Chiffrement Symétrique

Ludovic Perret

Université Paris VI
ludovic.perret@lip6.fr

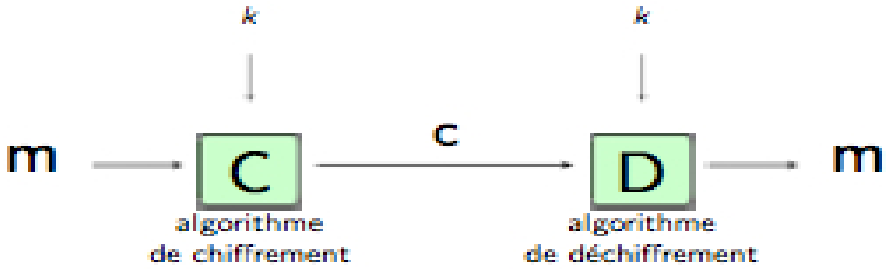
Premier Semestre 2009–2010

Plan du cours

Le chiffrement symétrique

Principe

La clef de chiffrement est la même que la clef de déchiffrement.



Le chiffrement symétrique : caractéristiques

Principe

La clef de chiffrement est la même que la clef de déchiffrement.

Avantages

- rapidité (« hardware » et « software »)
- taille de clefs relativement courtes (128 à 256 bits)

Inconvénients

- Gestion des clefs difficiles
- échange d'un secret préalable

Chiffrement par blocs

Le message est découpé en blocs de taille fixe (typiquement 64 bits). On chiffre ensuite bloc par bloc.

- DES (IBM &NSA, 1975) sur la base de Lucifer (IBM, 1974)
  - blocs de 64 bits, clefs de 56 bits
- IDEA (Lai-Massey, 1992)
  - blocs de 64 bits, clefs de 128 bits
- Rinjdaël (Daemen-Rijmen, 1998)
  - blocs/clefs de taille $\{128, 192, 256\}$

Principe de construction

Constat

- Chiffrement classique : Scytale, Vigenère, . . .
  - faible niveau de sécurité
- Chiffrement de Vernam
  - pas utilisable en pratique

Shannon

La combinaison de confusion et diffusion permet d'obtenir une sécurité convenable.

- confusion : masquer la relation entre message/chiffré
  - Substitution
- diffusion : éparpiller la redondance du message
  - Transposition

Plan du cours

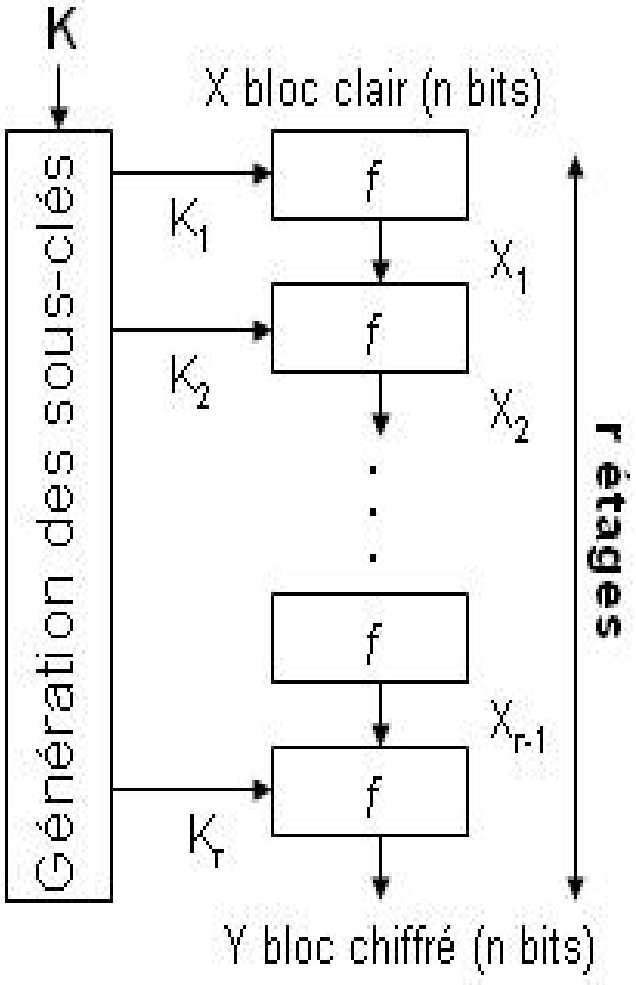
Construction d'un chiffrement par blocs

Principe général

Empilement de **tours**, utilisant chacun un **sous-clef**.

- construction itérative
  - application itérée d'une transformation à chaque tour
- combinaison de transformations élémentaires
  - substitution, transposition, opérations linéaires et arithmétiques
- chaque transformation de tour dépend d'une **sous-clef**
  - Les **sous-clefs** sont générées à partir d'une **clef maître**

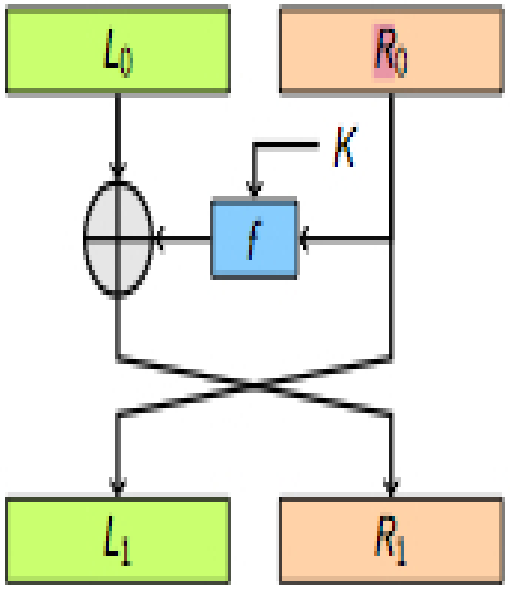
Construction d'un chiffrement par blocs – suite



Ce type de chiffrement est caractérisé par :

- nombre de tours **r**
- la taille des blocs **n**
- la taille de la clef **K**

Schéma de Feistel



Nous avons :

$$\begin{cases} L_1 = R_0 \\ R_1 = L_0 + F(R_0, K) \end{cases}$$

Ainsi :

$$\begin{cases} R_0 = L_1 \\ L_0 = R_1 - F(L_1, K) \end{cases}$$

La fonction F est appelée fontion de confusion.

Schéma de Feistel – (I)

Définition

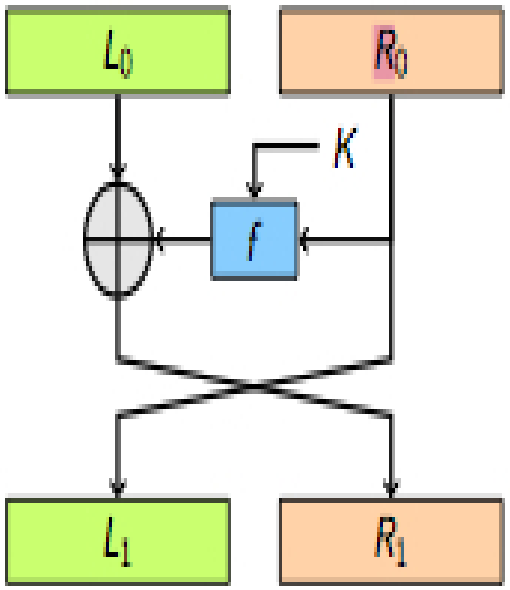
Un schéma de Feistel est un chiffrement itératif par blocs transformant un message $m = (L_0, R_0) \in \mathbb{F}_2^m \times \mathbb{F}_2^m$ en un chiffré $c = (L_{r-1}, R_{r-1}) \in \mathbb{F}_2^m \times \mathbb{F}_2^m$ par un procédé de $r \geq 1$ tours. Chaque tour transforme $(L_{i-1}, \mathrm{R}_{i-1})$ en (L_i, R_i) en utilisant une sous-clef K_i, et une fonction de confusion F, par :

$$L_i = R_{i-1} \qquad R_i = L_{i-1} \oplus F(R_{i-1}, K_i).$$

L'opération est inversible :

$$L_{i-1} = R_i \oplus F(L_i, K_i) \qquad R_{i-1} = L_i.$$

Schéma de Feistel – (II)



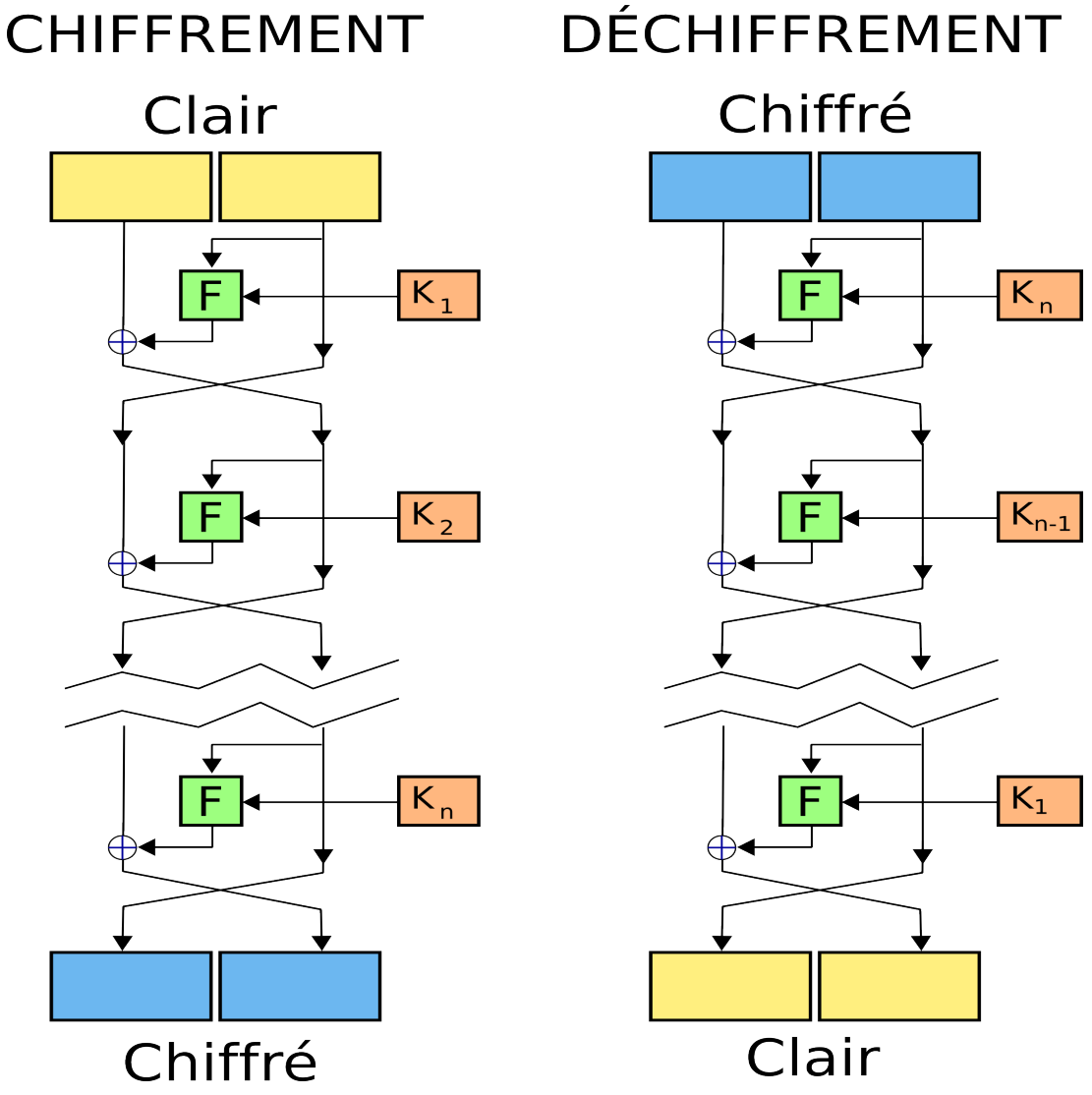
Nous avons :

$$\begin{cases} L_i = R_{i-1} \\ R_i = L_{i-1} \oplus f(R_{i-1}, K) \end{cases}$$

Ainsi :

$$\begin{cases} R_{i-1} = L_i \\ L_{i-1} = R_i \oplus f(L_i, K) \end{cases}$$

Réseau de Feistel



Plan du cours

Data Encryption Standard

DES

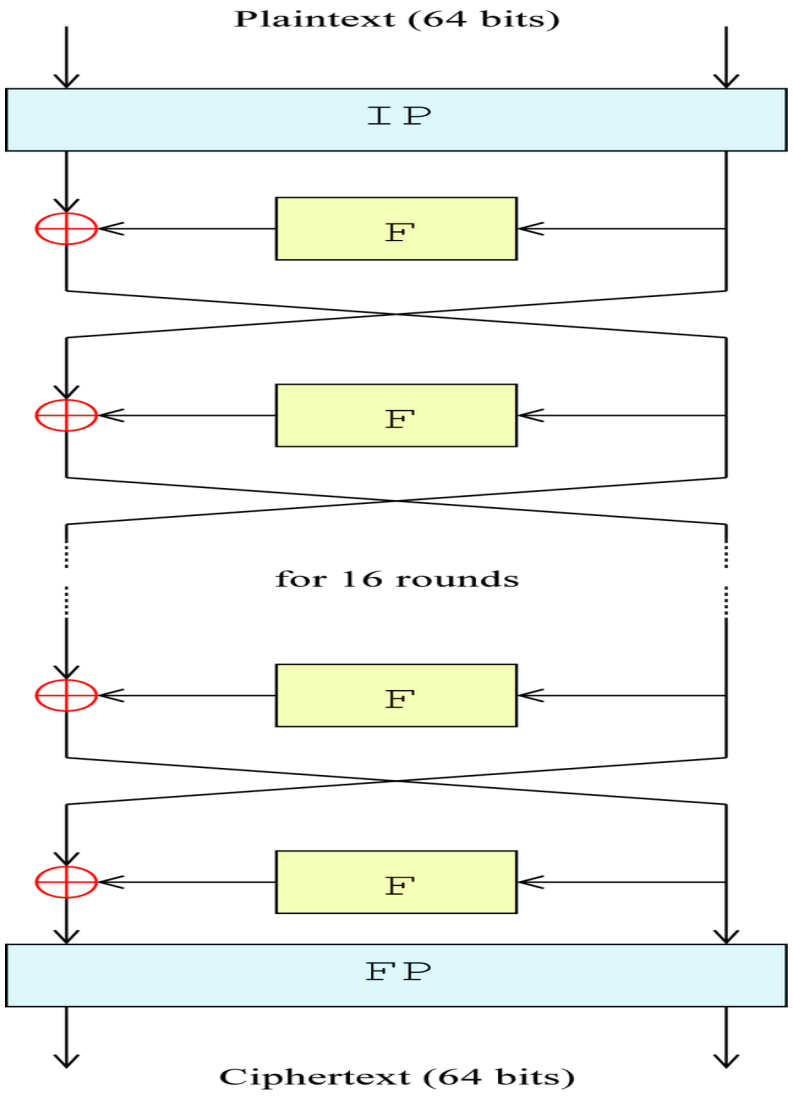
- IBM & NSA (1975) sur la base de Lucifer (IBM, 1974)
- utilise une clef de 56 bits et des blocs de 64 bits
- Feistel à 16 tours avec des sous-clefs de 48 bits
- standardisé en 1977 par le National Bureau of Standards

Controverse

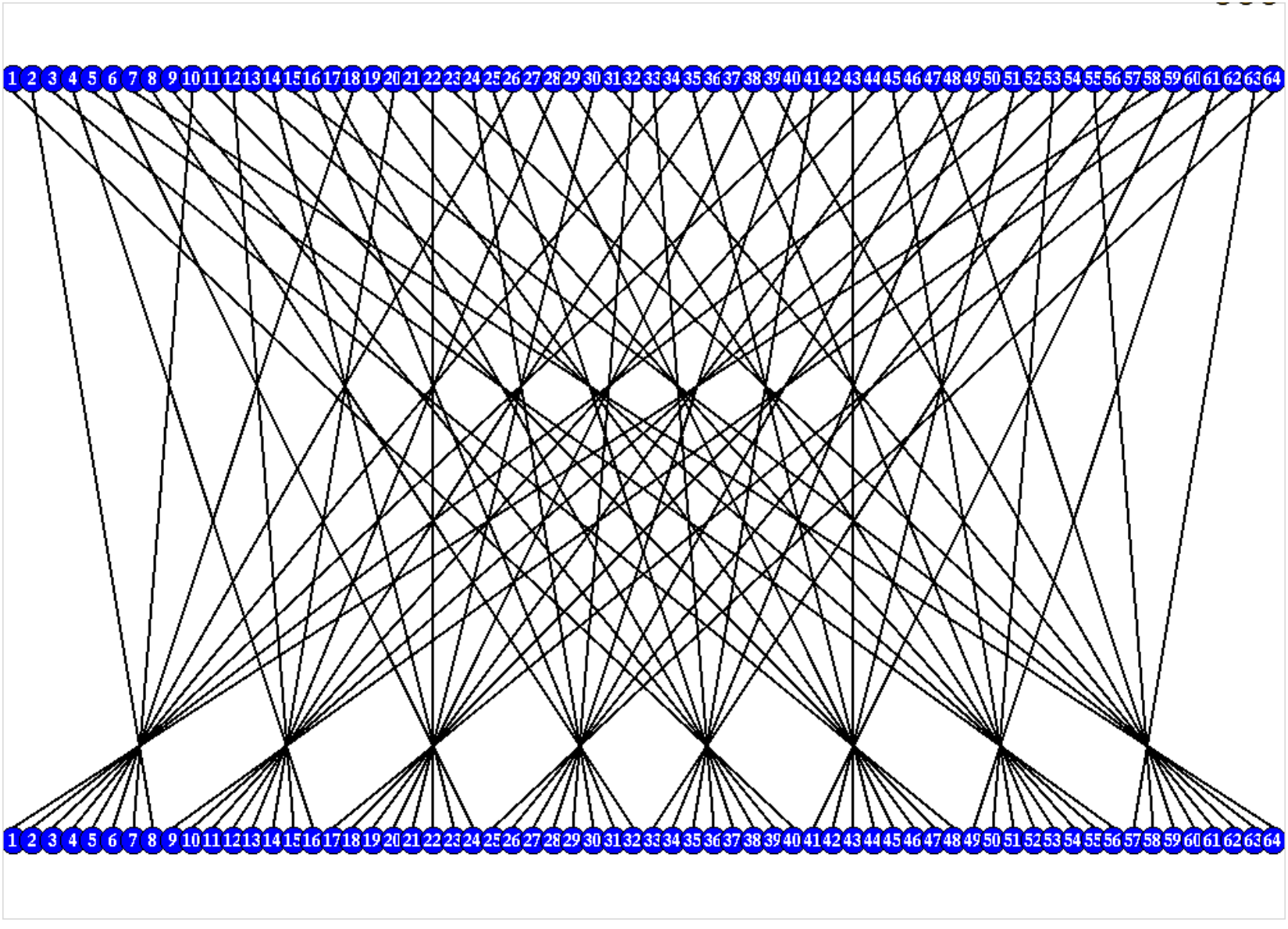
Algorithme repris et corrigé par la NSA (1975)

- modification des boîtes S
- clef réduite de 64 à 56 bits

Structure du DES



Permutation initiale

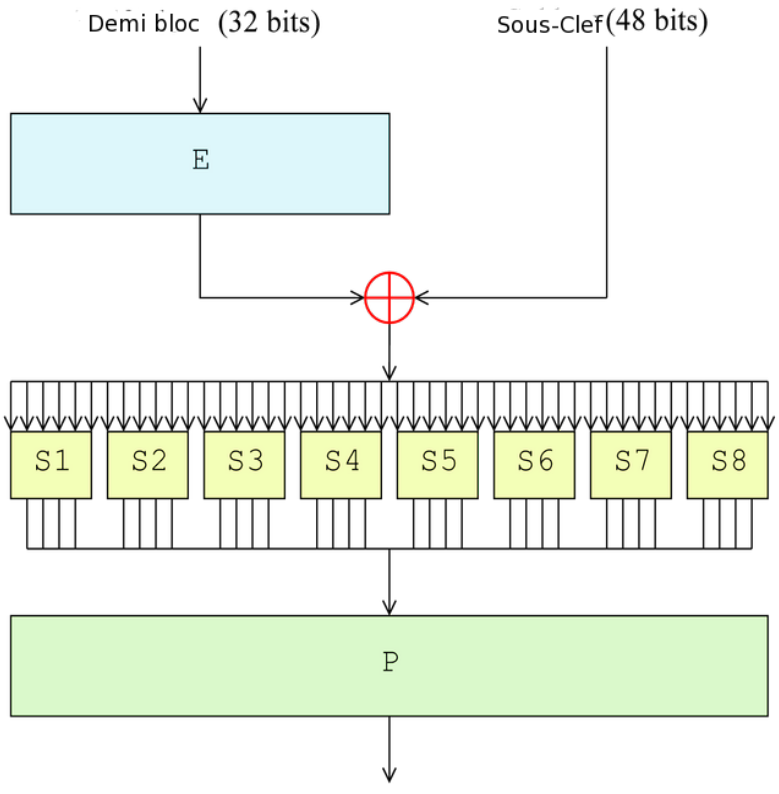


Permutation initiale – suite

$$\mathrm{IP} : \mathbf{x_1 x_2 \cdots x_{63} x_{64}} \in \mathbb{F}_2^{64} \mapsto \mathbf{x_{58} x_{50} \cdots x_{15} x_7} \in \mathbb{F}_2^{64}.$$

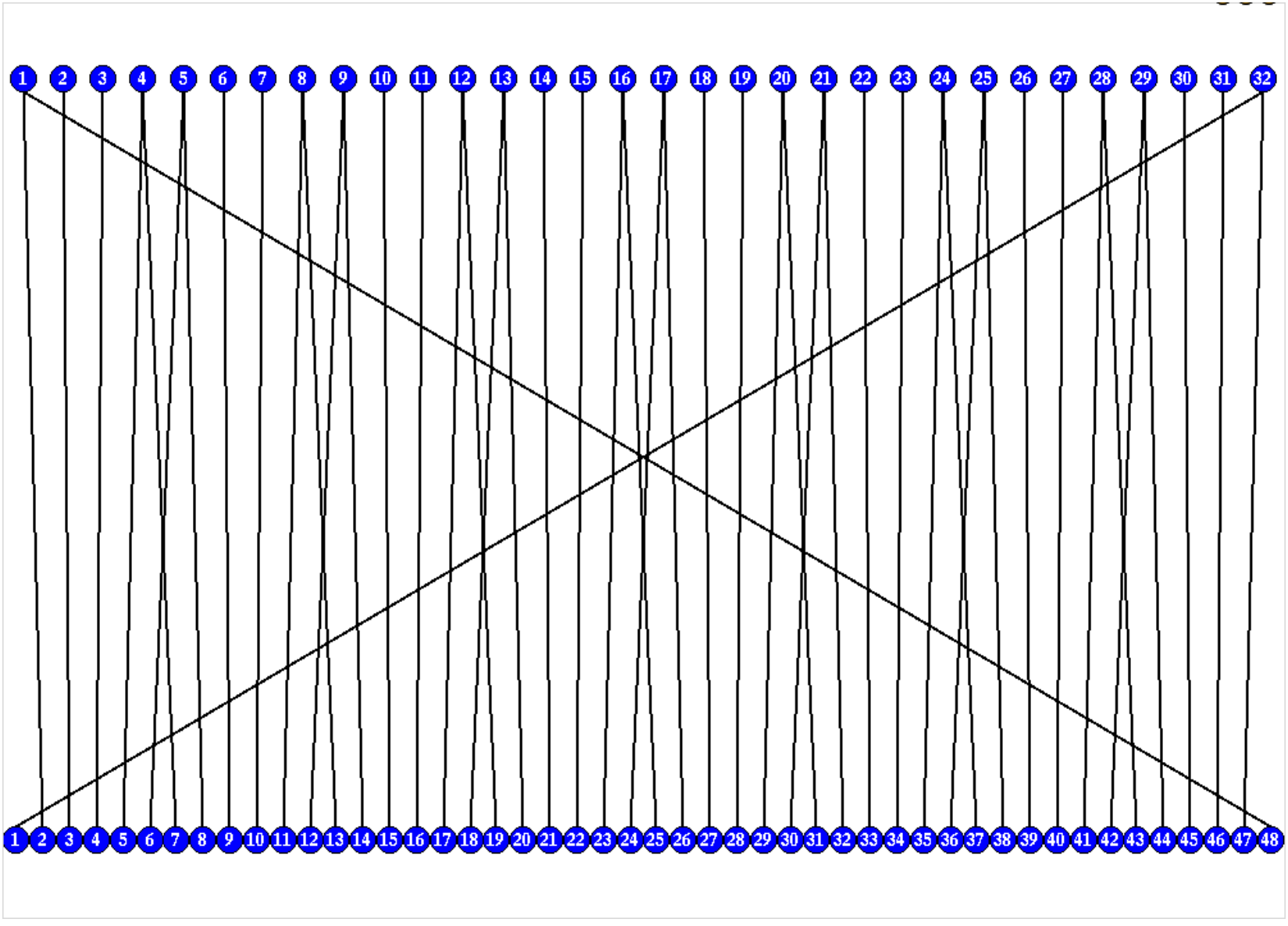
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

Structure de *F*



- expansion $\mathrm{E} : \mathbb{F}_2^{32} \mapsto \mathbb{F}_2^{48}$
- XOR avec la sous-clef
- boites S : $\mathbb{F}_2^{6} \mapsto \mathbb{F}_2^{4}$
- permutation $\mathrm{P} : \mathbb{F}_2^{32} \mapsto \mathbb{F}_2^{32}$

Expansion

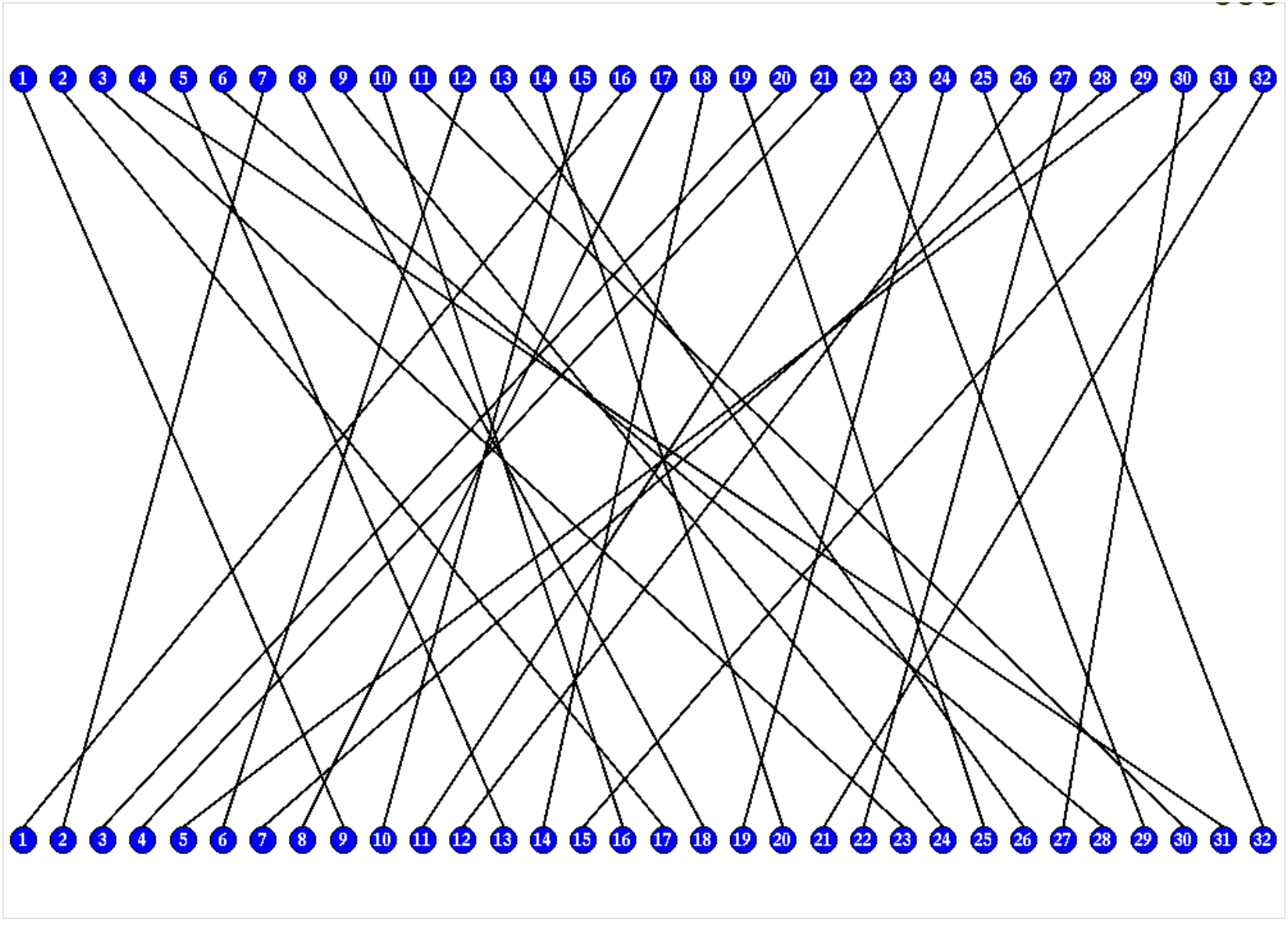

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48

Expansion – suite

$$\mathrm{E} : \mathbf{x_1 x_2} \cdots \mathbf{x_{31} x_{32}} \in \mathbb{F}_2^{32} \mapsto \mathbf{x_{32} x_1} \cdots \mathbf{x_{32} x_1} \in \mathbb{F}_2^{48}.$$

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

Permutation



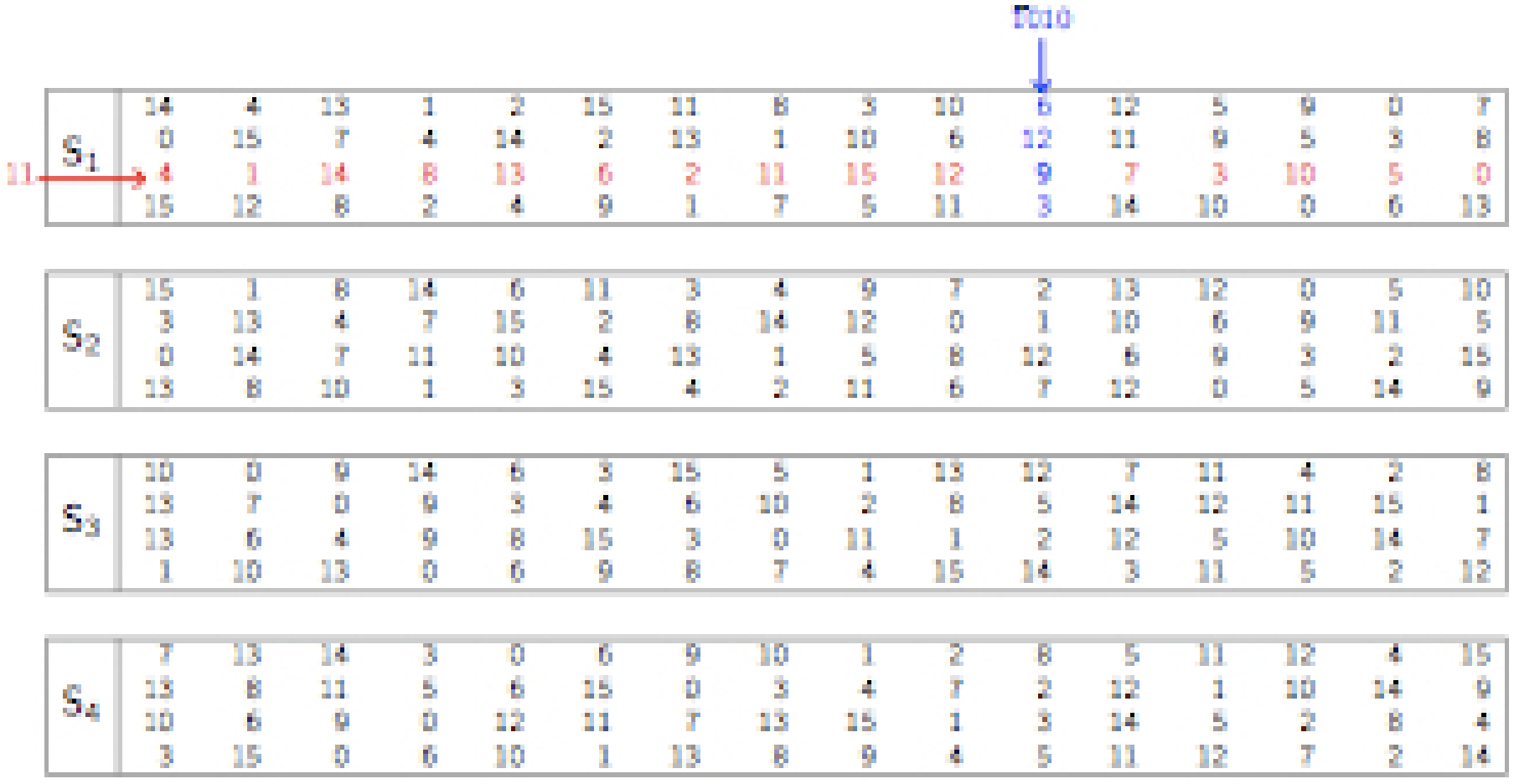
Permutation – suite

$$\mathrm{P} : \mathbf{x_1 x_2 \cdots x_{31} x_{32}} \in \mathbb{F}_2^{32} \mapsto \mathbf{x_{16} x_7 \cdots x_{11} x_{25}} \in \mathbb{F}_2^{32}.$$

16	7	20	21
29	12	28	17
1	15	23	26
2	18	31	10
19	13	13	6
22	11	11	25

Boîtes S

Les bits 1 et 6 désignent la ligne, les bits $2, 3, 4, 5$ la colonne.
Par exemple $S_1(110100) = (9)_{10} = (1001)_2$.



Boîtes *S* – suite

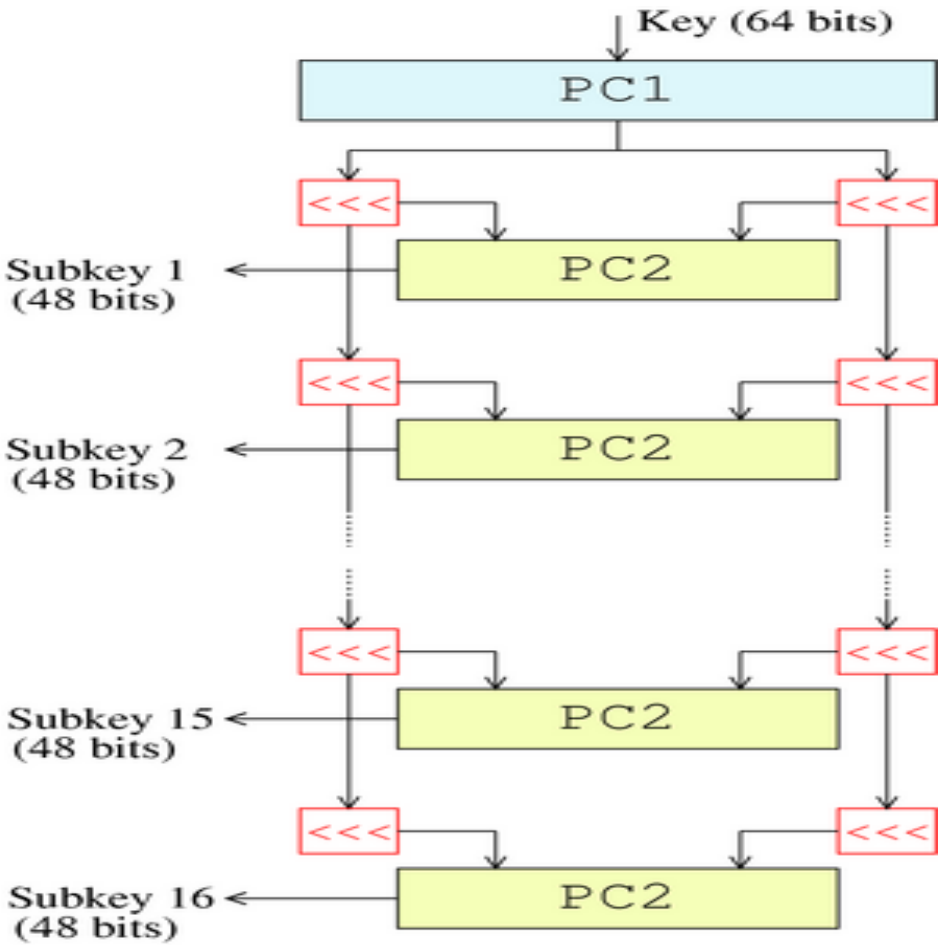
S_5	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

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

Dérivations des sous-clefs



Dérivations des sous-clefs – suite

Les bits $8, 16, 24, 32, 40, 48, 56, 64$ de la clef maître K servent à la détection d'erreurs.

$$\text{PC1} : \mathbf{x_1 x_2 \cdots x_{63} x_{64}} \in \mathbb{F}_2^{64} \mapsto \mathbf{x_{57} x_{49} \cdots x_{12} x_{44}} \in \mathbb{F}_2^{56}.$$

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	46
63	55	49	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	44

Dérivations des sous-clefs – (II)

On note C_0 (resp. D_0) les 28 premiers (resp. derniers) bits.
On calcule ensuite :

$$\begin{cases} C_i = Rol_{\alpha_i}(C_{i-1}) \\ D_i = Rol_{\alpha_i}(D_{i-1}) \end{cases}$$

tour	$[1-2]$	$[3-8]$	9	$[10-15]$	16
α_i	1	2	1	2	1

Dérivations des sous-clefs – (III)

Finalement $K_i = \mathrm{PC2}(C_i D_i)$, avec :

$$\mathrm{PC2} : \mathbf{x_1 x_2 \cdots x_{31} x_{56}} \in \mathbb{F}_2^{56} \mapsto \mathbf{x_{14} x_{17} \cdots x_{29} x_{32}} \in \mathbb{F}_2^{56}.$$

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

Sécurité du DES

- Attaques sur des versions réduites (moins de 16 tours)
- 1990 – 1992 : cryptanalyse différentielle (Biham-Shamir)
- 1993 : cryptanalyse linéaire (Matsui)

Limite du DES

La recherche exhaustive sur 56 bits (2^{56}) est maintenant réaliste.

Electronic Frontier Foundation

- Janvier 1998 : 39 j. avec 10000 Pentium
- Janvier 1998 : 3 j. avec une machine dédiée (coût 250 000 $)
- Janvier 1999 : 22h15m.
- Mars 2007 : 6.4 j, COPACOBANA (utilisation de FPGA) par l'université de Bochum et Kiel (coût 10 000 $)