République Algérienne Démocratique et Populaire Ministère de l'enseignement supérieur et de la recherche scientifique
UNIVERSITE Dr. TAHAR MOULAY SAIDA
FACULTE : TECHNOLOGIE
DEPARTEMENT D'INFORMATIQUE

Rapport de projet de fin d'étude

Master informatique

***Option :* Sécurité Informatique et Cryptographie**

Thème :

Cryptographie à base d'ADN dans l'IoT: Vers Un code Qr sécurisé

Présenté par :
BENDANI Fatima Djihed
SERRAR Maroua

Encadré par :
Dr Benyahia Kadda

Année universitaire : 2021/2022

Remerciements

Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail.

Nous tenons à exprimer notre reconnaissance à Monsieur BENYAHIA KADDA qui a bien voulu diriger ce travail de recherche. Nous lui présentons nos vifs remerciements pour sa disponibilité et ses conseils pertinents qui ont aidé de façon très significative à l'amélioration de ce mémoire.

Ainsi, nous remercions pour leur soutien tant moral, spirituel et matériel, nos parents

sans laisser de côté nos Sœurs bendani imene et serrar chaimaa

A nos familles nos tante et cousines surtout et nos amis d'enfance hanane et fatima aussi nos amis qu'on s'est fait a l'Université qui par leurs prières et leurs encouragements, on a pu surmonter tous les obstacles.

Nos remerciements s'étendent également à tous nos enseignants durant les années des études.

Enfin, nous tenons à remercier tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail.

Djihed & Maroua

Tables des matières

Introduction générale **8**

Chapitre 1 : La cryptographie

I.1. Introduction **11**

I.2. Cryptographie **12**

I.3. Objectifs de la cryptographie **12**

I.3.1. La confidentialité **12**

I.3.2. L'authentification **13**

I.3.3. L'intégrité **13**

I.3.4. Non répudiation **14**

I.4 Crypto-système **15**

I.4.1 Les éléments d'un crypto-système **15**

I.4.2 Types de crypto-systèmes **16**

1.4.2.1 Cryptographie symétrique **16**

I.4.2.2.1 Exemples des méthodes symétriques **17**

1.4.2.2 Cryptographie asymétrique. **18**

I.4.2.2.1 Exemples des méthodes asymétriques **19**

I.4.2.3 Comparaison entre Cryptographie symétrique et Cryptographie asymétrique **20**

I.5 Principes de Kerckhoffs **21**

I.14. Conclusion **21**

Chapitre 2 : La cryptographie à base d'ADN

II.1 Introduction **23**

II.2 L'ADN **23**

II.3 Fonctions de l'ADN **23**

II.3 Structure de l'ADN **24**

II.4. Réplication de l'ADN **26**

II.5. Cryptographie à base d'ADN **27**

II.6 Présentation de quelques travaux de cryptographie à base d'adn **28**

Conclusion **30**

Chapitre III : Implémentation et résultats

III.1 Introduction **32**

III.2 Présentation de l'approche **32**

III.3 Le chiffrement **33**

III.3.1 Codage en binaire **35**

III.3.2 Division en blocs **35**

III.3.3 Codage de huffman **35**

III.3.4 Codage en ADN **36**

III.3.5 Extraction des clés **36**

III.3.6 Xor biologique **37**

III.3.7 Brouillage **37**

III.3.8 La transcription **38**

III.3.9.La Traduction (Translation) **39**

III.3.9 Codage en QR **41**

1.Qu'est ce qu'un code rQ ? **41**

2. Codage en Qr **42**

III.4 Le déchiffrement **45**

III.5 Exemple explicatif **46**

III.6 Expérimentations et résultats **49**

III.6.1Environnement de travail **49**

III.6.2Tests et résultats **51**

III.6.2.1 Evaluation du temps d'exécution **51**

III.6.2.2 Analyse de l'espace de la clé **52**

III.7 Conclusion **53**

Conclusion Générale **55**

Références Bibliographiques **57**

Liste des Figures

Figure I.1 Aperçu sur la cryptologie **11**

Figure I.2: Chiffrement/déchiffrement **12**

Figure I.3 : La confidentialité. **13**

Figure I.4 : L'authentification. **13**

Figure I.5 : L'intégrité **14**

Figure 1.6 Schéma générale d'un crypto-système **15**

Figure I.7 : Le schéma général de la crypto-système symétrique. **16**

Figure I.8 : Chiffrement de César. **17**

Figure I.10 : Algorithme principal du DES. **18**

Figure I.11 : Le schéma général de la crypto-système asymétrique. **19**

Figure II.2 : Structure de l'ADN **24**

Figure II.2 : Les quatre nucléotides **25**

Figure II.3 :Chromosome **26**

Figure II.4 : Génome **26**

Figure II.5 : Réplication semi-consercative de l'ADN **27**

Figure III.1 Le crypto-système proposé **32**

Figure III.2 L'algorithme de chiffrement **33**

Figure III.3 Le processus de chiffrement **34**

Figure III.4 la table ASCII 35

FigIII.5: Transformation bits→bases. **36**

FigIII.7 Brouillage **38**

FigIII.8 Extrait des boites de Brouillage 38

FigIII.9: La transcription. **39**

Figure III.10 : Tableau du code génétique 39

Figure III.11 : boite de traduction. 41

Figure III.12 Structure d'un code QR **42**

Figure III.13 L'algorithme de déchiffrement **44**

Figure III.14 Processus de déchiffrement **45**

Figure III.15 Code Qr du texte chiffré **47**

Figure III.16 Raspberry Pi3 **49**

Figure III.17 variation de temps d'exécution en fonction du texte en claire **52**

Liste des Tableaux

Tableau 1.1 : Cryptographie symétrique Vs Cryptographie asymétrique **20**

Tableau III.1 : Xor biologique. **37**

Tableau 3.1 temps de chiffrement/déchiffrement en fonction de la taille de texte en claire **52**

Résumé

Avec l'augmentation quotidienne des appareils connecté dans l'IOT.La principale préoccupation est la sécurité des données échangées entre l'expéditeur et le récepteur. De nombreuses approches ont été proposées pour améliorer les crypto-systèmes dont la cryptographie ADN. Dans ce mémoire, une méthode cryptographique à clé symétrique basée sur l'ADN a été proposée qui combine la cryptographie ADN et le codage de huffman pour créer un code Qr sécurisé.

L'algorithme a été implémenté sur un Raspberry pi3 et les résultats obtenus ont montré sa robustesse.

Absract

With the daily increase of devices connected in the IOT, the main concern is the security of the data exchanged between the sender and the receiver. Many approaches have been proposed to improve crypto-systems including DNA cryptography. In this thesis, a DNA-based symmetric key cryptographic method was proposed that combines DNA cryptography and Huffman coding to create a secure Qr code.

The algorithm was implemented on a Raspberry pi3 and the results obtained showed its robustness.

ملخص

مع الزيادة اليومية للأجهزة المتصلة في إنترنت الأشياء ، فإن الشاغل الرئيسي هو أمن البيانات المتبادلة بين المرسل والمستقبل. تم اقتراح العديد من الأساليب لتحسين أنظمة التشفير بما في ذلك تشفير باستعمال الحمض النووي. في هذه المذكرة ، تم اقتراح طريقة تشفير بمفتاح متماثل تعتمد على الحمض النووي والتي تجمع بين تشفير الحمض النووي وتشفير هوفمان لإنشاء كود Qr آمن.

تم تنفيذ الخوارزمية على Raspberry pi3 وأظهرت النتائج التي تم الحصول عليها متانة الخوارزمية المقترحة

Introduction Générale

De nos jours, on entend de plus en plus parler de l'Internet des Objets (IoT), c'est l'ensemble des objets qui peuvent communiquer sans fil. Dans l'environnement IoT, plusieurs messages sont envoyés par les éditeurs, sur des canaux de transmission qui seront lus par les abonnés via un serveur qui se charge de les relier.

L’internet des objets

L'Internet des objets (IoT) est le réseau d'objets physiques de la vie quotidienne, d'appareils électroménagers, de véhicules, d'appareils et de nombreux éléments électroniques embarqués, actionneurs, capteurs, logiciels, matériels et connexions qui assure la communication de données entre ces objets pour créer la possibilité d'une forte intégration du monde physique dans le système numérique. Cela se traduit par des avantages économiques, une efficacité améliorée et une diminution des efforts humains.

L'Internet des objets englobe un réseau Internet étendu au-delà des appareils standard, tels que les ordinateurs de bureau, les ordinateurs portables, les tablettes et les smartphones, à toute gamme d'appareils physiques et d'objets quotidiens comme présenté dans la figure 1.

Intégrés à la technologie, les appareils IoT peuvent communiquer et interagir sur Internet, et ils sont surveillés et contrôlés à distance

Figure 1 L’internet des objets

Avec l'accélération d'Internet et des technologies de réseautage, de jour en jour, les menaces augmentent également pour les utilisateurs, en raison du grand nombre d'objets connectés et du flux important d'informations entre ces objets. Par conséquent, pour garantir que les informations parviennent à l'expéditeur et au destinataire prévus, toutes les faiblesses des systèmes de sécurité doivent être surmontées.

Plusieurs travaux de recherche ont été menés autour de la sécurité des données dans l'environnement IoT tout en cherchant à répondre aux exigences de temps et de flux de données, le chiffrement fait donc partie de ces solutions.

La cryptographie ADN résout ces problèmes et donne l'espoir de développer des algorithmes incassables. Les données sont sécurisées à l'aide de séquences d'ADN pour créer un texte crypté qui ne peut être décrypté que si la clé ou la séquence correcte est connue. La majorité des travaux de recherche a montré que la cryptographie basée (ADN) est plus résistante aux attaques cryptographiques dans ces situations.

Dans ce présent travail, nous présenterons un algorithme de chiffrement à base d'ADN qui : 1- Utilise les séquences ADN pour l'extraction des clés de chiffrement des blocs

2- Utilise les séquences ADN pour l'extraction des codes des bases azotiques en appliquant le codage de Huffman afin de chiffrer les textes en claire.

3- Générer et lire des codes Qr sécurisés

Organisation de mémoire :

Notre mémoire est répartie en 3 chapitres ; Le premier chapitre c'est une introductions à la sécurité et aux techniques de cryptographie , symétrique et asymétrique.

Dans le deuxième chapitre , nous présenterons un état de l'art autour de la cryptographie à base d'ADN , nous dériverons l'ADN , sa structure et ses différents composant. Ensuite nous présenterons la cryptographie à base d'ADN avec une bref description de quelques travaux dans l'axe.

Le quatrième chapitre fait l'objet de la description en détail de notre algorithme et de l'ensemble des expérimentations réalisées.

Chapitre I

La cryptographie

I.1. Introduction

La cryptologie peut être considérée à la fois comme un art ancien et une science nouvelle. Un art ancien car depuis le temps de César, on utilisait déjà des méthodes artisanales comme le décalage alphabétique, les permutations ou les substitutions pour cacher des informations. Une science nouvelle car ce n'est que depuis les années 1970, à la naissance du chiffrement à clef publique, qu'elle est devenue un thème de recherche académique lié à beaucoup d'autres domaines scientifiques, en particulier à la théorie des nombres, à l'algèbre, à la complexité, à la théorie de l'information, aux codes correcteurs d'erreurs.

La cryptologie, étymologiquement la science du secret, englobe la cryptographie, l'art des écritures cachées, et la cryptanalyse dont le but n'est autre que d'attaquer les méthodes cryptographiques. **[1]**

Le préfixe « crypto » provient du grec « kryptos » qui signifié « caché » ou « secret » donc la cryptologie signifié la science du caché ou la science du secrets. **[2]** Se décompose en **cryptographie** et **cryptanalyse**.

La figure 1.1 donne la vue d'ensemble du champ cryptologie dans sa totalité. En cryptologie, Le message original à envoyer est appelé texte clair, tandis que le message codé est appelé texte chiffré. Le processus que nous utilisons pour convertir le texte brut en texte chiffré s'appelle le chiffrement et le processus qui convertit le texte chiffré en texte clair est appelé décryptage. De nombreux régimes ont été utilisés pour le chiffrement qui constituent un domaine d'étude connu sous le nom de cryptographie et ces schémas sont appelés systèmes cryptographiques ou chiffrements.

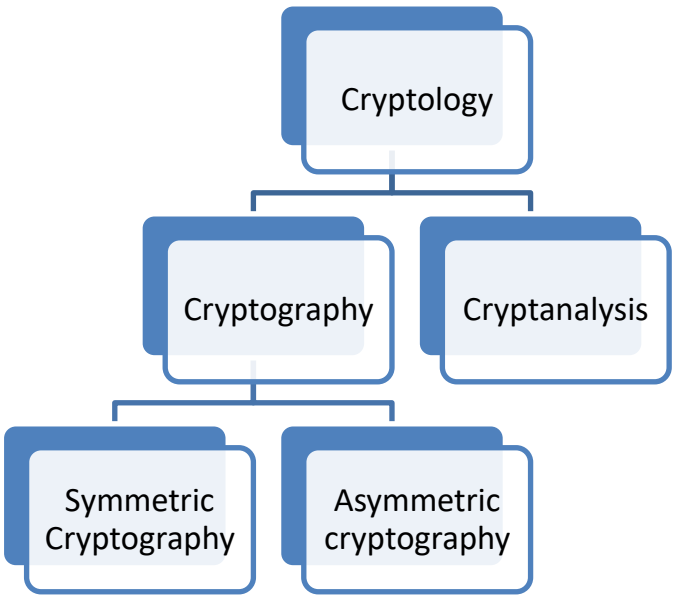


Figure I.1 Aperçu sur la cryptologie

Dans ce mémoire, nous nous concentrons uniquement sur la cryptographie. Dans ce qui suit nous présentons la cryptographie, son principe, les éléments d'un crypto-système et les deux types des crypto-systèmes (symétrique et asymétrique)

I.2. Cryptographie

Le mot vient du grec « crypto » qui signifié « caché » ou « secret », et « graphein » qui signifié « écriture », donc c'est une manière de masquer l'écriture tout en préservant un moyen de le retrouver**. [3]**

La cryptographie est la science qui consiste à écrire n'importe quel message dans un secret ou langage de code destiné à masquer le sens du message.

Le chiffrement consiste à transformer une donnée (texte, message, ...) afin de la rendre incompréhensible par une personne autre que celui qui a créé le message et celui qui en est le destinataire. La fonction permettant de retrouver le texte clair à partir du texte chiffré porte le nom de déchiffrement. **[4]**

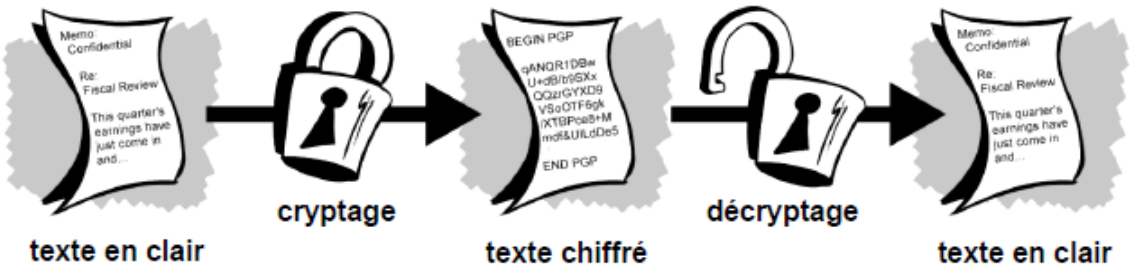


Figure I.2: Chiffrement/déchiffrement

I.3. Objectifs de la cryptographie

Le but principal de la cryptographie ancienne est d'élaborer une méthode permettant de transmettre des messages écrits sur un papier d'une manière confidentielle. Les objectifs de la cryptographie moderne sont plus complexes et plus nombreux, mais on peut en distinguer quatre principaux : la confidentialité, l'authentification, l'intégrité des données et la non répudiation. **[3]**

I.3.1. La confidentialité

C'est la propriété qu'une information n'est ni disponible ni divulguée aux personnes, entités ou processus non autorisés *(norme ISO 7498-2)* **[5]**. La confidentialité est le premier problème posé à la cryptographie, il se résout par l'opération de chiffrement. En effet, Si Bob et Alice s'échangent des informations secrètes en utilisant un algorithme de chiffrement, alors ils doivent avoir la

certitude que seul celui qui dispose de leur clef de déchiffrement peut déchiffrer ces messages.

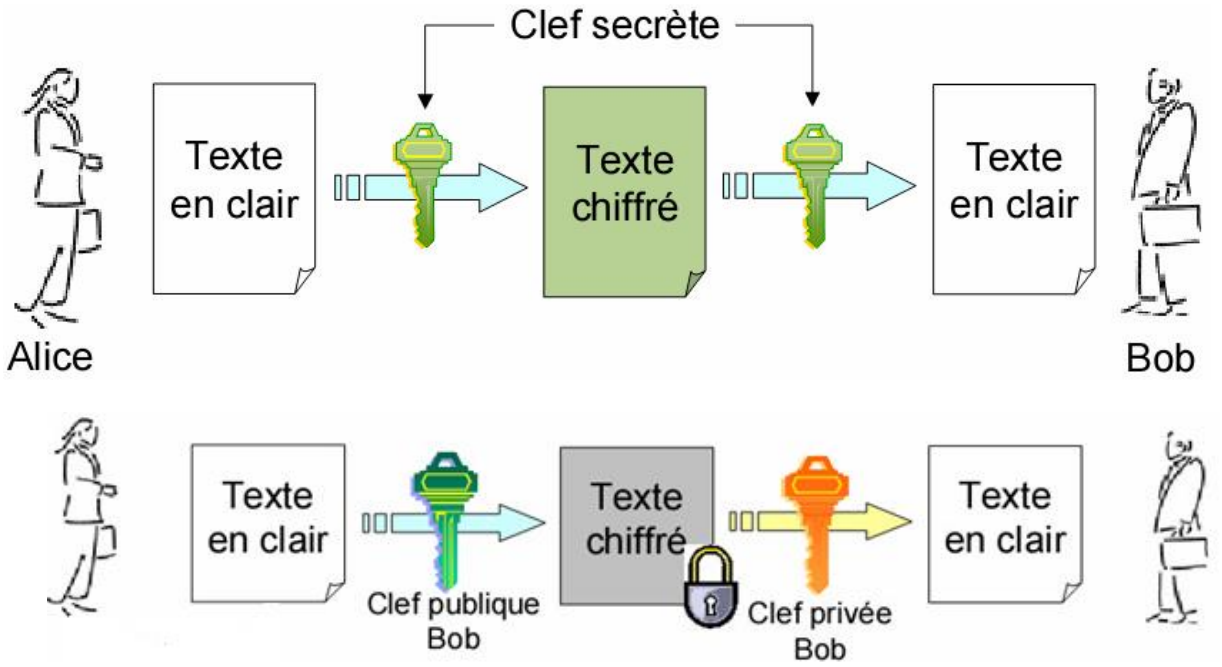


Figure I.3 : La confidentialité.

I.3.2. L'authentification

Le destinataire d'un message doit pouvoir s'assurer de son origine. Un intrus ne doit pas être capable de se faire passer pour quelqu'un d'autre.

L'émetteur est sûr de l'identité du destinataire c'est à dire que seul le destinataire pourra prendre connaissance du message car il est le seul à disposer de la clef de déchiffrement. Le receveur est sûr de l'identité de l'émetteur.**[5]**

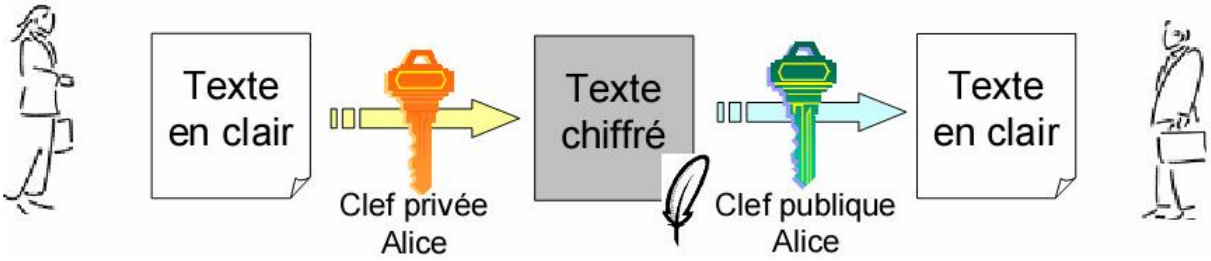


Figure I.4 : L'authentification.

I.3.3. L'intégrité

Le destinataire d'un message doit pouvoir vérifier que celui-ci n'a pas été modifié en chemin. Un intrus ne doit pas être capable de faire passer un faux message pour légitime.

L'intégrité du système et de l'information traitée garantit que ceux-ci ne sont modifiés que par une action volontaire et légitime. Lorsque l'information est échangée, l'intégrité s'étend à l'authentification du message, c'est à dire `a la garantie de son origine et de sa destination **[5]**

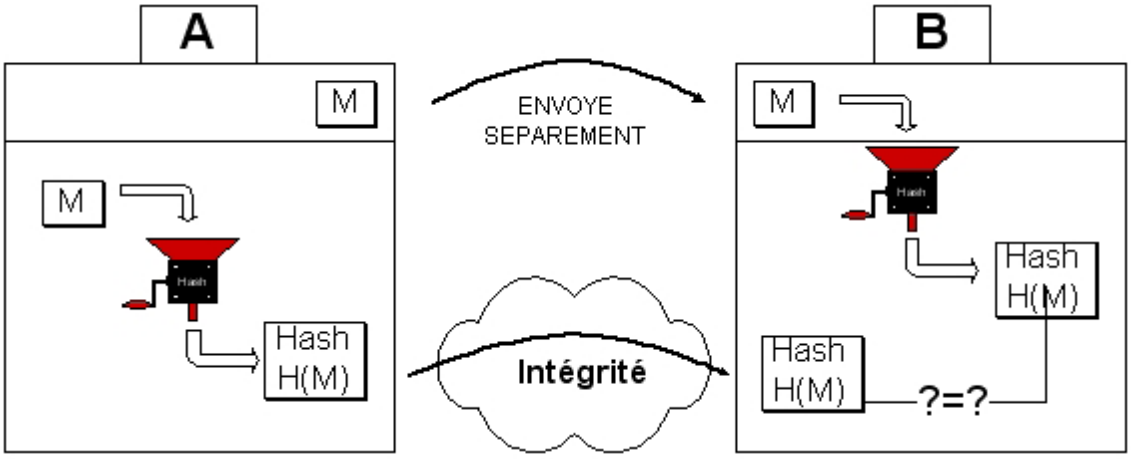


Figure I.5 : L'intégrité

I.3.4. Non répudiation

Un expéditeur ne doit pas pouvoir, par la suite, nier à tort avoir envoyé un message.

Non répudiation se décompose en trois:

- Non répudiation d'origine l'émetteur ne peut nier avoir écrit le message et il peut prouver qu'il ne l'a pas fait si c'est effectivement le cas.
- Non répudiation de réception le receveur ne peut nier avoir reçu le message et il peut prouver qu'il ne l'a pas recul si c'est effectivement le cas.
- Non répudiation de transmission l'émetteur du message ne peut nier avoir envoyé le message et il peut prouver qu'il ne l'a pas fait si c'est effectivement le cas. **[6]**

L'objectif fondamental de la cryptographie est de permettre à deux personnes appelées traditionnellement, **Alice** et **Bob** de communiquer à travers un canal peu sûr de telle sorte qu'un opposant passif **Ève** ne puisse pas comprendre ce qui est échangé et que les données échangées ne puissent pas être modifiées ou manipulées par un opposant actif **Martin**.

On peut regarder ces quatre qualités du point de vue de l'émetteur. **Alice** veut être certaine

- Qu'une personne non-autorisée **(Ève)** ne peut pas prendre connaissance des messages qu'elle envoie, **confidentialité**.
- Que ses messages ne seront pas falsifiés par un attaquant malveillant **(Martin), intégrité**.
- Que le destinataire **(Bob)** a bien pris connaissance de ses messages et ne pourra pas nier l'avoir reçu, **non répudiation**.

De plus elle veut être certaine que son message ne sera pas brouillé par les imperfections du canal de transmission (cette exigence ne relève pas du cryptage mais de la correction d'erreur).

Bob veut être certain

- Que personne d'autre que lui (et **Alice** bien sûr) n'a accès au contenu du message, **confidentialité**.
- Que le message reçu vient bien d'**Alice authentification**, par exemple qu'un attaquant malveillant **(Oscar)** ne puisse pas se faire passer pour **Alice**, **mascarade ou usurpation d'identité**
- Que le message n'a pas été falsifié par un attaquant malveillant **(Martin), intégrité des données**
- Que l'expéditeur **(Alice)** ne pourra pas nier avoir envoyé le message, **non répudiation. [6]**

I.4 Crypto-système

Un crypto-système est également appelé système de chiffrement. Il met en œuvre des techniques cryptographiques utilisant divers composants cryptographiques tels que le texte brut, l'algorithme de chiffrement, le texte chiffré, l'algorithme de déchiffrement et la clé de chiffrement pour fournir des services de sécurité de l'information.

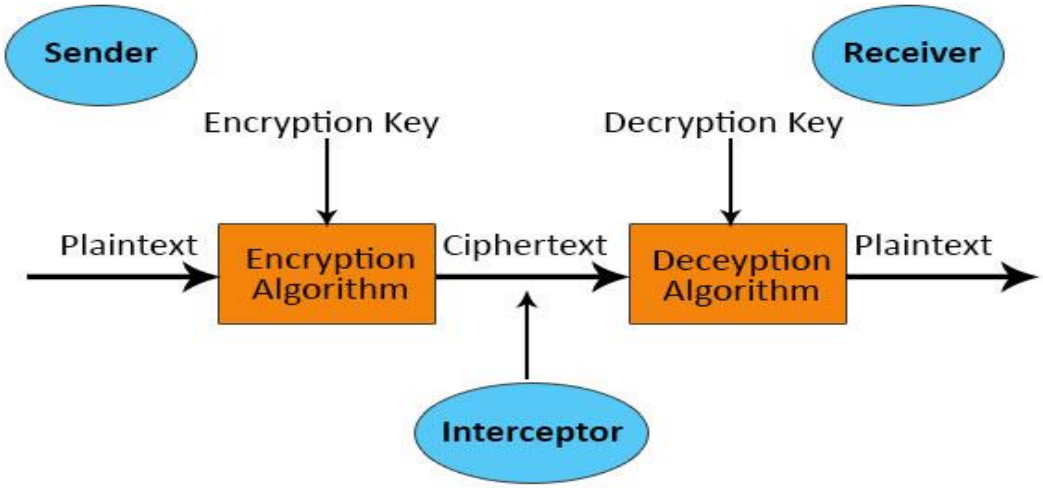


Figure 1.6 Schéma générale d'un crypto-système

I.4.1 Les éléments d'un crypto-système

1) Texte brut

Le texte brut est un message ou une donnée compréhensible par n'importe qui.

2) Texte chiffré

Le texte chiffré est un message ou des données qui ne sont pas lisibles ; il est accompli en exécutant l'algorithme de cryptage sur du texte brut à l'aide d'une clé de cryptage.

3) Algorithme de chiffrement

Il s'agit d'un processus de conversion de texte brut en texte chiffré à l'aide d'une clé de cryptage. Il faut deux entrées, c'est-à-dire du texte brut et une clé de chiffrement, pour produire un texte chiffré.

4) Algorithme de déchiffrement

C'est un processus opposé d'un algorithme de cryptage ; il convertit le texte chiffré en texte brut à l'aide de la clé de déchiffrement. Il faut deux entrées, c'est-à-dire le texte chiffré et la clé de déchiffrement, pour produire du texte brut.

5) Clé de chiffrement

Il s'agit d'une clé utilisée par l'expéditeur pour convertir le texte brut en texte chiffré.

6) Clé de déchiffrement

C'est une clé que le récepteur utilise pour convertir le texte chiffré en texte brut.

I.4.2 Types de crypto-systèmes

Il existe deux types de crypto-systèmes : le chiffrement à clé symétrique et le chiffrement à clé asymétrique. Discutons ces deux types en détail.

1.4.2.1 Cryptographie symétrique

De l'Antiquité jusqu'en 1976, la cryptographie utilisait exclusivement les méthodes symétriques. Le chiffrement symétrique est le type de chiffrement le plus simple et implique une seule clé pour chiffrer et déchiffrer les informations. Les algorithmes symétriques sont très rapides en termes de calcul. Ils sont classifiés en deux types :

1.Le chiffrement par flux :

Il se fait bit à bit sans attendre la réception entière des données, l'algorithme le plus connu est le RC4**.[7]**

2.Le chiffrement par bloc :

Consiste à diviser les données en blocs de taille fixe (64,128), chaque bloc ensuite sera chiffré**.[7]** Les algorithmes les plus connus sont : DES,3DES et AES.

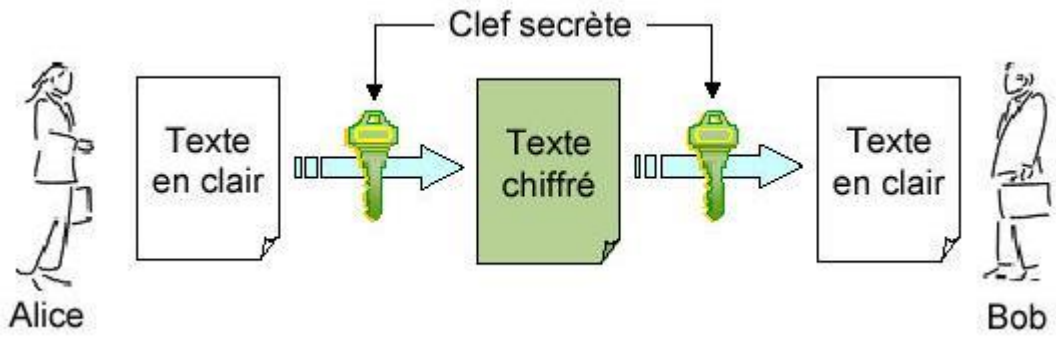


Figure I.7 : Le schéma général de la crypto-système symétrique.

I.4.2.2.1 Exemples des méthodes symétriques

A. Le chiffre de césar

La méthode de César est un système de chiffrement par substitution monoalphabétique. Ce procédé était utilisé dans les environs de 200 avant J.C et son fonctionnement consistait à décaler chaque lettre de l'alphabet par une autre de façon à rendre le message illisible. **[8]**

Le chiffre de César consiste simplement à décaler les lettres de l'alphabet de quelques crans vers la droite ou la gauche. Par exemple, décalons les lettres de 3 rangs vers la gauche, comme le faisait Jules César (d'où le nom de ce chiffre):

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
Chiffré	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

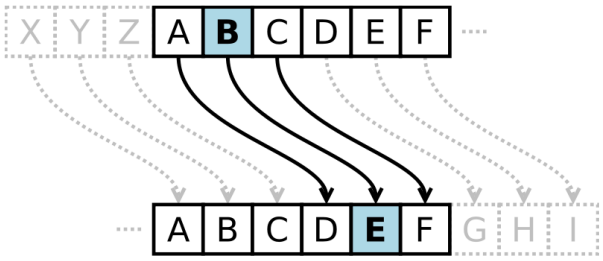


Figure I.8 : Chiffrement de César.

B. DES (Data Encryption Standard)

Le 23 Novembre 1976, le NIST (National Institute of Standards and Technology) adopte le standard de chiffrement DES]. Ce chiffrement conçu par IBM sous le nom de LUCIFER a été choisi par la NIST après quelques petites modifications. Ce chiffrement symétrique permet de chiffrer des messages de 64 bits avec une clef k de 56 bits. Pour chiffrer un texte, il faut d'abord le découper en blocs de 64 bits puis appliquer le chiffrement sur chacun des blocs. Ce procède est appelé mode de chiffrement par blocs. Ce chiffrement est constitue de 16 enchainements successifs d'opérations de transposition, de substitution et de chiffrement de Vernam. Les avancées matérielles en informatique permettent aujourd'hui, en un temps raisonnable de « casser » un message chiffre avec DES par « force brute », *i.e.* en testant toutes les clefs possibles grâce a une énumération exhaustive. En 1998, la NIST lança un appel d'offres pour choisir, l' « Advanced Encryption Standard » (AES), le successeur du DES, devenu trop sensible aux attaques par recherches exhaustives. **[8]**

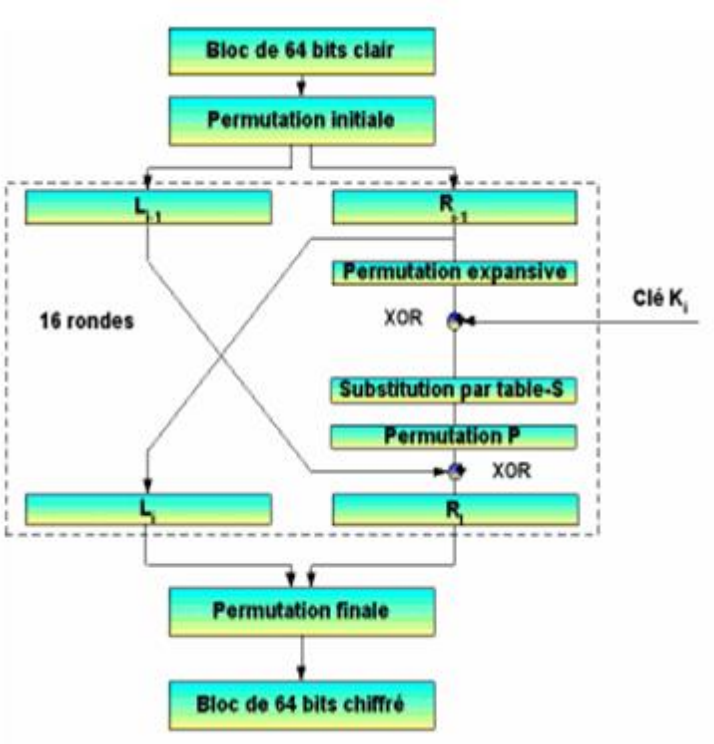


Figure I.10 : Algorithme principal du DES.

C. AES (Advanced Encryption Standard)

Le standard de chiffrement AES fut adopte en 2000 par le NIST en remplacement du DES. Le NIST nomma AES cet algorithme symétrique de Rijndael, conçu par Vincent Rijmen et Joan Daemen .Ce chiffrement est constitue de substitutions, de décalages, de «ou exclusif » et de multiplications dans un corps fini de polynômes fixes ; ces opérations sont élémentaires, simples et rapides a calculer. Il permet de crypter des blocs de 128, 192 ou 256 bits en utilisant des clefs symétriques de 128, 192 ou 256 bits. Le choix de la taille de la clef et de la taille des blocs sont indépendants, il y a donc au total 9 combinaisons possibles. Ceci laisse une plus grande flexibilité a l'utilisateur d'AES en fonction du niveau de sécurité et de la vitesse de calcul désires. **[9]**

1.4.2.2 Cryptographie asymétrique.

L'algorithme asymétrique dissocie les fonctions de chiffrement et de déchiffrement en deux clés. Ce que l'une chiffre, seule l'autre peut le déchiffrer. Aucune autre clé même celle qui a réalisé le chiffrement, ne peut y parvenir. Ainsi, chacun peut diffuser librement l'une de ses deux clés (dite publique) afin que n'importe qui puisse chiffrer un message à son attention. Seule la clé gardée secrète (dite privée) permet d'en prendre connaissance.

Un crypto-système à clef publique se comporte comme un coffre fort dont seul une personne possède la clef. Il laisse son coffre ouvert à disposition de toute personne désirant lui envoyer un message, celle ci referme lors la porte et seul le destinataire peut ensuite l'ouvrir. En pratique, le destinataire

publie à l'intention de ceux qui veulent lui envoyer des messages une méthode de chiffrement que lui seul est capable de déchiffrer. On voit donc bien pourquoi ces systèmes sont dits asymétriques. **[10]**

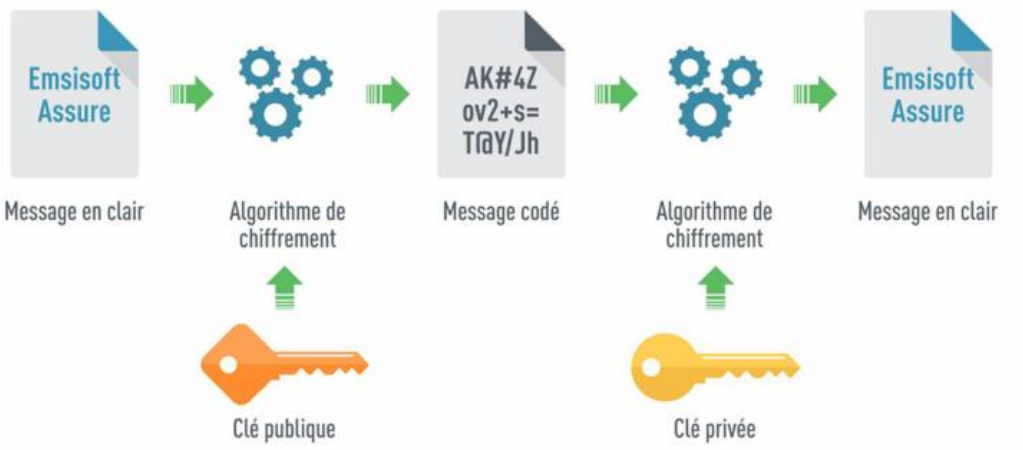


Figure I.11 : Le schéma général de la crypto-système asymétrique.

I.4.2.2.1 Exemples des méthodes asymétriques

A. L'algorithme RSA

L'algorithme RSA a été inventé en 1977 par Ron **R**ivest, Adi **S**hamir et Len **A**dleman. Au départ, les trois chercheurs voulaient prouver que tout système à clef publique possède une faille, mais ils ont terminé leurs travaux en inventant le système cryptographique à clef publique le plus utilisé jusqu'à nos jours. La clef utilisée est généralement de 1024 bits. **[11]**

Le principe de fonctionnement de cet algorithme est le suivant : Si Alice veut envoyer des messages à Bob, en utilisant le RSA, ils (Bob et Alice) procèdent de la façon suivante :

1) Bob génère quatre nombres **p**, **q**, **e**, et **d**, tel que :

- **p** et **q** sont deux grands nombres premiers distincts. Leur génération se fait au hasard, on pose **n** = **p*****q.**
- **e** est un entier premier avec le nombre (**p**-1)*(**q**-1).
- **d** est un nombre qui vérifie : **e*****d** ≡1 [(**p**-1) (**q**-1)], c'est-à-dire que le nombre (**e*****d**-1) est un multiple de (**p**-1)*(**q**-1).

2) Bob publie le couple **(n,e)** qui représente sa clef publique, et garde le couple **(n,d)** qui représente sa clef privée (**p** et **q** sont aussi gardés secrets).

3) Pour que Alice envoie un message à Bob, elle doit représenter ce message sous forme d'un ou plusieurs nombre **M**, tel que **M** < **n**. Elle calcule le nombre $\mathbf{C = M^e \bmod n}$. C'est ce dernier nombre qu'elle envoie à Bob.

4) Bob reçoit le message chiffré **C**, il calcul le nombre $\mathbf{D = C^d \bmod n}$. Il est démontré que $\mathbf{D = M^{e*d} \bmod n = M}$. donc, Bob à calculé le message initiale **M**. **[11]**

B. D'autres chiffrements asymétriques

Il existe de nombreuses méthodes de chiffrements asymétriques. Elles sont souvent rattachées à un problème mathématique difficile, comme le chiffrement de Rabin, invente en 1979, qui est base sur le problème difficile des racines carrées dans un corps fini. Citons une dernière méthode de chiffrement qui repose sur un problème difficile beaucoup plus complexe. Il s'agit de l'usage des courbes elliptiques en cryptographie propose en 1985 par MILLER.

I.4.2.3 Comparaison entre Cryptographie symétrique et Cryptographie asymétrique

Attribut	**Cryptographie symétrique**	**Cryptographie asymétrique**
Durée d'existence	Des milliers d'années	Moins de 50 ans
Utilisation actuelle	Chiffrement des données	Echange des clefs et signature numérique
Standard Actuel	Triple DES, AES	RSA, Diffie-Hellman, DSA
Vitesse de chiffrement/déchiffrement	Rapide	Lent
Clefs	*Secrète* (partagée généralement par deux personnes)	*Privée* : gardée secret. *Publique* : distribuée largement.
Echange de la clef	Transfert difficile et risqué	Simple et moins risqué (clef publique)
Longueur de la clef actuelle	128 bits 256 bits	1024 bits 2048 bits

Tableau 1.1 : Cryptographie symétrique Vs Cryptographie asymétrique

I.5 Principes de Kerckhoffs

En 1883 dans un article paru dans le Journal des sciences militaires, **[12]**, Auguste Kerckhoffs (1835-1903) posa les principes de la cryptographie moderne.

Ces principes et en particulier le second stipulent entre autre que la sécurité d'un cryptosystème ne doit pas reposer sur le secret de l'algorithme de codage mais qu'elle doit uniquement reposer sur la clef secrète du cryptosystème qui est un paramètre facile à changer, de taille réduite (actuellement de 64 à 2048 bits suivant le type de code et la sécurité demandée) et donc assez facile à transmettre secrètement.

Ce principe a été très exactement respecté pour le choix du dernier standard de chiffrement, l'algorithme symétrique AES, par le NIST. Ce dernier a été choisi à la suite d'un appel d'offre international et tous les détails de conception sont publics. Ce principe n'est que la transposition des remarques de bon sens suivantes:

•Un cryptosystème sera d'autant plus résistant et sur qu'il aura été conçu, choisi et implémenté avec la plus grande transparence et soumis ainsi à l'analyse de l'ensemble de la communauté cryptographique.

• Si un algorithme est supposé être secret, il se trouvera toujours quelqu'un soit pour vendre l'algorithme, soit pour le percer à jour, soit pour en découvrir une faiblesse ignorée de ses concepteurs. A ce moment l`a c'est tout le cryptosystème qui est à changer et pas seulement la clé.

Les systèmes conçus dans le secret révèlent souvent rapidement des défauts de sécurité qui n'avaient pas été envisagés par les concepteurs. **[12]**

I.14. Conclusion

Dans ce chapitre nous avons présenté les principaux aspects de la cryptographie, Nous avons décrire les composants d'un crypto-système avec ces deux types : symétrique et asymétrique, tout en expliquant leur fonctionnement à travers des exemples d'algorithmes cryptographiques anciens et modernes.

La cryptographie à base d'ADN qui est l'axe sur lequel notre sujet s'articule, Il utilise le calcul à l'ADN dans ces phases de chiffrement et déchiffrement, fait l'objet du deuxième chapitre suivant.

Chapitre II

La cryptographie à base d'ADN

II.1 Introduction

La cryptographie à l'ADN (DNA *cryptography*) est un nouvel axe de recherche en cryptographie. Grâce à sa capacité importante de stockage et le parallélisme massif dans le calcul, l'ADN peut être prometteur en cryptographie. Les méthodes cryptographiques à l'ADN utilisent l'ADN dans des laboratoires de haute technologie avec des équipements biologiques très sophistiqués, afin d'implémenter les fonctionnalités cryptographiques classiques (le chiffrement et le déchiffrement, l'authentification, la signature numérique…). Ces méthodes peuvent être plus efficaces en stockage, calcul et sécurité que les méthodes cryptographiques classiques. **[13]**

Dans ce chapitre, nous présenterons l'ADN, sa structure et ses caractéristiques. Ensuite la cryptographie à l'ADN, dans laquelle nous présenterons ce domaine et nous expliquerons comment les avantages de l'ADN sont exploités dans la cryptographique à l'ADN.

II.2 L'ADN

L'acide désoxyribonucléique (ADN) est une molécule biologique qui contient l'information génétique de tous les êtres vivants et qui définit leurs caractéristiques et fonctionnalités. C'est l'ADN qui rend chaque organisme différent de l'autre. L'ADN a été découvert pour la première fois en 1869 par le chimiste suisse Friedrich Miescher, mais le mérite de la découverte de l'ADN revient au biologiste américain James Watson et au physicien anglais Francis Crick en 1953 [14]. Selon Watson & Crick, chaque brin d'ADN comprend une structure en double hélice torsadée avec un squelette de phosphate de sucre. L'ADN est une molécule qui contient plusieurs nucléotides attachés au désoxyribose et chaque nucléotide contient trois composants : une base azotée, un sucre pentose (sucre à cinq carbones) et un ensemble de phosphate.[14]

II.3 Fonctions de l'ADN

L'ADN est le **support de l'hérédité**. Dans une molécule d'ADN, les nucléotides sont rangés dans un ordre précis à la manière des lettres de l'alphabet dans un texte. C'est cet ordre qui détermine l'information génétique.

Cette molécule assure également un certain nombre de fonctions au sein de la cellule. Elle gouverne notamment la **synthèse des protéines**. Pour ce faire, l'information contenue dans l'ADN est d'abord transférée à des molécules d'ARN qui servent de matrice pour produire les séquences d'acides aminés caractéristiques des protéines.

L'ADN permet aussi la **réplication des cellules** : quand une cellule doit se reproduire, elle se dédouble en se dupliquant. L'ADN de la cellule mère est reproduit à l'identique pour former celle de la cellule fille. **[MAN09]**

II.3 Structure de l'ADN

L'ADN a une structure en double hélice, elle est formée de quatre bases qui sont des pyrimidines. Il existe quatre bases azotées dans l'ADN, à savoir l'adénine (A), la cytosine (C), la guanine (G) et la thymine (T). L'adénine et la guanine sont appelées purines, tandis que la cytosine et la thymine sont appelées pyrimidines ; mais sont appariés comme Adénine avec Thymine et Cytosine avec Guanine selon la règle complémentaire de Watson-Crick. La paire Adénine-Thymine est liée par une double liaison hydrogène, tandis que le duo Cytosine-Guanine est lié par une triple liaison hydrogène.**[15]**. La structure de l'ADN d'une molécule est illustrée dans la figure II.1 ci-dessous

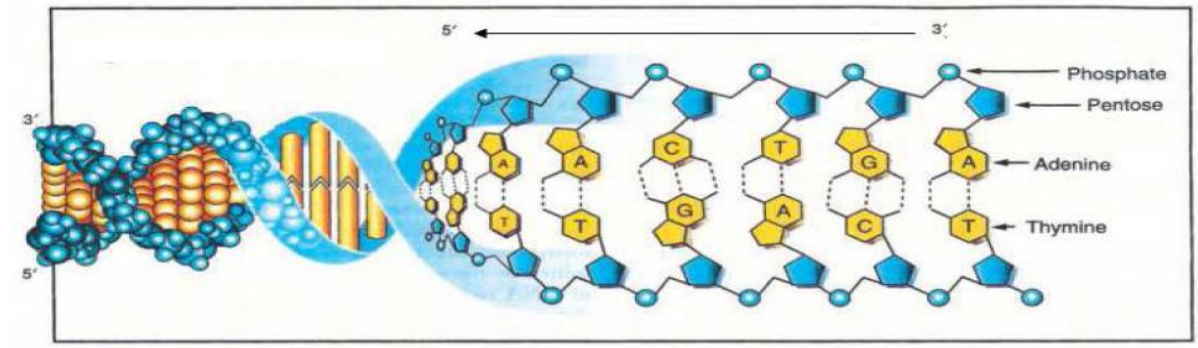


Figure II.2 : Structure de l'ADN

Un nucléotide est l'ensemble d'une base azotée, d'un sucre et d'un groupement de phosphate. Le sucre présent dans l'ADN est le *Désoxyribose*, ce sucre est relié à l'une des bases azotée (A, T, C, G). Si on enlève le groupement de phosphate au nucléotide, il devient un nucléoside. Un brin d'ADN est formé par la répétition ordonnée de ces nucléotides. **[MAN09]**

{« Sucre » + « Base Azoté »}⇒ ***Nucléoside***

{« Sucre » + « Base Azoté » + « groupement de phosphate »}⇒ ***Nucléotide***

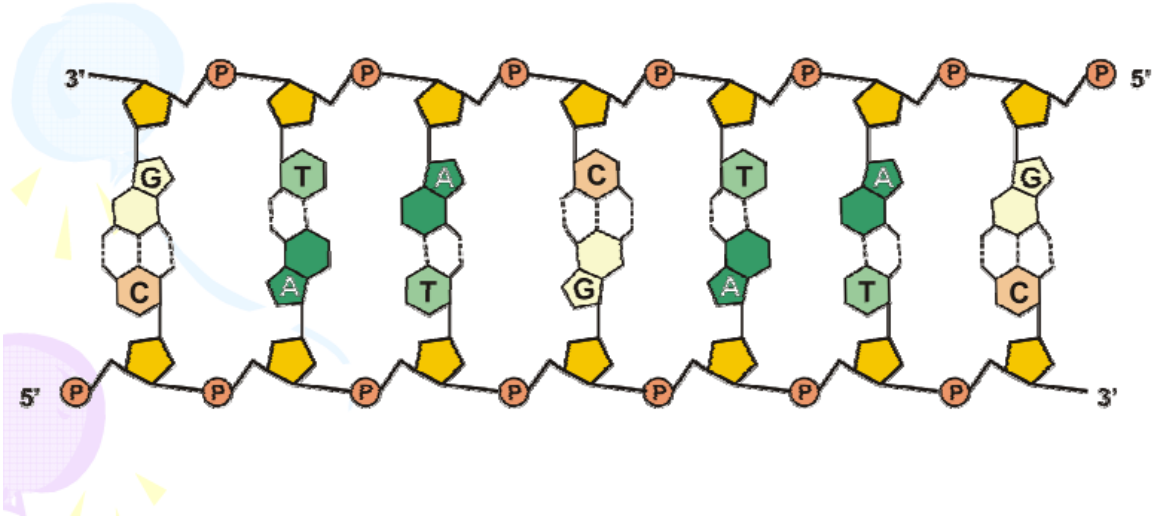


Figure II.2 : Les quatre nucléotides

Le mélange de plusieurs bases nucléotidiques forme ensemble un brin d'ADN. Lorsque les bases sont divisées en un groupe de trois, elles sont appelées codons. Enfin, la combinaison de plusieurs codons forme un gène, qui indique en fait à la cellule comment créer des protéines, qui expriment en fait une information génétique. Par conséquent, on peut dire que l'organisation de ces bases dans un certain ordre façonne des gènes uniques, et à travers les gènes, chaque être vivant se distingue des autres.[15]

Les chromosomes sont définis comme la grande structure organisée d'ADN enroulée autour de protéines qui englobent des gènes, d'autres séquences de nucléotides et des éléments de contrôle.

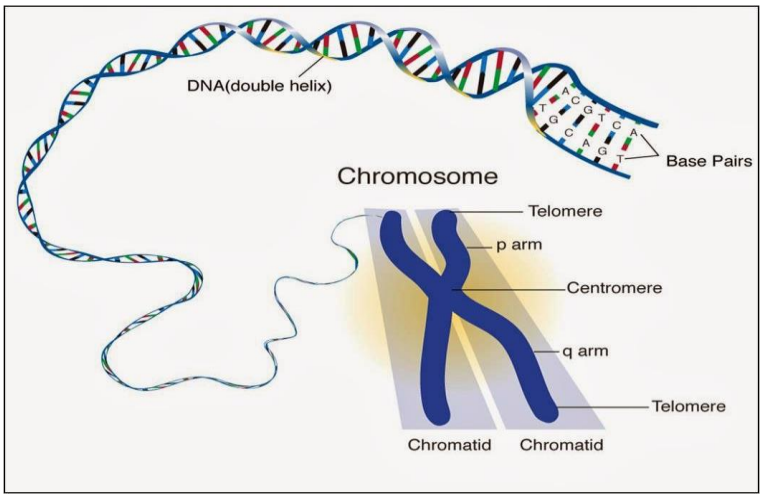


Figure II.3 : Chromosome

Une autre terminologie impérative de l'ADN est le génome, qui représente la séquence unique pour le contenu de l'ADN cellulaire de chaque organisme, c'est-à-dire les bases nucléotidiques, les gènes et les chromosomes.

Figure II.4 : Génome

II.4. Réplication de l'ADN

Le maintien de l'information génétique dans l'organisme et sa transmission sont assurés par la capacité de former deux molécules d'ADN ayant la même séquence, à partir d'une seule. Ce processus biochimique, appelé réplication, est fondé sur la propriété de complémentarité des bases A-T et G-C. Il fait intervenir un brin d'ADN, des bases A, G, T et C libres, et plusieurs enzymes catalysant cette réaction, les ADN polymérases.

La réplication de l'ADN a été démontrée par les expériences de *Meselson* et S*tahl*. Pour que la réplication puisse avoir lieu, les deux brins de la molécule d'ADN s'ouvrent, un peu comme une fermeture à glissière. Chaque brin est alors parcouru par une molécule d'une enzyme spécifique, l'ADN polymérase. Cette dernière synthétise un nouveau brin, complémentaire du premier, en accolant bout à bout des bases libres. A la fin du processus, au lieu d'une, il y a désormais deux molécules d'ADN : chacune est constituée d'un brin nouvellement formé et d'un brin ancien - la réplication est dite semi-conservative. **[16]**

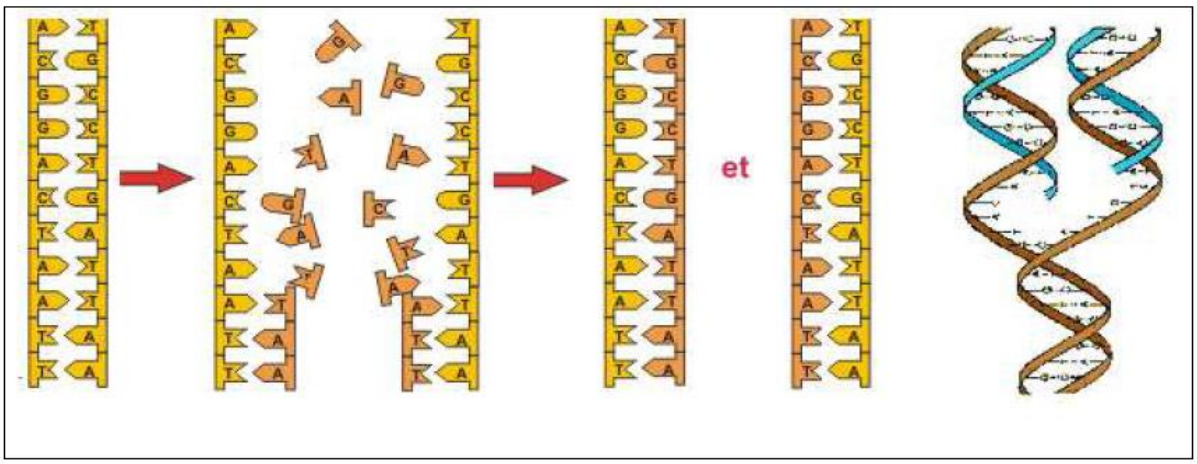


Figure II.5 : Réplication semi-consercative de l'ADN

II.5. Cryptographie à base d'ADN

La fusion des conceptions biologiques dans le cryptage s'appelle la cryptographie ADN. Au cours des dernières décennies, elle est devenue la technologie embryonnaire largement utilisée pour la sauvegarde des données cruciales. Il s'agit d'un nouveau domaine capable qui mélange les solutions cryptographiques conventionnelles et classiques au support génétique pour obtenir des chiffrements qui montrent une super force contre le attaques modernes. La forme la plus courante de la mise en œuvre est l'application d'opérations moléculaires sur des données suivies de processus conventionnels. Plus tard, les bases de données numériques d'ADN sont devenues courantes et les chercheurs ont commencé à les utiliser pour le cryptage de l'ADN en raison de leur facilité d'application et de leur large accessibilité.

Avantages :

Plusieurs motivations ont été à l'origine de l'apparition de la cryptographie à l'ADN, et ont permis à ce domaine de se propager dans plusieurs centres de recherche scientifique. Parmi ces motivations, on trouve :

1.En effet, actuellement plusieurs problèmes causés par les circuits à base de silicium sont apparus, comme par exemple, la distance entre les transistors qui s'approche de son seuil minimal, et le problème du bruit électromagnétique qui perturbe le bon fonctionnement de certains équipements électroniques

2. La capacité importante du stockage et le parallélisme massif de l'ADN Ont été les premières motivations de l'apparition de la cryptographie à l'ADN. En effet, un ordinateur à base d'ADN peut être 1 200 000 plus rapide qu'un ordinateur ordinaire et qu'un gramme d'ADN peut contenir 108 Téra octets. **[17]**

3.Le développement des outils de la biologie moléculaire ont permis un bon départ pour la cryptographie à l'ADN.

II.6 Présentation de quelques travaux de cryptographie à base d'adn

En 1994, Adleman [18] a introduit pour la première fois le concept de calcul de l'ADN en mettant en évidence les solutions aux «problèmes combinatoires» appliquant le calcul moléculaire tel que le problème du «chemin hamiltonien». Le graphique qui avait sept sommets a été codé en convertissant en arrangement moléculaire en vertu d'une procédure, après que les processus de calcul ont été réalisés par la technique de la force brute. Alors que le résultat d'occurrences à sept nœuds était insignifiant, c'était quand même la première expérience décrivant l'application de l'ADN dans l'informatique.

Lipton [19] en 1995 a détaillé les résultats d'Adleman en craquant un autre "problème complet NP" mieux connu sous le nom de "satisfaction" au moyen de fragments d'ADN dans un tube à essai pour convertir le graphique en nombres à 2 bits.

En 1997, Ouyang Qi, et. Al. [20] ont appliqué le concept de cryptographie de l'ADN pour créer l'explication du problème de la clique maximale, un problème « NP-complet ». Cela démontre l'efficacité de l'informatique ADN pour casser les problèmes difficiles .

En 2003, Jie Chen [21] a donné la méthode cryptographique de l'ADN fondée sur la théorie moléculaire et le tampon unique, réalisant ainsi l'encodage/décodage d'images bidimensionnelles. Les avantages de cette approche sont qu'elle peut stocker une grande

quantité de données dans un volume compact et qu'elle possède d'immenses compétences de traitement parallèle du calcul biomoléculaire.

En 2008, A.K. Verma, Mayank Dave et R.C. Joshi [22] a projeté une nouvelle approche du routage sécurisé dans les réseaux mobiles ad hoc (MANET). Ils ont utilisé la tactique de cryptoanalyse Quasi DNA pour sécuriser les réseaux Adhoc. L'approche de cryptographie du pseudo-ADN utilisée par eux a été construite sur le concept de biologie moléculaire dans lequel le texte d'entrée est stocké dans l'ADN puis transféré par transcription d'ARN, puis aux protéines par transformation, obtenant ainsi le texte chiffré. Le texte chiffré ainsi obtenu est ensuite envoyé sur le support sécurisé au récepteur .

En 2012, Yunpeng Zhang, Bochen Fu et Xianwei Zhang [23] ont défini un cryptosystème d'ADN utilisant l'assemblage de fragments d'ADN. Dans le protocole, il a été expliqué comment l'expéditeur traduit le texte d'entrée en une séquence binaire, puis en une chaîne d'ADN qui est à nouveau divisée en petites chaînes d'ADN. Dans les fragments, la clé de l'intégration de la chaîne courte se produit et est ensuite transférée au récepteur sous forme de texte codé. Le récepteur rassemble alors les fragments en les déchiffrant pour obtenir à nouveau le texte en clair. Ici, la longueur du texte chiffré est sécurisée et courte. Mais comme la dimension du fragment d'ADN est petite, l'intrus peut facilement l'identifier.

En 2013, Olga Tornea et Monica E. Borda [24] ont donné un code construit par ADN créé sur la base du schéma d'indexation de l'ADN. L'arrangement arbitraire de l'ADN obtenu à partir de la base de données génétiques a été utilisé comme clé à usage unique. Il est ensuite transféré au récepteur via un support de transmission sécurisé. Le processus d'encodage se produit en traduisant le texte d'entrée en son code ASCII correspondant, puis à nouveau en code binaire qui est finalement transformé en bases d'ADN. Maintenant, l'arrangement d'ADN ainsi conçu est recherché dans la séquence de clés et enregistre les numéros d'index. Le tableau de nombres entiers ainsi obtenu est le texte codé qui est déchiffré par le récepteur uniquement en utilisant la clé et le pointeur d'index.

En 2020, Benyahia et all , ont proposé un algorithme nommé « Stegano-DNA » à clé secrète agissant en bloc de 64 bits. Chaque bloc sera chiffré à l'aide d'une fonction de chiffrement qui produit un bloc chiffré de 64 bits à l'aide d'une clé de chiffrement composée de 4 parties. La phase de chiffrement est divisée en deux parties : la partie brouillage et la partie cryptage. Inspiré de plusieurs autres algorithmes déjà connus DES, ENIGMA et BOOK CYPHER.La clé de chiffrement est également divisée en deux parties : une partie choisie par le chiffreur et une partie générée par l'algorithme.[25]

En 2021 ; khabzaoui et all , ont proposé une méthode cryptographique à clé symétrique dans un environnement IoT basée sur l'ADN qui combine la cryptographie ADN et la clé d'échange Diffie-Hellman. Les sous-clés utilisées pour le chiffrement et le déchiffrement sont extraites d'une séquence de référence d'ADN sélectionnée au hasard à l'aide d'une graine secrète et continuellement modifiée partagée entre l'expéditeur et le destinataire selon l'algorithme d'échange de clés Diffie-Hellman qui conduit à une sécurité plus élevée, respectant les principes d'intégrité et de confidentialité.[26]

Conclusion

Dans ce chapitre, nous avons présenté l'ensemble des notions concernant l'ADN comme étant la base de la cryptographie ADN , sa définition et sa structure .Nous avons donné une vue sur la cryptographie à base d'ADN et ses avantages et Nous avons cité quelques travaux de chiffrement à bas d'adn. Nous proposons dans le chapitre suivant notre algorithme de chiffrement qui utilise les séquence d'ADN et le codage de huffman pour créer des codes qr sécurisés.

Chapitre III

Implémentation et résultats

III.1 Introduction

La sécurité dans l'internet des objets sert à sécuriser la communication entre les différents objets. Nous présentons dans ce chapitre un crypto-système qui constitue le moyen de chiffrer les différents messages circulant dans un réseau IoT. Notre algorithme utilise les séquence Adn pour chiffrer les messages avant de les transmettre aux récepteurs sous forme Code QR.

Tout au long de ce chapitre, nous présenterons en détails notre système et les différentes techniques utilisées dans ses différentes phases.

III.2 Présentation de l'approche

Le crypto-système que nous proposons est un algorithme symétrique (à clef secrète) agissant par bloc de 64 bits. La séquence ADN qui est généralement un chromosome fait l'objet ce système, c'est à partir de laquelle on fait l'extraction des codes binaires des bases azotiques (A ,T,C,G) et des clés de chiffrement/déchiffrement. La figure III.1 représente l'approche globale,

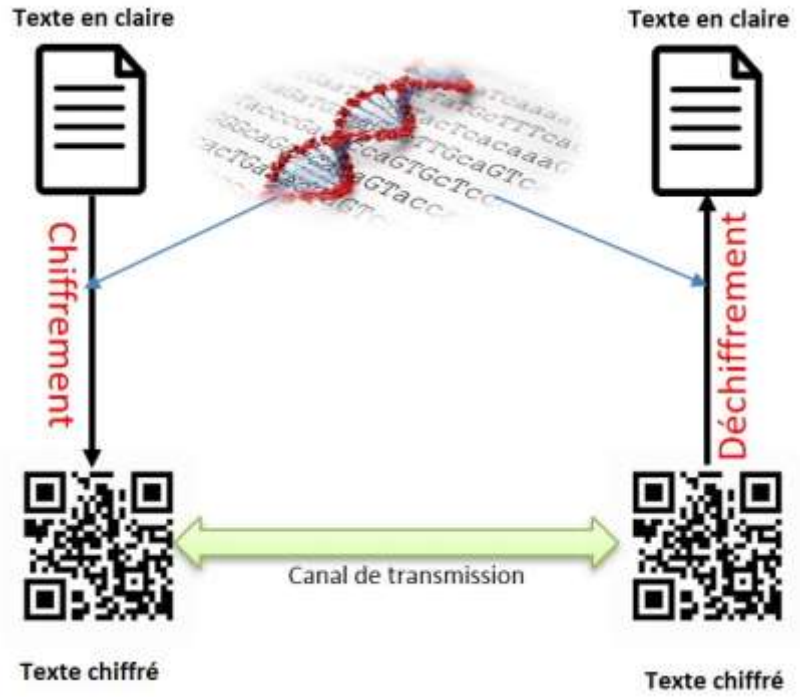


Figure III.1 Le crypto-système proposé

III.3 Le chiffrement

Nous détaillons le processus de chiffrement comme il est schématisé dans la figure III.3

1. Coder le texte en binaire

2. Diviser le texte en bloc de 64 bits qui soit kblocs

*3. Appliquer le codage de huffman sur la séquence d'adn (a partir d'une position P jusqu'au Kblocs*32)*
Et tirer les codes des bases A ,T,C,G

4. Coder les blocs en ADN

5. .A partir de la séquence ADN et la position P, tirer kblocs clés de 32 bases

6. Effectuer un « Ou Exclusif » entre le blocs i et la clé Ki ***(bloci ⊕ K*** $_i$*).*

7. Effectuer le brouillage selon la boite de brouillage Bt

8. Effectuer la phase de transcription

9.Effectuer la phase de translation

10. Coder le texte en code Qr

Figure III.2 L'algorithme de chiffrement

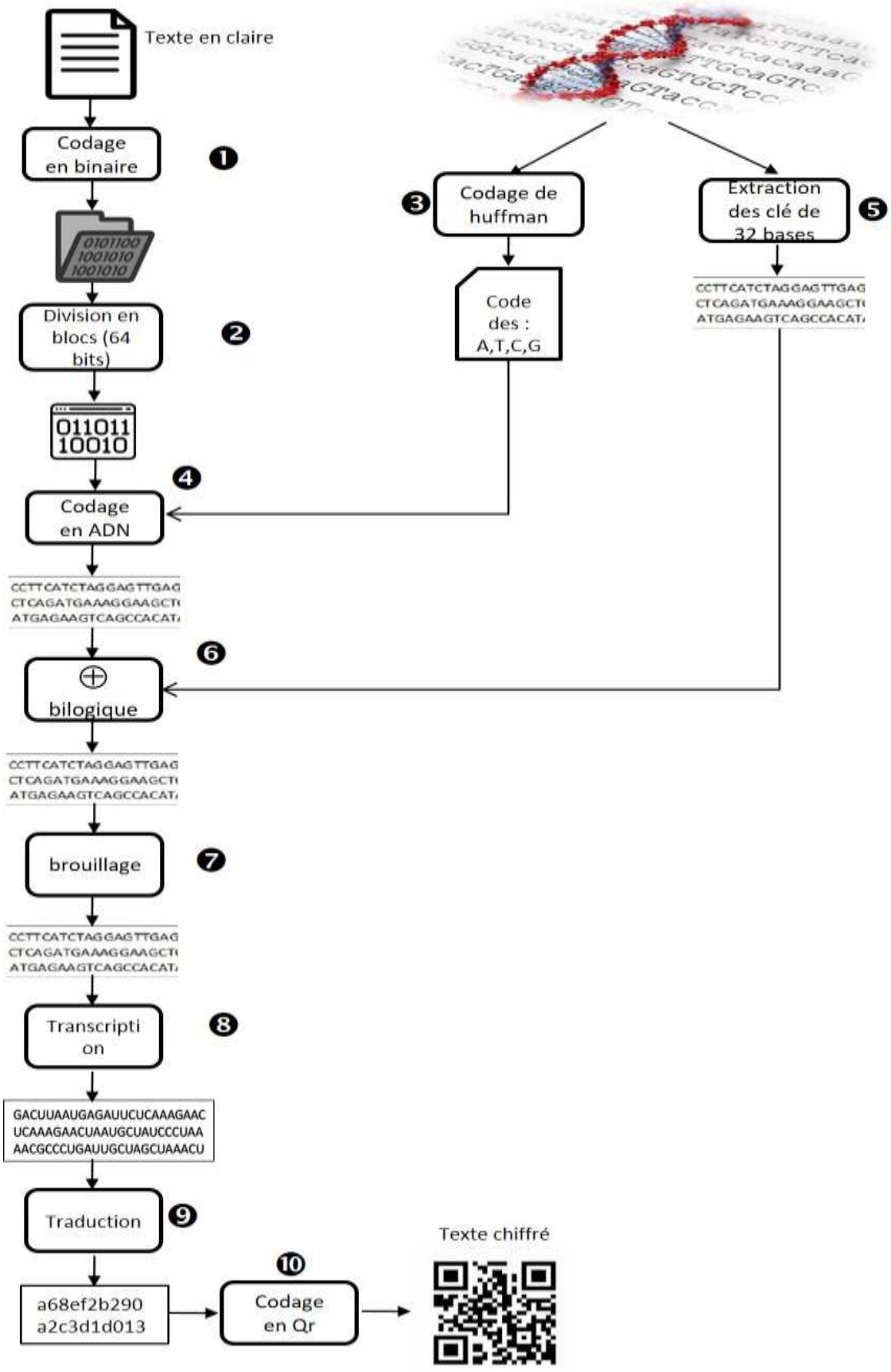

Texte en claire
Codage en binaire
1
Division en blocs (64 bits)
2
011011 10010
Codage de huffman
3
Code des : A,T,C,G
Extraction des clé de 32 bases
5
Codage en ADN
4
bilogique
6
brouillage
7
Transcripti on
8
GACUUAAUGAGAUUCUCAAAGAAC
UCAAAGAACUAAUGCUAUCCCUAA
AACGCCCUGAUUGCUAGCUAAACU
Traduction
9
a68ef2b290 a2c3d1d013
10
Codage en Qr
Texte chiffré

Figure III.3 Le processus de chiffrement

III.3.1 Codage en binaire

Le codage binaire permet de représenter le texte en claire, notamment des nombres ou des caractères textuel en suite de bits (0et 1).Chaque caractère est représenté par un nombre, un code numérique qui est le code ascii (Figure III.4) et chaque code sera remplacé par sa valeur binaire.

0	*NUL*	16	*DLE*	32	SPC	48	0	64	@	80	P	96	`	112	p
1	*SOH*	17	*DC1*	33	!	49	1	65	A	81	Q	97	a	113	q
2	*STX*	18	*DC2*	34	"	50	2	66	B	82	R	98	b	114	r
3	*ETX*	19	*DC3*	35	#	51	3	67	C	83	S	99	c	115	s
4	*EOT*	20	*DC4*	36	$	52	4	68	D	84	T	100	d	116	t
5	*ENQ*	21	*NAK*	37	%	53	5	69	E	85	U	101	e	117	u
6	*ACK*	22	*SYN*	38	&	54	6	70	F	86	V	102	f	118	v
7	*BEL*	23	*ETB*	39	'	55	7	71	G	87	W	103	g	119	w
8	*BS*	24	*CAN*	40	(	56	8	72	H	88	X	104	h	120	x
9	*HT*	25	*EM*	41	)	57	9	73	I	89	Y	105	i	121	y
10	*LF*	26	*SUB*	42	*	58	:	74	J	90	Z	106	j	122	z
11	*VT*	27	*ESC*	43	+	59	;	75	K	91	[	107	k	123	{
12	*FF*	28	*FS*	44	,	60	<	76	L	92	\	108	l	124	|
13	*CR*	29	*GS*	45	-	61	=	77	M	93	]	109	m	125	}
14	*SO*	30	*RS*	46	.	62	>	78	N	94	^	110	n	126	~
15	*SI*	31	*US*	47	/	63	?	79	O	95	_	111	o	127	DEL

Figure III.4 la table ASCII

III.3.2 Division en blocs

Cette étape consiste à diviser le texte binaire en blocs de 64 bits, le dernier bloc n'atteignant pas les 64 bits est complété par le caractère « espace » soit 32Dec soit 00100000Bin. Soit **Kblocs** le nombre des blocs obtenus.

III.3.3 Codage de huffman

Dans cette phase ; nous appliquons un codage de huffman sur une séquence adn , a partir d'une position p , et sur une longueur de **kblocs*32** bases azotique.

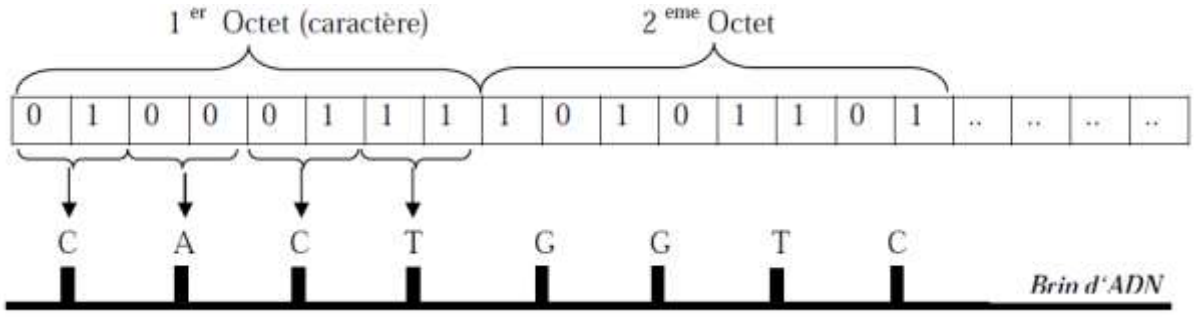
Le codage Huffman est un algorithme de compression de données sans perte. Dans cet algorithme, un code de longueur variable est attribué pour saisir différents bases azotique(A,T,C,G). La longueur du code est liée à la fréquence d'utilisation des caractères.

Les caractères les plus fréquents ont des codes plus petits et des codes plus longs pour les caractères les moins fréquents. Il y a principalement deux parties. Un premier pour créer l'arbre de Huffman, et un autre pour parcourir l'arbre pour trouver des codes.

A la fin de cette phase , nous obtiendrons les codes binaires corresponds de A ,T , et G

III.3.4 Codage en ADN

Dans cette phase, Nous transformons une bloc qui contient (64 bits) en brin d'ADN qui contient les quatre bases azotées A, C, G et T, selon les codes tirés de la phase précédente (codage de huffman) , on effectue la transformation suivante :



FigIII.5: Transformation bits→bases.

III.3.5 Extraction des clés

Une séquence ADN qui représente généralement un chromosome sert comme source des sous clés utilisées dans la phase de chiffrement. Un entier P_{dep} sert comme une position de départ à partir de laquelle on commence l'extraction des clés de 32 bases azotiques , le nombre des clés à extraire dépend du nombre des blocs détectés dans le texte en claire (K_{blocs}).

La position de départ sera calculée par la formule suivante :

$P_{dép}$ mod long(séq) (1)

Où : $P_{dép}$: Entier introduit par l'émetteur et Séq : est la séquence ADN

FigIII.6: Génération des sous clés

III.3.6 Xor biologique

C'est une opération définie entre les bases azotées selon la table de vérité suivante :

⊕	*A*	*C*	*G*	*T*
A	**A**	**C**	**G**	**T**
C	**C**	**A**	**T**	**G**
G	**G**	**T**	**A**	**C**
T	**T**	**G**	**C**	**A**

Tableau III.1 : Xor biologique.

Cette opération qui est définie sur les bases azotique respecte les mêmes critères d'un ou-exclusif entre les bits :

M ⊕ C =C' **(2)**

C'⊕ C = M **(3)**

Dans cette phase ,on réalise l'opération du Xor bilogique entre le bloc Bi et la cle Ki

$\mathbf{C_i = B_i \oplus K_i}$.

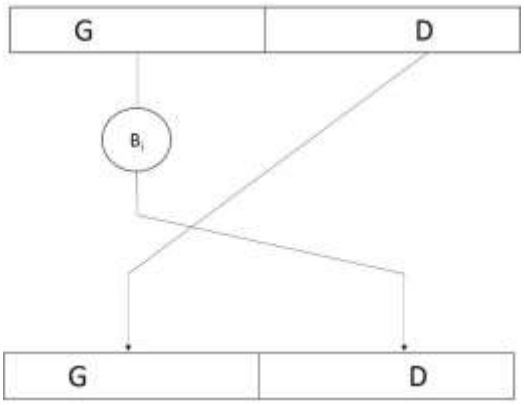
L'équation **(2)** est utilisée dans la phase de chiffrement

L'équation **(3)** est utilisée dans la phase de déchiffrement

III.3.7 Brouillage

Le brouillage sert comme moyen d'élimination d'ordre logique des bases azotique dans les blocs. Nous avons utilisé 16 boites de brouillage de 16 bases crées d'une manière aléatoire. (inspiré de DES) FigIII.8.

Nous appliquons dans cette phase un brouillage de 6 tours. C'est une inspiration du schéma de Feistel selon la **Figure III.7**



FigIII.7 Brouillage

Boite 01	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	4	9	8	3	15	7	10	2	14	6	11	16	1	12	5	13
Boite 02	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	9	11	1	8	10	2	13	7	12	15	3	16	14	5	6	4
Boite 03	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	8	7	14	1	9	6	13	2	10	15	11	3	16	12	4	5
Boite 04	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	7	1	9	2	8	3	10	4	14	5	13	15	11	16	12	6
Boite 05	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	10	9	16	8	7	15	6	5	14	13	4	3	12	11	2	1
Boite 06	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	10	8	11	2	6	5	14	13	4	3	15	12	1	16	9	7

FigIII.8 Extrait des boites de Brouillage

III.3.8 La transcription

En biologie, le processus de transcription commence quand une enzyme appelée ARN polymérase (ARN pol)se fixe au brin d'ADN matrice et commence à catalyser la production de l'ARN complémentaire, appelé le ARNm. Une copie d'un seul brin d'ADN est produite, prête pour le processus de traduction. Figure III.9

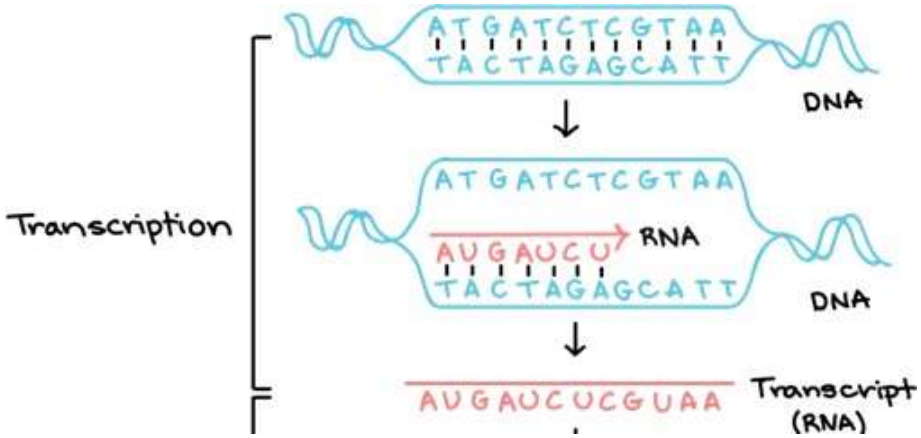
Dans notre algorithme, cette phase est une substitution mono-Alphabétique simulant le processus de transcription du dogme central.

A →T.

C →G.

G →C.

T →U(A).



FigIII.9: La transcription.

III.3.9.La Traduction (Translation)

Cette procédure est aussi une substitution qui simule la phase de translation du dogme central.

Rappel de La traduction dans l'ADN :

Dans ce processus, le ribosome (la molécule qui exécute la translation) lit trois premières bases (codant) de l'ARNm, et à l'aide de la table du code génétique, trouve l'acide aminé adéquat, puis les trois bases suivantes, ainsi de suite jusqu'à rencontrer une séquence stop qui indique la fin de la translation (Figure III..10)

NUCLÉOTIDE 1ère POSITION	NUCLÉOTIDE 2ème POSITION U		NUCLÉOTIDE 2ème POSITION C		NUCLÉOTIDE 2ème POSITION A		NUCLÉOTIDE 2ème POSITION G		NUCLÉOTIDE 3ème POSITION
U	UUU	phényl-alanine	UCU	sérine	UAU	tyrosine	UGU	cystéine	U
	UUC		UCC		UAC		UGC		C
	UUA	leucine	UCA		UAA	non-sens	UGA	non-sens	A
	UUG		UCG		UAG		UGG	tryptophane	G
C	CUU	leucine	CCU	proline	CAU	histidine	CGU	arginine	U
	CUC		CCC		CAC		CGC		C
	CUA		CCA		CAA	glutamine	CGA		A
	CUG		CCG		CAG		CGG		G
A	AUU	isoleucine	ACU	thréonine	AAU	asparagine	AGU	sérine	U
	AUC		ACC		AAC		AGC		C
	AUA		ACA		AAA	lysine	AGA	arginine	A
	AUG	méthionine	ACG		AAG		AGG		G
G	GUU	valine	GCU	alanine	GAU	acide aspartique	GGU	glycine	U
	GUC		GCC		GAC		GGC		C
	GUA		GCA		GAA	acide glutamique	GGA		A
	GUG		GCG		GAG		GGG		G

Figure III.10 : Tableau du code genetique

La traduction du code génétique de sa forme d'acide désoxyribonucléique consistant en une chaîne de quatre lettres répétées en un produit protéique final constitué d'acides aminés est un processus bien compris. C'est l'étape finale de la traduction d'une séquence d'ADN en une protéine fonctionnelle. Des molécules de complexe ARN /protéine appelées "ribosomes" se fixent sur le brin d'ARNm modifié et traduisent le brin en une chaîne de molécules protéiques. Ceci est accompli par des molécules d'ARN de transfert (ARNt) qui transportent des acides aminés spécifiques aux ribosomes où des codes à trois lettres sont lus et appariés avec des acides aminés spécifiques. Une fois la chaîne d'acides aminés synthétisée, elle se replie automatiquement en une conformation qui la rend fonctionnelle. C'est pourquoi une seule mutation de l'ADN peut être désastreuse. La mutation de l'ADN est transcrite en un code d'ARNm de trois lettres qui, à son tour, code pour le mauvais acide aminé. Cela empêche ainsi la chaîne d'acides aminés finale de se replier correctement dans une protéine fonctionnelle.

La phase de traduction :

Nous avons changé la table changé la table du code génétique en une table de dimension 16*16, tel que chaque colonne est indexée par deux bases et chaque ligne est indexée par deux bases. De plus, nous avons augmenté le nombre d'acides aminés à 256 au lieu de 21 (ou chaque acide est représenté par deux hexadécimaux. (Figure III.11)

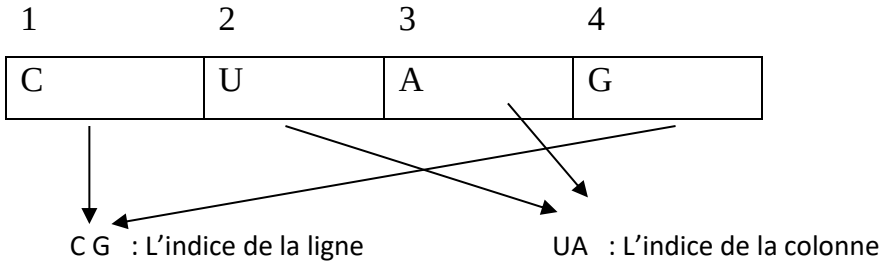
	A A	A C	A G	A U	C A	C C	C G	C U	G A	G C	G G	G U	U A	U C	U G	U U
AA	0F	6D	FD	70	1C	05	FE	BA	FB	73	74	FF	78	FC	60	63
AC	C0	1A	4D	48	20	50	88	E1	01	38	AA	0C	94	49	AB	02
AG	3D	E7	B0	0D	A0	D4	10	29	46	67	89	47	2E	B6	37	BF
AU	D2	9C	3F	D3	87	2D	BE	5C	D5	15	A9	56	C5	12	95	66
CA	65	F4	B8	19	D1	77	45	E0	31	8A	2A	E8	55	BD	54	04
CC	B1	9B	27	39	5B	C8	F5	B5	57	68	BC	0B	93	23	D0	79
CG	7F	1B	76	AE	11	44	8B	4F	14	3A	4A	E6	F6	B7	2B	AD
CU	3C	4B	A1	06	5A	6F	C9	A8	DF	51	AC	22	CA	53	92	36
GA	B2	8C	A7	4C	DE	21	75	1D	58	E5	C4	52	BB	1E	F1	A4
GC	C1	0E	43	9A	F9	2C	A2	B4	C2	00	5D	DC	16	CB	96	03
GG	7E	B3	7D	4E	6E	E9	32	40	25	F7	AA	69	EA	24	2F	7A
GU	D6	28	E2	07	DD	90	B9	3E	EB	91	F2	08	80	C3	DB	81
UA	64	A6	FA	8D	13	C7	1F	ED	99	EC	3B	C6	30	E4	0A	97
UC	85	D8	17	A3	86	26	83	59	CD	09	F3	03	98	18	EF	CF
UG	D7	9D	F0	6B	CC	8E	62	E3	EE	F8	9E	33	D9	DA	72	82
UU	CE	42	61	84	34	A5	6A	9F	5E	AF	71	8F	5F	41	7B	35

Figure III.11 : boite de traduction.

Pour appliquer une substitution :

- On lit les 4 premières bases du bloc
- La première sera remplacée par la case Tr[i][j] de la table génétique tel que :
  - L'indice i correspond au 1ère base + 4ème base
  - L'indice j correspond au 2ème base +3ème base
- Répéter ce procédé pour chaque chaîne (de quatre bases azotées) du bloc

Soit l'exemple illustré dans la Fig(*table de code génétique*) : nous lisons d'abord les quatre premières bases du bloc , par exemple CUAC ;



Ttr[CG ,UA] =16

Ces quatre premières bases du bloc seront replacées par le contenu de la case ***[CG, UA] = 16*** de la table génétique .Ce procédé est répété jusqu'à la fin du bloc.

III.3.9 Codage en QR

1.Qu'est ce qu'un code Qr ?

Un code QR est une version bidimensionnelle du code-barres, typiquement composée de pixels noirs et blancs. Denso Wave, filiale de Denso, fournisseur de pièces détachées pour Toyota, en 1994 a développé cette technologie pour accélérer les processus logistiques de sa production automobile (rappelons que QR sont les initiales de l'anglais « quick response », réponse rapide). Depuis l'avènement des smartphones, les codes QR font partie intégrante du marketing mobile.

Figure III.12 Structure d'un code QR

Comme il est indiqué dans la figure III.12 , un code Qr est structuré come suit :

– Les trois grands carrés surlignés en rouge sont les marqueurs de position. Ils indiquent au lecteur où se trouvent les bords du QR Code.

– Le petit carré rouge est un marqueur d'alignement. Il agit comme un point de référence pour le lecteur en s'assurant que tout s'aligne correctement. Sur des codes de plus grosses dimensions on trouve plusieurs de ces carrés.

– Les bandes rouges sont appelées « timing patterns ». Ils définissent le positionnement des lignes et des colonnes.

– Les sections vertes indiquent au lecteur le format du QR Code, il lui indique s'il s'agit d'un texte, d'un site web ou autres.

– Enfin les modules en bleu représentent le numéro de version à savoir que plus il y a de modules plus la version est grande. Jusqu'à v40 qui représente 177 * 177 modules.

– Les modules restants se regroupent par 8 formants donc un octet

2. Codage en Qr

Dans notre algorithme ; nous codons la série des code hexadécimaux issue de la dernière phase (translation) en code Qr qui représente notre texte chiffré qui sera transmis au récepteur.

III.4 Le déchiffrement

L'algorithme de déchiffrement sert à déchiffrer un bloc de 64 bits, qui a été chiffré par l'algorithme de chiffrement. La clef de chiffrement et de déchiffrement étant exactement les mêmes. L'algorithme de déchiffrement fait exactement l'inverse de l'algorithme de chiffrement E, c'est-à-dire, pour tous message en clair M, on a D (E(M)) = M.

Tous les modules qui existent dans le chiffrement sont inversés dans le déchiffrement. (Figure III.14) .

L'algorithme de déchiffrement a comme entrée un code Qr , les étapes de déchiffrement sont

1. Décoder le code Qr en texte (série hexadécimale)
2. Selon la boite de traduction utilisé dans le chiffrement ; effectuer le traduction inverse
3. Réaliser la transcription inverse
4. Effectuer le débrouillage selon la même boite de brouillage utilisé dans le chiffrement
5. Extraire de la séquence ADN , les sous clés de 32 bases azotiques Ki
6. Effectuer un « Xor bilogique » entre le bloc Bi et la sous clef *Ki*, $\boldsymbol{Bi \oplus K_i}$
7. Extraire les codes des A,T,C ,G selon le codage de Huffman sur la séquence ADN
8. Selon les codes extrait ; coder le bloc en binaire
9. Regrouper les blocs
10. Décoder le texte en claire (Binare-ASCII)

Figure III.13 L'algorithme de déchiffrement

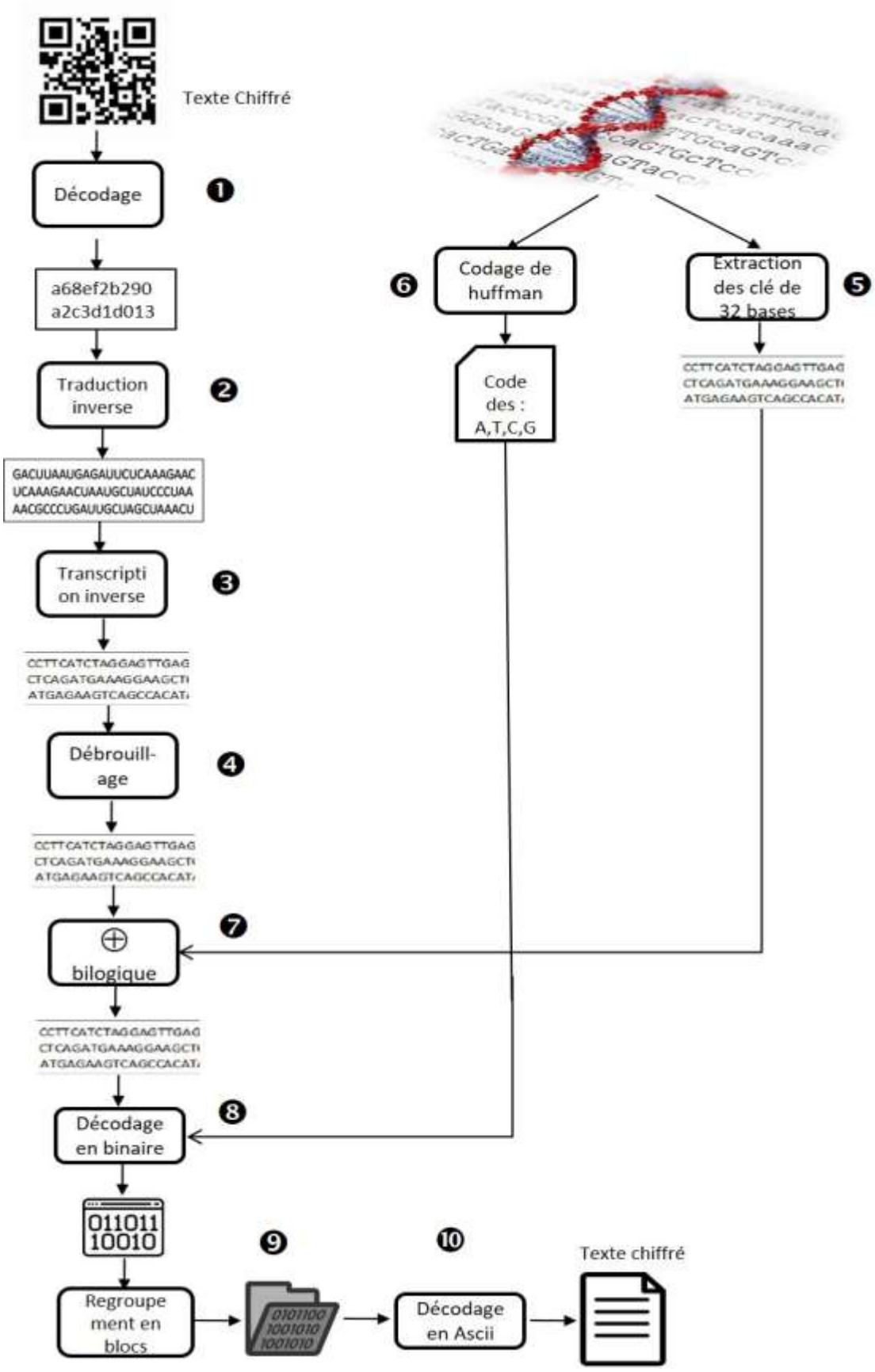


Figure III.14 Processus de déchiffrement

III.5 Exemple explicatif

Texte en claire : chiffrement sic2

1-Codage en binaire

01100011011010000110100101100110011001100111001001100101011011010110010101
10111001110100001000000111001101101001011000110011 0

2-Division en blocs

1er bloc : 0110001101101000011010010110011001100110011100100110010101101101

2ème bloc : 0110010101101110011101000010000001110011011010010110001100110010

3 : Codage en hufman

Séquence ADN

Chromosome 1

Pdep =2

Après l'application de huffman sur la séquence à partir de la position **Pdep** :

On obtient : A :10 ; T :11 ;C :01 ,G :00

4.Le codage en ADN :

01100011011010000110100101100110011001100111001001100101011011010

Bloc1 : CAGTCAAGCAACCACACACACTGACACCCATC

Bloc2 : CACCCATACTCGGAGGCTGTCAACCAGTGTGA

5. Extraction des sous clé :

Sous clé 1 : TCAAAAGTCTAGAGCCACCGTCCAGGGAGCA

Sous clé 2 : GGTAGCTGCTGGGCTCCGGGGACACTTTGCGT

6. Xor Biologique

Bloc1 : CAGTCAAGCAACCACACACACTGACACCCATC

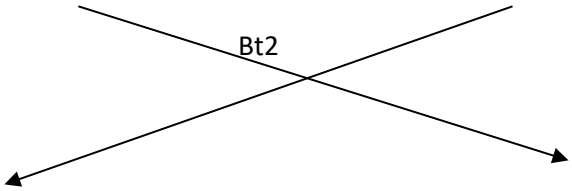
Cle1 : TCAAAAGTCTAGAGCCACCGTCCAGGGAGCA

=

GCGTCAGCATATCGACCCAGGGTATGTCTCTT

7.Brouillage avec la boite n° : 1

GCGTCAGCATATCGAC CCAGGGTATGTCTCTT



CCAGGGTATGTCTCTTT ACGAGTCGAACGTCC

Transcription :

GGUCCCAUACAGAGAAAUGCUCAGCUUGCAGG

Translation : avec la boite de translation n° :1

DC5AA0FBABCCAD76

Meme chose avec bloc 2 :

Codage en Qr :

Figure III.15 image Qr du texte chiffré

Déchiffrement :

1 Décodage :

DC5AA0FBABCCAD76 470876741074OF63

2 La traduction inverse :

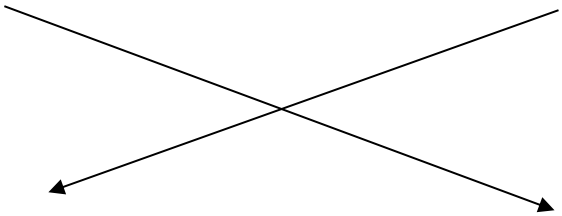
GGUCCCAUACAGAGAAAUGCUCAGCUUGCAGG

3 Transcription inverse

CCAGGGTATGTCTCTTTACGAGTCGAACGTCC

4 Débrouillage

CCAGGGTATGTCTCTT TACGAGTCGAACGTCC



GCGTCAGCATATCGAC CCAGGGTATGTCTCTT

5 Xor biologique

GCGTCAGCATATCGACCCAGGGTATGTCTCTT

Même traitement avec le bloc 2 :

6 Décodage en binaire

01100101011011100111010000100000011100110110100101100011001100 10

7 Regroupement en blocs

Le texte en claire est : chiffrement sic2

III.6 Expérimentations et résultats :

III.6.1 Environnement de travail :

-Equipement :

Figure III.16 Raspberry Pi3

Le Raspberry Pi est un ordinateur peu coûteux de la taille d'une carte de crédit qui se branche sur un écran d'ordinateur ou un téléviseur et utilise un clavier et une souris standard. C'est un petit appareil capable qui permet aux personnes de tous âges d'explorer l'informatique et d'apprendre à programmer dans des langages comme Scratch et Python. Il est capable de faire tout ce que vous attendez d'un ordinateur de bureau, de la navigation sur Internet à la lecture de vidéos haute définition, en passant par la création de feuilles de calcul, le traitement de texte et les jeux.

De plus, le Raspberry Pi a la capacité d'interagir avec le monde extérieur et a été utilisé dans un large éventail de projets de création numérique, des machines à musique et des détecteurs de parents aux stations météorologiques et aux nichoirs à gazouillis avec des caméras infrarouges. Nous voulons voir le Raspberry Pi utilisé par les enfants du monde entier pour apprendre à programmer et comprendre le fonctionnement des ordinateurs.

Caractéristiques et spécifications du Raspberry Pi 3

Taille : 8,56 cm x 5,65 cm x 1,7 cm

Poids : 45g

SoC : Broadcom BCM2837

Processeur : Processeur 4 cœurs 64 bits, architecture ARMv8, cadencé à 1.2 GHz

Mémoire : 1GB DDR2 (900 MHz)

GPU : Broadcom VideoCore IV avec 3D

Stockage : 1 emplacement de carte Micro-SD

Réseau : Port Ethernet 10/100 Mbits/s + Wi-Fi 802.11N + Bluetooth 4.1

Ports USB : 4 ports USB 2.0

Audio et vidéo : Jack 3.5mm + HDMI + CSI et DSI

Principales Entrées / Sorties : 40 broches GPIO

Système d'exploitation :

Raspbian 64 bits qui basé sur Linux Debian

Langage de programmation :

Nous avons utilisé le langage Python, qui est le langage de programmation open source le plus employé par les informaticiens. Ce langage s'est propulsé en tête de la gestion d'infrastructure, d'analyse de données ou dans le domaine du développement de logiciels. En effet, parmi ses qualités, Python permet notamment aux développeurs de se concentrer sur ce qu'ils font plutôt que sur la manière dont ils le font. Il a libéré les développeurs des contraintes de formes qui occupaient leur temps avec les langages plus anciens. Ainsi, développer du code avec Python est plus rapide qu'avec d'autres langages.

III.6.2Tests et résultats

III.6.2.1 Evaluation du temps d'exécution

Dans ce test, nous avons varié la taille du texte à chiffrer et nous avons calculé le yemps de chiffrement et déchiffrement correspond. Le tableau III.3 montre les résultats obtenus.

taille de texte en claire (octet)	temp de chiffrement (s)	temp de déchiffrement (s)
28	0,01779	0,011419
56	0,031048	0,017639
85	0,044074	0,027837
113	0,056583	0,028121
141	0,072511	0,045888
169	0,090625	0,052927
225	0,116294	0,072633
253	0,1351	0,080623
282	0,142952	0,089611
295	0,150717	0,090571
299	0,152235	0,096843

Tableau 3.1 temps de chiffrement/déchiffrement en fonction de la taille de texte en claire

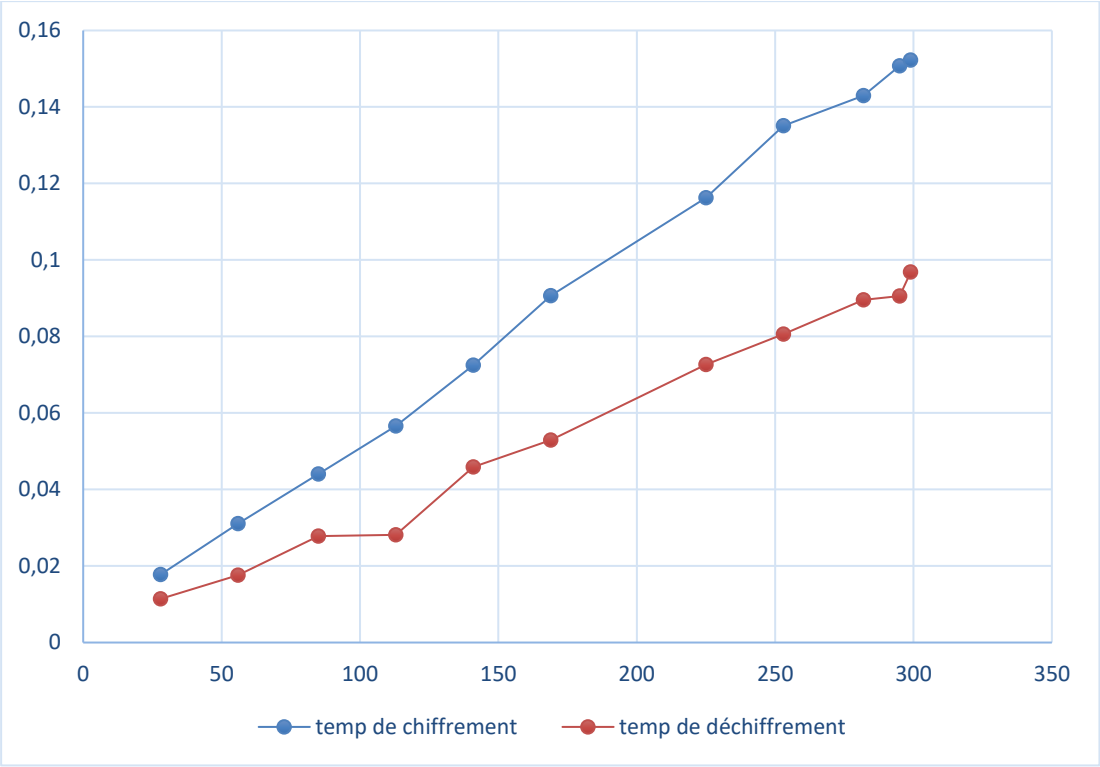


Figure III.17 Variation de temps d'exécution en fonction du texte en claire

Discussion :

On remarque que le temps de chiffrement (respectivement déchiffrement) augmente linéairement avec l'augmentation de la taille de texte en clair (respectivement texte chiffré), ce qui confirme les résultats théoriques de l'étude de la complexité de l'algorithme.

III.6.2.2 Analyse de l'espace de la clé

La clé adoptée pour notre algorithme est composée de :

Séquence ADN	**Position de départ**	**N° : boite de brouillage**	**N° : boite de traduction**

1- La séquence adn (collection de toutes les séquences ADN publiquement disponibles (Genbank) : 135440924

2- Boite de brouillage,(16 combinaisons de 0 à 16)

3-Position de départ (selon la taille de la séquence) ;

4-Boite de traduction : 256 combinaisons (de 0 à F) :

L'utilisation de séquences chromosomiques d'ADN rend l'approche proposée plus robuste et donc moins vulnérable aux différents types d'attaques notamment les attaques statistiques. Un adversaire doit avoir : (1) La position de départ, (2) la séquence d'ADN utilisée pour déterminer la position de départ de l'application de codage de huffman pour l'extraction des codes binaires des bases azotiques A,C,T et G d'un part et pour l'extraction des sous-clés d'autre part et (3) l'ordre de brouillage utilisé ainsi que la boite de traduction utilisée dans la phase de translation. Ainsi, il est presque impossible de deviner la clé de cryptage en raison du très grand nombre de séquences d'ADN et du changement fréquent de clé.

III.7 Conclusion

Dans ce chapitre, nous avons présenté notre algorithme avec une description détaillée de ces différentes phases. Ensuite, nous avons effectué plusieurs tests afin de bien évaluer sa performance, notamment son comportement quand les paramètres se varient. Et nous avons présenté les différents résultats obtenus après ces tests suivis des commentaires et justifications pratiques et théoriques.

Dans un premier lieu, nous avons présenté en détail notre algorithme avec un exemple explicatif, ensuite dans la phase des expérimentations, nous avons estimé le taux de chiffrement/déchiffrement de notre algorithme et ceci en effectuant plusieurs tests dont nous avons présenté une partie des résultats obtenus.

Nous avons aussi varié les paramètres de l'algorithme et nous avons suivi le comportement de temps d'exécution. Les résultats obtenus montrent que le temps de chiffrement/déchiffrement s'évolue linéairement avec l'augmentation de ces paramètres, ce qui est acceptable en termes de complexité.

Conclusion Générale

L'Internet des objets (IoT) représente un paradigme technologique florissant le plus rapide contenant de nombreux appareils qui produisent, traitent et communiquent une grande quantité de données. Par conséquent, plusieurs types d'intrusions et de menaces peuvent cibler les appareils IoT dans le réseau. Plusieurs approches ont été présentées pour améliorer la sécurité des communications dans ce type des réseaux.

Dans ce travail, nous avons présenté un algorithme qui sert à : **(1)** chiffrer et déchiffrer les messages circulants entre objets IoT. **(2)** Créer un code Qr sécurisé qui utilise des paramètres liés aux chromosome pour créer le code Qr d'un part et pour l'extraction du message inclut dans le code qr d'autre part.

Un ensemble des expérimentations ont été réalisé sur un Raspberry pi3 (come un objet IoT) , et le résultats obtenus ont montré la robustesse de l'algorithme de point de vue taille de clé , et temps de chiffrement tout en prend en considération les ressources limitées des appareils communicantes dans l'environnement IoT.

Nous envisageons de tester notre algorithme dans un environnement Arduino pour montrer les avantages de son utilisation dans des appareils IoT de ressources très limitées.

Références Bibliographiques

Références bibliographique

[1] : Dương Hiệu Phan, « Sécurité et efficacité des schémas cryptographiques », Doctorat de l'École polytechnique, École normale supérieure d'informatique, 2005.

[2] : Bouazza Med Abdeljalil, Charef Abderrahmane, « Etude et conception d'un crypto-système à clé secrète par le biais de la programmation cellulaire », mémoire de master, Université Djillali Liabès de Sidi Bel Abbes, 2012-2013.

[3] : H.X.Mel, Doris Baker, la cryptographie décryptée Compus Press, 2001.

[4] : Renaud Dumont, « Introduction à la Cryptographie et à la Sécurité informatique », Université de Liège, Faculté des Sciences Appliquées, 2007.

[5] : Robert Rolland *« Formation Générale en Cryptographie »* , 2002.

[6] : Daniel Barsky & Ghislain Dartois, cours de Cryptographie, 2010.

[7] : C.E.Shannon, A Mathematical Theory of Communication, Bell System Technical Journal, July 1948, P.623.

[8] : **Url:** http://dspace.univ-tlemcen.dz/bitstream/112/1076/5/chapitre1.pdf

Consulté le 08/12/2021 à 17:40.

[9] : Joan Daemen , Vincent Rijmen, AES proposal : Rijndael, 1999.

[10] : Network Associates International, Introduction à la cryptographie, Gatwickstraat 25 NL-1043 GL Amsterdam,

[11] : Lancelot Pecquet, Mathématiques du secret » Université Paris XII 2007.

[12] : Auguste Kerckhoffs, La cryptographie militaire, Journal des sciences militaires, vol. IX, pp. 5-83, Jan. 1883, pp. 161-191, Feb. 1883

[13] : XIAO Guozhen, LU Mingxin, QIN Lei & LAI Xuejia, New field of cryptography: 'DNA cryptography. Xidian University, Xi'an 710071, China; Chinese Science Bulletin 2006 Vol. 51 No. 12 1413—1420, January 16, 2006

[14] : E.G.Berger, T.Hennet, Le génome humain, Principe fondamental sur l'ADN. Forum Med Suisse N°23 le 6 juin 2001. [15] :

[15] : Akhil Kaushik, Dr. Vikas Thada, "The Evolution of DNA Cryptology - A Review", International Journal of Electrical Electronics & Computer Science Engineering (IJEECSE), Vol. 5, No. 2, Apr. 2018.

[16] : Mansouri Nabil, Gougache Mohamed, « Conception et implémentation d'une méthode cryptographique inspirée de l'ADN », Mémoire d'ingénieur d'état en informatique, Ecole nationale Supérieure d'Informatique (ESI) Oued-Smar, Alger, 2008-2009.

[17] : Ashish Gehani, Thomas H. LaBean, John H. Reif, DNA based cryptography 5th Annual DIMACS Meeting on DNA Based Computers (DNA 5), MIT, Cambridge, MA, June 1999

[18] : Adleman. M. L (1994), Molecular Computation of Solutions to Combinatorial Problems, Science, vol.266, pp.1021-1024.

[19] : J. Lipton. R (1995), Using DNA to Solve NP Complete Problems, Science, Vol.268, pp.542-545.

[20] : Ouyang Qi, D. Peter Kaplan, Liu Shumao and Albert Libchaber (1997), DNA Solution of the Maximal Clique Problem, Science 278, 5337, 446-449

[21] : . Chen Jie (2003), A DNA-based bio molecular cryptography design, Proceedings of IEEE International Symposium, Vol.3, pp.III-822.

[22] : A. K. Verma, Mayank Dave, R. C. Joshi (2008), DNA Cryptography: A Novel Paradigm for Secure Routing in Manets, Journal of Discrete Mathematical Sciences and Cryptography, Vol.11, No.4, pp.393-404

[23] : Yunpeng Zhang, Bochen Fu, and Xianwei Zhang (2012), DNA cryptography based on DNA Fragment assembly, Information Science and Digital Content Technology (ICIDT), 8th IEEE International Conference, Vol.1, pp.179-182

[24] : Olga Tornea and Borda E. Monica (2013), Security and Complexity Of A DNA-Based Cipher, Roedunet International Conference (Ro Edu Net), 11th IEEE International Conference, pp.1-5.

[25] : Benyahia, K., Mustapha, M., & Abdelkrim, L. (2021). A Bio-Inspired Algorithm for Symmetric Encryption. In *Research Anthology on Artificial Intelligence Applications in Security* (pp. 490-503). IGI Global.

[26] : Khobzaoui, A., Benyahia, K., Mansouri, B., & Boukli-Hacene, S. (2022). DNA-Based Cryptographic Method for the Internet of Things. *International Journal of Organizational and Collective Intelligence (IJOCI)*, *12*(1), 1-12.