Cryptographie

1-1 Algorithme de César

- On peut aussi attribuer à chaque lettre un numéro (A=0 …,Z=25) et le cryptogramme pourra être obtenu par ajout de k modulo 26 à chaque lettre du message.
- Ensemble des restes dans la division par n est $Z/nZ=\{0,1,\ldots,25\}$
- $(Z/nZ,+,x)$ est un anneau commutatif.

1-2 substitution et transposition

- Substitution: on substitue une lettre à une autre
- Transposition: la transposition consiste à décomposer le message en des blocs de longueur fixée d et ensuite à permuter les lettres de ce bloc. Il faut donc choisir un entier d et une permutation opérant sur d lettres.

1-2 substitution et transposition

- Exemple:

D=4 et la permutation

1 2 3 4

3 4 2 1

- Soit le message: NOUS ARRIVONS
- Découpage en blocs: NOUS ARRI VONS
- Substitution: SUNO IRAR SNVO
- Message Envoyé en préservant les espaces:
  - **SUNO IRARSNVO**

Algorithme de Vigenère

- Le message est divisé en blocs de longueur d
- d nombres entiers (entre 0 et 25) que l'on note k1, k2, …, kd
- Pour chaque bloc su message la i ème lettre

sera substituée selon l'algorithme de César avec comme clé ki.

- Exemple
- d = 3, (k1, k2, k3) = (7, 0, 12)
- Message à transmettre « **TO BE OR NOT TO BE »**
- **Découpage:** TOB EOR NOT TOB E
- **Cryptogramme:** « **AO NL OD UOF AO NL »**

Cryptanalyse de l'algorithme de César

- Essayer les 26 clés possibles
- Connaissant la langue du message, on peut utiliser un tableau des fréquences des lettres (cette technique a été introduite par Kasiski en 1863).

Formalisme Mathématique

$$\mathcal{M} \times \mathcal{K} \to \mathcal{C}$$
$$(M, K) \mapsto E(M, K).$$

- Pour k fixé

$$\mathcal{M} \to \mathcal{C}$$
$$M \mapsto E_K(M).$$

- Ek inversible $D_K \circ E_K(M) = M.$
- Algorithmes asymétriques à partir des années 70

Evaluation de la sécurité des algorithmes

- Types d'attaques sur un algorithme:
  - attaque « message chiffré connu » : E a réussi à capter un certain nombre de cryptogrammes (l'algorithme de César ne résiste pas à ce type d'attaque)
  - attaque « message clair connu » : E connaît des couples message et cryptogramme correspondant, mais n'a pas de contrôle sur ces données
  - attaque « message clair choisi » : connaît des couples messages et cryptogramme correspondant en ayant choisi

  les messages clairs ; cela revient, pour le cryptanalyste, à posséder la machine à chiffrer sous forme de « boîte noire » et à chercher à en déterminer le contenu (la clé).

Evaluation de la sécurité des algorithmes

- Types de sécurité
  - les algorithmes inconditionnellement sûrs ;
  - les algorithmes sûrs d'un point de vue informatique.
- Exemple du premier type d'algorithme :One time pad
  - Le message est une suite binaire m0 m1 … mN ;
  - Cryptogramme: c0 c1 … cN
  - $c_i = m_i \oplus k_i$ ($c_i = m_i + k_i$ modulo 2)
  - $D_k(c_i) = c_i + k_i \equiv m_i + k_i + k_i \equiv m_i$ modulo 2.
  - La clé est utilisée une fois

Algorithmes par bloc et en série

- Dans les algorithmes à clés secrètes nous distinguons les algorithmes en série (stream ciphers) et les algorithmes par blocs (block ciphers)
- Les premiers opèrent sur chaque unité d'information ; en général le bit, chiffre binaire ; les seconds opèrent sur des blocs plus longs, de taille fixée.
- Les algorithmes en série consistent à fabriquer, à partir d'une clé assez courte (quelques centaines de bits), une suite binaire
- « pseudo-aléatoire » de la longueur du message à chiffrer qui sera combinée par addition modulo 2 avec ce message.

DES

Algorithme à clé symétrique.

- (Data Encryption Standard) Conçu dans les années 70 IBM et destiné à être utilisé pour le chiffrement commercial.
- Opère sur des blocs de messages de 64 bits paramétré par une clé de 56 bits
- $\{0, 1\}^{64} \times \{0, 1\}^{56} \rightarrow \{0, 1\}^{64}$
- $Z /2Z^{64} \times Z/2Z^{56} \rightarrow Z /2Z^{64}$
  - $(M,K) \rightarrow C$

DES

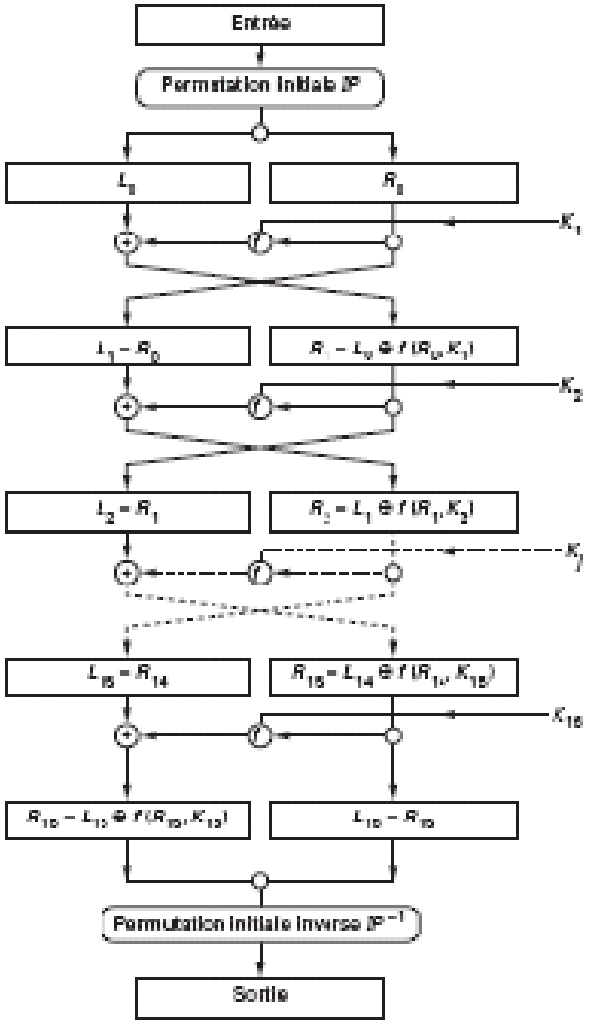
- DES_k $\mathbb{F}_2^{64} \to \mathbb{F}_2^{64}$

  $M \mapsto \mathrm{DES}(M, K)$.
- L'algorithme est la composition de 18 applications:
- *IP désigne une permutation des 64 bits d'entrée* et *IP – 1 son inverse* $\mathrm{DES}_K = (IP)^{-1} \circ \sigma_{16} \circ \ldots \circ \sigma_1 \circ (IP)$.
- Les notations Lj et Rj désignent des blocs de 32 bits
- à partir d'un bloc de 64 bits (b1, b2, … b64), Lj est le sous-bloc de gauche (b1, b2, … b32), tandis que Rj est le sous-bloc de droite (b33, b34, … b64).
- Les sous clés Kj (1 à 16) sont des blocs de 48 **sont fabriquées à partir** de la clé K

DES

- f est une application $\{0, 1\}^{32}$ x $\{0, 1\}^{48}$ → $\{0, 1\}^{32}$
- Le symbole $\oplus$ est le « ou exclusif ».

Schéma général DES



IP et IP^{-1}

Tableau 1 – Permutation initiale							
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

Tableau 2 – Inverse de la permutation initiale							
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

Construction des sous clés kj

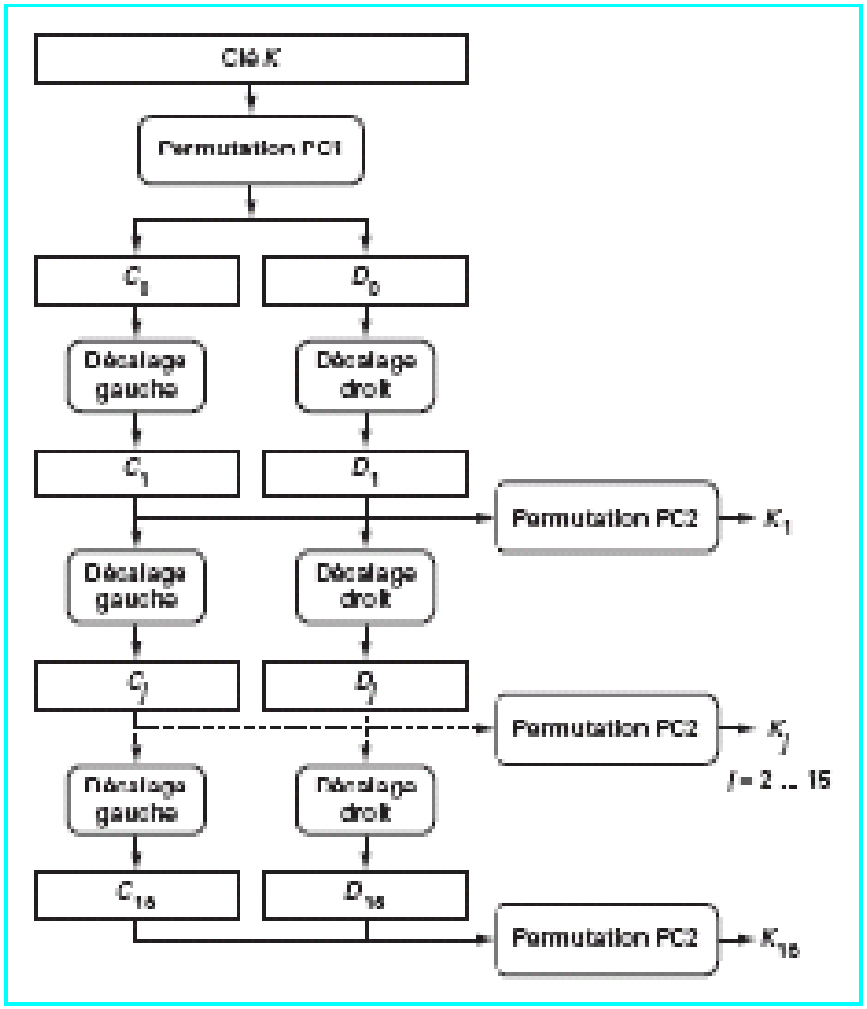
- Une clé DES possède 56 bits mais elle est notée sur 64 bits
- les 8 bits numérotés 8, 16, 24, 32, 40, 48, 56, 64 pouvant être pris comme des « bits
- de parité » (c'est-à-dire l'un est nul si les 7 bits précédents sont de somme paire et égal à 1 sinon) et n'intervenant pas dans les calculs qui suivent, comme le lecteur peut le vérifier.
- Comme le montre le schéma général du DES (figure **1), le clé K**
- intervient sous forme de ses sous-clés Kj, qui sont en fait des bits
- extraits de K de manière assez compliquée. La figure **2 donne le**
- schéma de **calcul de ces sous-clés Kj de 48 bits, chacune à partir**
- de la clé K.
- Sur cette figure, les symboles Cj et Dj (0 j 16) désignent des
- blocs de 28 bits.

Construction des sous clés kj

- On appelle décalage à gauche d'une position d'un tel bloc la
- permutation circulaire :
- (x1, x2, …, x28)→ (x2, …, x28, x1)
- Liste du nombre de décalages à gauche à effectuer :

  (1, 1, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 1) ;

Extraction des sous clés



PC1 et PC2

Tableau 3 – Choix permuté numéro 1						
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
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

Tableau 4 – Choix permuté numéro 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

Description de la j-ième étape

- σj : Cette application prend comme argument le couple
- (Lj – 1, Rj – 1) et produit le couple (Lj, Rj). On a :
- Lj = Rj – 1 et
- Rj = Lj – 1 + f (Rj – $_1$, Kj).
- L'application f va de {0, 1}32 x {0, 1}48 dans {0, 1}32.
- Remarque que, quelle que soit l'application f avec ces ensembles
- d'arrivée et de départ, ce schéma est inversible:
- Rj – 1 = Lj et
- Lj – 1 = Rj + f (Lj, Kj)
- Un tel schéma est appelé **schéma de Feistel .**

Calcul de f(R,kj)

- E désigne une application {0, 1}32 ® {0, 1}48 (« fonction d'expansion »)

- les 8 applications Si (1 i 8) : {0, 1}6 ® {0, 1}4 (sont souvent appelées boîtes S ; leurs propriétés sont cruciales pour la qualité cryptographique, mais rien n'a été dit par les concepteurs sur les raisons des choix faits
- l'application P est une permutation des 32 bits obtenus en sortie des 8 boîtes S.
- L'**application E « étend » le bloc R de 32 bits en un bloc de** 48 bits en répétant certains bits de l'argument suivant la table donnée tableau **5.**
- Les **boîtes S sont données par le tableau 6.**

Calcul de $f(R,K_j)$

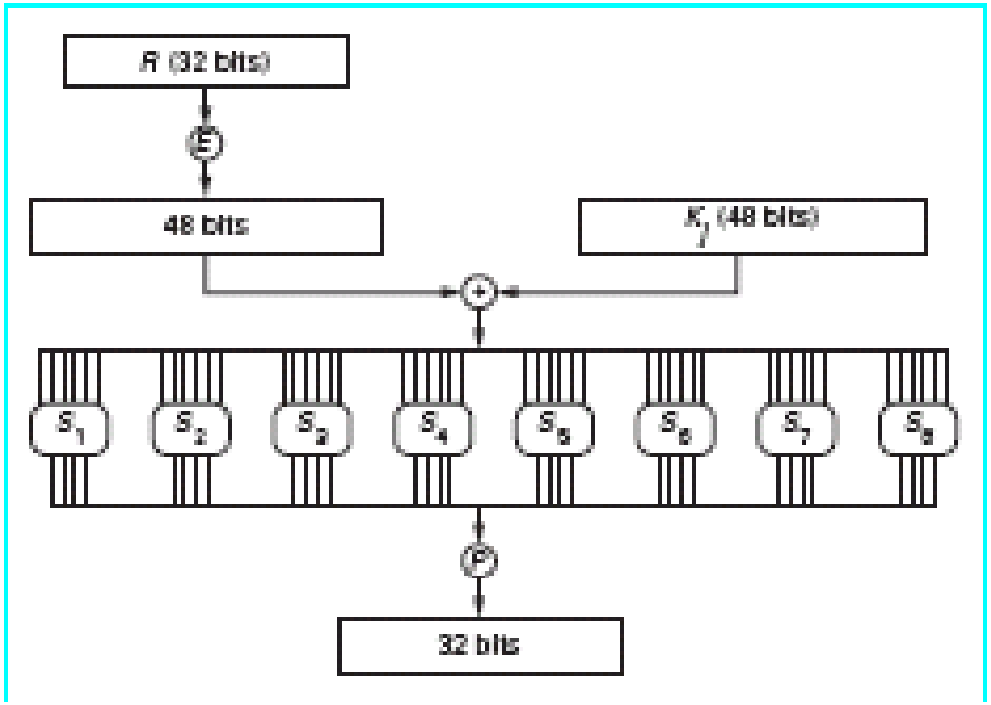


Tableau 5 – Table *E* de sélection de bits					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

Tableau 7 – Table de la permutation *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

Principe des boîtes S

- La boîte S1. prend, en entrée, un argument
  - B = b1 b2 b3 b4 b5 b6
- Le premier et le dernier représentent, en base 2, un entier
- entre 0 et 3 : b1 b6, qui donne un numéro de ligne.
- Sur cette ligne, on va à la colonne b2 b3 b4 b5 et on lit le nombre correspondant dont le développement binaire en 4 bits donne S1(B).
- Comme usuellement, l'argument de P est un bloc de 32 bits numérotés de gauche à droite de 1 à 32 l'application P les permute

Boîte S

Tableau 6 – Boites *S*

		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
	4	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
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

Déchiffrement de l'algorithme DES

- Le déchiffrement provient du caractère réversible du schéma de Feistel , Supposons que le clé K a été choisie et notons respectivement EK et DK les fonctions de chiffrement et de déchiffrement correspondantes.
- Supposons que le message est le bloc de 64 bits M et le cryptogramme reçu par le destinataire C avec, évidemment, les relations:
- C = EK (M) et M = DK (C).
- En recevant C et connaissant K, il s'agit, pour le destinataire, de calculer DK (C).
- On lit les opérations à effectuer sur le schéma général du DES (figure **1), mais en faisant les opérations à l'envers. On commence**
- donc par faire subir aux bits de C la permutation initiale *I P pour*
- inverser la dernière étape du chiffrement. On obtient ainsi le bloc de
- 64 bits R16 L16 (dans cet ordre), R16 et L16 désignant comme auparavant
- des « demi-blocs » de 32 bits ; ce sont ces deux blocs avec les
- mêmes noms qui sont apparus dans le chiffrement de M avant le
- passage dans l'inverse de la permutation initiale. À partir de là, on
- obtient le bloc L15 R15 en, inversant l'opération de chiffrement par la
- règle
- R15 = L16
- et
- L15 = R16 Å f (R15, K16).
- Un procédé identique permet de remonter à L14 R14 …, puis, finalement,
- à L0 R0 et à M en inversant la permutation initiale.
- Cet algorithme de déchiffrement pourrait donner lieu à un tableau
- comme celui explicitant le chiffrement ; la seule différence serait
- l'ordre d'intervention des sous-clés : dans le déchiffrement, on fait
- d'abord intervenir K16 puis K15 … et, enfin, K1.

Modes d’utilisation

- **mode dictionnaire ou mode ECB (de l’anglais « electronic codebook »).**
- **Le message se** présente comme une suite de blocs de 64 bits : M1, M2, M3
- Cn = EK (Mn).
- Dans certains contextes, ce mode d’utilisation peut présenter des dangers ; il est fréquent que, dans des applications, les mêmes blocs de message se répètent au cours d’un même échange ou au cours de communications différentes. Cela se traduit par des répétitions au niveau des blocs de cryptogramme et peut être utilisé par un observateur (ennemi). Cette propriété fait que, dans la réalité, ce mode d’utilisation n’est quasiment jamais pratiqué.
- **chaînage de blocs chiffrés » CBC (de l’anglais « Cipher**block chaining »).
- Le premier bloc de message clair de 64 bits M1 passe dans la machine à chiffrer EK et il en sort C1. Ensuite, on fait un « ou exclusif » entre le bloc de message clair suivant M2 et le cryptogramme C1 avant de le mettre dans la machine ; ainsi on a:
- C2 = EK (M2 Å C2)
- et on continue l’opération obtenant :
- Cn+1 = EK (Mn+1 Å C1) pour n > 0.
- À la réception de C1 le destinataire calcule
- M1 = DK (C1)
- puis
- M2 = DK (C2) Å C1

Modes d'utilisation

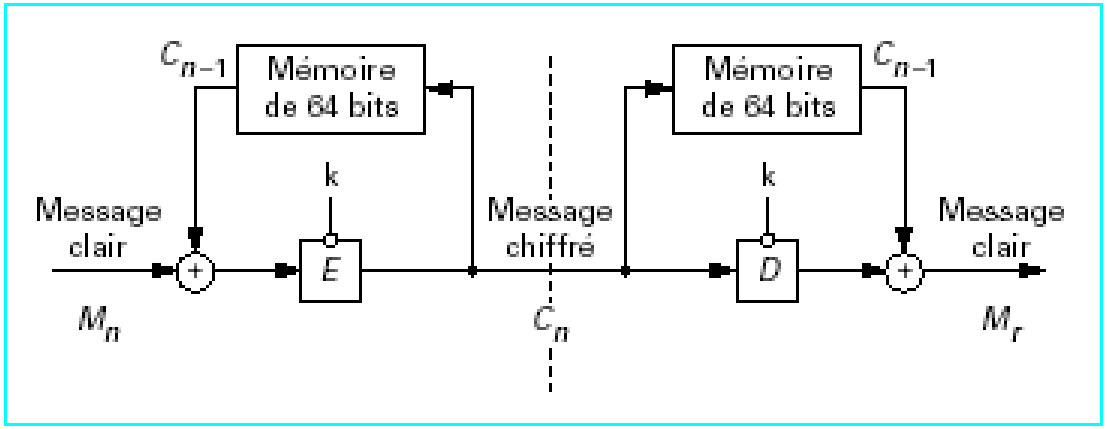


Figure 4 – Mode CBC (chaînage de blocs chiffrés) du DES