Chapitre 8

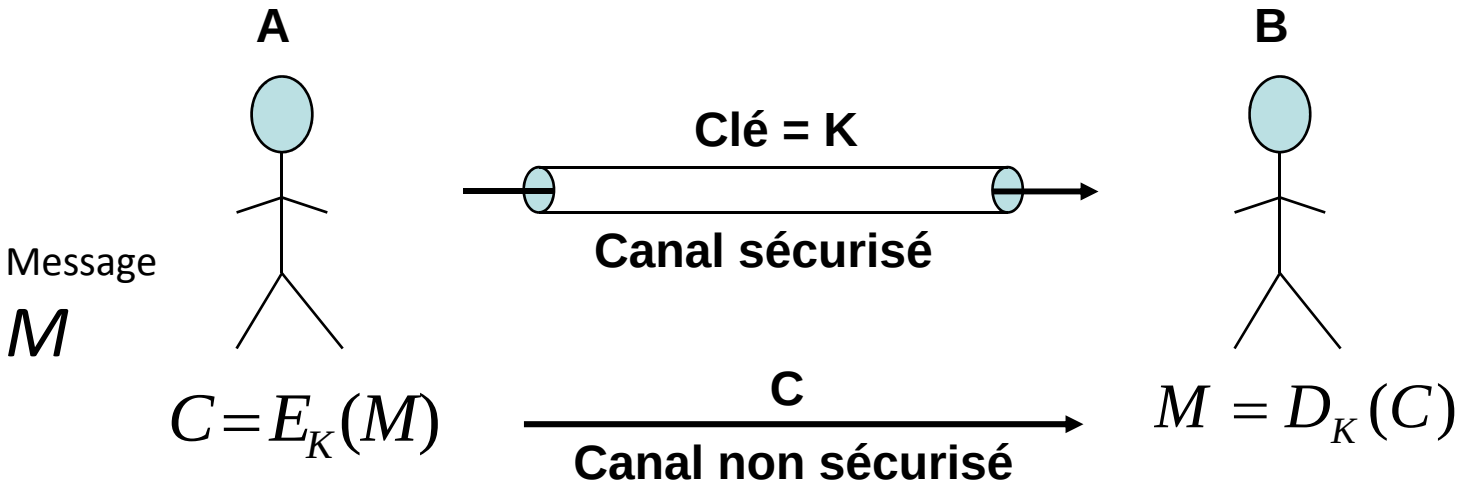
Crypto systèmes symétriques

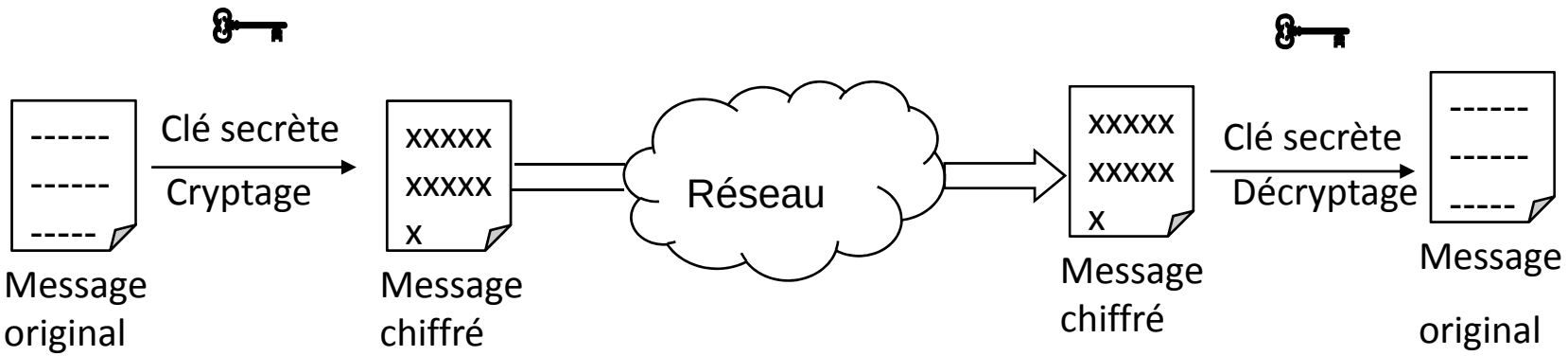
Remarques:

1) Ce document doit être complété par les notes de cours
2) La plupart des transparents sont basés sur le « handbook of applied cryptography »

Chiffrement symétrique: Rappel

- Schéma général du chiffrement symétrique





- Exemples d'algorithmes: DES, AES, IDEA…

Cryptosystèmes symetriques modernes

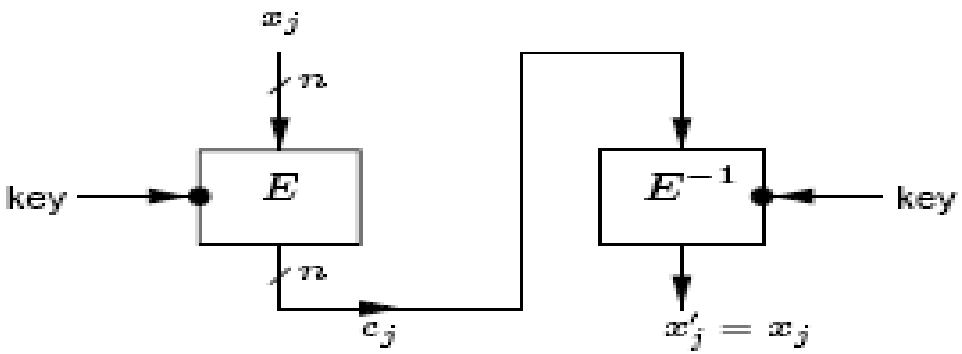
- Deux modes de chiffrement
  - En Stream
  - Par bloc
    - Segmentation du message M à chiffrer
      - M est scindé en un nombre de bloc de taille fixe
    - Cryptage des blocs
    - C est obtenu en concaténant les cryptogrammes des bloc

- Modes de chiffrement par bloc
  - ECB (Electronic CodeBook)
  - CBC(Cipher bloc Chaining)
  - CBF (Cipher FeedBack)
  - OFB (Output FeedBack)

Modes de chiffrement par bloc

- ECB (Electronic CodeBook)
  - Un bloc de texte se chiffre indépendamment de tout en un bloc de texte chiffré

a) Electronic Codebook (ECB)



(i) encipherment (ii) decipherment

7.11 Algorithm ECB mode of operation

INPUT: k-bit key K; n-bit plaintext blocks $x_1, \ldots, x_t$.
SUMMARY: produce ciphertext blocks $c_1, \ldots, c_t$; decrypt to recover plaintext.

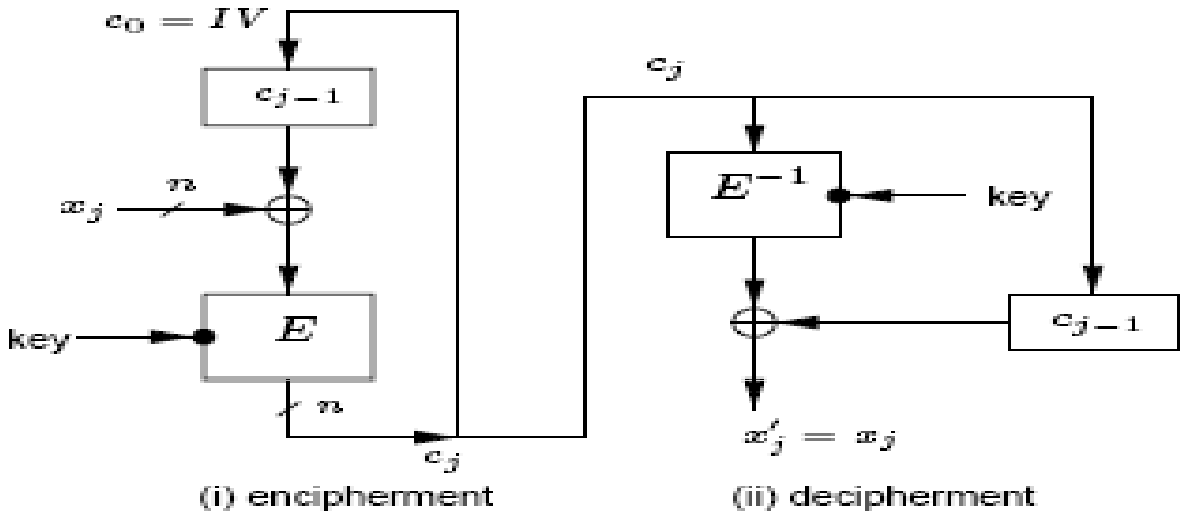
1. Encryption: for $1 \leq j \leq t$, $c_j \leftarrow E_K(x_j)$.
2. Decryption: for $1 \leq j \leq t$, $x_j \leftarrow E_K^{-1}(c_j)$.

Modes de chiffrement par bloc

CBC (Cipher Block Chaining)

- Chaque bloc du cryptogramme dépend du bloc de texte en clair et de tous les blocs précédents

b) Cipher-block Chaining (CBC)



7.13 Algorithm CBC mode of operation

INPUT: k-bit key K; n-bit IV; n-bit plaintext blocks $x_1, \ldots, x_t$.
SUMMARY: produce ciphertext blocks $c_1, \ldots, c_t$; decrypt to recover plaintext.

1. Encryption: $c_0 \leftarrow IV$. For $1 \leq j \leq t$, $c_j \leftarrow E_K(c_{j-1} \oplus x_j)$.
2. Decryption: $c_0 \leftarrow IV$. For $1 \leq j \leq t$, $x_j \leftarrow c_{j-1} \oplus E_K^{-1}(c_j)$.

Data Encryption Standard (DES)

- Généralités:
  - Chiffrement **par bloc** : 64bits
  - Clé de taille variable:
    - Entre 56 et 128 bits selon le niveau de sécurité désiré
    - Version initiale du DES utilise une clé de taille 64 bits dont 56 sont réellement utilisés
  - Utilise un **chiffrement produit**
    - Combine des algorithmes de **substitution** et des algorithmes de **transposition** ➔ maximiser la complexité de l'algorithme

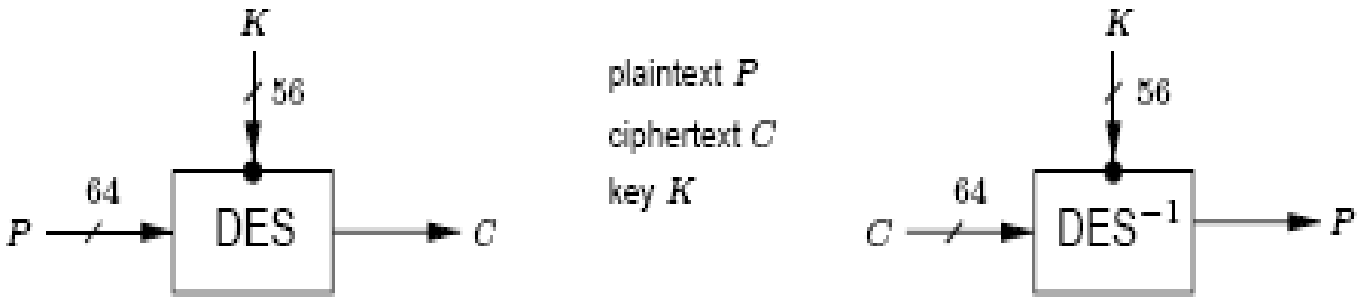
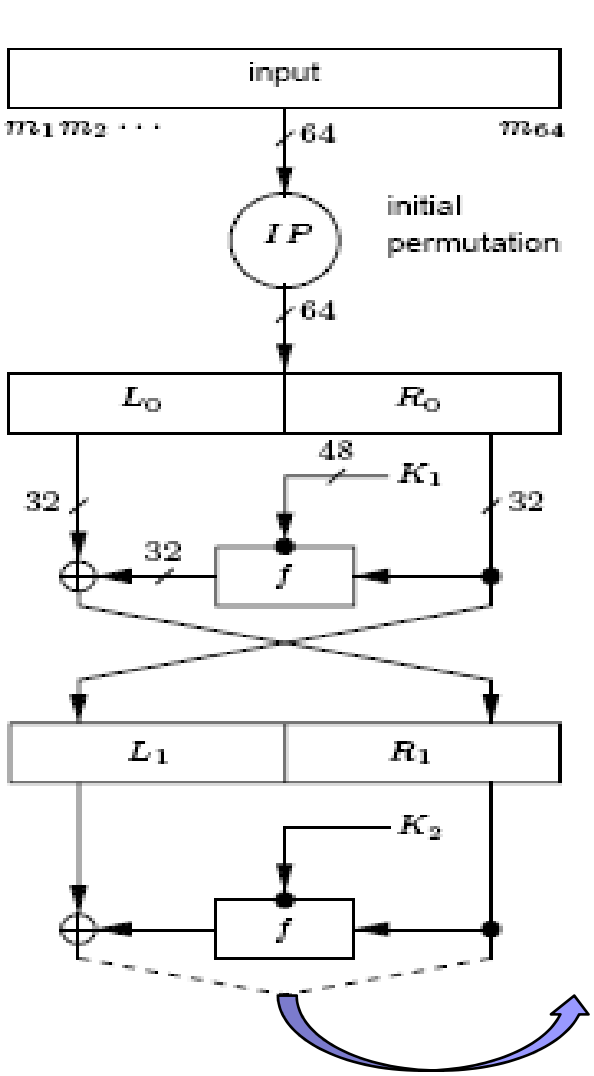
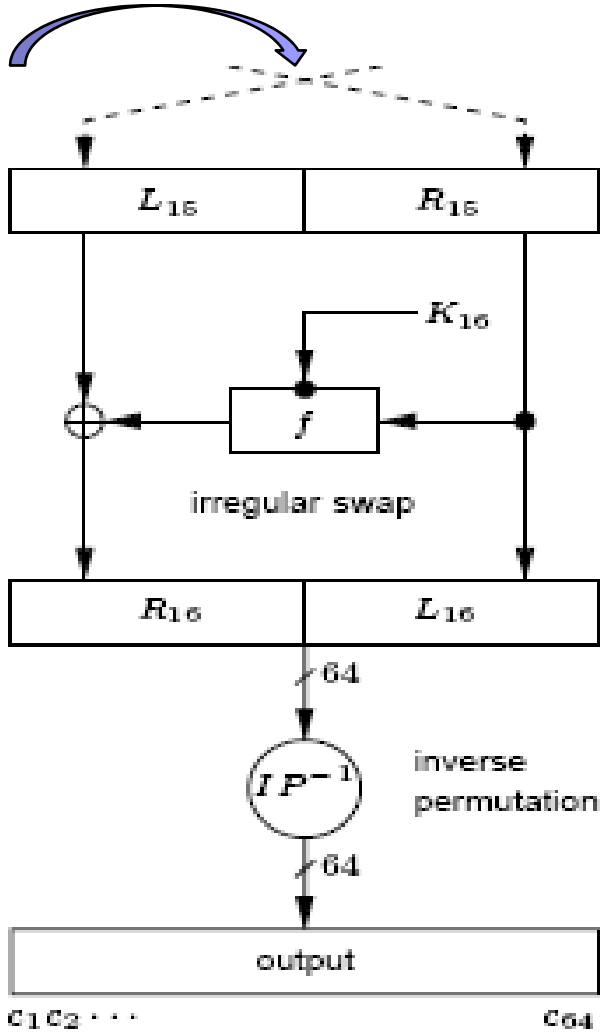


Figure 7.8: DES input-output.

Etapes du chiffrement DES





$$L_i = R_{i-1};$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i), \text{ where } f(R_{i-1}, K_i) = P(S(E(R_{i-1}) \oplus K_i))$$

DES: IP et IP^{-1} (inverse de IP)

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

IP^{-1}							
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

Table 7.2: DES initial permutation and inverse (IP and IP^{-1}).

Etapes du chiffrement DES

- E: Expansion
  - Permutation avec expansion: entrée 32bits ➔ sortie 48 bits
- S: substitution
  - 8 substitutions (S_1, S_2,S_3,S_4,S_5,S_6) 6to4 bits
- P: permutation fixe de 32 bits

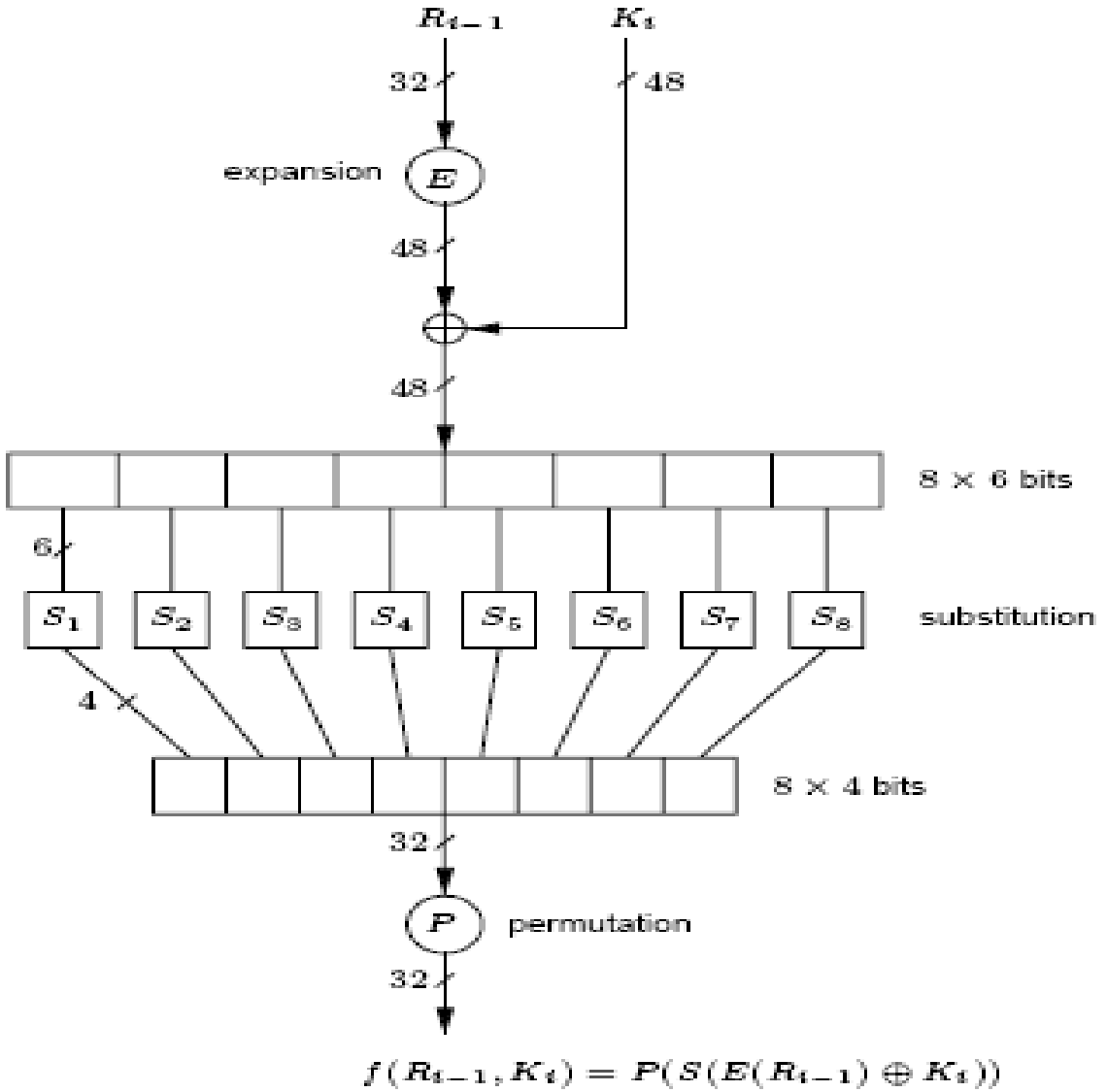


Figure 7.10: DES inner function f.

DES: Expansion (E), Permutation (P)

Expand $R_{i-1} = r_1 r_2 \ldots r_{32}$ from 32 to 48 bits using E per Table 7.3: $T \leftarrow E(R_{i-1})$. (Thus $T = r_{32} r_1 r_2 \ldots r_{32} r_1$.)

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

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

Table 7.3: DES per-round functions: expansion E and permutation P.

$$L_i = R_{i-1};$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i), \quad \text{where } f(R_{i-1}, K_i) = P(S(E(R_{i-1}) \oplus K_i))$$

DES: Substitutions (S)

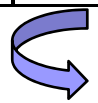
(c) $T' \leftarrow (S_1(B_1), S_2(B_2), \ldots S_8(B_8))$. (Here $S_i(B_i)$ maps $B_i = b_1 b_2 \ldots b_6$ to the 4-bit entry in row r and column c of S_i in Table 7.8, page 260 where $r = 2 \cdot b_1 + b_6$, and $b_2 b_3 b_4 b_5$ is the radix-2 representation of $0 \leq c \leq 15$. Thus $S_1(011011)$ yields $r = 1$, $c = 13$, and output 5, i.e., binary 0101.)

$$L_i = R_{i-1};$$

$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i), \quad \text{where } f(R_{i-1}, K_i) = P(S(E(R_{i-1}) \oplus K_i))$$

DES: S-boxes (substitutions)

row	column number															
	[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]
S_1																
[0]	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
[1]	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
[2]	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
[3]	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S_2																
[0]	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
[1]	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
[2]	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
[3]	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S_3																
[0]	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
[1]	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
[2]	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
[3]	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S_4																
[0]	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
[1]	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
[2]	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
[3]	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14



DES: S-boxes (substitutions)

	S_5															
[0]	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
[1]	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
[2]	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
[3]	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
	S_6															
[0]	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
[1]	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
[2]	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
[3]	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
	S_7															
[0]	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
[1]	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
[2]	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
[3]	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
	S_8															
[0]	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
[1]	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
[2]	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
[3]	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Table 7.8: *DES S-boxes.*

DES: calcul des sous clés

7.83 Algorithm DES key schedule

INPUT: 64-bit key $K = k_1 \ldots k_{64}$ (including 8 odd-parity bits).
OUTPUT: sixteen 48-bit keys K_i, $1 \leq i \leq 16$.

1. Define v_i, $1 \leq i \leq 16$ as follows: $v_i = 1$ for $i \in \{1, 2, 9, 16\}$; $v_i = 2$ otherwise. (These are left-shift values for 28-bit circular rotations below.)
2. $T \leftarrow \text{PC1}(K)$; represent T as 28-bit halves (C_0, D_0). (Use PC1 in Table 7.4 to select bits from K: $C_0 = k_{57}k_{49}\ldots k_{36}$, $D_0 = k_{63}k_{55}\ldots k_4$.)
3. For i from 1 to 16, compute K_i as follows: $C_i \leftarrow (C_{i-1} \hookleftarrow v_i)$, $D_i \leftarrow (D_{i-1} \hookleftarrow v_i)$, $K_i \leftarrow \text{PC2}(C_i, D_i)$. (Use PC2 in Table 7.4 to select 48 bits from the concatenation $b_1 b_2 \ldots b_{56}$ of C_i and D_i: $K_i = b_{14}b_{17}\ldots b_{32}$. '$\hookleftarrow$' denotes left circular shift.)

PC1						
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
above for C_i; below for D_i						
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

PC2					
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

Table 7.4: DES key schedule bit selections (PC1 and PC2).

DES: Algorithme

Algorithm Data Encryption Standard (DES)

INPUT: plaintext $m_1 \ldots m_{64}$; 64-bit key $K = k_1 \ldots k_{64}$ (includes 8 parity bits).
OUTPUT: 64-bit ciphertext block $C = c_1 \ldots c_{64}$. (For decryption, see Note 7.84.)

1. (key schedule) Compute sixteen 48-bit round keys K_i from K using Algorithm 7.83.
2. $(L_0, R_0) \leftarrow \mathrm{IP}(m_1 m_2 \ldots m_{64})$. (Use IP from Table 7.2 to permute bits; split the result into left and right 32-bit halves $L_0 = m_{58} m_{50} \ldots m_8$, $R_0 = m_{57} m_{49} \ldots m_7$.)
3. (16 rounds) for i from 1 to 16, compute L_i and R_i using Equations (7.4) and (7.5) above, computing $f(R_{i-1}, K_i) = P(S(E(R_{i-1}) \oplus K_i))$ as follows:
   (a) Expand $R_{i-1} = r_1 r_2 \ldots r_{32}$ from 32 to 48 bits using E per Table 7.3: $T \leftarrow E(R_{i-1})$. (Thus $T = r_{32} r_1 r_2 \ldots r_{32} r_1$.)
   (b) $T' \leftarrow T \oplus K_i$. Represent T' as eight 6-bit character strings: $(B_1, \ldots, B_8) = T'$.
   (c) $T'' \leftarrow (S_1(B_1), S_2(B_2), \ldots S_8(B_8))$. (Here $S_i(B_i)$ maps $B_i = b_1 b_2 \ldots b_6$ to the 4-bit entry in row r and column c of S_i in Table 7.8, page 260 where $r = 2 \cdot b_1 + b_6$, and $b_2 b_3 b_4 b_5$ is the radix-2 representation of $0 \leq c \leq 15$. Thus $S_1(011011)$ yields $r = 1$, $c = 13$, and output 5, i.e., binary 0101.)
   (d) $T''' \leftarrow P(T'')$. (Use P per Table 7.3 to permute the 32 bits of $T'' = t_1 t_2 \ldots t_{32}$, yielding $t_{16} t_7 \ldots t_{25}$.)
4. $b_1 b_2 \ldots b_{64} \leftarrow (R_{16}, L_{16})$. (Exchange final blocks L_{16}, R_{16}.)
5. $C \leftarrow \mathrm{IP}^{-1}(b_1 b_2 \ldots b_{64})$. (Transpose using IP^{-1} from Table 7.2; $C = b_{40} b_8 \ldots b_{25}$.)

DES: calcul des sous clés

- Exemple:
  - Pour mieux voir les permutations, considérons des chaines de caractères au lieu des bits. Les bits de parité sont représenté par des 0.

abcdefg0hijklmn0opqrstu0vwxyzAB0CDEFGHI0JKLMNOP0QRSTUVW0 XYZ12340

  - Après la permutation PC1, on obtient C_0 et D_0 suivants

C_0							D_0						
X	Q	J	C	v	o	h	4	W	P	I	B	u	n
a	Y	R	K	D	w	p	G	3	V	O	H	A	t
i	b	Z	S	L	E	x	m	f	2	U	N	G	z
q	j	e	1	T	M	F	s	l	e	y	r	k	d

DES: calcul des sous clés

- Après la permutation CP1, on obtient C_0 et D_0 suivants

C_0							D_0						
X	Q	J	C	v	o	h	4	W	P	I	B	u	n
a	Y	R	K	D	w	p	G	3	V	O	H	A	t
i	b	Z	S	L	E	x	m	f	2	U	N	G	z
q	j	e	1	T	M	F	s	l	e	y	r	k	d

- La première clé K1 =PC2(C_1 , D_1)= iSDlQoCXbhqKcEwvMYZaFxpJtyIVGdPAeUuz2sH4nrNml3Wb

i	S	D	l	Q	o
C	X	b	h	q	K
c	E	w	v	M	Y
Z	a	F	x	p	J
t	y	I	V	G	d
P	A	e	U	u	z
2	s	H	4	n	r
N	m	l	3	W	b

PC2					
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

DES: calcul des sous clés

- La première clé K1 =PC2(C_1 , D_1)=
  iSDlQoCXbhqKcEwvMYZaFxpJtyIVGdPAeUuz2sH4nrNml3Wb

➔ Les caractères R, L, j, T, g, O, f et k n'apparaissent pas dans K1

- La deuxième clés K2 sera:

K2 = bLwTJhvQZajD1xpoFRSYXqiCmrBOz4ItyNnsUIAWgkGfePu

➔ Les caractères absents dans K1 sont maintenant présents dans K2

Particularités du DES:

- Souplesse d'implémentation:
  - ECB ou CBC (en modifiant la phase de pré traitement des blocs de données)
  - Différentes implémentation en modifiants les fonction d'expansion ou de sélection

- Faiblesses
  - Conservation de la taille ➔ sensible aux attaques d'analyse de flux: on peut connaitre la taille exacte de chaque message
  - La clé est réduite à 56 bits ➔ réduit la sécurité de l'algorithme
  - Avec une clé de taille 128 bits ➔ algorithme couteux en temps
  - Peut être cassé par les processeurs actuels (exhaustive key search)
  - ➔ Triple DES (trois clés différentes)
  - ➔ AES : remplaçant du DES