

DES  
oo

permutations  
ooo

fonction F  
oooooooooooo

diversification clé  
oooooo

clés faibles  
ooooo

cryptanalyse  
ooooo

triple DES  
o

# Mathématiques-Cryptographie

Anne Garcia-Sanchez

M2i cyber2 dev - CFA CCI Avignon

14 mai 2024

# Chiffrement DES

DES = *Data Encryption Standard*

standard de chiffrement adopté par le gouvernement des États-Unis en 1977.

schéma de Feistel à 16 tours.

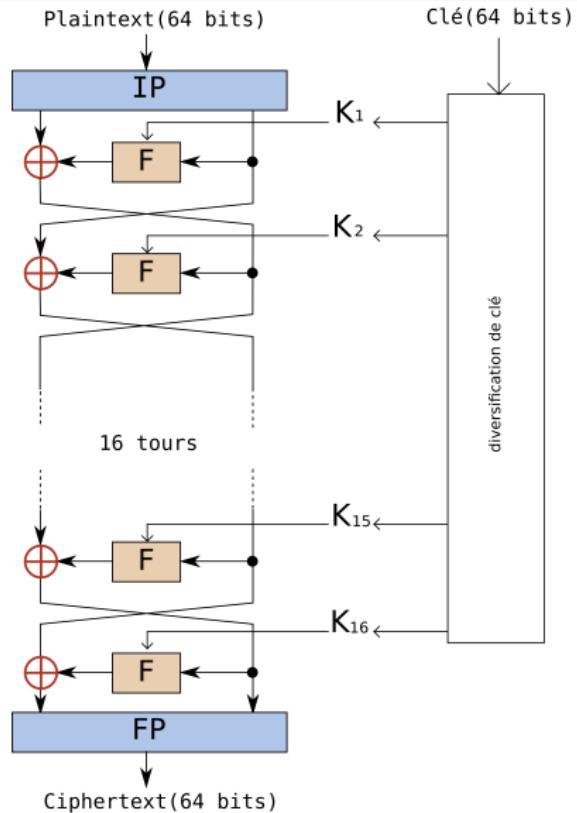
- blocs de 64 bits
- clé de 56 bits: 64 bits avec 8 bits de parité

bits de parité: permettent d'avoir un nombre impair de "1" dans chaque octet

Exemple:

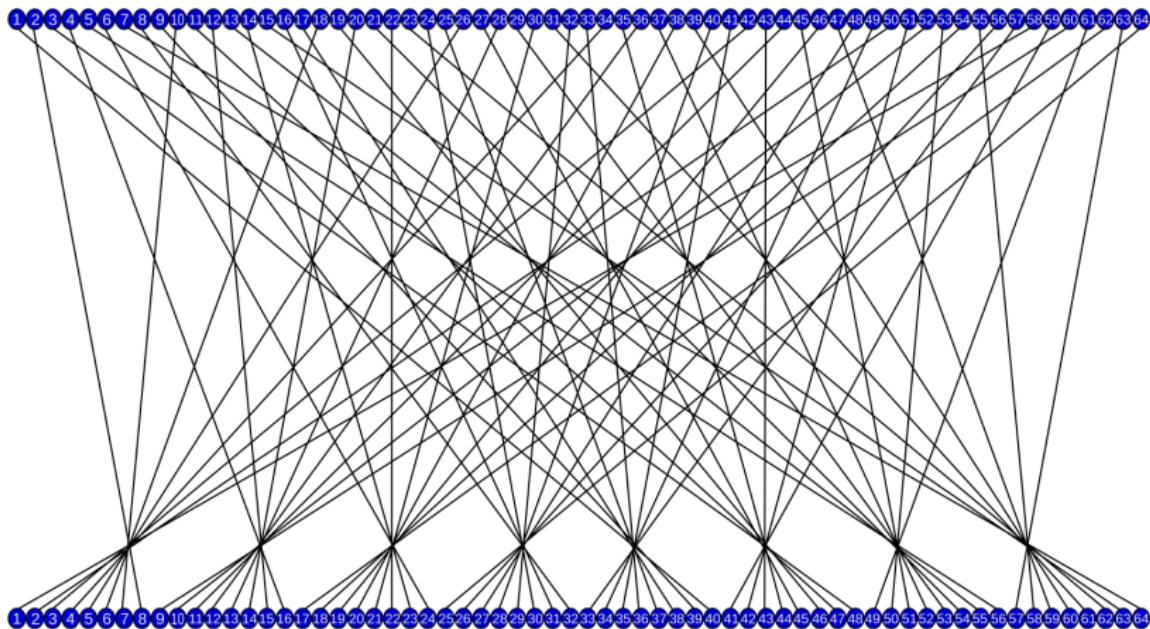
00000001 11111110 00000111 00001110 00011111 01010100  
11111101 11111000

# Chiffrement DES



## Chiffrement DES

### Permutation initiale - IP *Initial Permutation*



# Chiffrement DES

permutation donnée sous forme de tableau:

premier bit de sortie provient du 58ème bit d'entrée  
second bit provient du 50ème...

IP

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

# Chiffrement DES

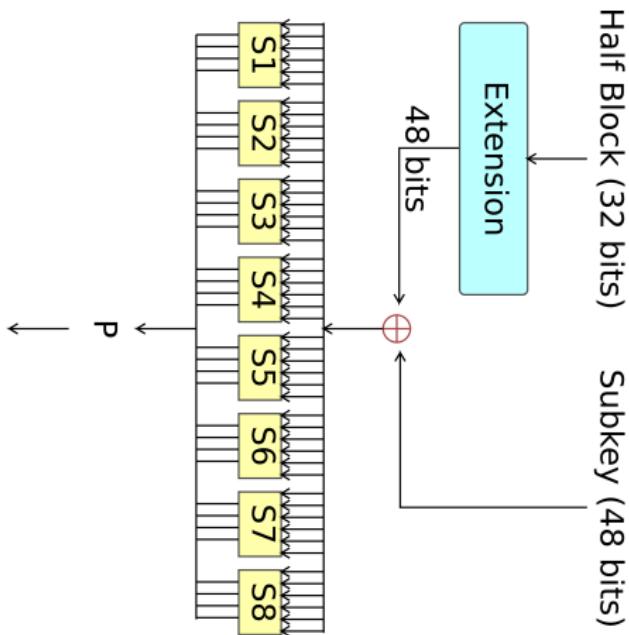
fonction inverse de la permutation initiale  $IP^{-1}$  (ou FP pour final permutation) donnée par le tableau:

**IP<sup>-1</sup>**

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

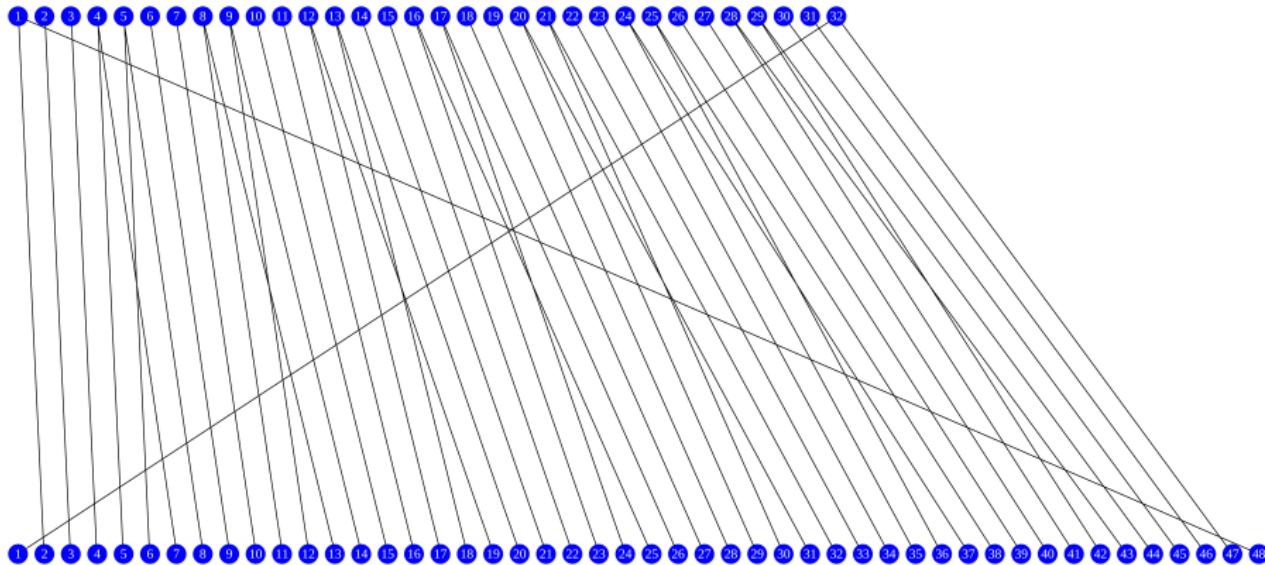
# Chiffrement DES - fonction F du schéma de Feistel

Lors d'un tour de DES, un bloc de 32 bits subit la transformation suivante (fonction F):



# Chiffrement DES - fonction F du schéma de Feistel

- les 32 bits sont étendus en 48 bits (16 sont dupliqués) par une fonction d'extension  $E$ :



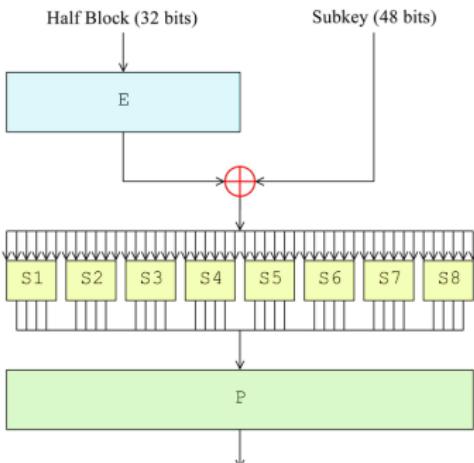
# Chiffrement DES - fonction F du schéma de Feistel

Cette fonction d'extension est donnée par le tableau suivant:

E					
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

# Chiffrement DES- fonction F du schéma de Feistel - S-Box

- ou exclusif bit à bit entre les 48 bits obtenus par l'extension et les 48 bits de la clé de tour
  - les 48 bits obtenus par ou exclusif sont découpés en 8 blocs de 6 bits
- chaque bloc passe par une boîte de substitution (boîtes S ou *S-Box*) qui substitue 6 bits par 4 bits.



# Chiffrement DES - S-Box

lecture tables de substitution des boîtes S:

- ligne : 2 bits aux extrémités
- colonne: 4 bits du centre.

Exemple: entrée "011011" = "0 1101 1"

ligne "01" / colonne "1101"

Sortie: "1001".

S <sub>5</sub>		4 bits au centre de l'entrée																	
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	1110	1111
Bits externes	00	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001	1100	
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110	0111	
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110	0011	
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011	0101	

# Chiffrement DES - S-Box

tables données en hexadécimal sur le document du Nist.

$S_5$		4 bits au centre de l'entrée															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Bits externes	00	0010	1100	0100	0001	0111	1010	1011	0110	1000	0101	0011	1111	1101	0000	1110	1001
	01	1110	1011	0010	1100	0100	0111	1101	0001	0101	0000	1111	1010	0011	1001	1000	0110
	10	0100	0010	0001	1011	1010	1101	0111	1000	1111	1001	1100	0101	0110	0011	0000	1110
	11	1011	1000	1100	0111	0001	1110	0010	1101	0110	1111	0000	1001	1010	0100	0101	0011

$S_5$

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

# Chiffrement DES - S-Box

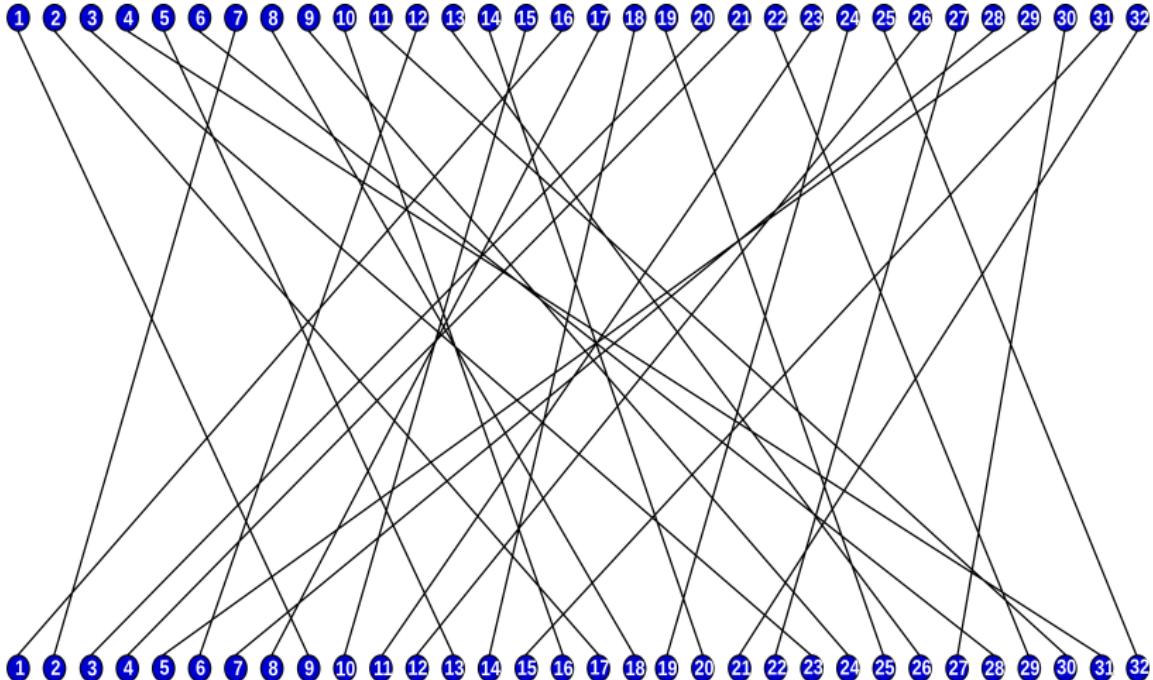
$S_1$	x0000x	x0001x	x0010x	x0011x	x0100x	x0101x	x0110x	x0111x	x1000x	x1001x	x1010x	x1011x	x1100x	x1101x	x1110x	x1111x
0yyyy0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0yyyy1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
1yyyy0	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
1yyyy1	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
$S_2$	x0000x	x0001x	x0010x	x0011x	x0100x	x0101x	x0110x	x0111x	x1000x	x1001x	x1010x	x1011x	x1100x	x1101x	x1110x	x1111x
0yyyy0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
0yyyy1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
1yyyy0	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
1yyyy1	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
$S_3$	x0000x	x0001x	x0010x	x0011x	x0100x	x0101x	x0110x	x0111x	x1000x	x1001x	x1010x	x1011x	x1100x	x1101x	x1110x	x1111x
0yyyy0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
0yyyy1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
1yyyy0	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1yyyy1	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
$S_4$	x0000x	x0001x	x0010x	x0011x	x0100x	x0101x	x0110x	x0111x	x1000x	x1001x	x1010x	x1011x	x1100x	x1101x	x1110x	x1111x
0yyyy0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
0yyyy1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
1yyyy0	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
1yyyy1	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

# Chiffrement DES - S-Box

$S_5$	$x0000x$	$x0001x$	$x0010x$	$x0011x$	$x0100x$	$x0101x$	$x0110x$	$x0111x$	$x1000x$	$x1001x$	$x1010x$	$x1011x$	$x1100x$	$x1101x$	$x1110x$	$x1111x$
0yyyy0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
0yyyy1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
1yyyy0	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
1yyyy1	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
$S_6$	$x0000x$	$x0001x$	$x0010x$	$x0011x$	$x0100x$	$x0101x$	$x0110x$	$x0111x$	$x1000x$	$x1001x$	$x1010x$	$x1011x$	$x1100x$	$x1101x$	$x1110x$	$x1111x$
0yyyy0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
0yyyy1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
1yyyy0	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
1yyyy1	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
$S_7$	$x0000x$	$x0001x$	$x0010x$	$x0011x$	$x0100x$	$x0101x$	$x0110x$	$x0111x$	$x1000x$	$x1001x$	$x1010x$	$x1011x$	$x1100x$	$x1101x$	$x1110x$	$x1111x$
0yyyy0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
0yyyy1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1yyyy0	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
1yyyy1	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
$S_8$	$x0000x$	$x0001x$	$x0010x$	$x0011x$	$x0100x$	$x0101x$	$x0110x$	$x0111x$	$x1000x$	$x1001x$	$x1010x$	$x1011x$	$x1100x$	$x1101x$	$x1110x$	$x1111x$
0yyyy0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
0yyyy1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
1yyyy0	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
1yyyy1	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

# Chiffrement DES - P-Box

- les 32 bits obtenus à la sortie des boîtes S subissent une permutation par une boîte de permutation: boîte P ou P-Box.

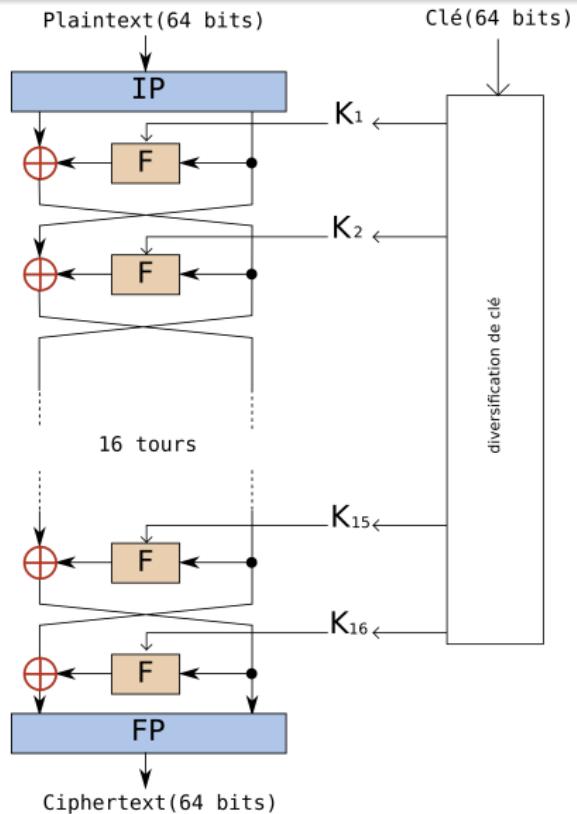


# Chiffrement DES - P-Box

Cette permutation est donnée par le tableau suivant:

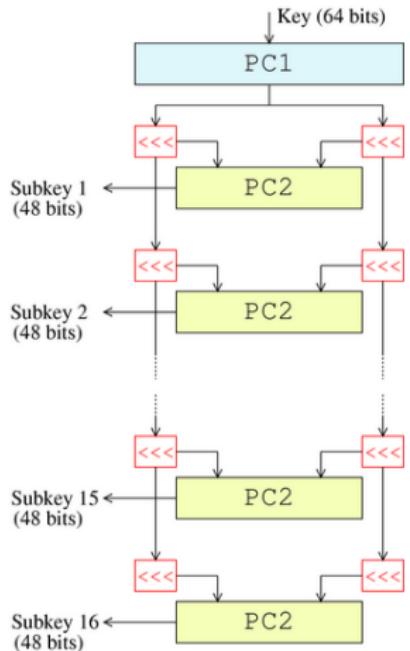
P			
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

# Chiffrement DES



# Chiffrement DES - Diversification de clé

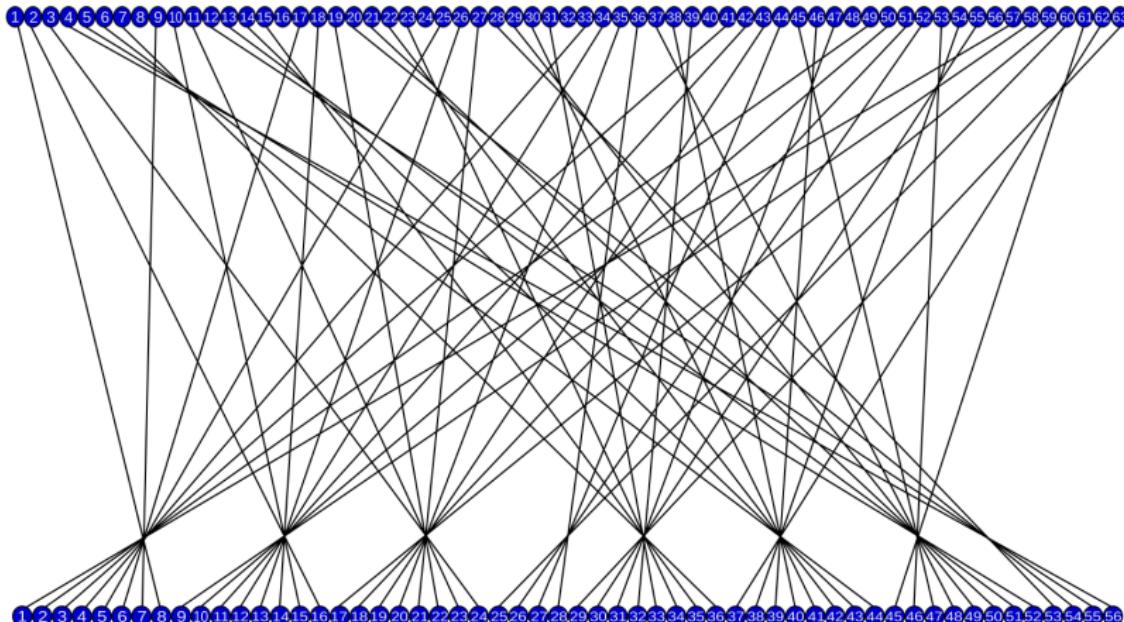
16 sous-clés de 48 bits obtenues à partir de la clé initiale de 64 bits dont 8 bits de parité.



# Chiffrement DES - Diversification de clé

clé initiale: permutation - fonction PC-1 *Permuted Choice 1.*

bits de parité positions 8, 16, 24, 32, 40, 48, 56 et 64 non utilisés.



# Chiffrement DES - Diversification de clé

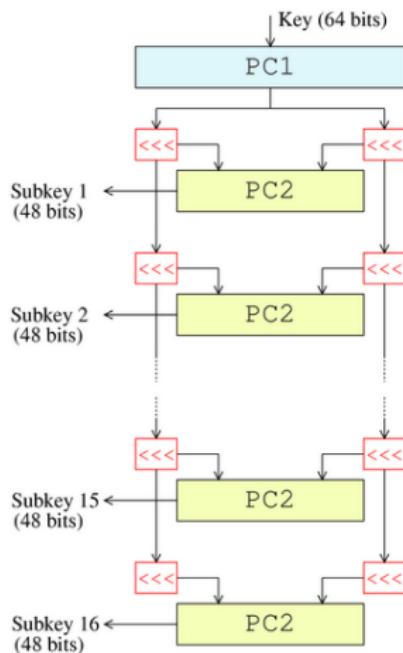
Cette permutation est donnée par le tableau suivant:

**PC-1**

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

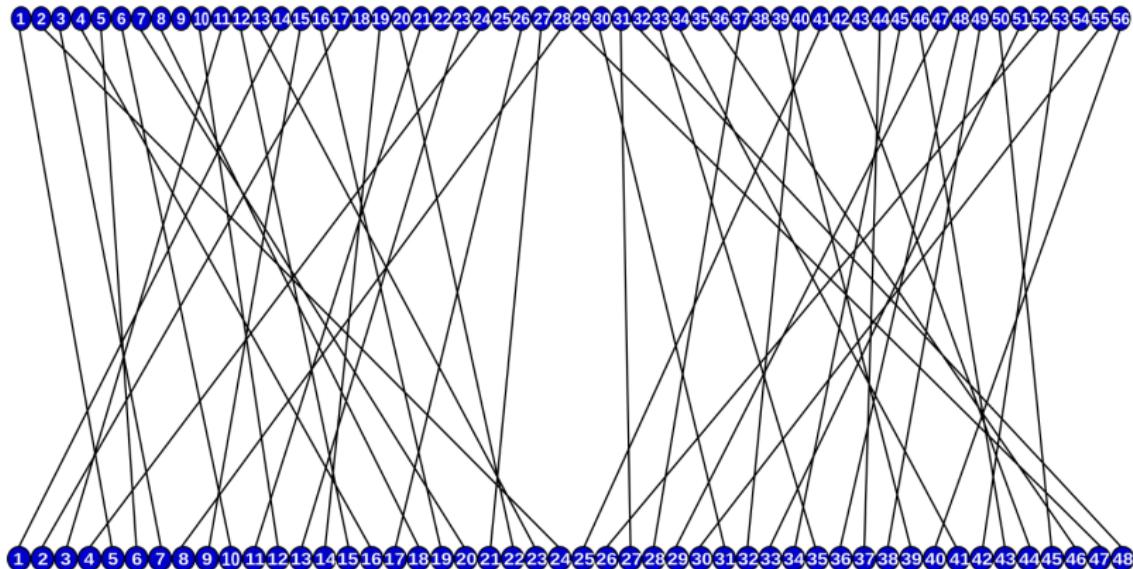
# Chiffrement DES - Diversification de clé

- chaque moitié de la clé: rotation à gauche d'un bit aux étapes 1, 2, 9 et 16 et de deux bits aux autres étapes.



## Chiffrement DES - Diversification de clé

fonction PC-2 *Permuted Choice 2* : clé partielle de 48 bits.



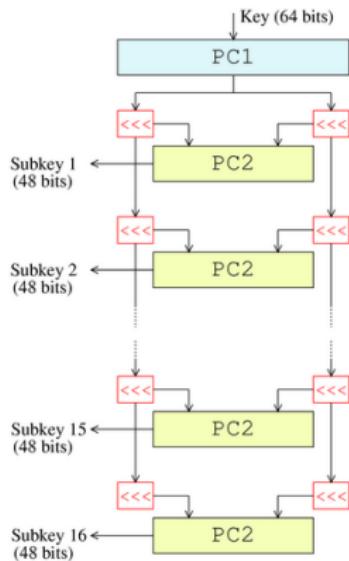
# Chiffrement DES - Diversification de clé

Cette permutation est donnée par le tableau suivant:

**PC-2**

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

# Chiffrement DES - Diversification de clé



Cherchons quelles sont les 16 sous-clés obtenues à partir de la clé  
0000 0001 0000 0001 0000 0001....0000 0001

# Chiffrement DES - Diversification de clé

la clé 0000 0001 0000 0001 0000 0001....0000 0001 donne 16 sous-clés identiques:

les bits de parités ne sont pas utilisés. Il ne reste que des 0.

Par permutations et rotations on n'obtient toujours que des 0.

0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

Cherchons une autre clé qui donnera 16 sous-clés identiques.

# Chiffrement DES - Diversification de clé

la clé 1111 1110 1111 1110 .... 1111 1110 donne 16 sous-clés identiques:

1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111

Cherchons d'autres clés qui donneront 16 sous-clés identiques.

# Chiffrement DES - Diversification de clé

la clé E0 E0 E0 E0 F1 F1 F1 F1 correspondant à  
1110 0000 1110 0000 1110 0000...1111 0001 1111 0001 1111 0001  
donne 16 sous-clés identiques:  
1111 1111 1111 1111 1111 1111 0000 0000 0000 0000 0000 0000

La clé 1F 1F 1F 1F 0E 0E 0E 0E  
donne 16 sous-clés identiques:  
0000 0000 0000 0000 0000 1111 1111 1111 1111 1111 1111 1111

# Chiffrement DES - Clés faibles et semi-faibles du DES

**clés faibles:** les 16 clés de tours obtenues sont toutes égales.

problème: chiffrement = déchiffrement

$$DES_K(DES_K(m)) = m \text{ pour tout bloc } m \text{ de 64 bits}$$

**clés semi-faibles:** six couples de clés  $(K_1, K_2)$  tels que

$$DES_{K_1}(DES_{K_2}(m)) = m \text{ pour tout bloc } m \text{ de 64 bits}$$

**Rappel:** il y a  $2^{56}$  clés possibles pour le DES.

# Chiffrement DES - Cryptanalyse: brute force

$2^{56}$  clés possibles: brute force possible

EFF DES cracker (surnommé Deep Crack) machine conçue par l'EFF (Electronic Frontier Foundation) en 1998 pour attaquer le DES

a permis de décrypter un message en 56 heures en juillet 1998.

# Chiffrement DES - Cryptanalyse différentielle

**cryptanalyse différentielle:** Eli Biham et Adi Shamir, 1991

permet de trouver la clé en utilisant  $2^{47}$  textes clairs choisis.

observation des différences entre les chiffrés obtenus pour des couples de textes clairs ayant une différence constante.

Une analyse statistique permet de retrouver la clé.

Le chiffrement DES a été conçu pour résister à la cryptanalyse différentielle.

# Chiffrement DES - Cryptanalyse linéaire

**cryptanalyse linéaire:** formalisée par Mitsuru Matsui en 1993.

idée: faire une approximation linéaire de certains tours de l'algorithme de chiffrement en le simplifiant.

Mitsuru Matsui a montré qu'on peut attaquer le DES en  $2^{43}$  chiffrements à l'aide de  $2^{43}$  couples de clairs/chiffrés.

DES  
oopermutations  
ooofonction F  
oooooooooooodiversification clé  
oooooooclés faibles  
ooooocryptanalyse  
oooo●○triple DES  
○

# Chiffrement DES - double-DES

Comment augmenter la sécurité du DES?

chiffrement double?

$c = E_{k_2}(E_{k_1}(m))$  avec  $k_1$  et  $k_2$  deux clés aléatoires indépendantes.

déchiffrement:  $m = D_{k_1}(D_{k_2}(c))$

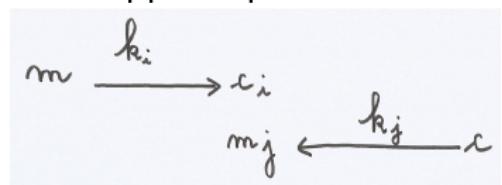
# Attaque par rencontre au milieu - Meet-in-the-middle

double DES = chiffrement avec une clé de taille  $56 \times 2 = 112$ ?

vulnérable à une attaque à clair connu:

**attaque par rencontre au milieu: meet-in-the-middle attack.**

ceci suppose que l'on connaît un couple (clair, chiffré):  $(m, c)$



Si  $c_i = m_j$ , les clés recherchées sont  $i, j$ .

Si on néglige la comparaison, la recherche demande  $2^{56} + 2^{56}$  opérations soit  $2^{57}$  et non  $2^{112}$

# triple DES

## chiffrement triple

La taille de la clé est de  $56 \times 3 = 168$  bits.

L'attaque *Meet-in-the-middle* permet de retrouver la clé en environ  $2^{112}$  chiffrements.

On utilise souvent seulement deux clés:  $c = E_{k_1}(D_{k_2}(E_{k_1}(m)))$   
chiffrement noté 3DES-EDE: a l'avantage d'être compatible avec le  
chiffrement DES si  $k_1 = k_2$

3DES jugé trop lent  $\rightarrow$  depuis 2000, nouveau standard = AES