q.

$$\forall n \geq 1, \quad f_n : \mathcal{K} \times \mathcal{P}^{n-1} \to \mathcal{L},$$

- $\forall z \in \mathcal{L}, \exists\, e_z : \mathcal{P} \to \mathcal{C}$ et $d_z : \mathcal{C} \to \mathcal{P}$ t.q. $d_z(e_z(x)) = x, \ \forall x$

A une suite $\mathbf{x} = x_1 x_2 x_3 \cdots$ d'éléments de $\mathcal{P}$ et à une clé $k \in \mathcal{K}$, correspond une unique suite de fonctions $\mathcal{F} = (f_1, f_2, f_3, \ldots)$.

Chaque f_n donne un $z_n \in \mathcal{L}$ et à ce z_n, il correspond e_{z_n} et d_{z_n}.

Le chiffrement de $\mathbf{x}$ est $e_{z_1}(x_1) e_{z_2}(x_2) e_{z_3}(x_3) \cdots$

POUR NOTRE EXEMPLE

$e_{z_1}(x_1) = x_1 + k \pmod{32}$ et
pour $n > 1$, $e_{z_n}(x_n) = x_n + x_{n-1} \pmod{32}$.

cas très particulier de chiffrement par flot.

En toute généralité, un chiffrement par flot admet plus de souplesse.

DEFINITION

Un chiffrement en chaîne est dit synchrone si la suite de clés $(z_n)_{n\geq 1}$ est indépendante du texte clair.

Un chiffrement est dit (ultimement) périodique si la suite $(z_n)_{n\geq 1}$ est (ultimement) périodique, i.e., s'il existe $N, p \geq 1$ tels que $z_n = z_{n+p}$ pour tout $n \geq N$.

DES - AES

Tous les chiffrements rencontrés jusqu'à présent sont à clé secrète (ou privée), i.e., la sécurité du système réside dans la non-divulgation de la clé $k \in \mathcal{K}$ choisie.

Les systèmes rencontrés sont aisément cryptanalysés.

DES : cryptosystème à clé secrète considéré jusqu'il y a peu comme sûr

Data Encryption Standard développé initialement par IBM (1977) comme une variante de LUCIFER.

DES - AES

Les cryptosystèmes à clé secrète (décalage, substitution, Hill, Vigenère) sont idempotents : *composer successivement deux cryptosystèmes du même type est encore un chiffrement de même type*
$\Rightarrow$ aucun intérêt à l'itérer

DES n'est pas idempotent (on le construit sur le produit de deux systèmes qui ne commutent pas).
$\Rightarrow$ La sécurité est augmentée en l'itérant. Ici, $16\times$

Clé de 64 bits

suite aléatoire s de 56 bits, $s = s_1 \cdots s_{56} \in \{0,1\}^{56}$.
On définit une suite de 64 bits $k = k_1 \cdots k_{64}$ t.q.

$$k_1, \ldots, k_7, k_9, \ldots, k_{15}, \ldots, k_{57}, \ldots, k_{63} = s_1, \ldots, s_{56}$$

et

$$\forall j \in \{0, \ldots, 7\}, \quad \sum_{i=1}^{8} k_{j8+i} \equiv 1 \mod 2.$$

Cette condition stipule simplement que la somme de 8 bits consécutifs est toujours impaire (détecter une erreur dans le stockage ou le transfert, bit de parité).

Ces 8 bits ajoutés aux 56 bits de départ ne jouent pas de rôle dans la suite.

On considère un texte clair de 64 bits $\mathbf{x} = x_1 \cdots x_{64} \in \{0,1\}^{64}$.
On lui applique une permutation ν de $\{1,\ldots,64\}$:

58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Ainsi, on considère $\nu(\mathbf{x}) = x_{\nu(1)} \cdots x_{\nu(64)} = x_{58}\, x_{50} \cdots x_7$.
Ce mot $\nu(\mathbf{x})$ est divisé en deux mots de longueur 32,

$$\nu(\mathbf{x}) = L_0 R_0 = \left(x_{\nu(1)} \cdots x_{\nu(32)}\right) \left(x_{\nu(33)} \cdots x_{\nu(64)}\right).$$

Connaissant L_n et R_n ($n \geq 0$), on calcule L_{n+1} et R_{n+1}

$$L_{n+1} = R_n \tag{1}$$

$$R_{n+1} = L_n \oplus f(R_n, K_{n+1}) \tag{2}$$

où $\oplus$ représente le “ou-exclusif”,

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

Jusqu’à obtenir le mot $L_{16}R_{16}$.

Le chiffrement de **x** est $\mathbf{y} = \nu^{-1}(L_{16}R_{16})$. On dit qu’on applique le DES en seize tours.

Comment obtenir les clés $K_1, \ldots, K_{16}$ (processus de diversification des clés) et la fonction f.

On définit d'abord K_0. On l'obtient à partir de k en sélectionnant dans l'ordre les bits (on ne prend pas en compte ici les bits de parité $k_{j8}, j = 1, \ldots, 8$) de k en suivant la permutation

57	49	41	33	25	17	9	1	58	50	42	34	26	18
10	2	59	51	43	35	27	19	11	3	60	52	44	36

pour former une chaîne de longueur 28 : $C_0 = k_{57}\, k_{49} \cdots k_{44}\, k_{36}$ et aussi la permutation

63	55	47	39	31	23	15	7	62	54	46	38	30	22
14	6	61	53	45	37	29	21	13	5	28	20	12	4

pour former une 2ème chaîne de longueur 28 : $D_0 = k_{63}\, k_{55} \cdots k_{12}\, k_4$. Ainsi, $K_0 = C_0 D_0$.

DES - AES

Pour construire K_{n+1}, $n \geq 0$, on procède en deux étapes. Disposant de C_n et D_n, on construit C_{n+1} et D_{n+1} en effectuant une permutation circulaire d'une ou de deux unités vers la gauche sur C_n et D_n de la manière suivante

$n+1$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
# permut.	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

$$C_1 = k_{49}\, k_{41} \cdots k_{44}\, k_{36}\, k_{57},\ C_2 = k_{41} \cdots k_{44}\, k_{36}\, k_{57}\, k_{49},\ \ldots$$

$$D_1 = k_{55}\, k_{47} \cdots k_{12}\, k_4\, k_{63},\ D_2 = k_{47} \cdots k_{12}\, k_4\, k_{63}\, k_{55},\ \ldots.$$

On obtient K_n en sélectionnant 48 des 56 bits de C_nD_n dans l'ordre suivant

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Passons à la définition de f qui prend comme argument un bloc R_n de 32 bits et une clé K_{n+1} de 48 bits pour produire un bloc de 32 bits.
On remplace R_n par un bloc R'_n de 48 bits en recopiant certains des bits de R_n plusieurs fois de la manière suivante (on parle de l'expansion de R_n)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Ainsi, si R_n est de la forme $r_1 r_2 \cdots r_{32}$, on obtient la suite

$$R'_n = r_{32}\, r_1\, r_2\, r_3\, r_4\, r_5\, r_4\, r_5 \cdots r_{31}\, r_{32}\, r_1.$$

On additionne à présent bit à bit R'_n et K_{n+1} modulo 2 pour obtenir B et cette suite est alors découpée en 8 blocs de 6 bits, i.e.,

$$B = R'_n \oplus K_{n+1} = \underbrace{b_1 \cdots b_6}_{B_1}\underbrace{b_7 \cdots b_{12}}_{B_2} \cdots \underbrace{b_{43} \cdots b_{48}}_{B_8}.$$

Transformer chacun des 8 blocs en 8 nouveaux blocs B'_i de longueur 4. La valeur de la fonction f est une permutation de $B'_1 \cdots B'_8$ qui est bien de longueur 32 : $\mu(B'_1 \cdots B'_8)$..
La transformation de B_i en B'_i est réalisée par une table S_i.

S_1 :

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

On l'utilise pour calculer B'_1 à partir de $B_1 = b_1b_2b_3b_4b_5b_6$. Le mot b_1b_6 (resp. $b_2b_3b_4b_5$) représente un entier $0 \leq x \leq 3$ (resp. $0 \leq y \leq 15$) écrit en base 2. Dans le tableau S_1, on considère l'élément $(S_1)_{x,y}$ se trouvant à la ligne x et à la colonne y. La représentation binaire de $(S_1)_{x,y}$ (éventuellement complétée par des zéros de tête pour obtenir un mot de longueur 4) est alors B'_1. On procède de manière semblable pour $B'_2, \ldots, B'_8$ avec les tables données ci-après. La dernière table reprend la permutation μ.

Exemple

Soit $B_1 = 100101$. Le mot $b_1 b_6 = 11$ (resp. $b_2 b_3 b_4 b_5 = 0010$) correspond à $x = 3$ (resp. $y = 2$). Dans la table S_1, cela correspond[1] à l'élément 8 dont la représentation binaire est 0100 qui est B'_1.

[1] Les lignes sont numérotées de 0 à 3 et les colonnes de 0 à 15.

DES - AES

Les tables S sont appelées S-boxes (ou substitution-boxes) : composantes non linéaires du cryptosystème. Aucune des boîtes n'est une fonction linéaire ou affine.

Avantage certain : rapidité. Implémentation réalisée de manière efficace logiciellement ou physiquement sur des circuits électroniques élémentaires à faible coût (exemple, placés sur une carte de crédit).

Critique principale : espace des clés de taille $2^{56} \simeq 7,2 \times 10^{16}$ jugée trop petite. En effet, en 1993, on estimait pouvoir construire une puce capable de tester 5×10^7 clés par seconde pour un coût proche de 8 euros. Ainsi, en regroupant un grand nombre de telles puces, on pourrait trouver la clé secrète en quelques heures.

DES - AES

S_2 :

15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S_3 :

10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S_4 :

7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S_5 :

2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

DES - AES

S_6 :

12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S_7 :

4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S_8 :

13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

μ :

16	7	20	21	29	12	28	17	1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

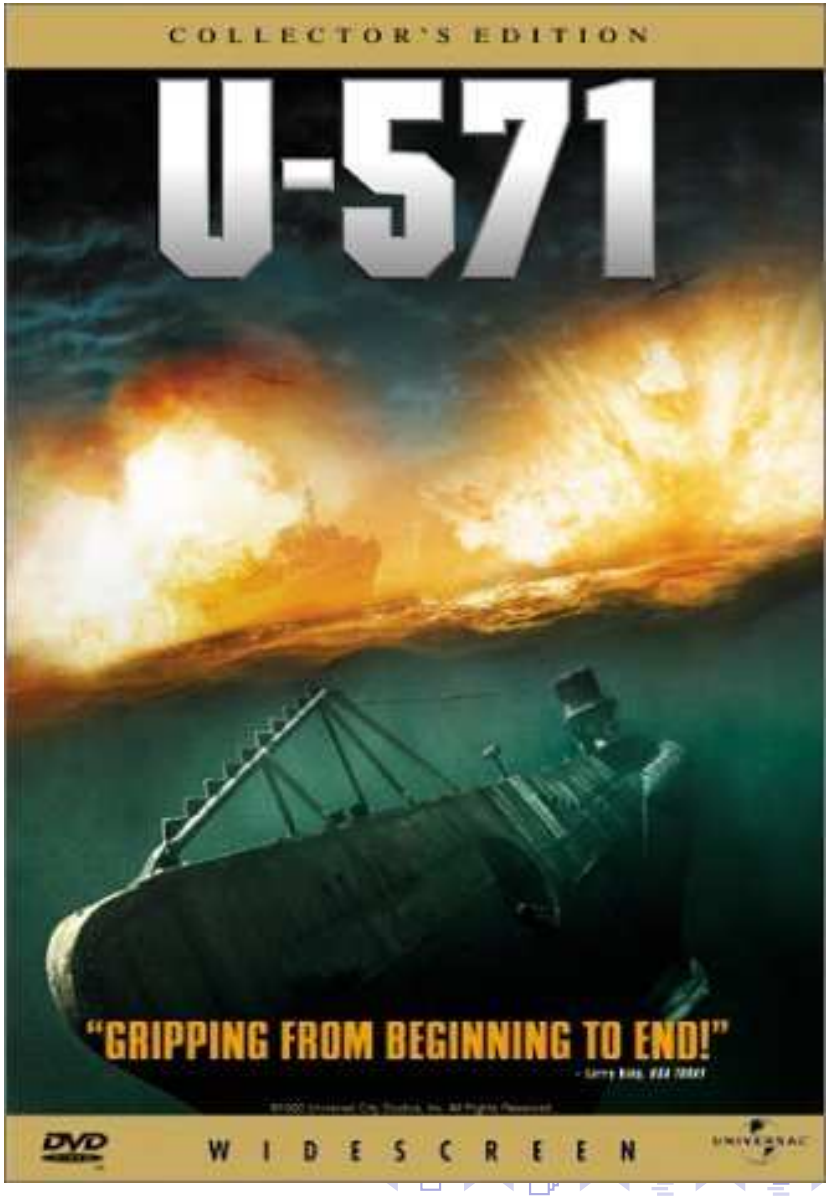
DES - AES

Depuis juin 2001, DES a été officiellement remplacé par l'AES (Advanced Encryption Standard) dont l'algorithme cryptographique *Rijndael* est d'origine belge (Joan Daemen, Vincent Rijmen)

La sélection de cet l'algorithme a été réalisée par un concours initié par le *National Institute of Standards and Technology* (NIST).

Pour palier à l'insécurité grandissante de DES, on a parfois recours à un triple DES.

Historique, la machine Enigma



FIN
de la
cryptographie
à clé secrète