الجمهورية الجزائرية الديمقراطية الشعبية
République algérienne démocratique et populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique
جامعة عين تموشنت بلحاج بوشعيب
Université–Ain Temouchent- Belhadj Bouchaib
Faculté des Sciences et de la Technologie
Département : d'électronique et des Télécommunications

Projet de Fin d'Études
Pour l'obtention du diplôme de Master en :
Domaine : SCIENCES & TECHNOLOGIES
Filière : TELECOMMUNICATION
Spécialité : RESEAU ET TELECOMMUNICATION
Thème

Analyse des performances des méthodes cryptographiques à clé secrète dans le cryptage des images

Présenté Par :
1) Melle CHERIF Meriem
2) Melle BOUOKKA Hassiba

Devant le jury composé de :

Dr BENGHENIA Hadj Abdelkader	MCB	UAT.B.B	(Ain Temouchent)	Président
Dr HAMLILI Heyem	MAB	UAT.B.B	(Ain Temouchent)	Examinateur
Dr BOUKHOBZA Abdelkader	MCA	UAT.B.B	(Ain Temouchent)	Encadrant

Année Universitaire 2023/2024

DEDICACES

À mes très chers parents, pour leur amour inconditionnel, leur soutien indéfectible et tous les sacrifices consentis pour me permettre d'avancer dans la vie.

À mes frères BOUHADJAR et ABDELMOUDJIB pour leur présence réconfortante et leurs encouragements perpétuels.

À mes proches amis, qui ont su égayer mon quotidien et me donner la force d'aller de l'avant.

À tous mes enseignants, qui ont consacré leur énergie à me transmettre leurs savoirs avec passion et dévouement.

Que cette réalisation soit le témoignage de ma profonde gratitude envers chacun d'entre vous.

CHERIF Meriem

DEDICACES

À mes très chers parents, pour leur amour inconditionnel, leur soutien indéfectible et tous les sacrifices consentis pour me permettre d'avancer dans la vie.

À mes frères et sœurs pour leur présence réconfortante et leurs encouragements perpétuels.

À mes proches amis, qui ont su égayer mon quotidien et me donner la force d'aller de l'avant.

À tous mes enseignants, qui ont consacré leur énergie à me transmettre leurs savoirs avec passion et dévouement.

Que cette réalisation soit le témoignage de ma profonde gratitude envers chacun d'entre vous.

BOUOKKA Hassiba

Remerciements

Avant toute chose, nous adressons nos plus sincères remerciements au Tout-Puissant qui nous a donné la chance de suivre le chemin de la science.

A l'issue de ce modeste travail, nous tenons à exprimer notre profonde gratitude à Notre encadreur, Monsieur BOUKHOBZA Abdelkader, pour ses précieux conseils, son suivi rigoureux et ses orientations avisées qui nous ont guidés tout au long de cette étude.

L'ensemble des enseignants du département de la Science et Technologie, filière Télécommunications, pour leur dévouement et les connaissances inestimables qu'ils nous ont transmises durant notre formation.

Nous adressons également nos sincères remerciements aux membres respectés du jury qui nous font l'honneur d'évaluer ce modeste travail.

Que toutes les personnes ayant contribué de près ou de loin à la réalisation de cette étude trouvent ici l'expression de notre profonde reconnaissance.

Résumé

Ce mémoire se concentre sur la cryptographie, qui est essentielle pour sécuriser les communications, les transactions et les données dans le monde numérique moderne. Il examine principalement trois algorithmes de cryptographie : DES, AES et RC4. L'étude évalue leurs performances et leur efficacité en chiffrement des images numériques. Les résultats montrent que l'algorithme AES surpasse largement DES en termes de sécurité et d'efficacité, même avec un nombre de rondes inférieur. RC4, bien que performant en termes de temps d'exécution, présente une entropie et des valeurs NPCR et UACI légèrement meilleures comparées à l'AES. Les algorithmes hybrides AES-RC4 et RC4-AES sont également explorés pour combiner les avantages des deux méthodes.

Mots clé : Cryptographie, DES**,** AES**,** RC4**,** Chiffrement**,** Algorithmes hybrides**,** cryptage d'image.

Abstract

This thesis focuses on cryptography, which is essential for securing communications, transactions, and data in the modern digital world. It primarily examines three cryptographic algorithms: DES, AES, and RC4. The study evaluates their performance and efficiency in encrypting digital images. The results show that the AES algorithm significantly outperforms DES in terms of security and efficiency, even with fewer rounds. RC4, while efficient in execution time, shows slightly better entropy and NPCR and UACI values compared to AES. Hybrid algorithms AES-RC4 and RC4-AES are also explored to combine the advantages of both methods.

Keywords: Cryptography, DES, AES, RC4, Encryption, Hybrid algorithms, Image Encryption.

ملخص

تركز هذه الأطروحة على التشفير، وهو أمر ضروري لتأمين الاتصالات والمعاملات والبيانات في العالم الرقمي الحديث. وينظر بشكل أساسي إلى ثلاث خوارزميات تشفير: وتقوم الدراسة بتقييم أدائها وفعاليتها في تشفير الصور الرقمية DES، وAES،RC4 .أظهرت النتائج أن خوارزمية AES تتفوق بشكل كبير على DES من حيث الأمان والكفاءة، حتى مع عدد أقل من الجولات. RC4، على الرغم من فعاليته من حيث وقت التنفيذ، إلا أنه يتمتع بقيم ENTROPIE وNPCR وUACI أفضل قليلاً مقارنة بـ AES.يتم أيضًا استكشاف خوارزميات Hybrid AES-RC4 وRC4-AES للجمع بين مزايا كلا الطريقتين.

الكلمات المفتاحية :علم التشفير , DES، AES، RC4 ، التشفير، الخوارزميات الهجينة، تشفير الصور

Table des matières

Chapitre 1 : Généralité sur la cryptographie

Chapitre 2: Les algorithmes de cryptographie à clé secrète

Chapitre3 : Résultats et discussion

Liste des figures

Chapitre 1 : Généralité sur la cryptographie

Chapitre 2: Les algorithmes de cryptographie à clé secrète

Chapitre3 : Résultats et discussion

Liste des tableaux

Liste des Abréviations

AES : Advanced Encryption Standard

CBC : Cipher Block Chaining

CFB : Cipher Feedback Block

DES : Data Encryption Standard

ECB : Electronic Codebook Block

FIPS : Fédéral Information Processing Standard

GF : Galois Field

IDEA : International Data Encryption Algorithm

KSA: Key SchedulingAlgorithm

NPCR : Number of Pixels Change Rate

OFB : Output Feedback (**OFB**) mode

PRGA: Pseudo-Random Génération Algorithm

RC4: Rivest Cipher 4

RVB : Rouge, Vert, Bleu

SPN : Permutation Substitution Network

UACI : Unified Average Changing Intensity

INTRODUCTION GÉNÉRALE

La cryptographie est l'étude des techniques permettant de sécuriser les communications et l'information. Elle joue un rôle essentiel dans notre monde numérique actuel en assurant la confidentialité, l'intégrité et l'authentification des données. Ses origines remontent à l'Antiquité avec les premiers codes secrets utilisés dans un contexte militaire. Au fil des siècles, la cryptographie s'est complexifiée, des méthodes manuelles aux algorithmes mathématiques complexes. Aujourd'hui, elle sous-tend de nombreux aspects de notre vie quotidienne comme les transactions bancaires, les communications chiffrées ou encore l'authentification en ligne. Bien que souvent invisible, la cryptographie constitue une composante fondamentale de la sécurité numérique en protégeant nos données sensibles des menaces internes et externes.

Dans l'ère numérique actuelle, la sécurité des données visuelles est devenue une préoccupation majeure. Avec la prolifération des images numériques dans divers domaines tels que la médecine, la défense, les réseaux sociaux et le commerce électronique, la protection de ces données contre l'accès non autorisé, la manipulation et le vol est primordiale. Le chiffrement d'images, une branche spécialisée de la cryptographie, répond à ce besoin en transformant les images en formes inintelligibles pour quiconque ne possède pas la clé de déchiffrement.

Cette étude se concentre sur l'évaluation et la comparaison des performances de quelques algorithmes de cryptographie à clé privée (ou symétrique) appliqués au chiffrement d'images. Nous examinons trois algorithmes classiques bien établis : le Data Encryption Standard (DES), l'Advanced Encryption Standard (AES), et le RC4 (RivestCipher 4). DES, introduit en 1977, a longtemps été la norme, mais sa clé de 56 bits est considérée comme faible aujourd'hui. AES, adopté en 2001, offre une sécurité renforcée avec des clés de 128, 192 ou 256 bits. RC4, conçu en 1987, est un algorithme de chiffrement de flux rapide, largement utilisé malgré certaines vulnérabilités connues.

Nous examinons l'impact du nombre de rondes dans AES et DES sur la sécurité des images, cherchant un équilibre optimal entre la force cryptographique et l'efficacité computationnelle. Cette étude vise non seulement à comparer ces algorithmes bien établis, mais aussi à explorer de nouvelles stratégies hybrides qui pourraient offrir une sécurité renforcée sans compromettre significativement les performances.

Inspirée par le concept de chiffrement hybride, qui combine les forces de différents algorithmes pour renforcer la sécurité, nous explorons également deux combinaisons : AES suivi de RC4, et l'inverse, RC4 suivi d'AES. Cette approche vise à exploiter la forte diffusion d'AES et l'excellente confusion de RC4, tout en réduisant le nombre de rondes d'AES pour améliorer la rapidité de chiffrement.

Ce mémoire est structuré en trois chapitres :

- Dans le premier chapitre, nous présentons les concepts fondamentaux et la terminologie de la cryptographie dans un premier temps, puis nous exposons ces différents types. Enfin, nous décrivons les méthodes de cryptage d'images et nous rappelons également la

définition de l'image numérique et ces différentes classes.

- Dans le deuxième chapitre nous allons focaliser sur les méthodes de cryptographie à clé secrète les plus populaires : DES, AES, RC4,….
- Le troisième chapitre nous avons commencé par la définition des critères employés pour l'évaluation des méthodes de chiffrement d'images, puis nous présentons et nous discutons les résultats de cryptage d'images.

CHAPITRE 1

Généralités sur la cryptographie

1.1 Introduction

La cryptographie est la science qui utilise les mathématiques pour crypter et décrypter des données. Il vous permet de stocker ou de transmettre des informations confidentielles sur un réseau non sécurisé (comme Internet) afin qu'elles ne puissent être lues par personne autre que le destinataire. La cryptographie consiste à protéger les données, tandis que la cryptanalyse consiste à étudier les informations cryptées pour découvrir leurs secrets. La cryptanalyse classique implique une combinaison intéressante de raisonnement analytique, d'application d'outils mathématiques, de découverte de modèles, de patience, de détermination et de chance. Ces cryptanalyses sont également connus sous le nom de hackers. La cryptographie comprend la cryptographie et la cryptanalyse, dans ce chapitre, nous introduisons les concepts de base de la cryptographie [1].

1.2 Historique de la Cryptographie

Une fois que les gens ont appris à communiquer, ils ont dû trouver des moyens de garantir la confidentialité de leurs communications. Il ne fait aucun doute que c'est là l'origine de la cryptographie. Environ 600 avant JC Au 4ème siècle avant JC, le roi de Babylone (Nabuchodonosor) écrivit des messages sur 4 444 crânes de ses esclaves. Il l'a renvoyée en attendant que ses cheveux repoussent. Lorsqu'il a lu le message, tout ce qu'il savait, c'était comment la raser à nouveau pour son esclave. Dans les temps anciens, 4 444 appareils appelés Syctale cryptaient les communications grecques .L'appareil était une fine bande de parchemin enroulée en spirale autour d'un cylindre en bois et ensuite écrite dessus. Pour lire le message, il fallait enrouler la bande autour d'un cylindre de même diamètre.

50 avant JC Au 4ème siècle avant JC, Jules César utilisa la substitution de l'alphabet pour cacher des messages. Chaque lettre a été remplacée par lettres alphabétiques différentes décalées d'un montant fixe. Sa faiblesse réside dans le grand nombre de façons de coder un message, mais l'alphabétisation d'une petite population le rend efficace. On assiste à des développements plus ou moins ingénieux de techniques cryptographiques expérimentales dont la sécurité repose fondamentalement sur la confiance des utilisateurs.

Plus tard, en 1467, Leone Battista propose la méthode de substitution poly-alphabétique c'est à dire de remplacer chaque lettre d'un autre alphabet. Vers les années 1500, une procédure de remplacer une lettre par un groupe de mots a été proposée par l'abbé Jean Tritheme.

L'inconvénient de la substitution alphabétique est la fréquence de chaque caractère. Cela signifie que même si la substitution alphabétique de Vigénere a été améliorée en 1586 en utilisant des clés littérales, il est toujours possible de déchiffrer le message grâce à une attaque statistique. Cependant, l'impossibilité de protéger la clé reste un inconvénient majeur et peut conduire à un décodage des messages.

En 1918, l'Allemand Arthur Scherbius a donné naissance à la fameuse machine « Enigma ». Le principe fut que chaque lettre est remplacée par une autre lettre, la règle de substitution avec cette machine est changée à chaque lettre. Cette procédure nous permet d'évincer le problème de fréquence ainsi le problème de Vigénère. Le développement de l'électronique ainsi que

l'apparition des ordinateurs puissants et le développement des techniques de communications, ont fait que la sécurité de l'information devienne un nouveau problème non seulement pour la confidentialité mais pour préserver le contenu des messages est assurer l'identité de l'émetteur et du récepteur.

En 1970, Horst Feistel de compagnie IBM, propose un projet de recherche qui consiste à trouver de nouvelles méthodes de chiffrement. Des efforts de travail ont conduit à l'élaboration du DES (Data Encryption Standard). En 1976, Whit Field et Martin Helmman proposent la cryptographie à clé publique. En 1978 les trois mathématiciens américaine Rivest, Shamir, et Adleman propose le système de chiffrement à clé publique RSA ce qui a mené à l'explosion des applications civiles de chiffrement. Ces deux derniers algorithmes de chiffrement à clé publique et à clé secrète font révolutionner le monde de la cryptographie à nos jours [2].

1.3 Définition et terminologie

La cryptographie est l'étude des techniques visant à sécuriser les communications et à protéger l'information contre l'accès non autorisé ou la modification. Elle repose sur l'utilisation de méthodes mathématiques et algorithmiques pour chiffrer (crypter) et déchiffrer (décrypter) des données assurant ainsi:

Confidentialité : Le texte chiffré ne doit être lisible que par les destinataires légitimes. Il ne doit pas pouvoir être lu par un intrus.

Intégrité : Seuls les utilisateurs autorisés peuvent modifier l'information. D'où la nécessité de vérifier si le message n'a pas subit de modifications durant la communication

Authentification : Il s'agit d'une propriété qui valide l'identité d'une entité avant d'autoriser l'accès à une ressource. Les entreprises doivent prouver leur identité. Tous les mécanismes de contrôle d'accès logique aux ressources informatiques nécessitent une gestion des identités et des authentifications.

Non-répudiation: C'est le fait de ne pas pouvoir nier qu'un évènement (action, transaction) a eu lieu. Elle contient :

a. **Non-répudiation de l'origine** : L'émetteur ne peut nier avoir écrit le message, et si tel est le cas, il peut prouver qu'il ne l'a pas écrit.
b. **Non-répudiation de réception :** Le receveur ne peuvent refuser de recevoir un message et peuvent prouver qu'ils ne réutilisent pas le message s'ils le réutilisent effectivement.

Dans ce domaine, on trouve généralement les termes suivant :

Texte brut : Informations qu'Alice souhaite transmettre.

Texte chiffré : Résultat du chiffrement.

Cryptage : Il s'agit d'un algorithme utilisé pour rendre le texte brut inintelligible.

Décryptage : Il s'agit de la méthode ou de l'algorithme utilisé pour reconstruire le texte en clair.

Clé : Il s'agit du secret partagé utilisé pour le cryptage et le déchiffrement.

Cryptage : Il s'agit de l'étude des méthodes permettant de transmettre des données de manière confidentielle sur un support particulier.

Cryptanalyse : L'objectif est de retrouver du texte clair à partir du texte chiffré en identifiant les erreurs dans les algorithmes utilisés.

Algorithmes cryptographiques : Fonctions mathématiques utilisées dans le processus de cryptage et de déchiffrement.

Cryptologie : Science mathématique comportant deux branches : la cryptographie et la cryptanalyse.

Encodage, Décodage : est une méthode ou un algorithme qui permet de changer le format d'un message sans introduire d'éléments secrets [3].

1.4 Mécanisme de cryptographie

Un système cryptographique ou crypto système est une description du processus de cryptage/déchiffrement. Il s'agit d'envoyer un message que seul le destinataire peut comprendre. Pour ce faire, indiquez votre secret à l'expéditeur du message. Convertit un message «texte clair» en un message « texte chiffré » codé à l'aide d'une « fonction de chiffrement » paramétrée par une « clé », et le message codé est paramétré par une fonction « clé » [4].

Chiffrement

Le cryptage est le processus de transformation d'un message M en un autre message non compréhensible. Il s'appuie sur une fonction de chiffrement E et une clé de chiffrement k.

$$C = E_k(M)$$

Déchiffrement

Le déchiffrement est le processus de reconstruction d'un message en clair à partir d'un message chiffré.Il s'appuie sur une fonction de décryptage D et une clé de décryptage k'.

$$M = D_{K'}(C)$$

1.5 Les applications de la cryptographie

La cryptographie, ou l'étude des techniques de communication sécurisée, a de nombreuses applications pratiques dans divers domaines. Voici quelques-unes des applications les plus importantes de la cryptographie :

a) **Sécurité des communications** : La cryptographie est utilisée pour sécuriser les communications à travers le chiffrement et la signature numérique. Cela garantit la confidentialité, l'intégrité et l'authenticité des données échangées sur des réseaux informatiques.
b) **Protection des données personnelles** : Les données personnelles stockées dans les bases de données, sur les appareils mobiles ou dans le cloud sont souvent chiffrées pour éviter tout accès non autorisé.

c) **Identité numérique** : La cryptographie est essentielle pour établir et vérifier l'identité numérique des utilisateurs. Les certificats numériques et les infrastructures à clés publiques (PKI) sont utilisés pour garantir l'authenticité des utilisateurs sur Internet.
d) **Sécurité des systèmes informatiques** : Les systèmes d'exploitation et les logiciels utilisent la cryptographie pour sécuriser les mots de passe, les fichiers sensibles et les communications internes.
e) **Sécurité des objets connectés (IoT)** : Avec la prolifération des appareils IoT, la cryptographie est utilisée pour sécuriser les données échangées entre les appareils connectés et pour protéger l'intégrité des commandes et des mises à jour logicielles.
f) **Télécommunications sécurisées** : Les réseaux de télécommunications utilisent la cryptographie pour protéger les appels vocaux, les messages SMS et les communications par courriel.
g) **Applications militaires et gouvernementales** : La cryptographie est cruciale pour les communications militaires et gouvernementales, où la sécurité et la confidentialité des informations sont d'une importance capitale.
h) **Blockchain et cryptomonnaies** : La technologie de la block chain repose sur des principes cryptographiques pour sécuriser les transactions et garantir l'immuabilité des registres distribués utilisés dans les cryptomonnaies comme Bitcoin et Ethereum.
i) **Protection de la propriété intellectuelle** : La cryptographie est utilisée pour protéger les droits d'auteur et les propriétés intellectuelles en permettant l'authentification et la signature numérique des créations.

En résumé, la cryptographie joue un rôle essentiel dans la sécurisation des communications, des transactions et des données dans le monde numérique moderne, contribuant ainsi à assurer la confidentialité, l'intégrité et l'authenticité des informations sensible.

1.6 Classification de la cryptograhie

Les systèmes cryptographiques peuvent être divisés en deux catégories.

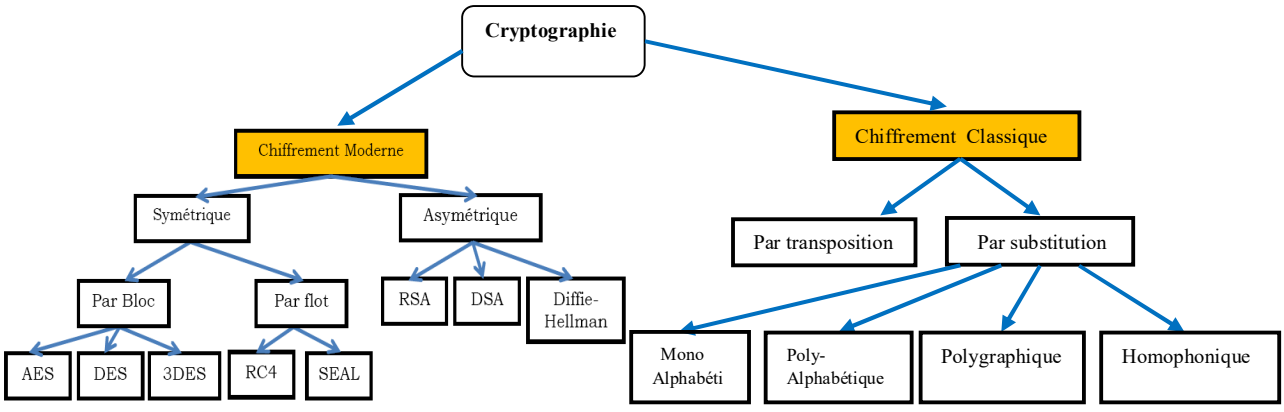


Figure 1.1 : Classification de la cryptographie [4]

1.7 La cryptographie Classique

Dans le cryptage classique, les méthodes et les clés de cryptage et de déchiffrement sont connues de l'émetteur et du destinataire.

La plupart des méthodes de décodage classiques reposent sur deux principes principaux: la substitution et la transposition.

Le chiffrement par substitution remplace une ou plusieurs entités (généralement des caractères) dans un message par une ou plusieurs autres entités :

- **La substitution mono alphabétique** : Il s'agit de remplacer chaque caractère d'un message par un autre caractère ou symbole. Certains des diagrammes les plus célèbres incluent le chiffre de César et le chiffre affines.

- **La substitution polygraphique**: Remplacement d'un ensemble de n caractères dans un message par un autre ensemble de n caractères ou symboles. Ce type de cryptage est également appelé substitution poly grammatique.

- **La substitution poly alphabétique** : Il consiste à utiliser une série de caractères alphanumériques régulièrement réutilisés. Je cite notamment le chiffre de Vigenère, qui utilise une clé qui définit le décalage de chaque caractère du message.
- **La substitution homophonique** : permet de faire correspondre à chaque lettre du message en clair un ensemble possible d'autres caractères.

Le chiffre de césar

Cet algorithme est l'un des plus anciens et fut utilisé par Jules César [5] [6]. Le codage consiste à ajouter une valeur constante à chaque caractère (plus précisément, le code ASCII) d'un message. Il s'agit de substitutions mono-alphabétique.

Chaque lettre peut être remplacée par une seule lettre différente (un « alphabet unique »), soit selon la modification de l'alphabet, soit à volonté (une « substitution »). Selon Suétone, César a utilisé des transformations à trois lettres, telles que A devenant D, B devenant E et C devenant F. Il composa donc le message comme d'habitude, en remplaçant chaque lettre par la lettre correspondante. Si l'ajout de valeurs entraîne plus de caractères que Z, continuez avec A. C'est la même chose que l'opération Modulo26.

CLAIR	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
-> décalage = 3																										
CODE	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Figure 1. 2 : Tableau du principe de César [7] [8]

Le caractère qui correspond à la valeur que vous ajoutez à un message pour effectuer le chiffrement est appelé clé. Dans ce cas, la clé est C, la troisième lettre de l'alphabet. Bien que ce système cryptographique soit certes simple à mettre en œuvre, il présente l'inconvénient d'être totalement symétrique. Autrement dit, il suffit d'effectuer la soustraction pour connaître le message original. La méthode principale consiste simplement à soustraire les nombres 1 à 26 et à voir si l'un de ces nombres fournit un message compréhensible.

1.8 La cryptographie Moderne

Les chiffrements modernes sont divisés en deux types les chiffrements symétriques et les chiffrements asymétriques.

1.8.1 Cryptographie symétrique (à clefs privés)

Le chiffrement par clé privée est également appelé chiffrement symétrique [7][8].Le chiffrement symétrique (également appelé clé privée ou chiffrement à clé privée) utilise la même clé pour le chiffrement et le déchiffrement. Le chiffrement consiste à effectuer des opérations entre une clé privée et les données à chiffrer pour rendre les données illisibles. Le décryptage consiste à effectuer le processus inverse .Récupérez le message original du message crypté à l'aide de la clé privée.

Figure 1. 3: Chiffrement symétrique [8]

Les principaux types de cryptographie à clé privée symétrique utilisés aujourd'hui se répartissent en deux grandes catégories de cryptographie moderne : les chiffrements par blocs et les chiffrements par flux.

1.8.1.1 Chiffrement par blocs

Le chiffrement par blocs est une technique de chiffrement utilisée en cryptographie où le texte en clair est divisé en blocs de taille fixe avant d'être chiffré. Chaque bloc est traité indépendamment des autres blocs, et la même clé de chiffrement est généralement utilisée pour chiffrer tous les blocs.

Une fonction cryptographique est typiquement une itération d'une fonction ronde g. Les paramètres d'un chiffrement par bloc itératif sont la taille du bloc n, le nombre de tours r et la clé K. Le chiffre est obtenu en appliquant de manière répétée une fonction ronde g à la même manière en fonction de la clé ronde. La clé ronde est dérivée de la clé K.

Donc pour un chiffrement par blocs itérés à r rondes :

C_i=**g**(C_{i-1}, K_i) pour i=1,…,r

Avec

C_0 : le clair,

g : la fonction de ronde,

Ki : les clés de ronde

Cr : le chiffré

Le déchiffrement se déroule suivant le processus inverse ⇒ g doit être inversib

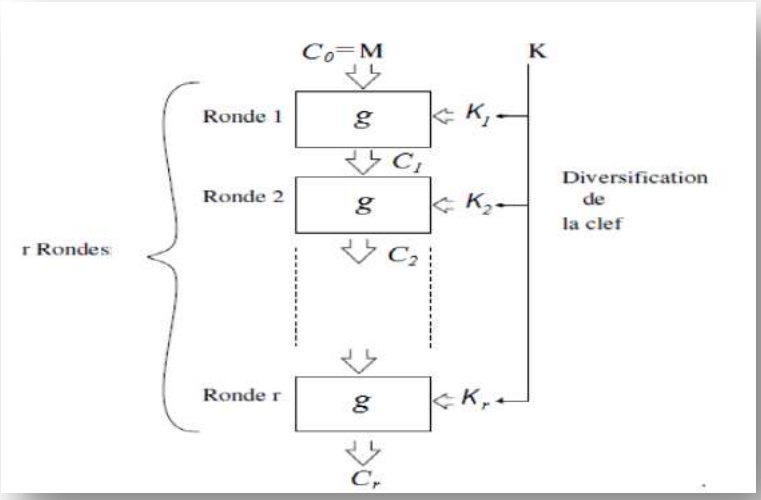


Figure 1. 4 : Principe du chiffrement par blocs itérés [2]

Il existe deux variantes de chiffrement par blocs :

- SPN.
- Feistel.

a. Réseau SPN (Permutation Substitution Network) :

Les réseaux SPN sont constitués d'une succession de :

- Substitution,
- Permutation,
- XOR bit à bit avec la clé.

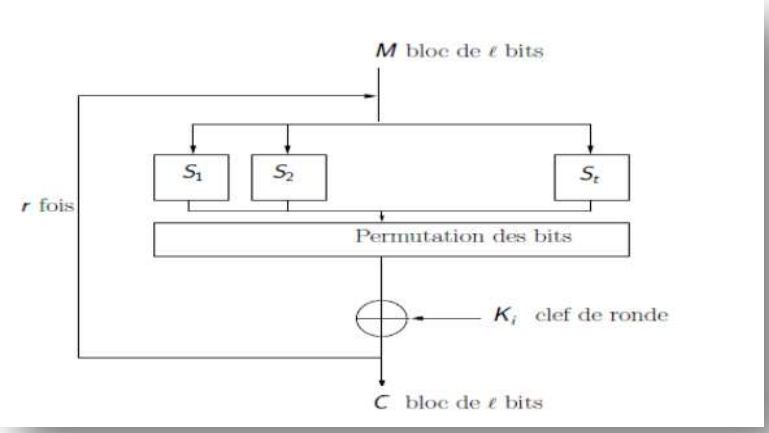


Figure 1.5: Réseau SPN [2]

b. Chiffrement de Feistel :

C'est effectivement la base de certains algorithmes modernes à clé privée (notamment DES) et a été proposé par Horst Feistel (IBM) en 1973.Il s'agit d'un chiffrement par blocs itératif qui opère sur des blocs de 2n bits [8].

Divisez le bloc de texte brut en deux parties, appliquez une transformation ronde (fonction f) à la partie droite et combinez le résultat avec la partie gauche par OU exclusif. Les deux moitiés sont ensuite retournées pour être utilisées au tour suivant. Le décryptage est structurellement identique au cryptage.

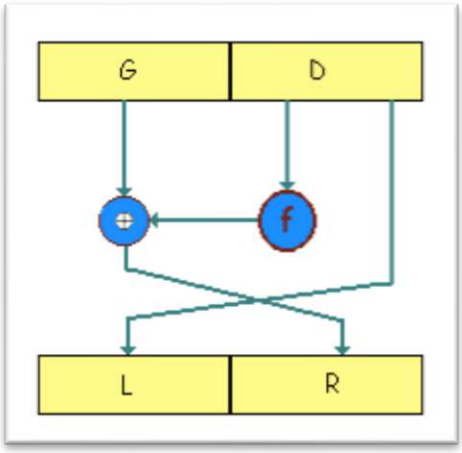


Figure 1.6 : Réseau de Feistel [7]

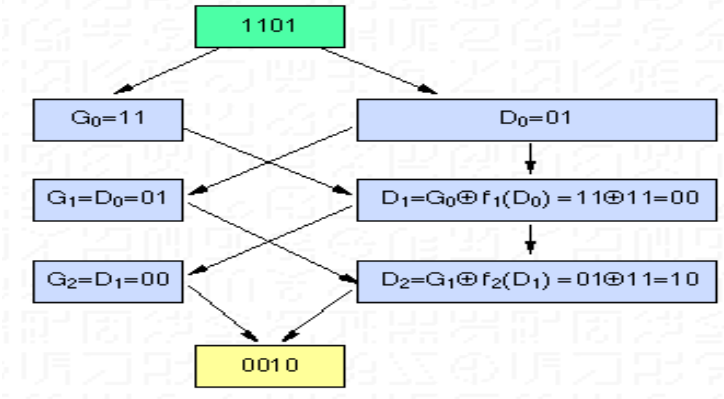
- **Exemple :**

Pour un chiffrement Feistel à deux rondes pour un message composé de 4 bits, considérons la fonction de ronde suivante :

entrée	f_1	sortie
00	→	01
01	→	11
10	→	10
11	→	01

entrée	f_2	sortie
00	→	11
01	→	00
10	→	00
11	→	01

Chiffrons le message 1101 :



message	résultat
0000	0100
0001	1100
0010	1010
0011	0111
0100	0011
0101	1001
0110	1111
0111	0000
1000	1101
1001	0101
1010	0001
1011	1110
1100	1000
1101	0010
1110	0110
1111	1011

1.8.1.2 Chiffrement par flux

Dans les systèmes de chiffrement de flux, le chiffrement des messages est effectué caractère par caractère ou bit par bit à l'aide de permutations générées aléatoirement. La taille de la clé correspond donc à la taille du message.

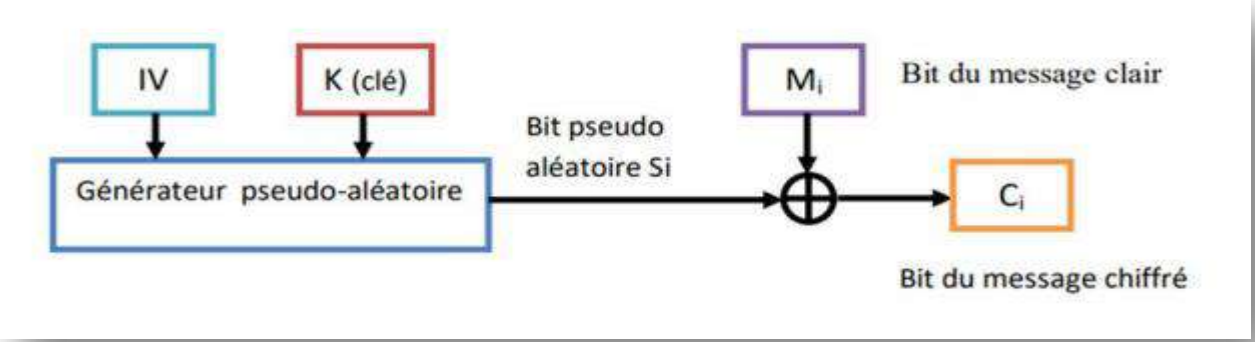


Figure I.7: Schéma de chiffrement par flux [7]

Le tableau 1.1 résume les avantages et les inconvénients du chiffrement par blocs et par flux

	Par blocs	**Par flots**
Avantages	- Réutilisation des clés	-Plus rapide -Moins de code d'implémentation
Inconvénients	-Plus lent	- Deux utilisations d'une même clé facilitent la cryptanalyse
Applications	- Transfert de fichiers	- Chiffrement de canal de communication

Tableau 1. 1 : Comparaison des chiffrements par blocs et par flots [9]

1.8.2 Cryptographie asymétrique (à clefs publiques)

La cryptographie asymétrique repose sur des problèmes mathématiques complexes tels que la factorisation de grands entiers et les équations logarithmiques discrètes. Le chiffrement asymétrique repose sur le principe de deux clés : une clé publique et une clé privée. La clé publique est mise à disposition de toute personne souhaitant chiffrer un message (tout le monde peut connaître cette clé). Ce dernier ne peut être déchiffré qu'à l'aide de votre clé privée. La clé privée est confidentielle et ne doit être connue que de son propriétaire.

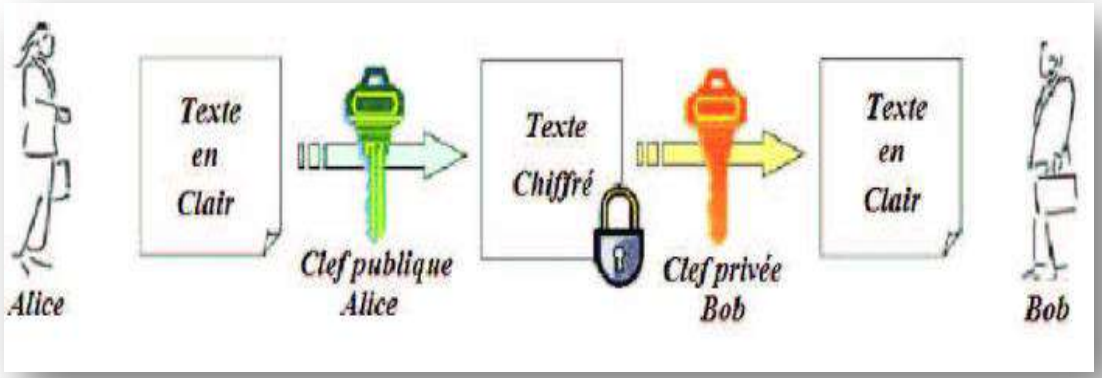


Figure 1. 8 : Chiffrement asymétrique [8]

Le tableau 1.1 compare les deux méthodes de chiffrement symétrique et asymétrique

	Les Avantages	Les inconvénients
symétrique	- Rapide - Facile - Clés relativement courtes (128 ou 256 bits).	-Gestion des clés difficiles (nombreuses clés). - Difficulté de distribuer la clé secrète -Ne permet pas de signature électronique
asymétrique	- Utilise deux clés différentes. - Fournit des garanties d'intégrité et de non répudiation par signature électronique. - Très utile pour échanger les clés.	- Des clés plus longues (1024 à 4096 bits) - Lenteur de calcul - Difficile

Tableau1. 2 : Comparaison entre la cryptographie symétrique et asymétrique. [9]

1.9 Cryptage des images numériques

Le but du chiffrement d'images est de garantir la sécurité visuelle du contenu en clair d'une image. Afin de transmettre des images de manière confidentielle, une variété de schémas de cryptage ont été proposée. [10]

Plusieurs classifications sont considérées :

- Les schémas peuvent être répartis en deux grandes catégories : les schémas du domaine spatial et ceux du domaine fréquentiel.
- Les méthodes peuvent être classées selon que l'image déchiffrée identique à l'image originale ou chiffré non. Notons par ailleurs que cette dernière exigence n'est pas nécessaire pour les images numériques. En effet, le niveau d'exigence de la perception humaine des images décryptée est plus faible et de ce fait une image décryptée contenant une faible distorsion est généralement acceptable.
- Le chiffrement des images peut être classé en chiffrement complet, chiffrement Partiel. Dans le chiffrement Complet, l'image brute ou compressée est chiffrée par une nouvelle méthode ou une méthode traditionnelle. Dans le chiffrement partiel (également appelé chiffrement sélectif), seules les parties significatives dans l'image sont chiffrées.

Dans notre travail expérimental représenté dans le chapitre trois, nous nous focalisons seulement sur les processus de chiffrement complet des images brutes. L'image sera traitée dans le domaine spatial et l'image déchiffrée est identique à l'image brute. Nous présentons ci-dessous un bref rappel sur l'image numérique et ces différents types.

1.9.1 Définition de l'image numérique

L'image numérique est définie par le nombre total de pixels qui constitue l'image calculé en multipliant le nombre de pixels de la colonne par le nombre de pixels de la ligne.

Par exemple; la définition d'une image avec 800 pixels de largeur et 600 est 480000.Cela nous conduit à clarifier la notion de pixel. Le mot Pixel est l'abréviation de «Picture Element »et représente la plus petite unité d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image. Les valeurs d'un pixel sont toujours des mots binaires de longueur k afin qu'il puisse représenter 2^k valeurs différentes. La valeur de k est souvent appelée «profondeur de l'image ». Pour une image couleur typique avec trois composantes RVB, le pixel entier est codé en 24 bits, en conséquence, cette image peut représenter 2^{24} couleurs différentes, ce qui équivaut à 16777216 couleurs différentes.

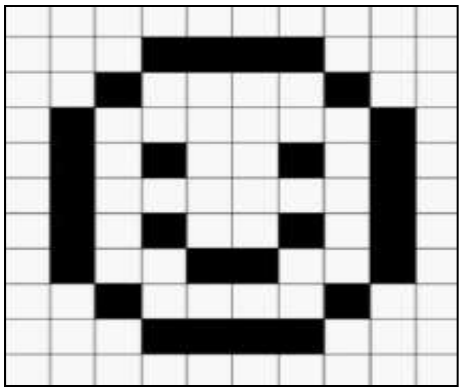


Figure 1. 10: Tableau à deux dimensions de l'image [17]

1.9.2 Types des images numériques

II existe différentes types des images selon le nombre de bits sur lequel est codée la valeur de chaque pixel [16] [17].

A. Image binaire

Les images binaires sont un type spécial d'image ou les pixels ne peuvent prendre qu'une des deux valeurs, noir ou blanc. Généralement codé en utilisant un seul bit (0/1) par pixel. C'est typiquement le type d'image que l'on utilise pour scanner du texte quand celui-ci est composé d'une seule couleur.

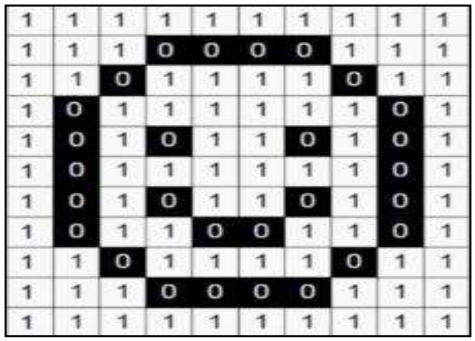


Figure 1.11: Image binaire [17]

B. Image en niveaux de gris

Une image en niveaux de gris a des couleurs qui sont des nuances de gris (Figure 1.9.b). Le niveau de gris est la valeur de l'intensité lumineuse d'un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires.

Une image typique en niveaux de gris utilise k = 8 bits par pixel, chaque pixel n'est donc plus représenté par un bit, mais par un octet. La valeur de l'intensité lumineuse correspondante est comprise entre 0 et 2^8- 1.ou 0 représente la luminosité minimale (noir) et 255 représente la luminosité maximale (blanc). Dans certains domaines tels que la photographie professionnelle, la médecine et l'astronomie, 8 bits par pixel ne sont pas suffisants, des profondeurs d'image de 12; 14 et même 16 bits sont souvent utilisées.

C. Image en couleurs

L'intérêt de ces images est de réduire l'espace de stockage même s'il est parfois utile de pouvoir présenter des images en noir et blanc ou en niveau de gris, les applications multimédias utilisent le plus souvent des images en couleurs (Figure1.9.c). La plupart des images en couleur sont basées sur les couleurs primaires rouge, vert et bleu (RV B), lesquels utilisent généralement 8 bits pour chaque composante de couleur. Cela signifie que chaque pixel nécessite 3×8 = 24 bits pour coder les trois composantes, et l'intervalle de chaque composante de couleur individuelle est de [0; 255]. Alors que les images couleurs avec30; 36 et 42 bits par pixel sont couramment utilisées dans les applications professionnelles. Les images couleurs indexées, dont une classe très spéciale d'image couleur, qui stockent Pour chaque pixel un numéro de couleur (son index), le quel fait référence à une couleur stockée séparément dans une palette. Par exemple, une image en 256 couleurs indexées occupera sensiblement la même place qu'une image en 256 niveaux de gris (la place occupée par la palette est négligeable par rapport à la taille de l'image).

Figure 1.12 : (a) Image binaire, (b) Image en niveau de gris, (c) Image couleur. [17]

1.10 Conclusion

Ce chapitre présente les bases de la cryptographie. Après un bref historique de la cryptographie et la terminologie spécifique au domaine, les différents objectifs de la cryptographie et les différentes techniques cryptographiques ont été exposés. Comme ce travail concerne les images, nous avons exposé les classes des méthodes de chiffrement d'images et présenté également la définition de l'image numérique ainsi que ces différentes classes, dans le chapitre suivant nous explorons les algorithmes de cryptographie symétrique tels que DES, AES et RC4 et Nous expliquons comment chaque algorithme fonctionne, et comment ils sont utilisés pour sécuriser les données.

Chapitre 2

Les algorithmes de cryptographie à clé secrète

2.1 Introduction

Le chiffrement à clé privée, également appelé chiffrement symétrique, est un concept fondamental dans le domaine de la sécurité de l'information. Cette méthode de cryptage est largement utilisée dans diverses applications telles que les canaux de communication sécurisés, le stockage de données et les protocoles d'authentification, ce chapitre explore les algorithmes de chiffrement symétriques tels que DES, AES et RC4 et comment chaque algorithme fonctionne.

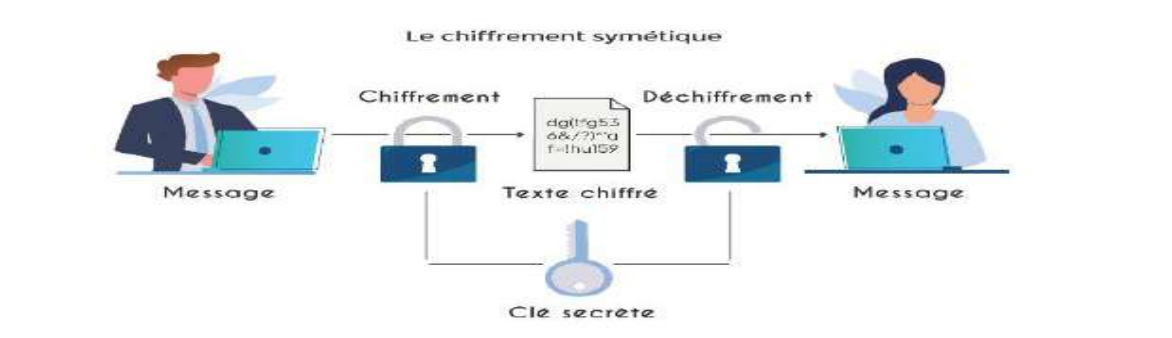


Figure 2. 1: cryptage à clé secrète [19]

Un algorithme symétrique est celui qui utilise une clé pour convertir le texte brut en texte chiffré, et la même clé pour reconvertir le texte chiffré en texte brut. La confidentialité des communications est garantie uniquement par les clés utilisées lors des phases de cryptage et de déchiffrement. Les algorithmes utilisés ne font pas partie du secret. Nous parlons d'algorithmes symétriques car la même clé est utilisée à la fois pour chiffrer et déchiffrer le message.

2.2 Méthodes de cryptographie symétrique par blocs

Un chiffrement par bloc est un type d'algorithme de chiffrement à clé symétrique qui convertit un bloc de données de texte brut de longueur fixe en un bloc de données de texte chiffré de même longueur. La longueur fixe est appelée taille de bloc.

Pour le chiffrement multibloc, la taille du bloc est de 64 bits ou 128 bits. Plus la taille du bloc est grande, plus le cryptage est efficace, mais plus les algorithmes et les dispositifs de cryptage et de déchiffrement sont complexes. Un exemple de chiffrement par bloc est la méthode (DES). Certains des algorithmes qui utilisent des chiffrements par blocs sont : DES, AES, IDEA…

2.2.1 Algorithme DES

Le Data Encryption Standard (DES) [11] [12] est un algorithme de cryptographie qui a été sélectionné comme un standard pour la Fédéral Information Processing Standard (FIPS) pour les Etats-unien 1976 et qui a connu un succès international par la suite.

Le DES opère sur des blocs de 64 bits avec des clés de 64 bits dont 8 bits de parité. Il est basé sur un ensemble de transformations composées de substitution, de transposition et d'opérations non linéaires (figure 2.2).

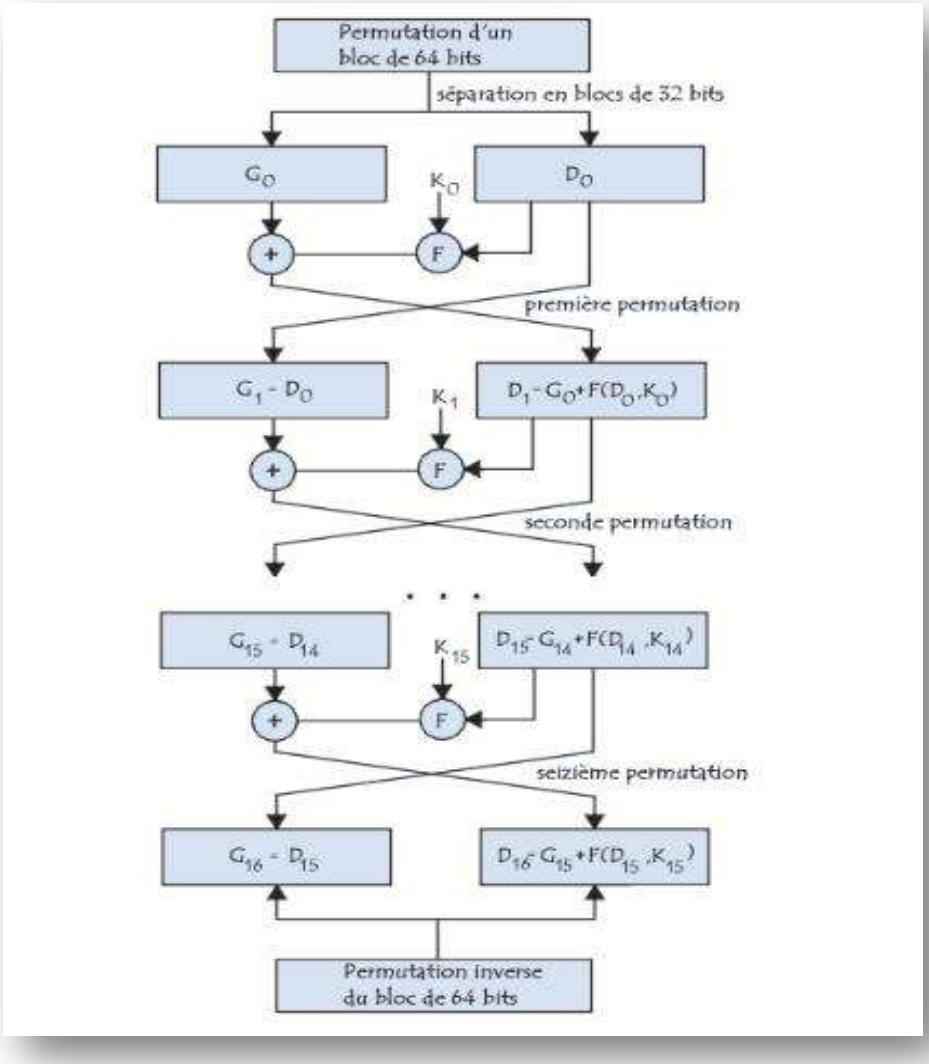


Figure 2.2 : Algorithme DES [11]

Après une permutation initiale IP, le bloc est découpé en deux parties gauche et droite de 32 bits. Il y aura 16 rondes où les données sont combinées avec la clé, après le $16^{éme}$ ronde, les parties gauches et droites sont transposées et rassemblées puis une permutation finale IP^{-1}.

Au cours de chaque ronde, la partie droite est soumise à une fonction F composée de quatre opérations :

- Extension à 48 bits en utilisant des tables E ;
- Combinaison avec la clé secondaire KI à l'aide d'un « ou exclusif » ;
- Décomposition en 8 mots de 6 bits associés à 8 tables de substitution S1 à S8, chaque mot servant à fournir les numéros de ligne et de colonne;
- Extraction de chaque table de substitution S1 à S8 d'un mot sur 4 bits. En termine par une permutation P des 32 bit obtenus.

A. Première étape : Permutation initiale

Les 64 bits du bloc d'entrée subissent la permutation initiale présentée par la matrice suivante :

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

Cette matrice de permutation indique, en parcourant la matrice de gauche à droite puis de haut en bas, que le ^{58ème} bit du bloc de texte de 64 bits se retrouve en première position, le $50^{ème}$ en seconde position et ainsi de suite.

B. De la 2 éme à la 17 éme étape (ronde)

- Les 64 bits initiaux de données sont divisés en 2 blocs (G et D).

G

58	52	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	28	30	22	14	6
64	56	48	40	32	24	16	8

D

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

- Les blocs G et D sont soumis à un ensemble de transformations itératives appelées rondes à l'aide des sous clés générées à partir de la clé principale. L'ensemble des transformations se traduit par les équations suivantes :
  - $G_i = D_{i-1}$
  - $G_i = G_{i-1} \oplus F(G_{i-1}, K_i)$.

  Où :

  i =1……..16.

F(de confusion): Fonction mathématique qui représente l'élément important sur lequel repose la sécurité du DES composée de :

1. Extension de D_{I-1} de 32 bits à 48 bits en utilisant la table E ;

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

2. La matrice résultante de 48 bits est appelée E $[D_{i-1}]$. L'algorithme DES procède ensuite à un OU exclusif entre la clé $K_i$et E $[D_{i-1}]$, le résultat est une matrice de 48 bits;
3. E $[D_{i-1}]$ est ensuite partagé en 8 blocs de 6 bits, noté Bj (j=1……8) qui sera l'entrée d'une table de substitution S ;
4. Chaque table reçoit une information de 6 bits et envoie en sortie une information de 4 bits.

Le bloc en entrée de 6 bits est traité comme suit : le premier et le dernier bit du bloc unis représentent un nombre compris entre 0 et 3 qui désignent le numéro de la ligne de la table S_i, les 4 bits restants représentent un nombre compris entre 0 et 15 qui donnent le numéro de la colonne.

On regroupe les mots de 4 bits de chaque table pour former un mot de 32 bit qui subira une permutation selon la table P.

P

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

C. Permutation finale

Le contenu du bloc de pré-sortie, est permuté une dernière fois. Cette permutation correspond à l'inverse de la permutation initiale.

IP-1

48	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	29
35	3	43	11	51	19	59	28
34	2	42	10	50	18	28	26
33	1	41	9	49	17	57	25

D. Génération des clés

La clé initiale est de 64 bits, à partir de laquelle en génère 16 sous clés K_i chacune sur 48 bits, en suivant les étapes ci-dessous :

1. Enlever les bits de parités afin d'obtenir une clé de longueur de 56 bits ;
2. Application d'une première permutation notée CP^{-1} dont la matrice est présentée ci-dessous ;

CP^{-1}

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

3. On divise la matrice CP^{-1} en deux matrices G et D de 28 bits chacune ;

G

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

D

63	55	47	39	31	33	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

4. Les blocs subissent un décalage à gauche, puis regrouper pour former un bloc de 56 bits. Ce dernier subira une permutation CP-2 fournissant en sortie un bloc de 48 bits, représentant la clé Ki.

CP-2

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	24	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 itérations de l'algorithme permettent de donner les 16 sous clés K1 à K16 utilisées dans l'algorithme DES.

2.2.2 Algorithme Triple DES

Triple DES [12] (également connu sous le nom de 3 DES) est un algorithme de chiffrement symétrique qui concatène trois applications consécutives de l'algorithme DES sur le même bloc de données de 64 bits à l'aide de deux ou trois clés DES différentes. Cette utilisation des trois chiffres DES a été développée par Walter TUCHMAN. Bien que le DES améliore la sécurité,

son inconvénient majeur est que le chiffrement et le déchiffrement nécessitent davantage de ressources. [11]

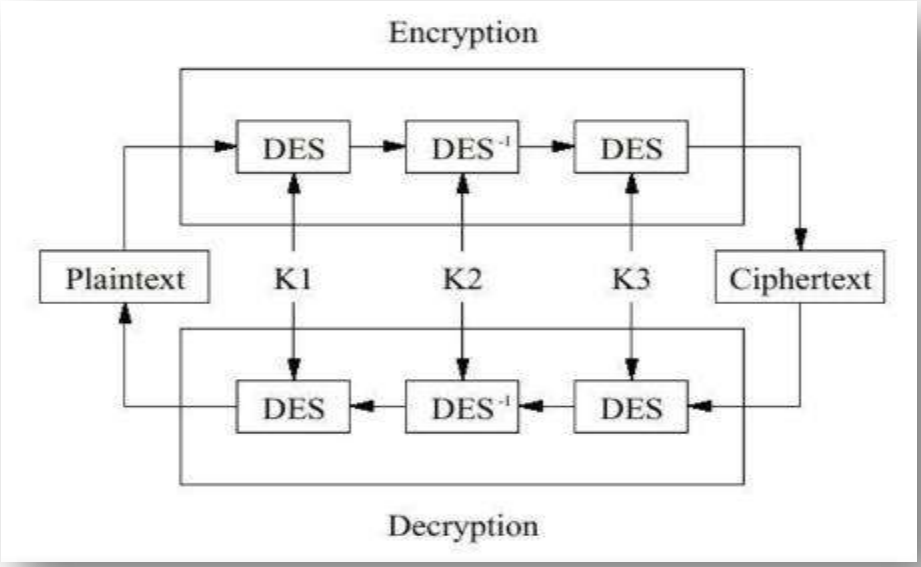


Figure 2. 3 : Triple DES [11]

2.2.3 Algorithme IDEA

IDEA for International Data Encryption Algorithm est un algorithme de décryptage développé par Xuejia Lai et James Massey et introduit pour la première fois en 1991. IDEA est un algorithme de chiffrement par bloc symétrique de 64 bits (ou 8 octets) qui fonctionne avec une clé de 128 bits. Une particularité de cet algorithme est que les opérations utilisées en interne sont très facilement adaptables à la programmation informatique sur des machines 16 bits. [12]

2.2.4 Algorithme AES

AES est un algorithme de chiffrement symétrique basé sur le système Rijndael développé par Joan Daemen et Vincent Rijmen. Vous pouvez chiffrer et déchiffrer 4 444 éléments d'information par blocs de 128 bits (16 octets binaires) à l'aide d'une clé de 128, 192 ou 256 bits.AES est effectué en plusieurs tours comprenant plusieurs transformations. Le nombre de tours Nr dépend de la taille de la clé, les clés de 128 bits nécessitent 10 tours, les clés de 192 bits nécessitent 12 tours et les clés de 256 bits nécessitent 14 tours.

A. Chiffrement et déchiffrement avec l'AES

Dans ce qui suit, nous détaillons l'AES-128, où les 128 bits de données sont répartis en 16 blocs de 8 bits (8 bits = 1 octet), eux-mêmes « dispatchés » dans un tableau 4×4. Même les 128 bits de la clé sont organisés sous forme matricielle. Pour d'évidentes raisons de taille, les valeurs binaires seront notées sous forme hexadécimale. Le déchiffrement est l'opération inverse du chiffrement et les transformations se réalisent dans le sens inverse. La figure 2.3 montre les différentes opérations effectuées dans chaque round pendant le processus de chiffrement et de déchiffrement de l'AES.

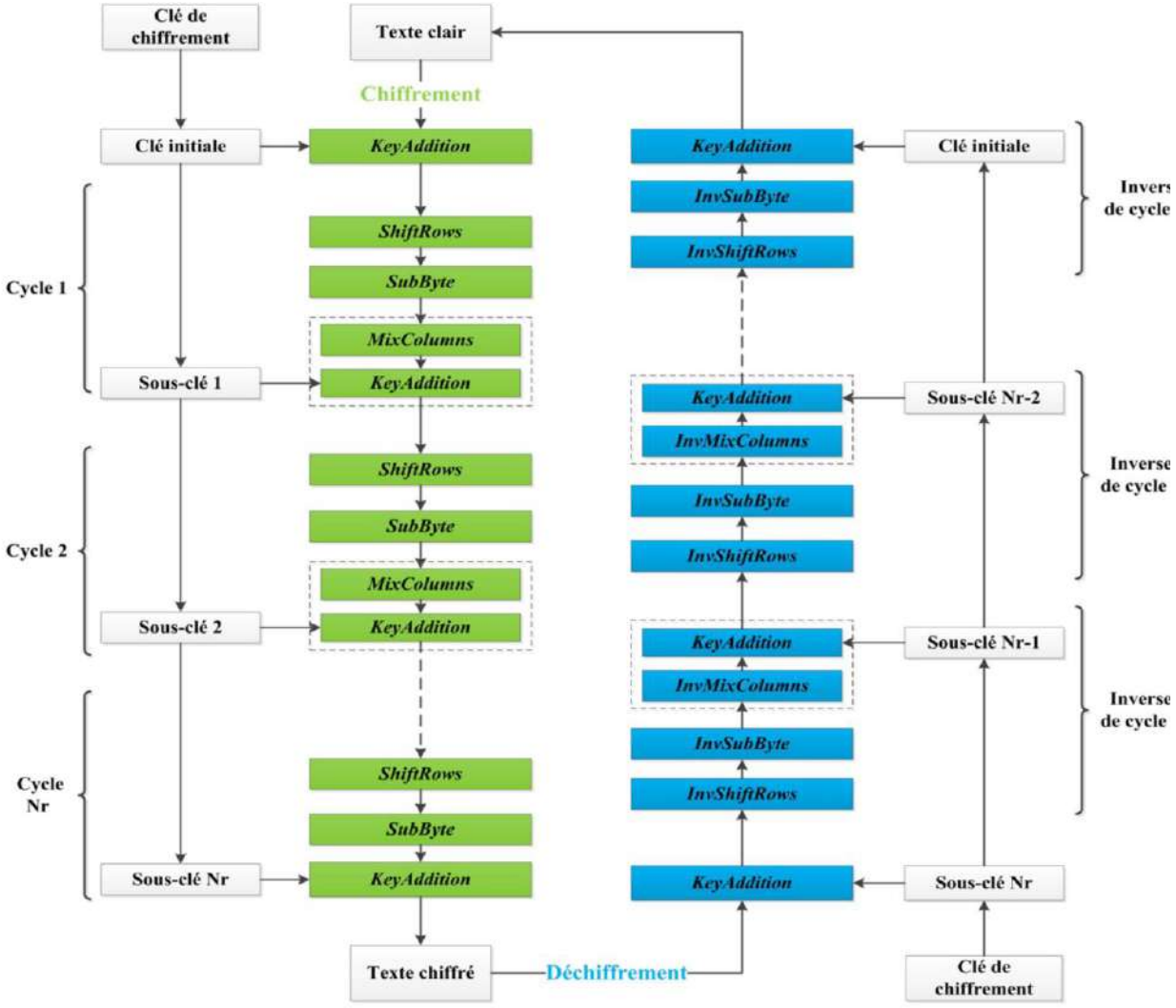


Figure 2. 4 : Transformations de l'AES [11]

La première étape de chiffrement consiste à combiner la matrice State (le bloc de texte clair) avec la clé. Cette opération s'appelle AddRound Key. À chaque tour, quatre transformations sont appliquées SubBytes, ShiftRows, MixColumn et AddRoundKey sauf pour le dernier tour, l'opération MixColumns n'est pas effectuée. Chaque tour utilise son propre sous clé qui est générée par l'opération Key Expansion à partir de la clé maitresse [8].

❖ SubBytes

L'étape SubBytes correspond à la seule transformation non linéaire de l'algorithme [13]. Dans cette étape, chaque élément de la matrice d'état est trié selon une table de permutation réversible appelée S-Box, comme le montre la figure 2.5.

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Figure 2. 5 : Table S-Box de l'AES [13]

La figure 2.6 illustre un exemple de transformation de l'élément S_{11} en l'élément S'_{11}.

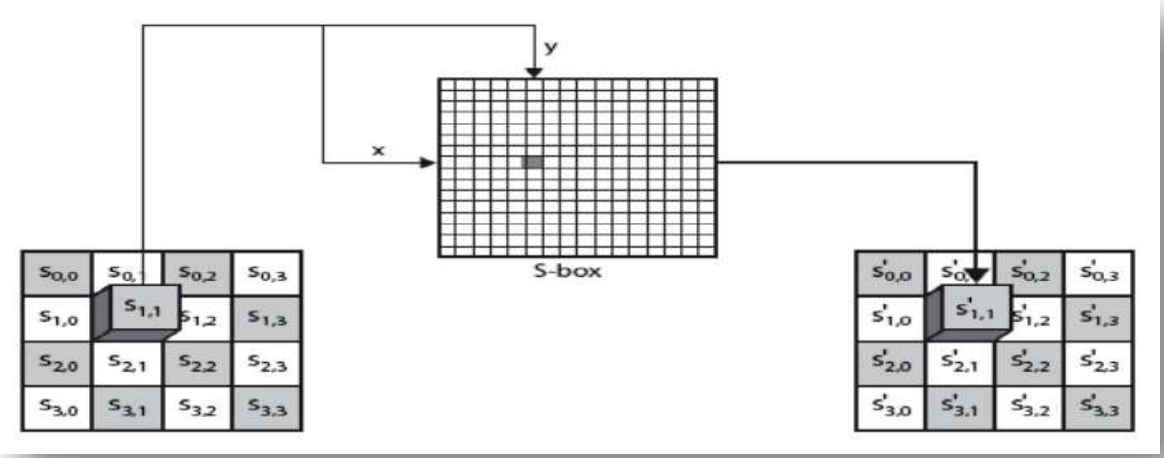


Figure 2.6 : Transformation SubBytes [13]

Le tableau S-box est dérivé de la fonction inverse g : a → a^{-1} de GF [2^8].Cette fonction est connue pour ses excellentes propriétés non linéaires. Pour éviter les attaques basées sur des propriétés algébriques simples, la table S-box est construite en combinant sa fonction inverse avec une transformation affine réversible f.

Donc : S-Box[a] = f (g(a)), ∀ a ∈ un GF [2^8].

La fonction affine f : GF [2^8] → GF[2^8] est définie comme :

$$b = f(a) \equiv \begin{pmatrix} b_7 \\ b_6 \\ b_5 \\ b_4 \\ b_3 \\ b_2 \\ b_1 \\ b_0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \times \begin{pmatrix} a_7 \\ a_6 \\ a_5 \\ a_4 \\ a_3 \\ a_2 \\ a_1 \\ a_0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \qquad (2.1)$$

❖ ShiftRows

Cette étape effectue un décalage circulaire vers la gauche sur chaque élément de la ligne [13]. Le nombre de décalages varie d'une ligne à l'autre. La première ligne est inchangée, la deuxième ligne est déplacée vers la gauche de 1, la troisième ligne est déplacée vers la gauche de 2 et la quatrième ligne est déplacée vers la gauche de 3. L'opération inverse consiste à effectuer le même décalage sur toutes les lignes dans le sens opposé, comme le montre la figure 2.7, qui montre ShiftRow et son inversion.

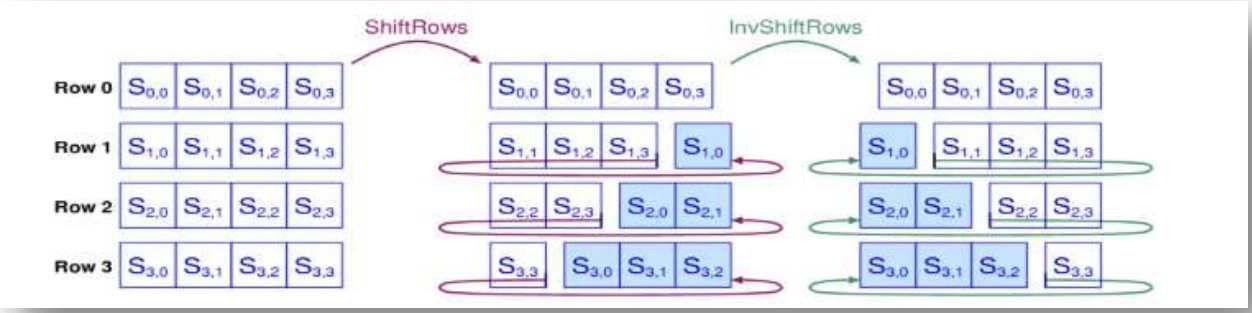


Figure 2.7: ShiftRows et Inverse ShiftRows [13]

❖ MixColumns

Cette étape est une transformation linéaire qui remanie chaque colonne de la matrice d'état.

Chaque octet d'entrée affecte 4 octets de sortie, et chaque colonne de 4 octets est considérée comme un vecteur et multipliée par une matrice fixe (4×4) définie comme :

$$\begin{bmatrix} C_0 \\ C_1 \\ C_2 \\ C_3 \end{bmatrix} = \begin{bmatrix} B_0 \\ B_5 \\ B_{10} \\ B_{15} \end{bmatrix} \times \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \quad (2.2)$$

L'opération inverse consiste à multiplier chaque colonne de la matrice state par ma matrice prédéfinie ci-dessous ;

$$\begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \quad (2.3)$$

La multiplication et l'addition se font dans GF [2^8].

❖ AddRoundKey

Les deux entrées de l'opération AddRoundKey sont la matrice d'état et la sous-clé. Les deux entrées sont combinées par une opération XOR par octet. Notez que l'opération XOR équivaut à l'ajout du champ de Galois GF (2^8).

B. KeyExpansion

C'est l'algorithme de dérivation des sous-clés à partir de la clé secrète. Le nombre de sous-clés est égal au nombre de tours plus un. Les sous-clés sont calculées de manière récursive, afin de dériver une sous-clé K_i, la sous-clé K_{i-1} doit être connue. Le nombre de cycle du protocole cryptographique AES dépend de la taille de la clé utilisée et le processus de génération de clé diffère aussi selon la taille de cette clé.

❖ Clés de 128 bits

La génération de clé est orientée mot, 1 mot = 32 bits. Les 11 sous-clés sont stockées dans une matrice d'expansion de clés w composée de mots (W[0], ..., W[43]), comme le montre la figure 2.8 [18].

La première sous-clé est la clé AES originale, qui est copiée dans les quatre premiers éléments de W. Les autres éléments de la matrice sont calculés comme suit. Chaque mot W[i] dépend de W[i-1] et W[i-4]. Pour les mots dont les positions sont des multiples de 4, une fonction g plus complexe est utilisée.

g() est une fonction non linéaire d'entrée/sortie de 4 octets qui décale d'un octet, remplace chaque octet à l'aide de la table Sbox et ajoute un coefficient RC. Le facteur RC n'est ajouté qu'à l'octet le plus à gauche de la fonction g().

Ce coefficient varie d'un tour à un autre selon la règle suivante :

$$\begin{aligned} RC[1] &= x^0 = (0000\ 0001)_2 \\ RC[2] &= x^1 = (0000\ 0010)_2 \\ RC[3] &= x^2 = (0000\ 0100)_2 \\ &\vdots \\ RC[10] &= x^9 = (0011\ 0110)_2 \end{aligned} \qquad (2.4)$$

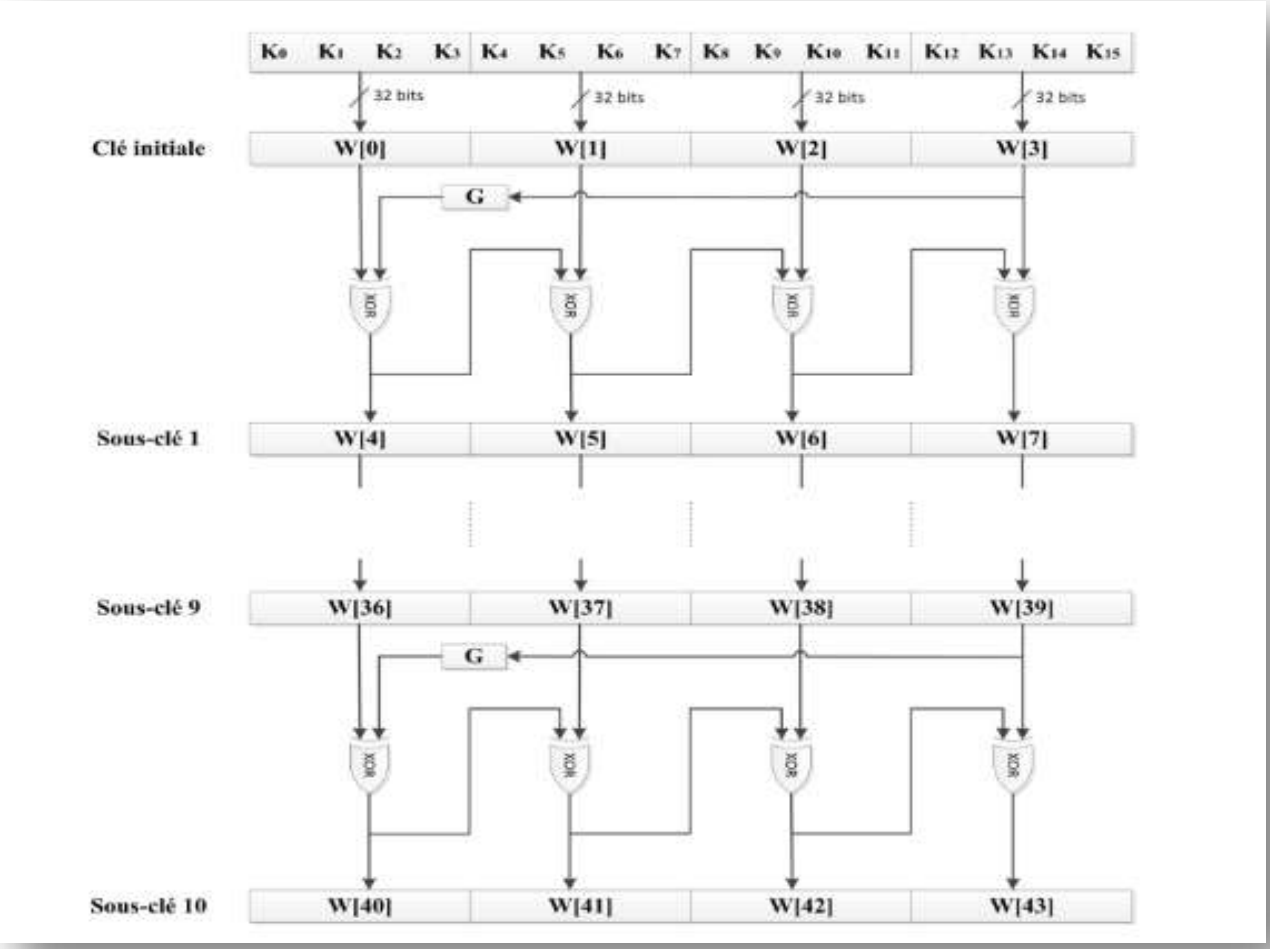


Figure 2. 8 : Génération des sous clés à partir d'une clé de 128 bits [13]

❖ Clés de 192 bits

Pour une clé de 192 bits, le processus de cryptage/déchiffrement prend 12 cycles et 13 sous-clés de 128 bits chacune [13].

Le calcul de la sous-clé est similaire à celui d'une clé de 128 bits, avec 8 itérations d'extension de clé. A chaque itération, six nouveaux mots sont calculés à partir du tableau W. Le premier tour de sous-clés est formé par les éléments du tableau (W[0], W[1], W[2], W[3]), et le deuxième tour de sous-clés est formé par les éléments du tableau.

- Saisir les éléments (W[4], W[5], W[6], W[7]), etc. Huit coefficients RC[i] sont requis dans la fonction g(). Ceux-ci sont calculés comme dans le cas 128 bits et placés à partir de RC[1]. .., RC [8].

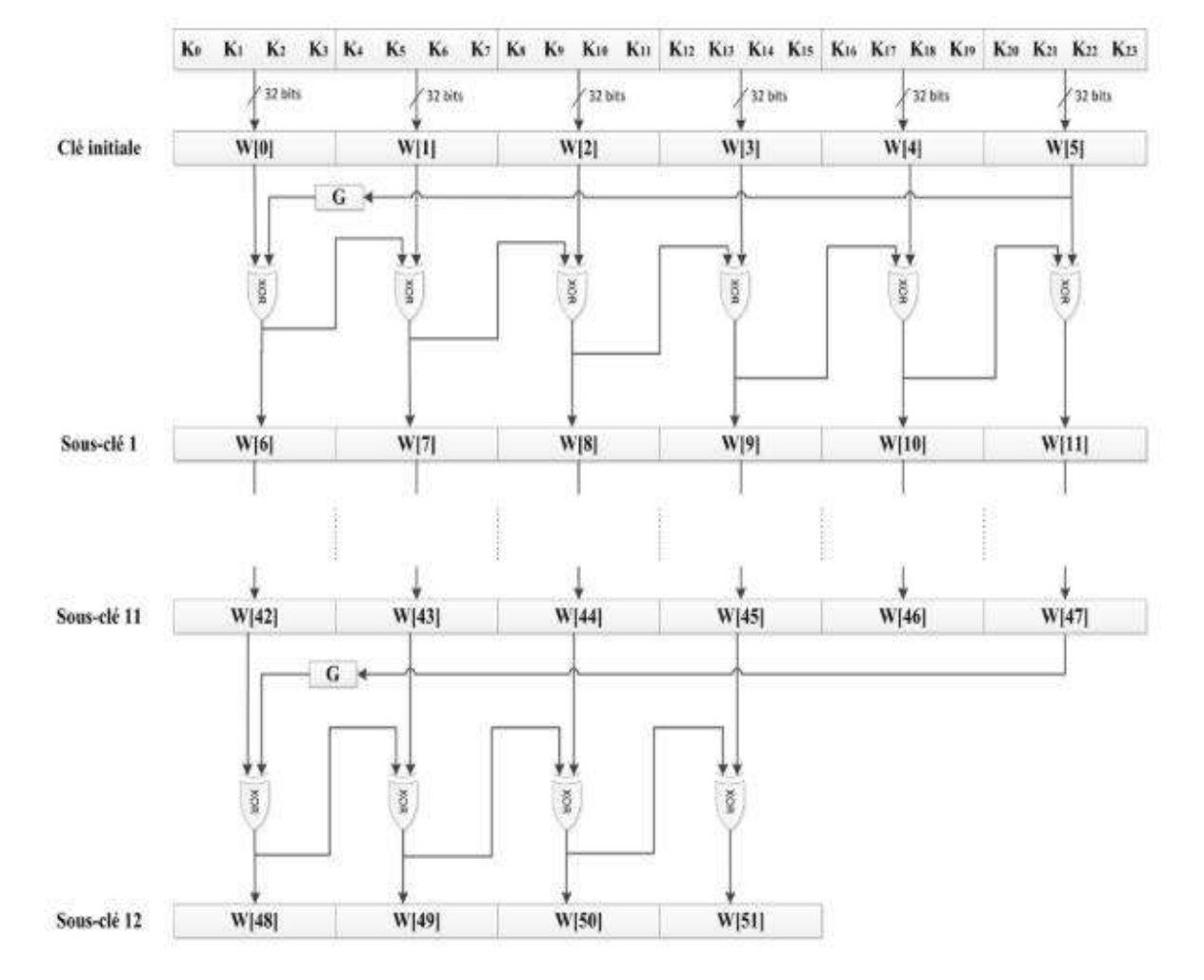


Figure 2.9: Génération de clés pour 192 bits [13]

❖ Clés de 256 bits

AES avec une clé de 256 bits a besoin de 15 sous-clés [14]. Les sous-clés sont stockées dans les 60 mots W[0], . . ., W[59]. Le calcul des éléments du tableau est similaire à celui de la clé de 128 bits et le Key Schedule comprend 7 itérations, où chacune calcule huit mots pour les sous-clés. La sous-clé pour le premier tour est formée par les éléments du tableau (W[0], W[1], W[2], W[3]), la deuxième sous-clé par les éléments (W[4], W[5], W[6], W[7]), et ainsi de suite. Il y a sept coefficients RC[1] . . . RC[7], nécessaires dans la fonction g(), qui sont calculés comme dans le cas de 128 bits. Le Key Schedule a une fonction h() avec des entrées /sortie de 4 octets.

La fonction applique la S-Box à tous les quatre octets d'entrée. La figure 2.10 montre le processus de génération de clé dans le cas de 256 bit.

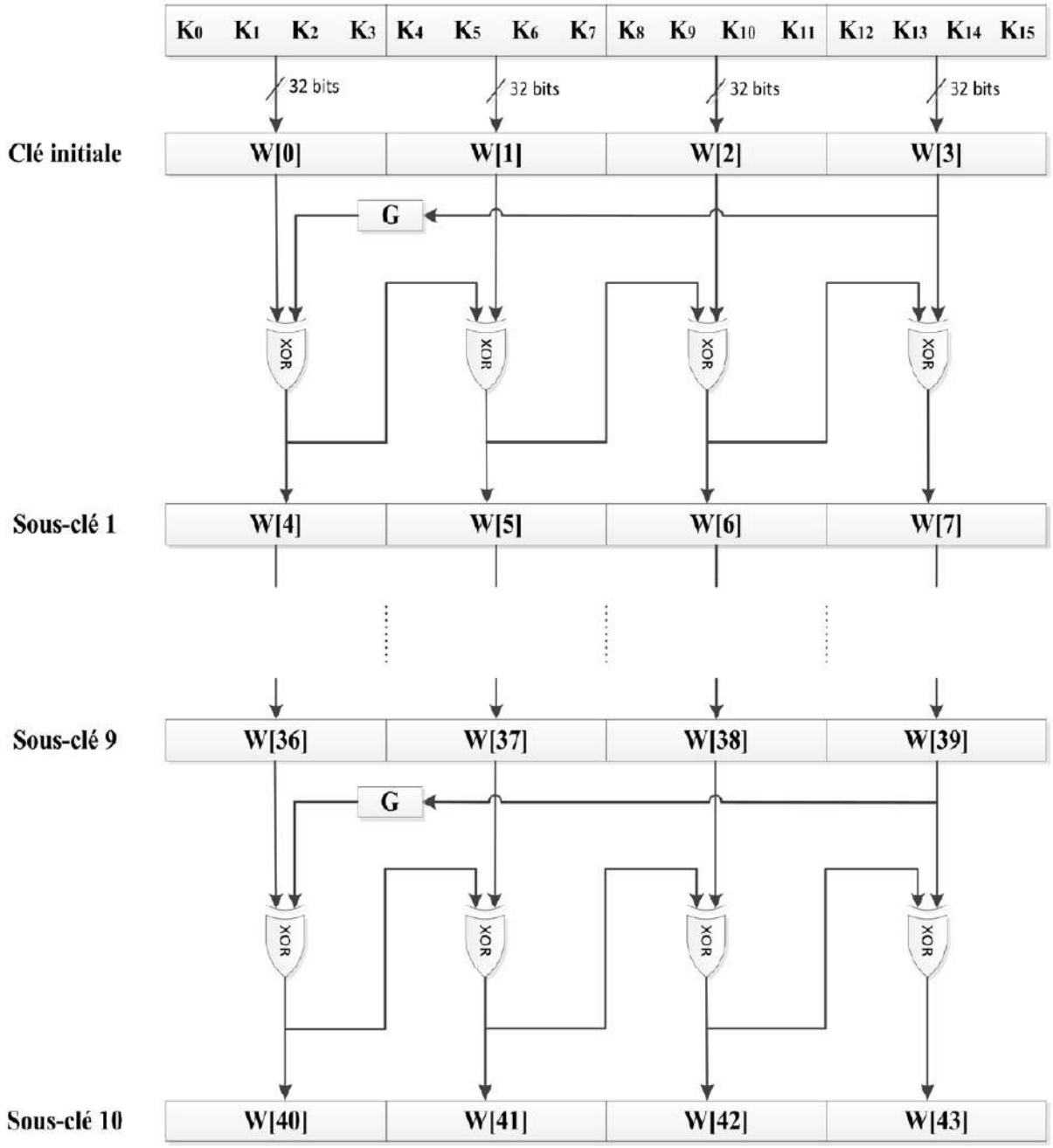


Figure 2. 10: Processus de génération de clé dans le cas de 256 bits. [13]

2.2.5 Les modes d'opération

Un mode de fonctionnement en cryptographie est la manière dont un bloc de texte clair chiffré est traité dans un algorithme de chiffrement par bloc [8].Cet algorithme est combiné à une série d'opérations simples pour améliorer la sécurité sans compromettre l'efficacité de l'algorithme. Cette combinaison est appelée mode de chiffrement.

Les quatre modes les plus courants sont décrits ci-dessous :

A. Le mode ECB

Le livre de codes électronique illustré à la figure 2.11, « Electronic Code Book » (ECB) en anglais: Le message à chiffrer est divisé en plusieurs blocs, qui sont cryptés individuellement en séquence. L'inconvénient majeur de cette méthode est que deux blocs ayant le même contenu sont chiffrés de la même manière, ce qui permet d'extraire des informations du texte chiffré en recherchant des séquences identiques.

Ensuite, nous obtenons un « dictionnaire de codes » contenant la correspondance entre texte en clair et chiffrement. D'où le terme livre de codes.

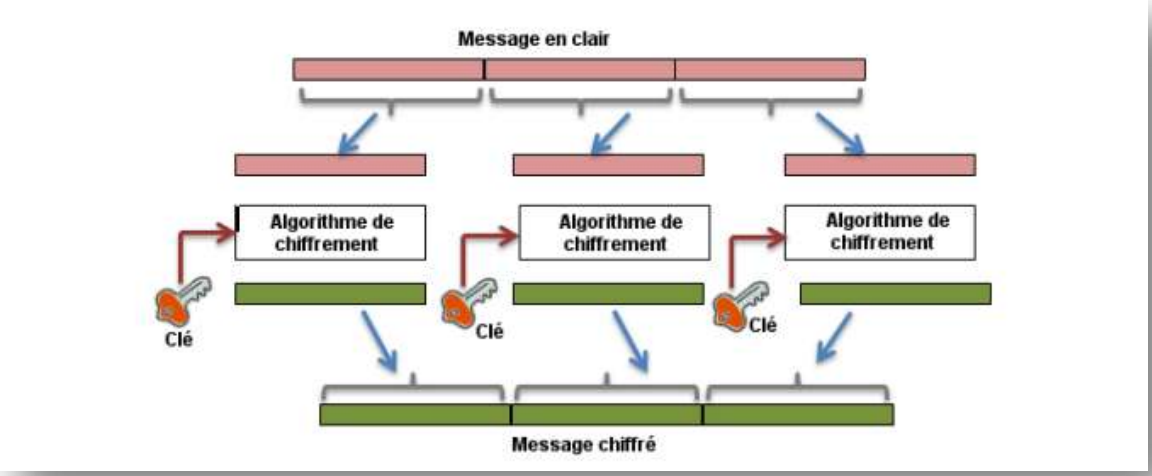


Figure 2. 11 : Le mode ECB [8]

Un problème de sécurité connu avec ce mode est que si vous utilisez deux fois le même texte brut et la même clé de chiffrement, le résultat du chiffrement sera le même. Les régularités du texte en clair sont également reproduites en texte chiffré.

Par conséquent, le texte chiffré vous permet d'obtenir des informations sur le texte en clair sans pouvoir le déchiffrer. Pour ces raisons, nous vous déconseillons fortement d'utiliser ce mode dans les applications cryptographiques. Le seul avantage qu'il peut offrir est la possibilité d'accéder rapidement à n'importe quelle région du texte chiffré et la possibilité de décrypter seulement une partie des données.

B. Le mode CBC

Les modes de chiffrement de la blockchain sont illustrés dans la figure 2.12 «Crypto Blockchain» (CBC). Fournit des solutions à la plupart des problèmes en mode ECB **[**12**]**. La chaîne utilise une méthode de rétroaction car le résultat du chiffrement du bloc précédent est réutilisé pour chiffrer le bloc actuel. Plus spécifiquement, un opérateur binaire XOR est appliqué entre le bloc de texte en clair actuel et le bloc de texte chiffré précédent, et un algorithme de chiffrement est appliqué au résultat de cette opération. Pour le tout premier bloc : un bloc au contenu aléatoire appelé vecteur d'initialisation (VI) est généré et utilisé pour appliquer l'opération XOR. Son rôle est d'empêcher les textes chiffrés correspondants de commencer de la même manière si deux textes en clair commencent de la même manière.

Elle doit donc être différente pour chaque message chiffré avec la même clé. Par conséquent, chaque bloc de chiffrement dépend non seulement du bloc de texte en clair correspondant mais également de tous les blocs de chiffrement qui le précèdent, comme le montre la figure 2.12.

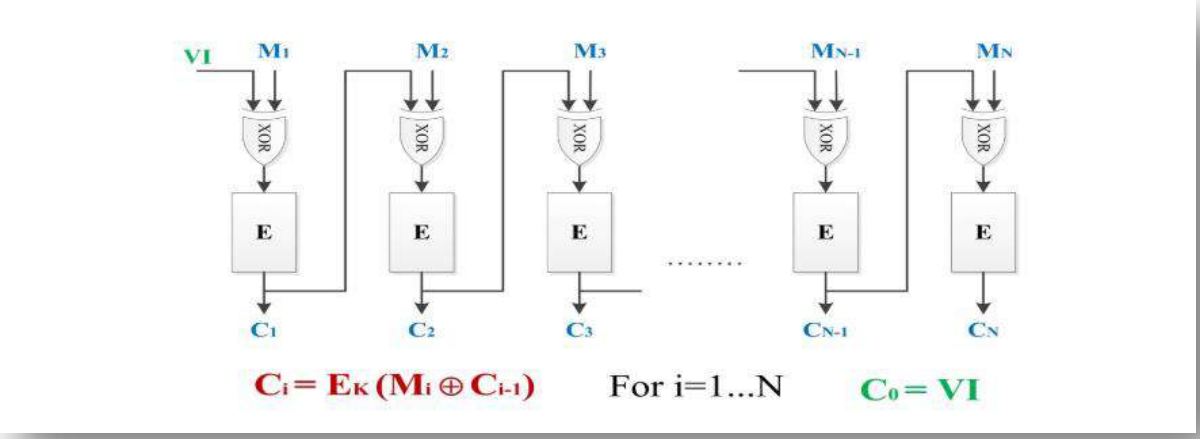


Figure 2. 12 : Le mode CBC [8]

C. Le mode CFB

Le mode de chiffrement des commentaires est illustré dans la figure 2.13, « Cypher Feedback » (CFB) **[**8**]**. Le premier bloc de ce mode est un bloc avec un vecteur d'initialisation (VI) qui est chiffré avec l'algorithme de chiffrement utilisé et combiné avec le texte en clair par OU exclusif

pour obtenir le texte chiffré pouvant être transmis. Le résultat de cette combinaison est réutilisé pour chiffrer le bloc suivant. Plus précisément, ils sont chiffrés grâce au même algorithme de chiffrement et combinés à la clé par OU exclusif.

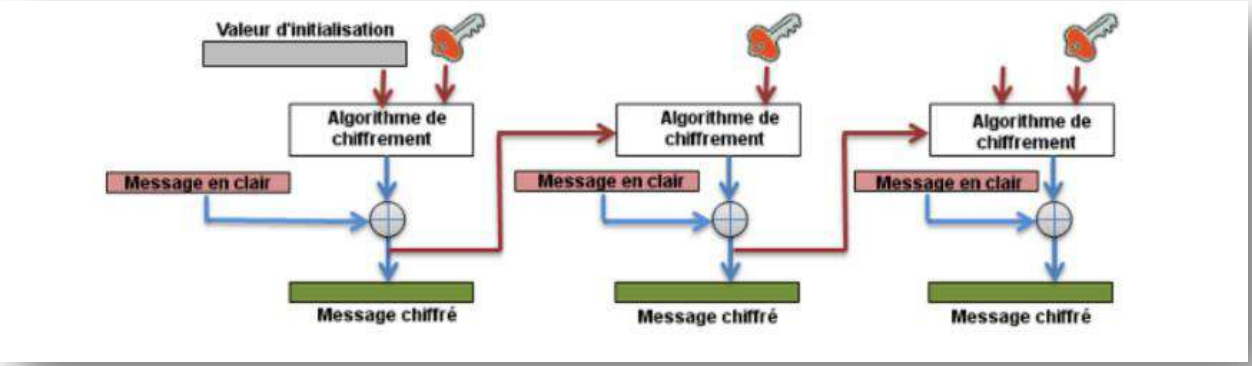


Figure 2. 13: Le mode CFB [8]

Le mode CFB offre une sécurité supérieure. Le seul problème est que le texte brut est simplement XOR. Si le texte en clair est connu, il peut être remplacé par un texte en clair entièrement différent en inversant les bits du texte chiffré aux mêmes endroits que les bits du texte en clair.

D. Le mode OFB

Le mode de retour de sortie (OFB) est très similaire au mode CFB [8], sauf que le résultat du chiffrement est le vecteur d'initialisation du bloc précédent, qui est réutilisé pour chiffrer le bloc courant.

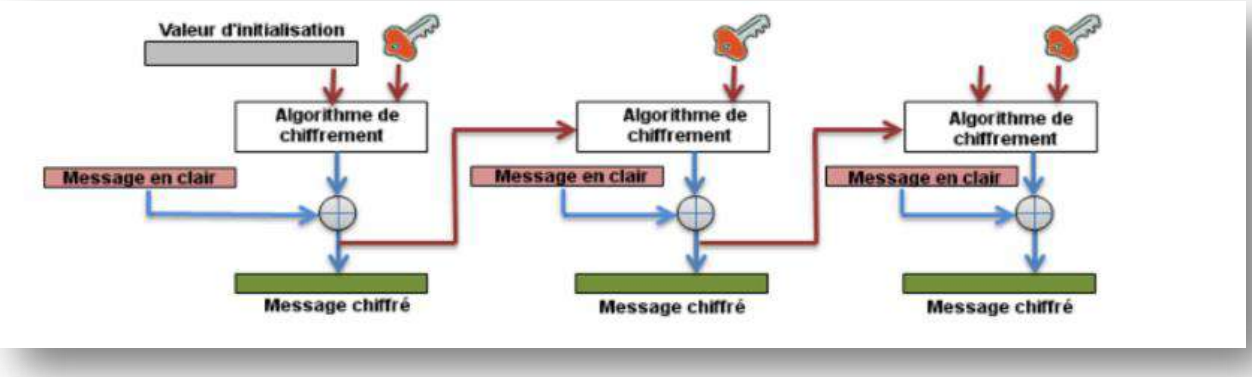


Figure 2. 14: Le mode OFB [8]

2.3 Méthodes de cryptographie symétrique par flots

Les chiffrements par flots fonctionnent généralement sur de petites unités de texte en clair, généralement des bits. Ainsi, les chiffrements par flots sont beaucoup plus rapides qu'un chiffrement par blocs typique. Généralement, un chiffrement par flots génère une séquence de bits en tant que clé (appelée flux de clé) en utilisant un générateur de nombres pseudo-aléatoires (PRNG) qui étend une courte clé secrète (par exemple 128 bits) en une longue chaîne de bits (flux de clé). Le chiffrement est effectué en combinant le flux de clé avec le texte en clair. Habituellement, l'opération XOR bit à bit est choisie essentiellement pour sa simplicité à effectuer ce chiffrement. Parmi les algorithmes qui utilisent chiffrement par flot : RC4, A5/1, E0 …. etc.

2.3.1 L'algorithme RC4

L'algorithme RC4 (Rivest Cipher 4) est un chiffre de flux symétrique utilisé pour le chiffrement des données [15]. Il a été conçu en 1987 par Ron Rivest de la société RSA Security. Malgré des faiblesses connues, il reste populaire pour sa simplicité et sa rapidité. Voici les étapes du RC4 :

- Étant donné un message clair à chiffrer
- Choisir une clé de chiffrement;
- Créer deux tableaux S et T de 256 cases chacun devant contenir des octets;
- Initialiser S avec les nombres de 0 à 255 dans cet ordre;
- Remplir T avec la clé;
- Effectuer des permutations aléatoires d'éléments de S en fonction d'éléments de T;
- Ré-effectuer des permutations aléatoires d'éléments de S suivant un schéma basé sur la configuration actuelle de S pour obtenir le flux de clés final;
- Effectuer un ou-exclusif entre le flot de clés final et le message clair à chiffrer.

A. Key Scheduling Algorithm (KSA)

Cet algorithme fonctionne sur les octets [18]. Ainsi, la clé, de longueur variable, peut avoir une taille comprise entre 1 et 256 octets (de 8 à 2048 bits). Elle est utilisée pour initialiser un vecteur S de 256 octets. A tout moment, S contient une permutation de toutes les cellules le composant. La clé est utilisée pour mélanger le tableau le mieux possible (On échange chaque élément avec un autre élément choisi en fonction de la clé); ainsi le tableau (par permutations) évolue à chaque tour. Cet algorithme est appelé Key Scheduling Algorithm, ou KSA.

Initialisation de la permutation identité

Initialement, les cellules de S reçoivent une valeur égale à leur position (i.e., S[0]=0, S[i]=i, ...

S

0	1	2	3	4	...	253	254	255

Ainsi la première instruction de notre algorithme sera:

For i = 0 to 255 do

S[i]:= i; // permutation identity

Un vecteur temporaire T (de longueur égale à celle de S) est également créé et destiné à recevoir la clé.

Si la longueur 'len' de la clé K est égale à 256 octets, K est simplement transféré dansT.

Si K est inférieur à 256 octets, il est recopié dans T jusqu'à atteindre la taille de T.

Dans notre algorithme, ce qui précède s'écrit (pour une clé K de longueur I):

T[i] = K [i mod len];

Le vecteur temporaire T est ensuite utilisé pour produire la permutation initiale de S. Pour chaque cellule S[i] de S, celle-ci sera échangée avec une autre cellule de S selon un calcul basé sur la valeur comprise dans la cellule T[i] correspondante.

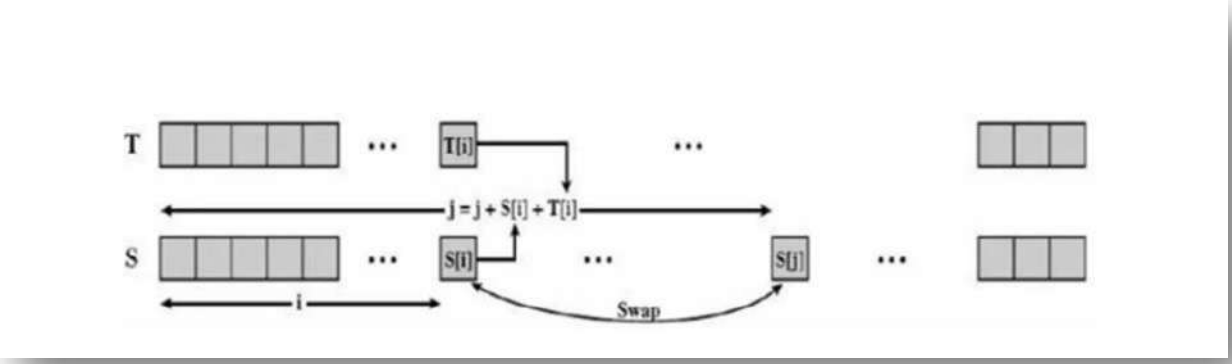


Figure 2.15 : Algorithme KSA [18]

Tout comme l'initialisation des vecteurs, on peut représenter la permutation initiale de S comme suit:

```
j = 0;
FOR i = 0 TO 255 DO
j = (j + S[i] + T[i]) mod 256;
SWAP(S[i], S[j]);
```

B. Random Génération Algorithm (PRGA)

Il s'agit de la Génération du flux pseudo aléatoire. A partir de cet instant, la clé d'entrée n'est plus utilisée. Pour chaque S[i], on procèdera à un échange avec un autre octet de S, selon un schéma basé sur la configuration courante de S. Une fois arrivé à S[255], le processus redémarre à la cellule S[0]. La figure suivante présente la procédure.

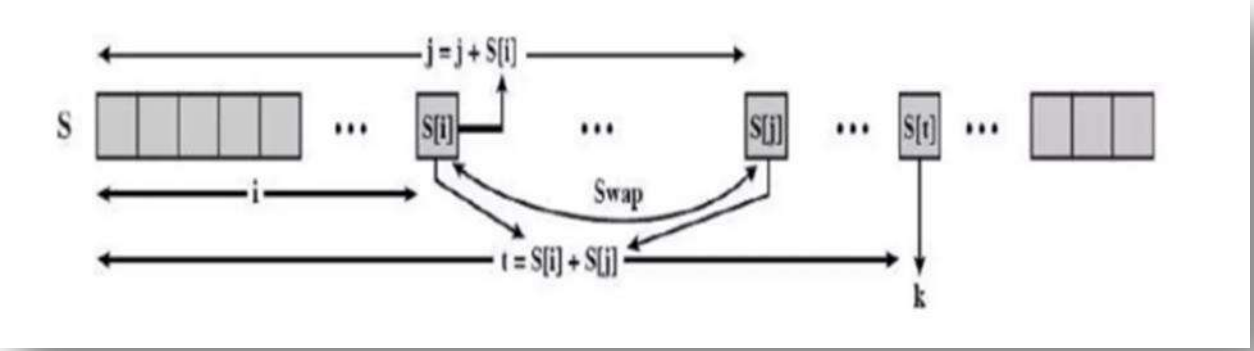


Figure 2. 16 : Random Génération Algorithm (PRGA) [18]

A nouveau, on peut illustrer algorithmiquement la méthode (on parle de PRGA pour Pseudo-random Generation Algorithm):

```
i, j=0;
WHILE génération DO // tant qu'il y'a un octet à chiffrer
i = (i+1) mod 256;
j=(j+S[i]) mod 256;
SWAP (S[i], S[j]);
t = (S[i] + S[j]) mod 256;
k = S[t];
```

C. Chiffrement proprement dit

La valeur de k est alors utilisée pour le chiffrement (avec le prochain octet de texte clair).

D. Déchiffrement

La même valeur de k est utilisée pour le déchiffrement ($\oplus$ avec le prochain octet de texte chiffré).Ceci se justifie par le fait que$\oplus$ est une loi de groupe qui est son propre inverse.

2.4 Conclusion

En résumé, ce chapitre a présenté en détail les principaux algorithmes de cryptographie à clé secrète ou symétrique tels que DES, Triple DES, IDEA, AES, RC4. Bien que des algorithmes plus anciens comme DES et IDEA montrent aujourd'hui des limites, l'AES reste l'algorithme de référence actuel pour le chiffrement robuste des données grâce à sa solidité face aux attaques connues. La compréhension du fonctionnement de ces algorithmes symétriques largement utilisés pour assurer la confidentialité des échanges est essentielle, dans le chapitre suivant nous présentons et discutons les résultats de notre étude sur la performance des algorithmes de cryptographie. Nous évaluons ces algorithmes selon plusieurs critères, en analysant leur efficacité et leur sécurité pour le chiffrement des images.

Chapitre 3 : Résultats et discussion

3.1 Introduction

Après avoir présenté dans les chapitres précédents les différents algorithmes de chiffrement et les concepts fondamentaux de la cryptographie, nous entamons maintenant la partie pratique de ce mémoire, consacrée à l'implémentation et à l'analyse des performances de ces algorithmes en cryptage d'images.

Dans ce chapitre, nous mettons en œuvre trois algorithmes de chiffrement largement utilisés DES, AES et RC4 pour assurer la confidentialité des images numériques. L'objectif est d'évaluer les performances et l'efficacité de ces algorithmes dans le contexte spécifique du cryptage d'images.

3.2 Critères d'évaluation

Pour démontrer la sécurité et l'efficacité de l'algorithme de chiffrement proposé face aux attaques connues, nous allons évaluer ses performances en utilisant différentes mesures. Nous nous concentrerons sur les critères d'évaluation les plus importants, à savoir les tests statistiques tels que l'histogramme, la corrélation et l'entropie, ainsi que les tests différentiels comme le NPCR (Number of Pixels Change Rate) et l'UACI (Unified Average Changing Intensity). De plus, nous analyserons la taille de l'espace des clés et la sensibilité de l'algorithme aux changements de clé, qui sont des facteurs cruciaux pour résister aux attaques par force brute et assurer une diffusion adéquate. En examinant ces critères, nous pourrons déterminer si l'algorithme proposé offre un niveau de sécurité suffisant contre les différents types d'attaques connus, tout en maintenant une efficacité acceptable pour le chiffrement d'images.

A. Corrélation

La corrélation mesure la relation linéaire entre les pixels de l'image originale et de l'image chiffrée. Elle est calculée en utilisant le coefficient de corrélation de Pearson, qui varie entre -1 et 1. Une valeur proche de 0 indique l'absence de relation linéaire statistiquement significative entre les deux images, ce qui est souhaitable pour un bon algorithme de chiffrement.

Formule du coefficient de corrélation de Pearson :

$$r = \frac{\sum[(x-\mu x)(y-\mu y)]}{\sqrt{[\sum(x-\mu x)^2 \times \sum(y-\mu y)^2]}} \qquad (3.1)$$

Où x et y représentent les valeurs des pixels correspondants dans l'image originale et l'image chiffrée, respectivement, et μx et μy sont les moyennes de ces valeurs.

B. Entropie

L'entropie mesure le degré d'aléatoire dans la distribution des pixels de l'image chiffrée. Elle est calculée en utilisant la formule de l'entropie de Shannon :

$$H(s) = \sum p(si) \log 2 \left(\frac{1}{p(si)}\right) \qquad (3.2)$$

Où $p(si)$ représente la probabilité d'occurrence du niveau de gris si dans l'image chiffrée.

Pour une image en niveaux de gris de 8 bits, la valeur maximale d'entropie est de 8, ce qui correspond à une distribution parfaitement aléatoire des pixels.

C. NPCR (Number of Pixels Change Rate)

Le NPCR mesure la proportion de pixels modifiés dans l'image chiffrée lorsqu'un seul pixel est modifié dans l'image originale [14]. Il est calculé en comparant deux images chiffrées correspondant à deux images originales différant d'un seul pixel, en utilisant la formule suivante :

$$NPCR = \sum \frac{D(i,j)}{(W \times H)} \times 100 \quad (3.3)$$

Où D(i, j) est une fonction qui renvoie 0 si les pixels aux coordonnées (i,j) sont identiques dans les deux images chiffrées, et 1 sinon. W et H sont la largeur et la hauteur des images, respectivement.

D (i, j) est défini comme suit:

$$D(j,i) = \begin{cases} 0 & if \quad C_1(i,j) = C_2(i,j) \\ 1 & if \quad C_1(i,j) \neq C_2(i,j) \end{cases} \quad (3.4)$$

Une valeur de NPCR proche de 100% indique une bonne propriété de diffusion, ce qui signifie que la moindre modification dans l'image originale entraîne un changement important dans l'image chiffrée

D. UACI (Unified Average Changing Intensity)

L'UACI mesure l'intensité moyenne des changements de pixels entre deux images chiffrées correspondant à des images originales différant d'un seul pixel [14]. Il est calculé en utilisant la formule suivante :

$$UACI = \sum \frac{|C_1(i,j) - C_2(i,j)|}{(W \times H \times 255)} \times 100 \quad (3.5)$$

Où $C_1(i,j)$ et $C_2(i,j)$ représentent les valeurs des pixels aux coordonnées (i, j) dans les deux images chiffrées, et 255 est la valeur maximale d'un pixel en niveaux de gris de 8 bits.

Une valeur d'UACI proche de 33,33% pour des images en niveaux de gris indique une bonne capacité de diffusion de l'algorithme de chiffrement.

E. Histogramme

L'histogramme représente la distribution des niveaux de gris dans l'image chiffrée. Une distribution uniforme de l'histogramme est souhaitable pour résister aux attaques statistiques. L'histogramme représente la distribution des niveaux de gris (ou des valeurs de pixels) dans une image. Pour une image en niveaux de gris codée sur 8 bits, il y a 256 niveaux de gris possibles (de 0 à 255).

3.3 Schéma de chiffrement

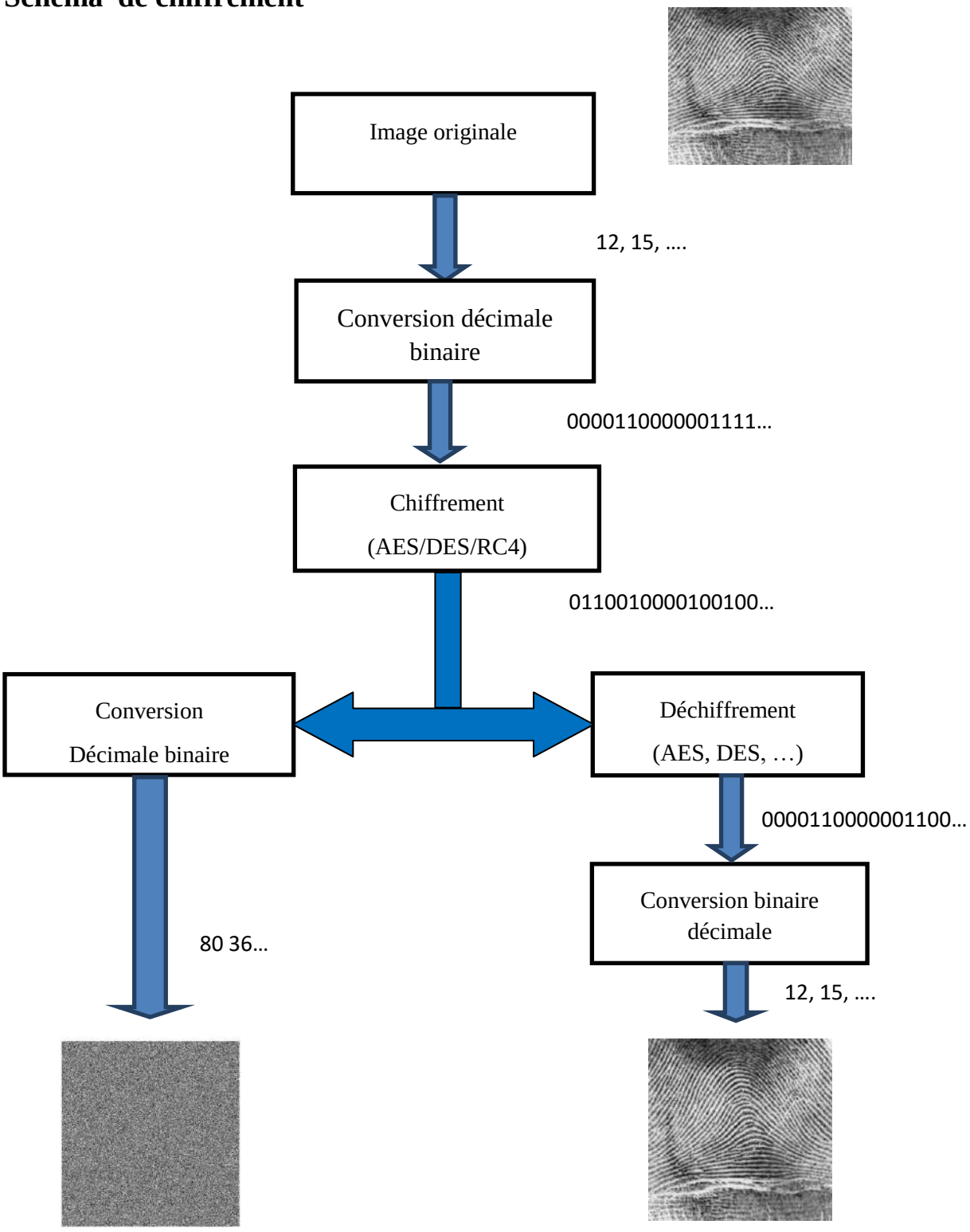


Figure 3.1: Schéma de chiffrement

3.4 Résultats expérimentaux

Pour la mise en œuvre pratique du cryptage des images, nous avons utilisé le logitiel MATLAB, un environnement de calcul numérique et de programmation largement utilisé pour le traitement d'images et les applications de cryptographie. MATLAB offre une multitude de fonctions et d'outils qui facilitent la manipulation des images ainsi que l'application des algorithmes de cryptage et de décryptage. Les algorithmes DES, AES et RC4 ont été implémentés et testés à l'aide des scripts MATLAB, permettant ainsi une évaluation rigoureuse de leur performance et efficacité en termes de temps de traitement et de sécurité. Les images à niveaux de gris populaires : Lena, Barbara, Aerial, et Fingerprint, sont employés dans ce travail.

3.4.1 Algorithme DES

Images	Nombre de rondes	Entropie	NPCR	UACI	Temps
Lena	16	7.976	99.606	34.246	490.710
	8	7.981	99.599	33.479	286.412
	4	7.980	99.579	33.932	180.793
Barbara	16	7.991	99.632	34.049	548.013
	8	7.992	99.625	33.984	336.554
	4	7.991	99.608	34.047	181.186
Aerial	16	7.998	99.603	33.518	735.939
	8	7.999	99.623	33.566	312.521
	4	7.997	99.608	33.580	174.241
Fingerprint	16	7.999	99.590	33.517	509.101
	8	7.999	99.580	33.454	278.458
	4	7.999	99.604	33.531	179.576

Tableau 3. 1: Résultats de chiffrement de l'algorithme DES

Le tableau 3.1 expose les résultats de l'algorithme DES. Nous pouvons remarquer facilement sur ce tableau que les valeurs d'entropie sont proches de 8 pour toutes les images et tous les nombres de rondes, ce qui suggère une bonne distribution aléatoire des pixels dans les images chiffrées.

Les valeurs de NPCR sont très proches de 100% pour toutes les images et tous les nombres de rondes, ce qui indique que la moindre modification dans l'image originale entraîne un changement important dans l'image chiffrée.

Les valeurs de l'UACI sont généralement comprises entre 33% et 34% pour toutes les images et tous les nombres de rondes, ce qui suggère une bonne capacité de diffusion de l'algorithme de chiffrement.

La figure représente les valeurs de l'histogramme ne sont pas fournies pour les résultats avec 16 et 8 rondes, mais pour 4 rondes, elles semblent être similaires pour toutes les images, ce qui indique une bonne uniformité de la distribution des niveaux de gris dans les images chiffrées.

Dans l'ensemble, a l'exception de la rapidité d'exécution, ces résultats suggèrent que l'algorithme de chiffrement utilisé a de bonnes performances en termes de diffusion, de confusion et de résistance aux attaques statistiques pour toutes les images testées, indépendamment du nombre de rondes utilisées.

3.4.2 Algorithme AES

Le tableau 3.2 expose les résultats de chiffrement de l'algorithme AES. Les valeurs d'entropie sont également très proches de 8 (maximum théorique pour des images à 8 bits). Cela suggère que les images chiffrées ont une distribution de pixels aléatoire, ce qui est souhaitable pour un algorithme de chiffrement sécurisé.

images	Nombre de rondes	Entropy	NPCR	UACI	Temps (sec.)
Lena	8	7.999	99.620	33.545	131.818
	4	7.9993	99.630	33.789	82.437
	2	7.9993	99.617	33.894	58.389
Barbara	8	7.9993	99.626	33.670	128.773
	4	7.9993	99.608	33.697	80.970
	2	7.9993	99.591	33.797	58.333
Aerial	8	7.9993	99.606	33.437	127.413
	4	7.9993	99.620	33.449	81.160
	2	7.9993	99.630	33.643	58.851
Fingerprint	8	7.9993	99.611	33.519	131.416
	4	7.9993	99.618	33.538	81.191
	2	7.9993	99.595	33.412	61.663

Tableau 3. 2 : Résultats de chiffrement de l'algorithme AES

Les valeurs de NPCR sont très proches de l'idéal théorique de 99,609% pour un chiffrement parfait, indiquant une grande différence entre les images originales et chiffrées. Les valeurs de UACI sont également élevées, autour de 33%, ce qui signifie que les intensités des pixels changent considérablement après le chiffrement.

La figure 3.2 représente les images, les images chiffrées, et les histogrammes correspondants. Un bon algorithme de chiffrement devrait produire des histogrammes uniformes pour les images chiffrées, ce qui indiquerait une distribution aléatoire des pixels.

Dans l'ensemble, ces résultats suggèrent que l'algorithme de chiffrement AES fonctionne bien pour ces images, produisant des images chiffrées avec une entropie élevée, des taux de changement de pixels et d'intensité élevés, ce qui indique un chiffrement sécurisé et efficace.

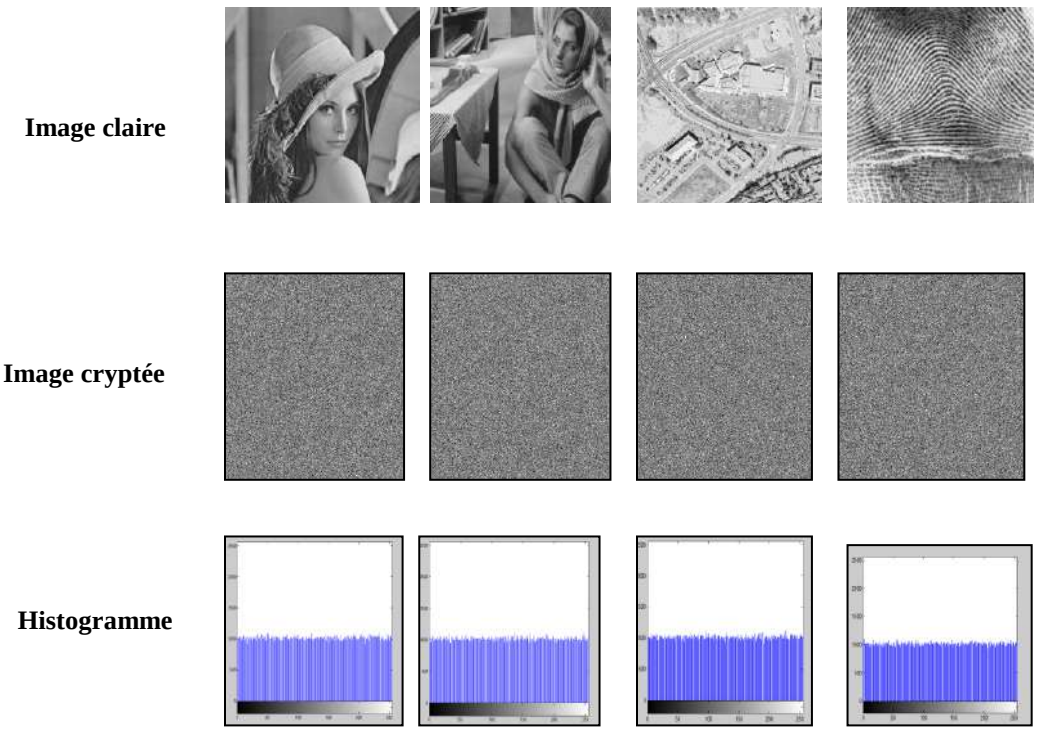


Figure 3. 2: Images cryptées par AES et les histogrammes correspondants

Comparaison entre DES et AES

A partir des tableaux 3.1 et 3.2, nous avons constaté que AES atteint une entropie de 7,9993 pour toutes les images à 2 rondes, surpassant DES qui nécessite 16 rondes pour s'approcher de cette valeur (valeurs entre 7.976 et -7.9994). En termes de NPCR, AES (99,617% - 99,630%) à 2 et 4 rondes) surpasse DES (99,578% - 99,608% à 4 rondes), montrant une meilleure sensibilité aux changements de pixels.

En terme de UACI, pour l'AES, nous avons enregistré une variation entre de 33,415% à 33,894% pour seulement 2 rondes, ce qui est généralement supérieur à DES 33,517% à 33,932%, 16 rondes.

En résumé, AES surpasse clairement DES dans tous les aspects, même avec beaucoup moins de rondes. AES à 2 ou 4 rondes offre une meilleure sécurité que DES à 16 rondes.

3.4.3 Algorithme RC4

A partir du tableau 3.3 et tableau 3.2, nous avons constaté que le temps d'exécution du RC4 est tés faible par rapport à l'AES. En plus, les valeurs de l'entropie obtenus avec les deux méthodes RC4 et AES sont très proches et sont presque parfaites (proches de 8).

Images	Entropie	NPCR	UACI	Temps
Lena	7.999	99.737	34.0142	4.761
Barbara	7.999	99.727	33.4839	4.640
Aerial	7.999	99.683	32.2512	4.732
Fingerprint	7.999	99.643	33.3765	5.644

Tableau 3. 3 : Résultats de chiffrement de l'algorithme RC4

Pour les deux métriques NPCR et UACI, nous avons obtenu un léger avantage de l'AES par rapport à RC4.

3.4.4 Algorithmes hybrides AES-RC4 et RC4-AES

La figure 3.2 représente le schéma de chiffrement hybride AES-RC4. Le tableau 3.4 expose les résultats de chiffrement correspondants [15].

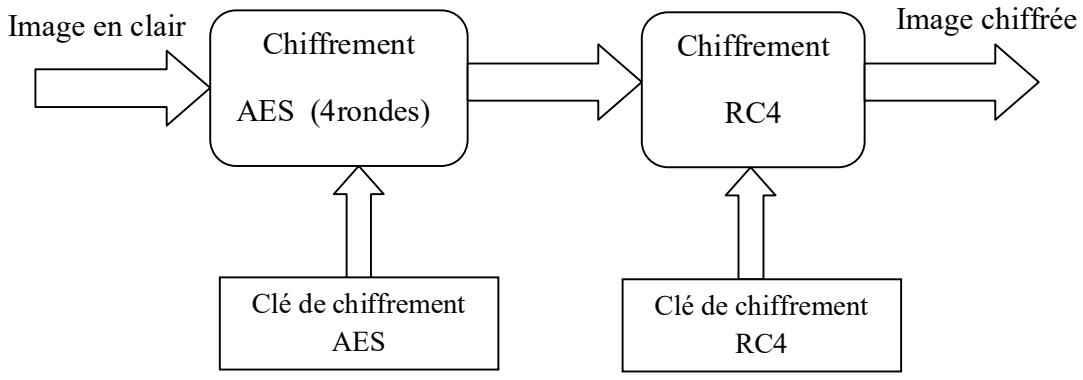


Figure 3. 1 : Algorithme de chiffrement hybride AES-RC4

Images		Entropie	NPCR	UACI	Temps (sec.)
Lena	AES-RC4	7.999	99.620	33.484	89.659
	AES	7.999	99.620	33.545	131.819
Barbara	AES-RC4	7.999	99.620	33.499	86.438
	AES	7.999	99.626	33.670	128.773
Aerial	AES-RC4	7.999	99.6201	3.526	80.467
	AES	7.999	99.606	33.437	127.413
Fingerprint	AES-RC4	7.999	99.6201	33.521	79.627
	AES	7.999	99.611	33.519	131.416

Tableau3.4 : Résultats de chiffrement de l'algorithme AES- RC4

La figure 3.3 représente le schéma de chiffrement hybride RC4-AES. Le tableau 3.5 expose les résultats de chiffrement correspondants.

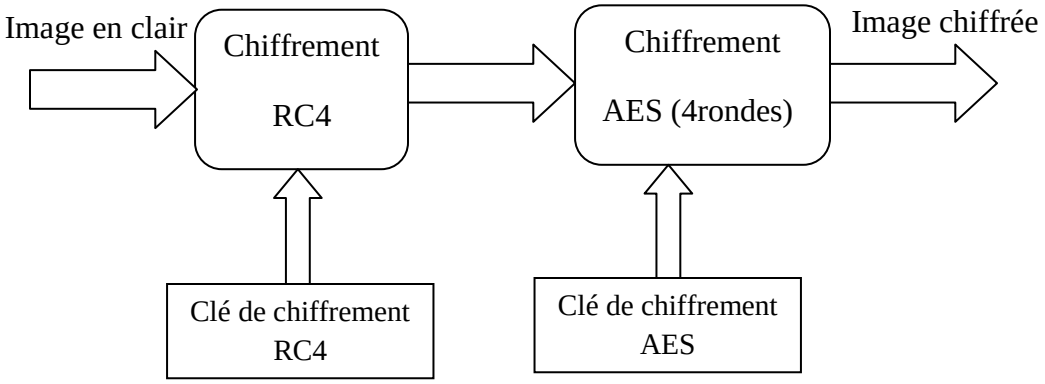


Figure 3. 3: Schéma de chiffrement hybride RC4-AES

Images		Entropie	NPCR	UACI	Temps (sec.)
Lena	AES-RC4	7.999	99.617	33.447	83.295
	AES	7.999	99.620	33.545	131.819
Barbara	AES-RC4	7.999	99.6120	33.456	84.839
	AES	7.999	99.626	33.670	128.773
Aerial	AES-RC4	7.999	99.6109	33.415	86.7593
	AES	7.999	99.606	33.437	127.413
Fingerprint	AES-RC4	7.999	99.609	33.421	84.725
	AES	7.999	99.611	33.519	131.416

Tableau 3. 5 : Résultats de chiffrement de l'algorithme RC4-AES

Les résultats montrent que pour la plupart des images, l'algorithme AES et les algorithmes hybrides AES-RC4 et RC4-AES atteignent de bonnes valeurs pour l'entropie et optimales (proches de 8). Pour les deux métriques NPCR et UACI, les résultats de l'AES et l'algorithme hybride AES-RC4 sont proches. Cependant, les algorithmes hybrides sont caractérisés par un temps d'exécution largement plus faible par rapport à celui de l'AES.

Dans l'ensemble, les deux algorithmes hybrides semblent fournir un bon niveau de sécurité, avec un bon compromis entre sécurité et vitesse.

3.5 Conclusion

Dans ce chapitre nous avons évalué et comparer les performances des algorithmes de chiffrement DES, AES et RC4 dans le contexte spécifique du cryptage d'images numériques. En employant divers critères d'évaluation tels que l'entropie, la corrélation, le NPCR, l'UACI, ainsi que le temps d'exécution, nous avons pu mesurer l'efficacité et le niveau de sécurité relative de chaque algorithme. Malgré que l'AES reste la référence en termes de robustesse et de sécurité des données, les résultats montrent que les algorithmes hybrides comme AES-RC4 offrent des temps d'exécution plus rapides. Ainsi, ce chapitre confirme l'importance de choisir l'algorithme de chiffrement approprié en fonction des exigences spécifiques de sécurité et de performance dans le domaine du cryptage d'images.

Conclusion générale

Cette étude comparative des performances des méthodes de cryptographie à clé privée, à savoir, DES, AES, et RC4 a permis d'apporter un éclairage significatif sur l'efficacité et la sécurité de ces algorithmes en cryptographie d'images. De plus, l'exploration d'approches hybrides combinant AES (4 rondes) et RC4 dans différents ordres a ouvert de nouvelles perspectives prometteuses.

Nos expérimentations ont révélé que chaque algorithme présente des forces et des faiblesses distinctes. DES, bien que historiquement important, a montré ses limites en termes de sécurité face aux capacités de calcul modernes. AES, avec sa robustesse cryptographique supérieure, s'est imposé comme le choix le plus sûr, particulièrement dans sa version standard. Cependant, cette sécurité accrue s'accompagne d'un coût de calcul plus élevé, ce qui peut être problématique pour le traitement d'images volumineuses. RC4, quant à lui, a démontré son excellente performance en termes de vitesse, le rendant attrayant pour les applications en temps réel, mais sa sécurité a été remise en question par des vulnérabilités connues.

L'innovation majeure de ce travail réside dans l'exploration d'approches hybrides. La combinaison d'AES réduit à 4 rondes avec RC4 visait à exploiter la rapidité de RC4 et la robustesse d'AES, tout en minimisant leurs faiblesses respectives. Nos résultats ont montré que ces combinaisons offrent effectivement un équilibre intéressant. La séquence RC4 suivi d'AES (4 rondes) s'est particulièrement distinguée, offrant une sécurité supérieure à RC4 seul, tout en conservant une grande partie de sa rapidité. Cette approche s'avère prometteuse pour les applications nécessitant un traitement rapide sans compromettre la sécurité.

En conclusion, cette étude souligne qu'il n'existe pas de solution universelle en cryptographie d'images. Le choix de l'algorithme doit être guidé par le contexte spécifique : AES pour une sécurité maximale, RC4 pour la performance, et potentiellement des approches hybrides comme RC4 suivi d'AES (4 rondes) pour un compromis optimal. Notre travail ouvre également la voie à de futures recherches sur les algorithmes hybrides, suggérant que la combinaison judicieuse de différentes techniques pourrait être la clé pour répondre aux défis croissants de la sécurité des images dans notre monde numérique.

Références

[1] **J. Buchmann.** Introduction à la cryptographie, cours et exercices corrigés, Dunod, 2006.

[2] **M. BOUTORA, D. BENAMI.** Conception, Etude et Réalisation d'un Cryptosystème Hybride de Transmission d'Images. Mémoire de Master, UMMTO, 2015.

[3] **L. Koceila.** Conception et réalisation d'un cryptosystème hybride pour la transmission sécurisé d'image. Mémoire de master UMMTO, 2016.

[4] **M. Choumaissa, M. Ahlam.** Cryptage d'image par un signal unidimensionnel quelconque. Mémoire de master de Université Larbi Ben M'hidi Oum El Bouaghi, 2019.

[5] **B. Schneier.** Cryptographie appliquée. Algorithmes, protocoles et codes source en C. 2ième édition, Vuibert,1996.

[6] **A. François.** Théorie des nombres et Cryptographie. Université de Limoges cour de D.E.A.

[7] **I. Belkadi, A. narimen.** Cryptage d'image par considération des plans de bits des pixels séparément par ordre de leurs poids avec une clé publique de taille libre. Mémoire de Master de l'université larbi ben m'hidi Oum El Bouaghi, 2018.

[8] http://e-biblio.univ-mosta.dz/handle/123456789/19428 consulté le 30 mars 2024

[9] **H. Khouildat.** Méthode de cryptage d'image basée sur la permutation et la matrice de Householder. Thèse de l'université KASDI-MERBAH Ouargla, 2019.

[10] **J. KATZ, Y. LINDELL.** Introduction to modern cryptography. CRC press Lian, S Multimedeo content encryption: techniques and application: auerbach publication. 2014.

[11] **A. Said, A. Kahina**. Cryptographie et sécurité des Réseaux : Implémentation de l'AES sous MATLAB. 2008.

[12]CHAP1.6b_FLOT_RC4.pdf./pluginfile.php/13921/mod_resource/content/0/CHAP1.6b_FLOT_RC4.pdf.

[14] **Y. Wu, J. P. Noonan, S. Agaian**. NPCR and UACI randomness tests for image encryption. Cyber journals: multidisciplinary journals in science and technology, Journal of Selected Areas in Telecommunications (JSAT), 1(2), 31-38. 2011.

[15]https://fastercapital.com/fr/contenu/Cryptographie-a-cle-secrete---securisation-des-messages-avec-des-codes-d-authentification.html consulté 25 mai 2024

[16] **N. LOUZZANI**. Contribution à l'amélioration de la transmission sécurisée des images à base du chaos. Thèse de Doctorat, Université Badji Mokhtar-Annaba. 2022.

[17] **G. Simon.** Tutoriel de la cryptographie. Département des mathématiques et de l'informatique Université du Québec à Trois-Rivières, 2001/2002.

[18] **T. M. Hussein, T. Hayet.** Conception d'une architecture matérielle pour l'AES et implémentation sur circuit FPGA. Thése de doctorat, université mohamed bougara - boumerdes, 2016.