

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE MINISTÈRE DE
L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITÉ ABDERRAHMANE MIRA - BEJAIA
FACULTÉ DE SCIENCES EXACTES
DÉPARTEMENT INFORMATIQUE



MÉMOIRE DE FIN D'ÉTUDES
EN VUE D'OBTENTION DU DIPLÔME DE MASTER
OPTIONS RÉSEAUX ET SÉCURITÉ

Proposition d'une nouvelle approche Cryptographique

Réalisé par :
Mr.MAMMERI Malek

Encadré par :
Mme.SABRI Salima

Devant le jury composé de :

Présidente : Mme.TASSOULT Nadia

Examinatrice : Mme.BOUADEM Nassima

Promotion 2023 - 2024

REMERCIEMENTS

Mes remerciements vont tout premièrement à Dieu tout puissant pour la volonté, la santé et la patience qu'il m'a donné durant toutes mes années d'études.

Ma gratitude envers moi-même, reconnaissant de tous les efforts déployés pour parvenir à cette réussite.

Mes sincères remerciements à mon encadrante, **Mme SABRI**, pour avoir accepté ce sujet captivant et pour ses précieux conseils et encouragements. Je vous suis reconnaissant pour votre confiance, votre disponibilité et le temps que vous avez gracieusement consacré à l'achèvement de ce mémoire.

Mes remerciements vont également à tous les membres du jury qui ont bien voulu consacrer leur temps précieux pour l'examen de ce mémoire.

J'exprime ma profonde gratitude à **Mme.TASSOULT**, de nous avoir fait l'honneur de présider le jury.

Je tiens à remercier chaleureusement **Mme.BOUADEM**, d'avoir accepté d'examiner mon travail.

Mes Remerciements à tous mes enseignants tout au long de mon parcours, pour m'avoir appris le goût de l'effort et du travail durant cette période qui revêt une très grande valeur pour moi.

À mes chers parents, c'est grâce à eux que j'en suis arrivé à cette étape aujourd'hui, ainsi qu'à toute ma famille, mes amis et tous ceux qui ont contribué à ma réussite.

Encore une fois, merci du fond du cœur pour votre soutien inestimable.

DÉDICACES

La meilleure chose dans la vie, est d'aller à l'intérieur de soi, et de combattre les mauvais jours, pour arriver à réaliser le meilleur rêve de la vie.

Je dédie ce travail :

À ma très chère mère Nadia

La lumière de mes jours, la source de mes efforts, A celle qui m'a arrosé de tendresse et d'espoirs, à la mère des sentiments fragiles qui ma bénie toujours par ces prières. Ton affection me couvre, ta bienveillance me guide et ta présence à mes côtés a toujours été ma source de force pour affronter les différents obstacles .

À mon très cher père Ikhlef

De tous les pères, tu es le meilleur. aucune dédicace ne saurait exprimer ma reconnaissance et mon profond amour. Que ce travail traduit ma gratitude et mon affection pour vous et qu'il puisse vous rendre fière. Merci pour l'éducation et le soutien permanent venu de toi.

À vous, ma famille

Je dédie ce mémoire avec une profonde gratitude. Votre amour, votre soutien et vos encouragements ont été ma source de force et d'inspiration tout au long de ce voyage.

Enfin, je dédie ce travail a toutes les personnes qui m'ont aimé et soutenu, qui ont cru en moi et m'ont encouragé. votre confiance en moi a été ma plus grande motivation et je vous en suis infiniment reconnaissant.

TABLE DES MATIÈRES

Table des matières	I
Table des figures	IV
Liste des tableaux	VI
Listes des abréviations	VII
Introduction générale	1
1 Généralités sur la cryptographie	3
1.1 Introduction a la cryptographie	3
1.1.1 Définition de la cryptographie	3
1.1.2 Fonctions de la cryptographie	3
1.1.3 Terminologie	4
1.2 Evolution de la cryptographie	5
1.2.1 Histoire depuis l'antiquite	5
1.2.2 Transition de cryptographie classique vers moderne	6
1.2.3 Domaine d'application de la cryptographie	7
1.3 Attaques	7
1.3.1 Définition	7
1.3.2 Types d'attaques et techniques de cryptanalyse	7
1.4 Normes et protocole de sécurité	9
1.4.1 La Sécurité Informatique	9
1.4.2 Mécanismes de sécurité	9
1.4.2.1 Mot de passe	9

1.4.2.2	Pare-feu	9
1.4.2.3	VPN(Virtual Private Network)	10
1.4.2.4	IDS(Intrusion Detection System)	10
1.4.2.5	Signature, Fonction de hachage, Certificats	10
1.4.2.6	Les applications sécurisées	11
1.5	Conclusion	12
2	La cryptographie classique	13
2.1	Introduction	13
2.2	Définition de la cryptographie classique	13
2.3	Techniques de Chiffrement	13
2.3.1	Par Substitution	13
2.3.2	La Scytale	14
2.3.3	Chiffre de César	15
2.3.4	Chiffre de Vigenère	15
2.3.5	Chiffrement affine	17
2.3.6	Chiffrement de Hill	18
2.3.7	Machine Enigma	18
2.3.8	Par Transposition	19
2.3.8.1	Transposition simple par colonne	19
2.3.8.2	Transposition complexe par colonne	21
2.3.8.3	Transposition par carre-polybique	22
2.4	Limitations et Vulnérabilités	22
2.5	Conclusion	23
3	La cryptographie moderne	24
3.1	Introduction	24
3.2	La cryptographie à clé symétrique (dits à clé secrète)	25
3.2.1	Chiffrement par Blocs (Block Cipher)	26
3.2.2	DES (Data Encryption Standard)	28
3.2.3	AES(Advanced Encryption Standard)	31
3.2.4	Chiffrement par flux (Stream Cipher)	31
3.2.5	RC4 (Rivest Cipher 4)	32
3.3	La Cryptographie à clé asymétrique (dite publique)	33
3.3.1	Définition	33
3.3.2	Le système RSA	34
3.3.3	Le cryptosystème ELGamal	35

3.3.4	Protocole Diffie-Hellman	36
3.3.5	ECC (pour Elliptic Curve Cryptographie)	37
3.4	la cryptographie Hybride	38
3.4.1	Définition	38
3.4.2	PGP : description et fonctionnement	39
3.5	La Cryptographie quantique	41
3.5.1	Définition	41
3.5.2	Historique	41
3.5.3	Notion D'un Qubit	41
3.5.4	Nature du photon	42
3.5.5	Protocole BB84	42
3.6	Limitations et vulnérabilités	43
3.7	Conclusion	43
4	Proposition d'un système de cryptographie hybride	44
4.1	Introduction	44
4.2	Évaluation des techniques de cryptographie moderne	44
4.3	Structuration de Notre cryptosystème HAE	54
4.4	Implementations et Résultats	56
4.4.1	L'environnement de développement	56
4.4.2	Tests et Résultats	57
4.4.2.1	Résultats pour les algorithmes Asymétrique	57
4.4.2.2	Résultats pour les algorithmes symétrique	61
4.4.3	Résultats pour la proposition hybride	63
4.5	Conclusion	65
	Conclusion générale	66
	Bibliographie	67
	Resumé	70
	Abstract	70

TABLE DES FIGURES

1.1	Histoire de la cryptographie	5
1.2	Transition de la cryptographie a travers le temps	6
1.3	parefeu	9
1.4	fonctionnement d'une signature numérique	11
2.1	La Scytale	14
2.2	César	15
2.3	Tableau de Vigenere	16
2.4	La machine Enigma	19
2.5	déchiffrement complexe par colonne	22
3.1	taxonomie des techniques de cryptographie	24
3.2	la cryptographie symétrique	25
3.3	la cryptographie symétrique	26
3.4	mode ECB	27
3.5	mode CBC	27
3.6	mode CFB	28
3.7	Algorithme de DES	30
3.8	programme de clé	33
3.9	Chiffrement Asymétrique	34
3.10	fonctionnement PGP	39
3.11	polarisation d'un photon	42
4.1	Chiffrement AES	46
4.2	Transformation de substitution	47

4.3	Table de boîte de substitution AES	47
4.4	La transformation ShiftRows	48
4.5	La transformation Mix-colonne	48
4.6	La transformation Addroundkey	49
4.7	Expansion de la clé AES	50
4.8	Déchiffrement AES	51
4.9	Exemple d'une courbe ECC	52
4.10	jupyter notebook	56
4.11	comparaison entre RSA et ELGamal par rapport au temps de chiffrement et déchiffrement	57
4.12	comparaison entre RSA et ELGamal en temps de génération de clé	58
4.13	comparaison entre RSA et ELGamal en terme d'espace mémoire	59
4.14	comparaison entre RSA et ECC en terme de temps	60
4.15	comparaison entre RSA et ECC en terme d'espace mémoire	60
4.16	comparaison entre DES et AES par rapport au temps de chiffrement et déchiffrement	61
4.17	comparaison entre DES et AES en terme d'espace mémoire	62
4.18	comparaison entre AES et RC4 en terme de temps de chiffrement et déchiffrement	62
4.19	comparaison entre AES et RC4 en terme d'espace mémoire	63
4.20	comparaison entre l'approche proposée et AES	64

LISTE DES TABLEAUX

2.1	méthode de chiffrement de vigènère	17
2.3	Déchiffrement simple par colonne	20
2.2	chiffrement simple par colonne	20
2.4	Chiffrement complexe par colonne	21
3.1	les avantages et les Inconvénients symétrique et asymétriques	38
4.1	comparaison entre RSA et ElGamal	57
4.2	comparaison entre RSA et ElGamal en terme du temps de génération de clés	58
4.3	comparaison entre RSA et ECC	59
4.4	comparaison entre DES et AES	61

LISTES DES ABRÉVATIONS

AES : Advanced Encryption Standard

CA : Certification Authority

CFB : Cipher Feedback Block

CBC : Cipher Block Chaining

DES : Data Encryption Standard

DH : Diffie Hellman

DL : Discrete Logarithm

ECC : Elliptic Curve Cryptography

ECB : Electronic Codebook Block

SSL : Secure Socket Layer

IBM : International Business Machines

OFB : Output Feedback Block

HCF : Highest Common Factor

PGP : Pretty Good Privacy

PKI : Public Key Infrastructure

RSA : Rivest Shamir Aldeman

RC4 : Rivest Cipher 4

XOR : Exclusive Oipher R

HAE : Hybrid Advanced Elliptic

INTRODUCTION GÉNÉRALE

L'information est un élément constitutif et déterminant dans tous les domaines. Depuis l'invention de l'écriture, l'humanité exprime le besoin de transmettre leurs informations de manière sécurisée en les rendant incompréhensibles pour toute personne étrangère à l'échange, c'est-à-dire que les messages ne peuvent pas être compris par l'ennemi, même s'ils sont interceptés. Ils se servaient donc d'outils permettant de garder leurs confidences hors d'atteinte des yeux indiscrets : signes et symboles incompréhensibles, figures ou couleurs, usage d'expression ou phrases convenues d'avoir un sens spécifique qui diffère de l'ordinaire. La progression de ces outils primitifs à travers le temps, a permis de concevoir des règles de sécurité plus efficaces et plus logiques qui ont donné naissance à la cryptologie.

La cryptologie est une science mathématique qui étudie les communications secrètes. Elle est composée de deux domaines d'étude complémentaires : la cryptographie (chiffrement des données) et la cryptanalyse (analyse des techniques de chiffrement pour les attaquer). Le rôle des cryptographes est de construire des systèmes de chiffrement robustes pour protéger les données sensibles. L'objectif des cryptanalyses est de casser ces systèmes.[13]

La cryptographie constitue l'art et la science de sécuriser les communications face à des parties adverses. En se fondant sur l'usage de codes, elle garantit la protection des informations, restreignant l'accès exclusivement aux destinataires désignés. Les piliers de la cryptographie englobent la confidentialité, l'intégrité, la non-répudiation et l'authentification des données. Son influence s'étend largement, touchant des secteurs variés tels que les transactions en ligne sécurisées, les signatures numériques et la protection des mots de passe.[48]

Au fil du temps, la cryptographie a évolué avec l'évènement de la machine Enigma pendant la Seconde Guerre mondiale, représentant un jalon dans la cryptographie mécanique. Cependant, l'essor de l'informatique a ouvert la voie à la cryptographie moderne. Les années 1970 ont vu l'émergence d'algorithmes à clé publique, comme le RSA, révolutionnant la sécurité des communications numériques.[48]

Aujourd'hui, la cryptographie moderne englobe un large éventail de techniques, y compris les algorithmes de chiffrement symétrique et asymétrique, les fonctions de hachage et les protocoles de sécurité.

Le choix entre les algorithmes de cryptographie symétrique et asymétrique reste une question pour les professionnels de la sécurité informatique. Chaque type d'algorithme présente des avantages et des inconvénients distincts en termes de sécurité, de performances et de complexité.

L'objectif de notre mémoire est de faire une comparaison entre les algorithmes de cryptographie symétrique et asymétrique afin de déterminer leurs forces et leurs faiblesses respectives. En outre, nous proposons une nouvelle approche cryptographique. En examinant les aspects tels que le temps de chiffrement et de déchiffrement, la taille de la clé et la mémoire. Les résultats de cette recherche fourniront des informations précieuses pour les décideurs et les professionnels de la sécurité informatique, les aidant à choisir les solutions de cryptographie les plus adaptées à leurs besoins

spécifiques.

Ce mémoire est organisé en cinq chapitres :

Le premier chapitre explore l'évolution de la cryptographie depuis ses débuts jusqu'à aujourd'hui, en mettant en avant les concepts clés de ce domaine. Nous abordons la cryptographie, les notions essentielles sur les diverses fonctions utilisées en cryptographie et les stratégies d'attaques potentielles, ainsi que les mécanismes de sécurité.

Le deuxième chapitre traite la cryptographie classique et explore ses diverses techniques. Nous examinons en détail les méthodes traditionnelles de cryptographie, en mettant l'accent sur les techniques utilisées pour chiffrer et déchiffrer les messages.

Le troisième chapitre met l'accent sur la cryptographie moderne et ses multiples approches, en mettant en avant quelques algorithmes les plus réputés soit symétrique et asymétrique.

Dans le quatrième chapitre, on propose une approche hybride qui combine les bénéfices de la cryptographie symétrique et asymétrique. Par la suite, nous exposons les résultats de nos analyses et tests expérimentaux.

Ce mémoire se termine par une conclusion générale qui résume tout ce qui a été accompli ou obtenu, tout en évoquant les améliorations qui pourraient être apportées et les perspectives qu'elles offrent.

CHAPITRE 1

GÉNÉRALITÉS SUR LA CRYPTOGRAPHIE

1.1 Introduction a la cryptographie

Depuis l'antiquité, l'homme a perçu le besoin de cacher des informations personnelles ou confidentielles en utilisant des codes qui ont servi à protéger le contenu de certains messages des inévitables curieux d'où l'apparence de la cryptographie. Avant d'expliquer la cryptographie il faut tout d'abord la distinguer du terme « Stéganographie » et « cryptographie » car les deux méthodes visent à protéger l'information mais chacune à sa façon :

Stéganographie : cacher le message pour que l'ennemi ne le trouve pas.

Cryptographie : rendre le message incompréhensible par l'ennemi.

1.1.1 Définition de la cryptographie

Le terme cryptographie vient en effet de deux mots grecs : Kruptus qu'on peut traduire comme secret et Graphein pour écriture. La cryptographie est l'art de cacher l'information pour qu'elle soit incompréhensible, elle désigne l'ensemble des techniques qui permettent de chiffrer les messages.

il existe deux types de cryptographies : la cryptographie classique et la cryptographie moderne.[48].

1.1.2 Fonctions de la cryptographie

La cryptographie permet à deux personnes de communiquer entre eux à travers un canal peu sécurisé de telle sorte qu'on opposant ne puisse pas comprendre ce qui est échangé, en utilisant une clé appelée clé de chiffrement pour le processus de chiffrement, et pour rendre l'information à nouveau compréhensible on utilise une clé appelée clé de déchiffrement pour le processus de déchiffrement.[28]

Les principaux objectifs garantis par l'application de la cryptographie sont :

- **La confidentialité** : Le message chiffré ne doit pas être compréhensible que par les destinataires légitimes. Il ne peut pas être déchiffré par un intrus.
- **L'intégrité** : Le destinataire peut vérifier le message reçu qui n'a pas été modifié en chemin par l'utilisation de mécanisme de la signature électronique.
- **L'authentification** : Un mécanisme pour permettre d'authentifier les utilisateurs de façon à limiter l'accès aux données, serveurs et ressources aux seules personnes autorisées.
- **Non-répudiation** : L'information est l'assurance qu'aucun correspondant ne peut nier la transaction. Elle se décompose :
 - Non-répudiation d'origine : l'émetteur ne peut nier avoir écrit le message.
 - Non-répudiation de réception : le receveur ne peut nier avoir reçu le message .
 - Non-répudiation de transmission : l'émetteur du message ne peut nier avoir envoyé le message.

1.1.3 Terminologie

Les principaux termes utilisés dans la cryptographie sont[28] :

Cryptologie : c'est une science mathématique regroupant la cryptographie et la cryptanalyse.

Cryptographie : est l'étude des méthodes donnant la possibilité d'envoyer des données de manière confidentielle sur un rapport donné.

Cryptanalyse : opposée à la cryptographie, elle a pour but de retrouver le texte en clair à partir de textes chiffrés en déterminant les failles des algorithmes utilisés.

Crypto-système : un crypto système est constitué d'un algorithme cryptographique ainsi toutes les clés possibles et tous les protocoles qui le font fonctionner.

Cryptogramme : Texte chiffré : est le résultat de l'application d'un chiffrement d'un texte clair.

Texte clair : est le message à chiffrer.

Chiffrement : la fonction permettant de transformer une donnée (texte, message, ...) afin de la rendre incompréhensible par une personne autre que celui qui a créé le message et ainsi que le destinataire.

Déchiffrement : la fonction permettant de retrouver le texte clair à partir du texte chiffré.

Crypter : synonyme de chiffrer.

Coder/Décoder : c'est une méthode ou un algorithme permettant de modifier la mise en forme d'un message sans introduire d'élément secret.

Clé : est un paramètre utilisé en entrée d'une opération cryptographique. On distingue deux types de clés :

Clés symétriques : il s'agit de clés utilisées pour le chiffrement ainsi que pour le déchiffrement. On parle alors de chiffrement symétrique ou à clé secrète.

Clés asymétriques : il s'agit de clés utilisées dans le cas du chiffrement asymé-

trique ou à clé publique. Dans ce cas, une clé différente est utilisée pour le chiffrement et pour le déchiffrement.

1.2 Evolution de la cryptographie

1.2.1 Histoire depuis l'antiquité

L'antiquité vers 600ans avant j.-C, le roi de Babylone Nabuchodonosor écrivait le message qu'il souhaitait transmettre à ses généraux, sur le crane préalablement rasé de ses esclaves. Il attendait que leurs cheveux repoussent avant de les envoyer chez ces généraux, qui rasent de nouveau les cheveux des messages pour lire le texte. Dans le Xème et VIIème siècle avant j.-C les Grecs ont utilisé le chiffrement de la scytale spartiate c'est un exemple de chiffrement par transposition. Des lettres étaient écrites sur une longue et mince bande de cuir enveloppée autour d'un cylindre, pour déchiffrer ces lettres, il devait faire un cylindre d'un diamètre identique à celui utilisé pour le chiffrement, il lui suffit d'enrouler la scytale autour de ce cylindre pour obtenir les lettres en clair, le diamètre du cylindre était la clé. Dans 200avant j.-C apparait les premiers systèmes de cryptographie, ce sont les chiffrements par substitutions, Dans le 1er siècle avant j.-C lorsque Jules César envoyait des messages à ses généraux, il ne faisait pas confiance à ses messagers. Il remplaçait donc tous les A contenus dans ses messages par des D, les B par E, et ainsi de suite pour tout l'alphabet. Seule la personne connaît la règle " décalage par trois " pouvait déchiffrer ses messages. Et voilà comment tout a commencé.[24]

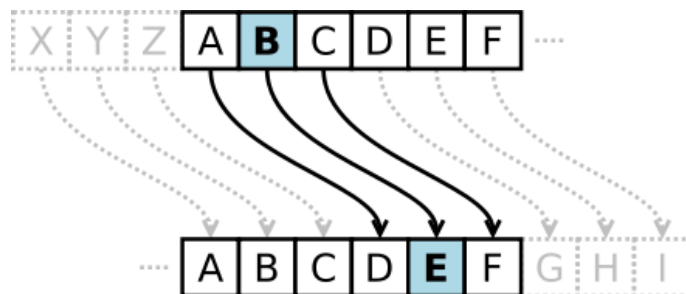


FIGURE 1.1 – Histoire de la cryptographie [8]

En 1918, L'allemand Arthur Scherius donne naissance à la célèbre machine Enigma, dont le principe est de remplacer une lettre par une lettre, et les règles de remplacement changent à chaque lettre.

EN 1976, Martin Hellman ont proposé la cryptographie à clé publique.

En 1978, trois mathématiciens américains, Rivest, Shamir, Adleman, ont proposé le système de cryptage à clé publique RSA, qui a conduit à la naissance explosive des applications de cryptage civiles.

En 1984, L'algorithme ROT13 est utilisé dans le USENET et devient le premier exemple de l'utilisation des clés publiques.

En 1987, Le RC4 est développé par Ronald L. Rivest pour la RSA Security et sera gardé secret jusqu'en 1994, où l'algorithme est rendu public.

En 1998, L'algorithme Rijndael est final et soumis au NIST pour devenir le nouveau

standard du chiffrement avancé : l'AES. Quinze autres algorithmes font partie du groupe dont MARS, RC6, Serpent et Twofish.

En 2000, Rijndael devient l'AES, le standard du chiffrement avancé.[35]

1.2.2 Transition de cryptographie classique vers moderne

La transition de la cryptographie classique à la cryptographie moderne marque une évolution significative dans l'histoire de la sécurité des communications. Alors que la cryptographie classique se concentrait principalement sur des méthodes telles que le chiffrement par substitution et le chiffrement par transposition, la cryptographie moderne a émergé avec l'évènement de techniques plus avancées et complexes, telles que le chiffrement à clé publique et les algorithmes de hachage cryptographique.

Cette transition a été motivée par la nécessité de sécuriser les communications dans un environnement numérique en constante évolution, où les menaces sont devenues plus sophistiquées.[45]

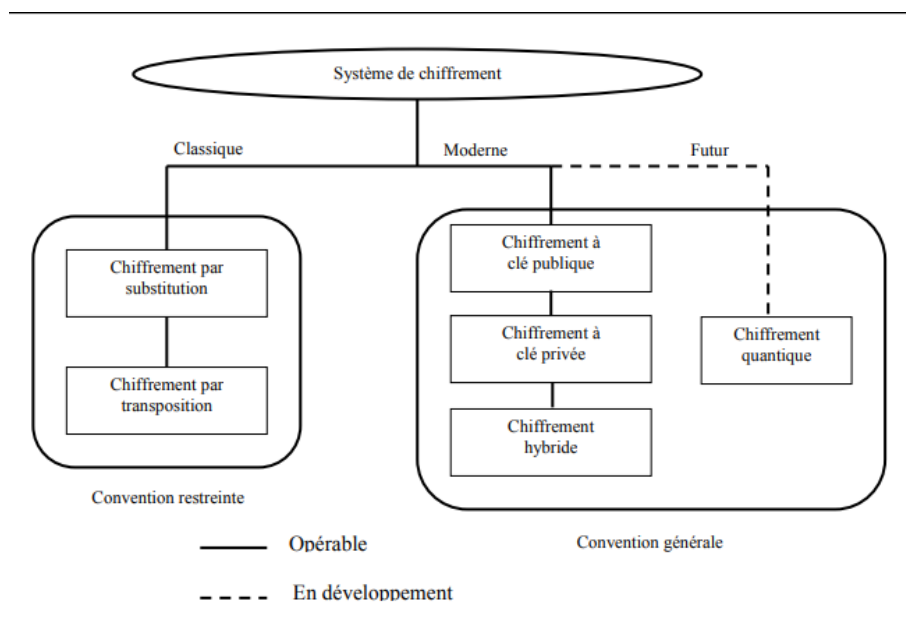


FIGURE 1.2 – Transition de la cryptographie a travers le temps [5]

Cette figure décrit deux périodes de La cryptographie, la période avant les ordinateurs durant laquelle, les principaux outils utilisés consistent à remplacer des caractères par d'autres et les transposer dans des ordres différents tout en gardant secrètes les procédures de chiffrement ou de déchiffrement. Sans cela le système est complètement inefficace, puisque n'importe qui peut déchiffrer le message codé. On appelle généralement cette classe de méthodes le chiffrement à usage(Convention restreinte).

La deuxième période, celle de la cryptographie moderne, repose sur des principes mathématiques avancés, tels que la théorie des nombres, l'algèbre linéaire et la théorie de l'information.

1.2.3 Domaine d'application de la cryptographie

La cryptographie est une discipline qui se trouve dans de nombreux domaines et industries. Voici quelques-uns des principaux domaines d'utilisation de la cryptographie :

- **Sécurité des communications** : La cryptographie est largement utilisée pour sécuriser les communications sur Internet, y compris les transactions financières en ligne, les communications par e-mail, les discussions instantanées, etc. Les protocoles tels que SSL/TLS utilisent la cryptographie pour chiffrer les données transmises sur le réseau, assurant ainsi la confidentialité et l'intégrité des informations.
- **Sécurité des données** : La cryptographie est utilisée pour protéger les données sensibles, telles que les informations personnelles, les données médicales, les informations gouvernementales, etc.
- **Sécurité des transactions financières** : Les systèmes de paiement électronique et les transactions financières en ligne utilisent la cryptographie pour sécuriser les transactions et prévenir la fraude. Les protocoles de sécurité tels qu'EMV (Europay, MasterCard, Visa) et les crypto-monnaies comme Bitcoin reposent sur des techniques de cryptographie.
- **Sécurité des objets connectés** : la cryptographie est devenue essentielle pour assurer la sécurité des données échangées entre ces appareils et les serveurs.

Maintenant, passons à un survol général de la sécurité et les attaques informatiques.

1.3 Attaques

1.3.1 Définition

Une attaque est l'exploitation d'une faille d'un système informatique (système d'exploitation, logiciel ou bien même de l'utilisateur) à des fins non connues par l'exploitant des systèmes et généralement préjudiciables. Ces attaques sont pour la plupart lancées automatiquement à partir de machines infectées (par des virus, chevaux de Troie, vers, etc).[44]

1.3.2 Types d'attaques et techniques de cryptanalyse

Sont des techniques utilisées par les attaquants pour tenter de compromettre les systèmes de cryptographie et de contourner les mécanismes de sécurité mis en place.[44]

Voici quelques-unes des principales attaques cryptographiques :

- **Attaques par force brute** : Consiste à essayer toutes les combinaisons possibles de clés pour chiffrer un message chiffré. Cette attaque est généralement inefficace contre les systèmes de cryptographie modernes en raison de la longueur des clés.

● **Attaques par recherche exhaustive** : Recherche systématique de vulnérabilités dans les algorithmes cryptographiques en analysant toutes les possibilités. Cela pour trouver des faiblesses dans les algorithmes.

● **Attaques par analyse de fréquence** : Consiste à analyser les fréquences d'apparition de certains symboles (lettres, chiffres, etc.) dans un message chiffré pour essayer de deviner la clé de chiffrement utilisée. Cette méthode peut être efficace pour certains types de chiffrement à substitution simple.

● **Attaques par collision** : Chercher à trouver deux messages distincts qui produisent la même valeur de hachage ou la même empreinte digitale. Les attaques par collision sont souvent utilisées pour contourner les fonctions de hachage cryptographie.

● **Attaque par canaux auxiliaires** : sont une famille d'attaques consistant à extraire une information par la récupération et l'interprétation de signaux émis « involontairement » par un système (vibrations, activité, consommation électrique, temps d'exécution. . .). Elles portent dans la majorité des cas sur l'extraction d'informations relevant de la cryptographie, comme les clés de chiffrement.

● **Attaque de l'homme du milieu** : Intercepter et éventuellement modifier les communications entre deux parties sans qu'elles en soient conscientes.

Techniques de Cryptanalyse

La cryptanalyse permet d'étudier la sécurité des procédés de chiffrement utilisés en cryptographie. Ainsi, elle désigne habituellement les techniques qui permettent d'extraire de l'information sur des secrets en observant uniquement les données publiques d'un crypto-système.[41]

Et suivant les données qu'elle nécessite, on distingue habituellement quatre méthodes de cryptanalyse :

● **Attaque sur texte chiffré seul (Ciphertext-Only Attack)** : le cryptanalyste possédant des exemplaires chiffrés des messages, essaye de faire des hypothèses sur les messages originaux qu'il ne possède pas en vue de retrouver la clé de déchiffrement. Dans ce cas, la cryptanalyse sera difficile à cause du manque d'informations à disposition.

● **Attaque à texte clair connu (known-plaintext attack)** : le cryptanalyste essaye de retrouver la clé de déchiffrement à partir de messages ou de parties de messages en clair possédés et de leurs versions chiffrées correspondantes.

● **Attaque à texte clair choisi (chosen-plaintext attack)** : Consiste à retrouver la clé de déchiffrement à partir de messages en clair, et en ayant la possibilité de générer les versions chiffrées de ces messages avec un algorithme considéré comme une boîte noire.

● **Attaque à texte chiffré choisi (chosen-ciphertext attack)** : le cryptanalyste possède des messages chiffrés et demande la version en clair de certains de ces messages pour mener l'attaque (retrouver la clé de déchiffrement).

1.4 Normes et protocole de sécurité

1.4.1 La Sécurité Informatique

La sécurité informatique est l'ensemble de politiques et de mécanismes de protection et de contrôle mis en œuvre pour réduire les vulnérabilités d'un système contre les menaces accidentelles ou intentionnelles pour éviter les erreurs, afin d'assurer le bon fonctionnement de système.[44]

1.4.2 Mécanismes de sécurité

les mécanismes sont conçus pour détecter, prévenir et lutter contre une attaque de sécurité.[32]

Parmi les mécanismes de sécurité existants, les plus utilisés on trouve :

1.4.2.1 Mot de passe

Un mot de passe est une suite de caractères, pas forcément constituée que de chiffres et de lettres, avec ou sans Signification, qui doit être tenue secrète pour éviter qu'une entité non autorisée puisse accéder à une ressource ou un service.[50]

1.4.2.2 Pare-feu

Un pare-feu (connu sous le nom de firewall), est un système permettant de protéger un ordinateur, ou un réseau d'ordinateurs des intrusions provenant d'un réseau tiers (notamment Internet).[50] Le pare-feu est un système permettant de filtrer les paquets de données échangés avec le réseau, il s'agit ainsi d'une passerelle filtrante comportant au minimum les interfaces réseau suivantes :

- Une interface pour le réseau à protéger (réseau interne).
- Une interface pour le réseau externe.

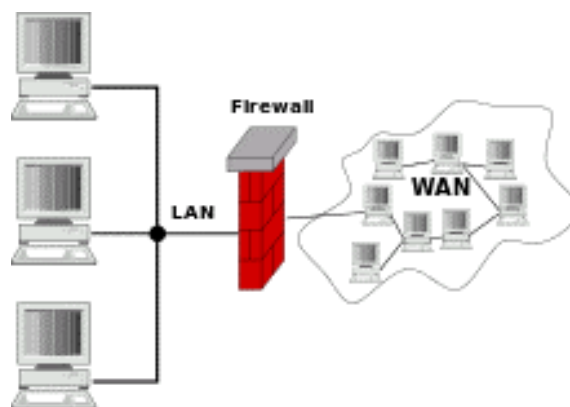


FIGURE 1.3 – parefeu
[8]

1.4.2.3 VPN(Virtual Private Network)

Un réseau privé virtuel (VPN) est constitué de liaisons virtuelles sur Internet entre des sites distants appartenant à une même société ou à un même organisme. En chiffrant les données, tout se passe exactement comme si la connexion se faisait en dehors d'Internet. Un des grands intérêts des VPN est de réaliser des réseaux privés à moindre coût en chiffrant les données.[49]

Les principaux protocoles permettant l'établissement d'un VPN :

- le Protocole **PPTP**(Point to Point Tunnelling Protocol) mis au point par la société Microsoft ;
- le Protocole **L2TP** (Layer Two Tunnelling Protocol) proposé par IETF(Internet Engineering Task Force).
- le Protocole **IPsec** (Internet Protocol Security) proposé par IETF.

1.4.2.4 IDS(Intrusion Detection System)

Est un mécanisme destiné à repérer des activités anormales ou suspectes sur la cible analysée. Il offre ainsi un moyen de prévention face aux risques d'intrusion. Il existe deux grandes familles distinctes d'IDS :

- Les N-IDS (Network-based Intrusion Détection System), ils assurent la sécurité au niveau du réseau.
- Les H-IDS (Host-based Intrusion Détection System), ils assurent la sécurité au niveau des hôtes.[47]

1.4.2.5 Signature, Fonction de hachage, Certificats

L'utilisation des techniques de signature électronique et des certificats sont la base d'un commerce électronique sécurisé :

- **Signature numérique** : L'un des principaux avantages de la cryptographie à clé publique est qu'elle offre la possibilité d'utiliser des signatures numériques, qui garantissent l'authentification, le contrôle de l'intégrité des données et la non-répudiation de l'information[17]

La figure 1.4 suivante illustre le processus de fonctionnement d'une signature numérique. Elle montre comment un fichier est chiffré avec une clé privée, puis déchiffré avec la clé publique correspondante, garantissant ainsi l'authenticité et l'intégrité des données.

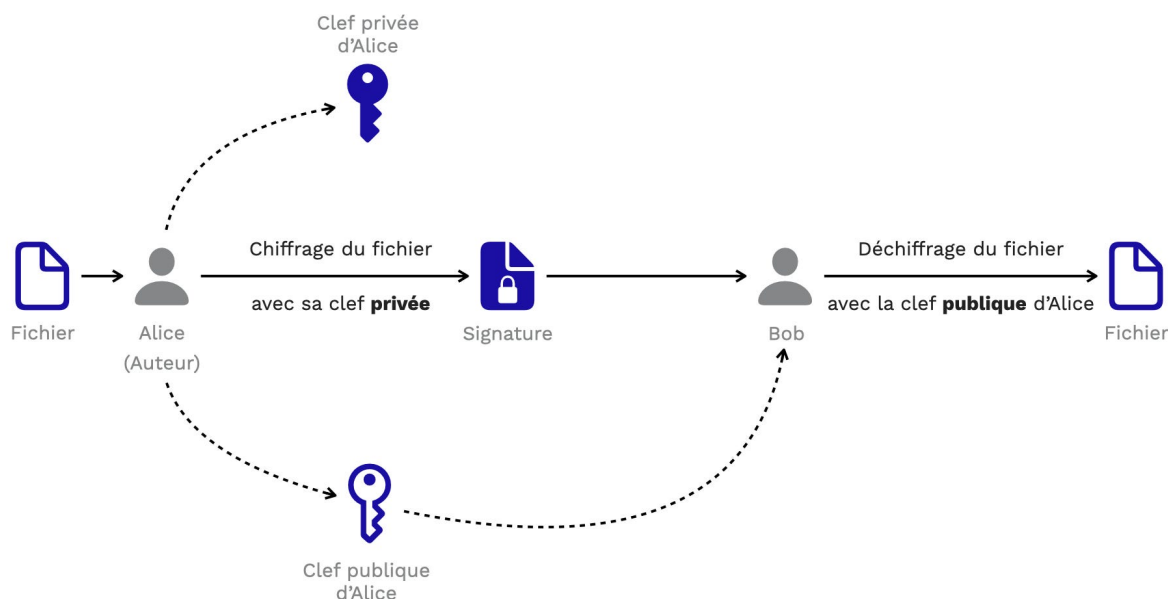


FIGURE 1.4 – fonctionnement d’une signature numérique [4]

- **Les fonctions de hachage** : appelée fonction de condensation, est une fonction qui convertit une chaîne de longueur quelconque en une chaîne de taille inférieure et généralement fixe, la chaîne résultante est appelée empreinte ou condensé de la chaîne initiale à sens unique, c’est-à-dire qu’elle doit permettre de trouver facilement l’empreinte à partir du message, et d’empêcher de retrouver le message à partir de l’empreinte.[20]
- **Certificat électronique** : Les certificats électroniques sont utilisés principalement pour assurer l’authentification. Le certificat est en quelque sorte une carte d’identité numérique. Pour en obtenir un, il faut s’adresser à une autorité de certification (Certificate Authority, ou CA). « Un certificat est un document numérique qui contient toutes les coordonnées d’un interlocuteur utiles pour communiquer avec d’autres, ainsi que sa clé publique ».

1.4.2.6 Les applications sécurisées

PGP (Pretty Good Privacy)

PGP est la solution la plus connue des usagers pour rendre confidentielle la transmission de messages et authentifier l’émetteur. En 1991, Philip Zimmermann a développé et partagé sur le web sa création.

S/MIME (Secure / Multipurpose Internet Mail Extensions)

S/MIME (pour Secure MIME, que l’on pourrait traduire par extensions du courrier électronique à but multiples et sécurisées). est une extension sécurisée qui propose, comme PGP des services d’authentification et de confidentialité. Ainsi, S/MIME permet de chiffrer tout type de contenu ainsi que les clés de chiffrement à destination de un ou de divers destinataires.

SSH (Secure Shell)

Il s'agit d'un protocole permettant à un client (un utilisateur ou bien même une machine) d'ouvrir une session interactive sur une machine distante (serveur) afin d'envoyer des commandes ou des fichiers de manière sécurisé.

SSL (Secure Socket Layer)

Le protocole SSL a été conçu pour assurer une communication confidentielle et fiable entre deux applications (un client et un serveur), pour identifier le serveur et parfois le client. SSL nécessite un protocole de transport sûr pour la transmission et la réception de données.

SET (Secure Electronic Transaction)

SET est basé sur l'utilisation d'une signature électronique au niveau de l'acheteur et une transaction mettant en jeu non seulement l'acheteur et le vendeur, mais aussi leurs banques respectives.

1.5 Conclusion

Dans ce chapitre, nous avons abordé la cryptographie dans son ensemble, en mettant particulièrement l'accent sur les attaques et la sécurité informatique. Nous reviendrons sur ces sujets dans le prochain chapitre pour explorer la cryptographie classique et ses différentes techniques.

CHAPITRE 2

LA CRYPTOGRAPHIE CLASSIQUE

2.1 Introduction

Dans ce chapitre, nous allons effectuer un survol de la cryptographie classique, en examinant ses techniques de chiffrement ainsi que les méthodes de cryptanalyse associées. Nous aborderons également quelques algorithmes classiques tels que le Chiffre de César, le chiffre de Hill, etc. tout en mettant en lumière les limitations et les vulnérabilités à la cryptographie classique.

2.2 Définition de la cryptographie classique

La cryptographie classique concerne la période de l'antiquité jusqu'à l'apparition des ordinateurs. Elle traite des systèmes reposant sur les lettres et les caractères d'une langue naturelle (allemand, anglais, français...). Et inclut tous les mécanismes et algorithmes basés sur des fonctions mathématiques ou logiques (exemples : César, Vigenere). [32]

Ces méthodes se décomposent en deux grandes familles de chiffrement :

- Par substitution
- Par transposition ou par permutation

2.3 Techniques de Chiffrement

2.3.1 Par Substitution

Principe

Un chiffrement par substitution est un algorithme par lequel chaque caractère du message en clair (écrit dans un alphabet donné) est substitué (remplacé) par un autre caractère dans le message chiffré (qui peut être écrit dans un alphabet différent de

celui du message clair) selon une règle convenue.

En cryptographie classique, On distingue plusieurs types de crypto systèmes par substitution[20] :

- **La substitution simple ou mono-alphabétique** : Chaque caractère du texte en clair est remplacé par un caractère correspondant dans le texte chiffre. Par exemple, le chiffrement de César, le chiffrement d’affine.
- **La substitution poly-alphabétique** : Chaque caractère du texte en clair est remplacé par un caractère correspondant dans un alphabet différent qui est choisi en fonction de la position du caractère dans le texte. Par exemple, le chiffrement de Vigenère, Playfair.
- **La substitution homophonique** : Permet de faire correspondre à chaque lettre du message en clair un ensemble possible d’autres caractères.
- **La substitution polygamique** : Chaque groupe de caractères du texte en clair est remplacé par un groupe de caractères correspondants dans le texte chiffre. Par exemple, le chiffrement de Hill.

Examinons quelques-uns des chiffrements classiques utilisés dans le domaine de la cryptographie :

2.3.2 La Scytale



FIGURE 2.1 – La Scytale
[8]

Entre le Xe et le VIIe siècle avant Jules César, les Grecs utilisaient des scytales, des sortes de bâtons en bois. Quand l’émetteur voulait communiquer, il enroulait une bande de cuir sur la scytale et y inscrivait le message (une lettre par bout de bande). Une fois la bande déroulée, les lettres n’étaient plus ordonnées et n’avaient donc plus aucun sens. Le seul moyen de pouvoir comprendre le message était d’enrouler la bande sur une scytale de même diamètre pour que les lettres puissent s’aligner correctement.[21]

Exemple :

Le message en Clair est. : KILL KING TOMORROW MIDNIGHT

Le message crypté est : KTMIOILMDLONKRIIRGNOHGWT

2.3.3 Chiffre de César



FIGURE 2.2 – César

[45]

Le Chiffre de César est la méthode de cryptographie, par substitution mono-alphabétique, la plus ancienne (Ier siècle av. J-C.). Cette méthode est utilisée dans l'armée romaine.

Son principe est simple, il fonctionne par décalage des lettres de l'alphabet de n positions, par exemple décalage de 3 positions :

Texte 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
Texte 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

Exemple :

Le message clair est : CECI EST UN CRYPTOGRAMME

On décale les lettres de 3 positions :

Le texte chiffré est : FHFL HVW XQ FUBSWRJUDPPH

Malheureusement, on comprendra que ce système est très peu sûr, puisqu'il n'y a que 26 lettres dans l'alphabet, donc seulement 25 façons de chiffrer un message avec le code de César (on ne peut substituer une lettre par elle-même).[21]

2.3.4 Chiffre de Vigenère

C'est un système de chiffrement par substitution polyalphabétique. La grande force du chiffre de Vigenère est que la même lettre sera chiffrée de différentes manières (une même lettre du message clair peut, suivant sa position dans celui-ci, être remplacée par des lettres différentes). Il consiste à utiliser 26 alphabets décalés pour chiffrer un message. Les 26 alphabets décalés sont représentés dans ce qu'on appelle un carré de Vigenère. Ce chiffre utilise une clé qui définit le décalage pour chaque lettre du message. Une clé se présente généralement sous la forme d'un mot ou d'une phrase.[25]

Principe de Chiffrement

A chaque lettre en clair, on sélectionne la colonne correspondante tandis que la lettre de la clé se sélectionne par ligne, au croisement de la ligne et de la colonne on trouve la lettre chiffrée.

Principe de Déchiffrement

On regarde pour chaque lettre de la clé répétée, la ligne correspondante sur laquelle on cherche la lettre chiffrée. La première lettre de la colonne que l'on trouve ainsi est la lettre déchiffrée.

Formellement

Les opérations de chiffrement et de déchiffrement sont, pour chaque lettre, celles du code de César. En désignant la i ème lettre du texte clair par $Texte[i]$, la i ème du chiffré par $Chiffré[i]$, et la i ème lettre de la clé, répétée suffisamment de fois, par $Clés[i]$, elle se formalise par :

$$\begin{aligned} Chiffré[i] &= (Texte[i] + Clés[i]) \pmod{26} \\ Texte[i] &= (Chiffré[i] - Clés[i]) \pmod{26} \end{aligned} \tag{2.1}$$

Pour le chiffrement : il suffit d'effectuer l'addition des deux lettres puis de soustraire 26 si le résultat dépasse 26.

Pour le déchiffrement : il suffit d'effectuer la soustraction et d'ajouter 26 si le résultat est négatif.

	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
A	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
B	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	A
C	C	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
D	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
E	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	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	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

FIGURE 2.3 – Tableau de Vigenere [41]

Application

Chiffrement du message en clair "Bonjour" pour le mot clé "mot" :

Texte en clair	B	O	N	J	O	U	R
clé	M	O	T	M	O	T	M
$(L[i]+clé[i])\text{mod}26$	13	2	6	21	2	13	3
Texte Chiffre	N	C	G	V	C	N	D

TABLE 2.1 – méthode de chiffrement de vigénère

2.3.5 Chiffrement affine

1.Principe

Le chiffrement affine est une méthode de chiffrement par substitution mono-alphabétique. Il s'agit d'une version améliorée du chiffre de César. Ce chiffrement est réalisé à l'aide d'une fonction appelée fonction d' affine :

$$F(x) = a \cdot x + b \tag{2.2}$$

On choisit deux entiers a et b, de tel sorte que

$$(a, b) \in [0,25] \quad \text{et} \quad \text{PGCD}(a, 26) = 1$$

2.Chiffrement

$$C = (a \cdot L + b) \quad \text{mod } 26 \tag{2.3}$$

3.Déchiffrement

Pour que l'on puisse déchiffrer, il faut que l'opération inverse soit possible. C'est-à-dire que l'équation :

$$y = a * x + b \quad \text{mod } 26 \tag{2.4}$$

ait une solution unique (cela revient à dire que la fonction f doit être injective).

Théorème

L'équation $y = a * x + b \text{ mod } 26$ admet une solution unique $x \in \mathbb{Z}/26$ pour tout $b \in \mathbb{Z}/26$ si $\text{PGCD}(a, 26) = 1$. On a donc 12 possibilités pour choisir a (1,3,5,7,9,11,15,17,19,21,23,25). Par contre, b doit être quelconque. On a donc 12x26 clés possibles pour le chiffrement d' affine.

Considérons maintenant la fonction f du chiffrement et $m = 26$. On suppose que $\text{PGCD}(a, 26) = 1$. L'équation $y = a * x + b \text{ mod } 26$ admet alors une solution. Pour la déterminer il faut faire appel à la notion d'inverse modulaire définie par :

Soit $a \in \mathbb{Z}/\mathbb{Z}26$, l'inverse de a est un élément $a^{-1} \in \mathbb{Z}/\mathbb{Z}26$ tel que :

$$a * a^{-1} = 1 \quad \text{mod } 26. \tag{2.5}$$

La fonction de déchiffrement est :

$$L = a^{-1} \times (C - b) \quad \text{mod } 26. \tag{2.6}$$

2.3.6 Chiffrement de Hill

Il consiste à chiffrer le message en substituant les lettres du message, non plus lettre par lettre, mais par groupe de lettres. Il permet ainsi de rendre plus difficile de le casser par observation des fréquences. C'est un chiffrement à base de l'algèbre matricielle, la substitution se fait à l'aide de m équations linéaires. L'algorithme remplace m lettres successives par m lettres chiffrées.[42]

1.Chiffrement

Les lettres sont remplacées par leur rang suivant l'alphabet. On choisit une clé k sous forme d'une matrice de 2×2 telle que $\text{PGCD}(\det(k), 26) = 1$ Chaque paire de lettres L_k et L_{k+1} du message en clair sont remplacées par C_k et C_{k+1} :

$$\begin{pmatrix} C_k \\ C_{k+1} \end{pmatrix} = k * \begin{pmatrix} L_k \\ L_{k+1} \end{pmatrix} \pmod{26} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} * \begin{pmatrix} L_k \\ L_{k+1} \end{pmatrix} \pmod{26} \quad (2.7)$$

2.Déchiffrement

Pour déchiffrer, le principe est le même que pour le chiffrement : on prend les lettres deux par deux, puis on les multiplie par une matrice. Cette matrice doit être l'inverse de matrice de chiffrement modulo 26.

$$\begin{pmatrix} L_k \\ L_{k+1} \end{pmatrix} \begin{pmatrix} C_k \\ C_{k+1} \end{pmatrix} \pmod{26} = \begin{pmatrix} d & -b \\ -c & a \end{pmatrix} * \begin{pmatrix} C_k \\ C_{k+1} \end{pmatrix} \pmod{26} \quad (2.8)$$

2.3.7 Machine Enigma

Chaque lettre est remplacée par une autre, l'astuce est que la substitution change d'une lettre à l'autre. La machine est alimentée par une pile électrique. Quand on appuie sur une touche du clavier, un circuit électrique est fermé, et une lampe s'allume qui indique quelle lettre codée l'on substitue. Concrètement, le circuit électrique est constitué de plusieurs éléments en chaîne :[45]

- Le tableau de connexions, où des paires de lettres sont échangées au moyen de fiches. Ce tableau permet d'effectuer jusqu'à 6 paires d'échanges, constituant ainsi une permutation particulière.
- les rotors : Un rotor est également une permutation mais cette fois quelconque. À chaque lettre en entrée correspond une autre lettre.
- Le réflecteur, situé à la fin des rotors, permet de renvoyer le flux de lettres à travers les rotors et le tableau de connexions dans le sens inverse.
En résumé, le codage d'une lettre sur la machine Enigma implique son passage à travers le tableau de connexions, les rotors et le réflecteur. À chaque étape, la lettre est substituée selon les permutations effectuées par ces éléments, avant d'être renvoyée dans le sens inverse pour être affichée.

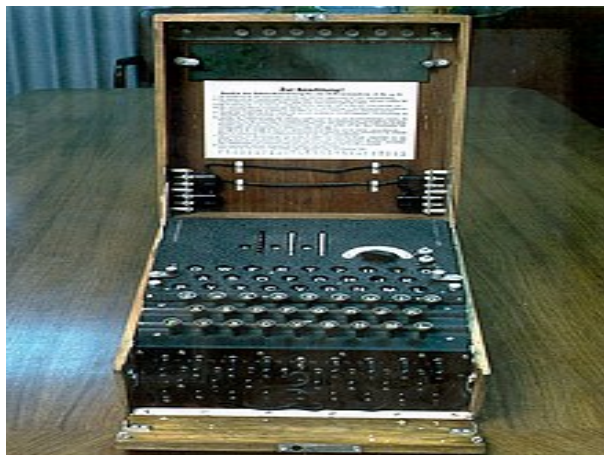


FIGURE 2.4 – La machine Enigma
[3]

La figure 2.4 illustre le fonctionnement de base de la machine Enigma pour le chiffrement de messages. Enigma est une machine électromécanique utilisée par les forces armées allemandes pendant la Seconde Guerre mondiale pour chiffrer leurs communications de manière polyalphabétique.

Lorsqu’une lettre est tapée sur le clavier, un signal électrique parcourt un circuit complexe à travers un tableau de connexions, des rotors qui tournent d’un cran à chaque lettre et un réflecteur. À chaque étape, le câblage transforme la lettre en une autre selon un alphabet de substitution. La lettre chiffrée correspondante s’allume alors sur le tableau lumineux en haut de la machine.

Le déchiffrement se fait de manière identique, en tapant le message chiffré. Pour un réglage donné de la machine, la frappe d’une lettre allume toujours la même lettre chiffrée, et inversement.

2.3.8 Par Transposition

Principe de transposition

C’est une technique de cryptographie classique qui ne modifie pas les lettres du message initial, peut être rendue très sécurisée en utilisant des méthodes de permutation sophistiquées. Il s’agit généralement de réarranger les données pour les rendre visuellement inexploitable. Avec le principe de la transposition, toutes les lettres du message sont présentes, mais dans un ordre différent.[35]

2.3.8.1 Transposition simple par colonne

On écrit le message horizontalement dans une matrice prédéfinie, et pour retrouver le texte chiffré, on lit la grille verticalement. Le procédé inverse représente le procédé de déchiffrement.[29]

Chiffrement

T	R	A	N	S
P	O	S	I	T
I	O	N	S	I
M	P	L	E	

TABLE 2.3 – Déchiffrement simple par colonne

- disposer les lettres du message horizontalement sur la matrice de longueur n ;
- collecter les lettres verticalement ;

Pour un message de longueur m , on aura deux cas de figures :

- $m \bmod n = 0$: dans ce cas, toutes les colonnes ont la même hauteur $m \div n$.
- $m \bmod n = i$ tel que $i > 0$: dans ce cas,
 - La hauteur des i premières colonnes est $(m \div n) + 1$.
 - La hauteur des $n-i$ colonnes restantes est $m \div n$.

Exemple : chiffrer le message $M = \text{“CHIFFREMENTDECESAR”}$, sur une matrice de longueur $n=5$;

- le nombre de lettres du message est 18.
- la hauteur des colonnes : $18 \bmod 5 = 3$ donc :
- les 3 premières colonnes auront la hauteur de $(18 \div 5) + 3 = 4$;
- la hauteur des deux dernières colonnes est $18 \div 5 = 3$;

Pour chiffrer le message, on dispose les lettres horizontalement, puis on récupère les lettres verticalement comme suit :

C	H	I	F	F
R	E	M	E	N
T	D	E	C	E
S	A	R		

TABLE 2.2 – chiffrement simple par colonne

Donc le message chiffré est le suivant : CRTSHEDAIMERFECFNE.

Déchiffrement

- Disposer les lettres du message verticalement sur la matrice de longueur n ;
- Collecter les lettres horizontalement ;
- La construction de la matrice se fait de la même manière.

Exemple : Déchiffrer le message $M = \text{“TPIMROOPASNLNISESTI”}$, sur une matrice de longueur $n=5$;

- Le nombre de lettres du message est =19.
- La hauteur des colonnes : $19 \bmod 5 = 4$:
- Les quatre premières colonnes auront la hauteur de $(19 \div 5) + 1 = 4$;
- La hauteur de la dernière colonne est $19 \div 5 = 3$;

Pour déchiffrer le message, on récupère les lettres horizontalement comme suit : D’où le message est : Transposition simple.

2.3.8.2 Transposition complexe par colonne

Un mot clé secret est utilisé pour dériver une séquence de chiffres commençant par 1 et finissant par le nombre de lettres composant le mot clé. Cette séquence est obtenue en numérotant les lettres du mot clé en partant de la gauche vers la droite et en donnant l'ordre d'apparition dans l'alphabet.

Une fois la séquence de transposition obtenue, on chiffre en écrivant d'abord le message par lignes dans un rectangle, puis on lit le texte par colonnes en suivant l'ordre déterminé par la séquence.[41].

Chiffrement

- la clé du message donne le nombre de colonnes, et l'ordre de la récolte des lettres, de la même manière que la transposition simple, on dispose horizontalement les lettres du message.
- On collecte verticalement les lettres suivant l'ordre croissant des lettres de la clé par rapport a l'alphabet.

Exemple :

Chiffrer le message M="CHIFFREMENTDAFFINE", avec la clé ="AVRIL".

- Le nombre de lettres du message est=18, la taille de la clé est 5.
- la hauteur des colonnes : $18 \bmod 5 = 3$:
- les trois premières colonnes auront une hauteur de $(18 \div 5) + 1 = 4$.
- la hauteur des deux dernières colonnes est $18 \div 5 = 3$;
- Pour chiffrer le message, on dispose les lettres du message horizontalement, puis on récupère les lettres verticalement selon le rend des lettres de la clé.ce tableau suivant montre comment on récupère les lettres :

A	V	R	I	L
1	5	4	2	3
C	H	I	F	F
R	E	M	E	N
T	D	A	F	F
I	N	E		

TABLE 2.4 – Chiffrement complexe par colonne

Le résultat est le suivant : CRTIFEFFNFIMAEHEDN.

Déchiffrement

- On effectue l'opération d'inverse pour déchiffrer le message.
- On dispose verticalement les lettres du message chiffré suivant l'ordre croissant des lettres du mot clé.
- Collecter horizontalement les lettres.

Exemple : déchiffrer le message M=" CRTSFECFNEIMERHEDA", avec la clé ="AVRIL".

Et voici le message déchiffrée : CHIFFREMENT DE CÉSAR.

A	V	R	I	L
1	5	4	2	3
C	H	I	F	F
R	E	M	E	N
T	D	A	F	F
I	N	E		

FIGURE 2.5 – déchiffrement complexe par colonne

2.3.8.3 Transposition par carre-polybique

Un mot clé secret est utilisé pour construire un alphabet dans un tableau, permettant d'extraire les coordonnées des lignes et des colonnes correspondant aux lettres du texte à chiffrer. Ainsi, chaque lettre du texte en clair est représentée par deux chiffres écrits verticalement sur deux lignes. L'étape qui suit, consiste à concaténer les deux lignes obtenues précédemment pour obtenir une seule ligne de chiffres, puis a recombinaison ces chiffres deux par deux. Ces nouvelles combinaisons de chiffres représentant les coordonnées de lignes et de colonnes de texte chiffré.[41]

2.4 Limitations et Vulnérabilités

La cryptographie classique présente plusieurs limitations et vulnérabilités, qui sont principalement dues à sa simplicité et à son manque de robustesse par rapport aux méthodes de cryptographie modernes. Voici quelques-unes des principales limitations et vulnérabilités de la cryptographie classique :

- **Faible résistance aux attaques par force brute :** Comme le chiffrement de César ou le chiffre de Vigenère, ils sont souvent vulnérables aux attaques par force brute. Étant donné le petit espace de clé, il est relativement facile pour un attaquant de tester toutes les clés possibles jusqu'à ce qu'il trouve la bonne.
- **Dépendance de la clé :** Dans de nombreux systèmes de cryptographie classiques, la sécurité repose entièrement sur la clé utilisée pour chiffrer et déchiffrer les messages. Si la clé est compromise, tout le système de cryptographie devient vulnérable.
- **Manque de diffusion :** Les méthodes de cryptographie classique ont souvent un manque de diffusion, ce qui signifie que de petits changements dans le texte en clair peuvent entraîner des changements mineurs dans le texte chiffré. Cela rend ces systèmes plus vulnérables aux attaques telles que l'analyse des coïncidences.
- **Manque de sécurité contre les attaques modernes :** Les techniques de cryptographie classiques ne sont généralement pas conçues pour résister aux attaques modernes, telles que les attaques par ordinateur quantique ou les attaques par canal auxiliaire, ce qui les rend inadaptées à une utilisation dans des environnements où la sécurité est critique.

2.5 Conclusion

La cryptographie classique permet de chiffrer des textes seulement. Dans ce chapitre, nous avons présenté certaines méthodes les plus connues. La cryptographie classique a assuré la confidentialité des messages pendant des siècles. Néanmoins, avec l'avènement de l'informatique, ces méthodes ne sont pas efficaces face à cette technologie. Dans le chapitre suivant, nous allons présenter la cryptographie moderne.

CHAPITRE 3

LA CRYPTOGRAPHIE MODERNE

3.1 Introduction

La cryptographie classique cherche à concevoir des techniques pour transmettre des données de façon confidentielle, tandis que la cryptographie moderne se concentre plus largement sur les aspects de sécurité des communications. Cela implique l'utilisation de divers mécanismes basés sur des algorithmes cryptographiques. La sécurité des données chiffrées repose sur plusieurs éléments clés tels que :

- La robustesse de l'algorithme cryptographie (difficile à casser).
- La confidentialité de la clé.

La figure suivante décrit la taxonomie des techniques de cryptographie symétriques et asymétriques :

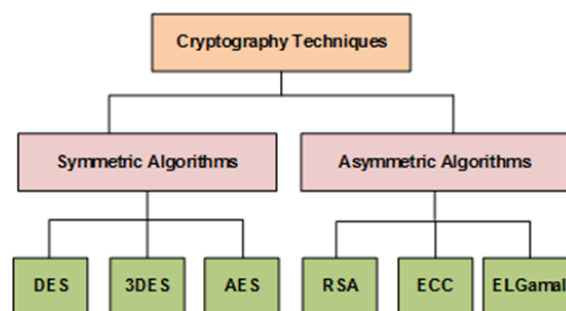


FIGURE 3.1 – taxonomie des techniques de cryptographie [9]

Elle présente une classification des algorithmes en fonction de leur type, qui peut être symétrique ou asymétrique, que nous détaillerons dans les prochaines sections.

3.2 La cryptographie à clé symétrique (dits à clé secrète)

Les algorithmes symétriques actuels utilisent une succession de transpositions et de substitutions complexes des valeurs du message, basées sur des opérations mathématiques et réalisées en plusieurs passes. La clé faisant partie intégrante de la fonction, il est impossible d'inverser l'algorithme sans elle, et les seules attaques envisageables consistent souvent à essayer toutes les valeurs de clés possibles. C'est la raison pour laquelle une clé symétrique suffisamment importante (128 bits) et bien choisie est considérée comme sûre.[48]. La figure suivante illustre l'utilisation de la même clé pour le chiffrement et le déchiffrement.

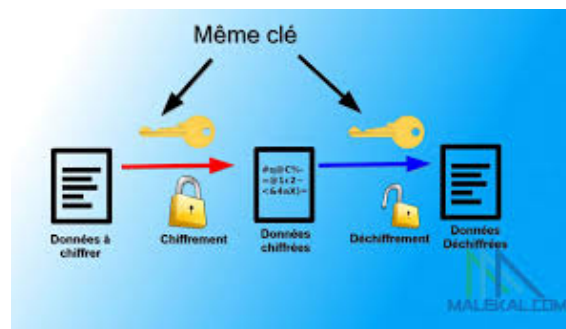


FIGURE 3.2 – la cryptographie symétrique [17]

L'algorithme symétrique le plus célèbre est le DES (Data Encryptions Standard, qui fonctionnait avec des clés de 64 bits) remplacé depuis par l'AES (Advanced Encryptions System, qui fonctionne avec des clés allant jusqu'à 256 bits). Les chiffrements symétriques exigent toutefois que les deux correspondants échangent au préalable la clé secrète par un canal sûr.[48]

Le partage de clés

Il arrive parfois que la clé soit connue par plusieurs personnes ou soit présente sur plusieurs serveurs du propriétaire. Le transfert de la clé doit absolument être sécurisé. Elle doit être échangée par un canal confidentiel pour que le chiffement ne soit pas compromis. Plusieurs stratégies existent et une des plus efficaces est d'utiliser un autre type de chiffement pour chiffrer la clé secrète et d'envoyer ce message au destinataire qui pourra récupérer la clé secrète en toute sécurité. De plus, Un grand nombre de clés lors de partages deux à deux entre de nombreuses personnes est généré par la formule suivante :

$$\text{clés} = \frac{N \times (N - 1)}{2} \quad (3.1)$$

ou N représente le nombre de personnes.

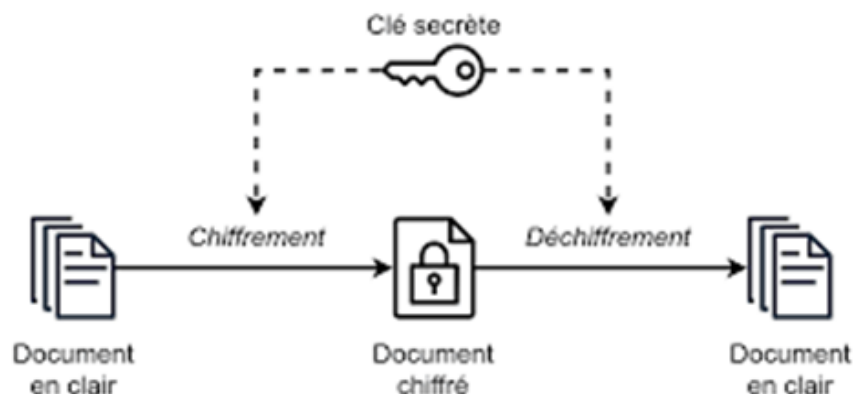


FIGURE 3.3 – la cryptographie symétrique
[8]

Caractéristiques :[45]

- Les clés sont identiques : $KE = KD = K$.
- La clé doit rester secrète.
- Les algorithmes les plus répandus sont le DES, AES, 3DES.
- Au niveau de la génération des clés, elle est choisie aléatoirement dans l'espace des clés.
- Ces algorithmes sont basés sur des opérations de transposition et de substitution des bits du texte clair en fonction de la clé.
- La taille des clés est souvent de l'ordre de 128 bits. Le DES en utilise 56, mais l'AES peut aller jusque 256,
- L'avantage principal de ce type est sa rapidité.

3.2.1 Chiffrement par Blocs (Block Cipher)

C'est une des deux grandes catégories de chiffrements modernes en cryptographie symétrique. Il consiste à un découpage des données en blocs de taille généralement fixe (souvent une puissance de deux comprise entre 32 et 512 bits). Les blocs sont ensuite chiffrés les uns après les autres. Le chiffrement par bloc utilise quatre modes opératoires : Electronic Code Book (ECB), Cipher Block Chaining (CBC), Output Feedback (OFB) et Cipher Feedback (CFB).[23]

Le mode ECB (Electronic Codebook)

Ce mode est le plus simple : un même bloc est toujours codé de la même manière. Il n'y a pas de rétroaction de l'entrée ou de la sortie sur la fonction de chiffrement.

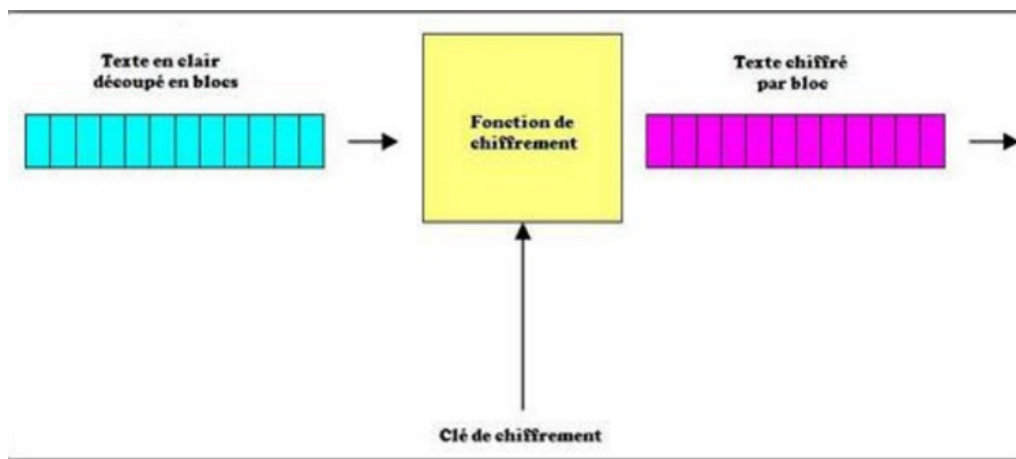


FIGURE 3.4 – mode ECB
[8]

Le mode CBC (Cipher Bloc Chaining)

Dans ce mode de chiffrement, chaque bloc de texte en clair est d'abord combiné par un ou exclusif avec le dernier bloc du texte chiffré. La sortie de ce ou exclusif est ensuite appliquée à la fonction de chiffrement. Ce mode de chiffrement dispose en plus d'un vecteur d'initialisation appelé IV (pour Initialisation Vector) qui permet d'initialiser le processus quand aucun bloc n'a encore été chiffré.[23]

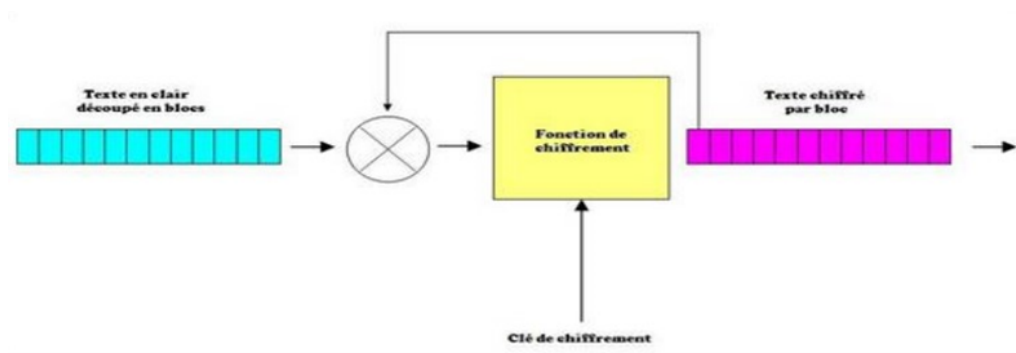


FIGURE 3.5 – mode CBC
[8]

Le mode CFB (Cipher FeedBack)

Les modes ECB et CBC travaillent avec sur des blocs de texte en clair (64 bits par exemple). Ces modes ne sont pas utilisables lorsque le chiffrement ne peut débuter que lorsqu'un bloc est complet. Sur des applications réseau, cela peut poser des problèmes, car les valeurs à chiffrer arrivent de manière asynchrone sous forme d'octets et doivent être transmises immédiatement (cas du protocole Telnet par exemple). Le registre à décalage est initialisé avec un vecteur d'initialisation. Le bloc complet est alors chiffré. L'octet de poids faible du texte chiffré est combiné par un ou exclusif avec l'octet de texte en clair. Le résultat de cette opération est alors transmis en même temps qu'il est injecté dans le registre à décalage.[23]

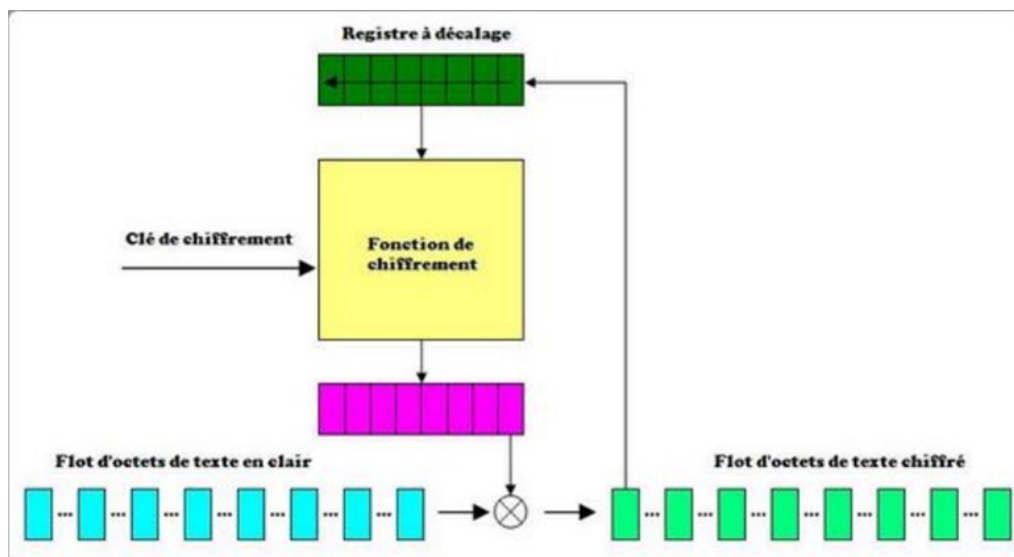


FIGURE 3.6 – mode CFB
[8]

Le Mode OFB (Output Feedback)

Dans ce mode, un vecteur initial est initialement chiffré pour démarrer le processus. Le flux de clé en sortie de ce bloc sera réinjecté en entrée pour calculer le prochain flux de clé.[19].

Nous allons maintenant présenter les algorithmes symétriques les plus connus et les plus utilisés :

3.2.2 DES (Data Encryption Standard)

Le Data Encryption Standard (standard de chiffrement de données) a été publié en 1977, et fut ainsi le premier algorithme de cryptographie à petite clé secrète (56 bits) à avoir été rendu public. Le DES consiste en un réseau de Feistel de 16 tours : le message à chiffrer est découpé en blocs de 64 bits, chacun d'eux étant séparé en deux sous-blocs de 32 bits.[48].

Historique de DES

Le Data Encryption Standard (DES) est un algorithme de chiffrement symétrique qui a été développé dans les années 1970 par IBM et adopté comme standard fédéral de traitement de l'information aux États-Unis en 1977 par le National Institute of Standards and Technology (NIST). L'algorithme DES utilise une clé de 56 bits pour chiffrer des blocs de données de 64 bits. Sa conception a été influencée par les travaux de Horst Feistel, qui a conçu un réseau de substitution-permutation complexe pour améliorer la sécurité du chiffrement[18]

Principe du DES

Le DES est un code à blocs de 64 bits. Le fichier clair est donc découpé en plusieurs blocs de 64 bits. La transformation d'un bloc comporte 16 itérations d'un processus de codage, qui effectue respectivement une étape de confusion, puis une étape de

diffusion. En effet, la sécurité des données cryptées repose sur une clé secrète de 64 bits (succession de 0 et de 1), mais en fait, seuls 56 bits servent réellement à définir la clé. Les bits 8, 16, 24, 32, 40, 48, 56, 64 sont des bits de parité (bits de détection d'erreur). Le 8e bit est fait en sorte que sur les 8 premiers bits, il y ait un nombre impair de 1. Par exemple, si les 7 premiers bits sont 1010001, le 8e bit est 0. Ceci permet d'éviter les erreurs de transmission.[37]

Génération des clés

La clé initiale est de 64 bits, à partir de laquelle on 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 :

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

4. Les blocs subissent un décalage à gauche, puis regroupés 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 K_i .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
```

Les grandes lignes de l'algorithme sont :

phase 1(préparation- diversification de la clé) :

Le texte est découpé en blocs de 64 bits. On diversifie aussi la clé K, c'est-à-dire qu'on fabrique à partir de K 16 sous-clés K_1, \dots, K_{16} à 48 bits. Les K_i sont composés de 48 bits de K, pris dans un certain ordre.

Phase 2 (Permutation initial) :

Pour chaque bloc de 64 bits x du texte, on calcule une permutation finie $y=P(x)$. y est Représenté sous la forme $y=G0D0$, G0 étant les 32 bits à gauche de y, D0 les 32 bits à droite.

phase 3 (Itération) :

On applique 16 rondes d'une même fonction. A partir de $G_{i-1}D_{i-1}$ (pour i de 1 à 16),

on calcule GiDi en posant :

$$G_i = D_{i-1} \tag{3.2}$$

$$D_i = G_{i-1} \oplus f(D_{i-1}, K_i) \tag{3.3}$$

XOR est le ou exclusif bit à bit, et f est une fonction de confusion, suite de substitution et de permutations.

phase 4(permutation finale) :

l'application de l'inverse de la permutation initiale sur le bloc $G_{16}D_{16}$ pour obtenir le bloc chiffré Z de 64 bits à partir de x :

$$Z = P^{-1}(G - 16D - 16) \tag{3.4}$$

Voilà en résumant Les grandes lignes de l'algorithme comme suit :

- Fractionnement du texte en blocs de 64 bits (8 octets).
- Permutation initiale des blocs.
- Découpage des blocs en deux parties : gauche et droite, nommées G et D.
- Étapes de permutation et de substitution répétées 16 fois (appelées rondes).
- Recollement des parties gauche et droite puis permutation initiale inverse.

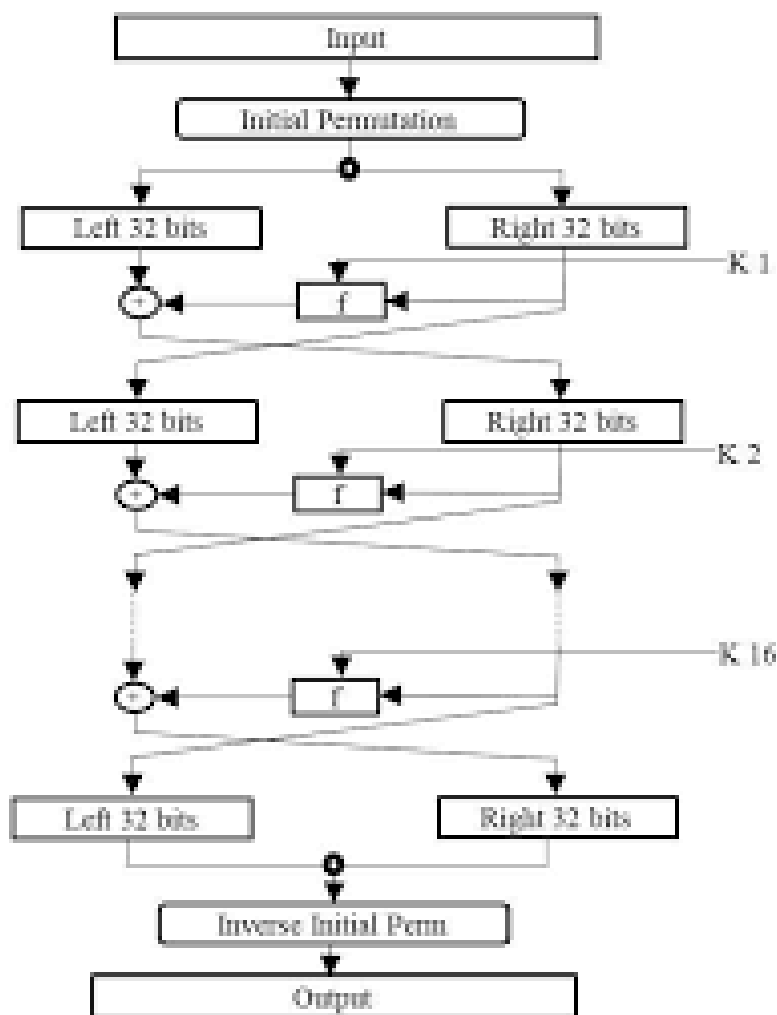


FIGURE 3.7 – Algorithme de DES
[12]

Déchiffrement de DES

Le déchiffrement du DES s'effectue en appliquant les 16 clés inversement.

La faiblesse majeure de DES est la taille relativement courte de sa clé de 56 bits. Avec l'augmentation de la puissance de calcul, il est devenu possible de casser DES par force brute en testant toutes les clés possibles. Cela conduit au développement de l'AES.[30]

3.2.3 AES(Advanced Encryption Standard)

Le NIST a lancé le nouveau standard de chiffrement, le Concours AES, en 1997. Est un algorithme de chiffrement symétrique utilisé pour sécuriser les données. Il utilise des blocs de données de 128 bits et des clés de chiffrement de longueurs variables (128, 192 ou 256 bits).[15]

Bref historique de L'AES

L'algorithme AES est l'un des algorithmes de chiffrement par bloc qui a été publié par l'Institut national des normes et de la technologie (NIST) en 2000. Les principaux objectifs de cet algorithme étaient de remplacer l'algorithme DES après avoir mis en évidence certains aspects vulnérables de cet algorithme. Le NIST a invité des experts qui travaillent sur le cryptage et la sécurité des données pour présenter un algorithme innovant de chiffrement par blocs pour décrypter les données avec une structure puissante et complexe. Le NIST a accepté cinq algorithmes pour l'évaluation. Après avoir évalué divers critères et paramètres de sécurité, ils ont sélectionné l'un d'entre eux. Il a sélectionné l'un des cinq algorithmes de cryptage proposés par deux cryptographes belges, Joan Daeman et Vincent Rijmen. Le nom original de l'algorithme AES est l'algorithme de Rijndel. Cependant, ce nom n'est pas devenu un nom populaire pour cet algorithme, mais il est reconnu comme l'algorithme AES (Advanced Encryption Standard (AES) dans le monde entier.[6]

Les étapes de cet algorithme sont résumées comme suit :

- **Permutation** : Un bloc de données de 16, 24 ou 32 octets sont permutés ensuite placés dans une matrice.
- **L'opération SubBytes** : consiste à substituer chaque élément de la matrice via une SBox.
- **L'opération Shiftrows** : Cette étape implique un décalage à gauche sur les éléments de la matrice.
- **L'opération MixColumns** : en effectuant une opération mathématique sur chaque colonne de la matrice de données et mettant le résultat dans une nouvelle matrice.
- **L'opération Addroundkey** : Cette étape consiste à faire un XOR entre la matrice qui contient la clé et le bloc de données.

On va détailler cet algorithme dans le chapitre 4 suivant.

3.2.4 Chiffrement par flux (Stream Cipher)

Le chiffrement par flot (Stream Cipher), également appelé chiffrement par flux, est une des deux grandes catégories de chiffrements modernes. Un chiffrement par

flot permet de traiter des données de longueur arbitraire sans avoir à les découper (il s'agit d'un chiffrement d'une suite de caractères un à la fois, à l'aide d'une transformation qui varie au fur et à mesure du texte).[19].

Exemple :

Si on souhaite chiffrer le message « Bonjour » avec la clé K, un algorithme de chiffrement effectue les opérations suivantes :

- La génération du flot de clés : à partir de K, on construit une clé $k_1 \dots k_n$ de même longueur que le message.
- pour chaque caractère m_i du message, on calcule le caractère chiffré correspondant $c_i = E(m_i, k_i)$ où E est une fonction qui prend en entrée un caractère m et une clé k et retourne un caractère chiffré $E(m, k)$.
- Le message chiffré est $c_1 \dots c_n$.

Exemple :

Message en clair "Bonjour" : 01010011 01000001 01001100 01010101 01010100

Clé (générée aléatoirement) : 01110111 01110111 00100100 00011111 00011010

Message Chiffré "\$6jJM" : 00100100 00110110 01101000 01001010 01001110

L'avantage du chiffrement par flot est qu'il :

- très rapide (en matériel et en logiciel)
- implémentation matérielle avec peu de portes

L'inconvénient du chiffrement par flot :

- propagation d'erreurs (prob. de synchronisation)
- la sécurité difficile à atteindre (pas de preuve).

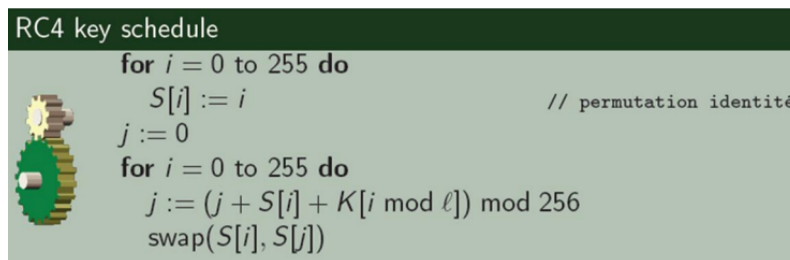
Parmi les algorithmes les plus utilisés dans la cryptographie symétrique par flot, on trouve :

3.2.5 RC4 (Rivest Cipher 4)

RC4 a été conçu par Ronald Rivest (le R de RSA) en 1987. Officiellement nommé Rivest Cipher 4, l'acronyme RC est aussi surnommé Ron's Code. Il est utilisé dans des protocoles comme WEP, WPA ainsi que TLS. Les raisons de son succès sont liées à sa grande simplicité et à sa vitesse de chiffrement. Les implémentations matérielles ou logicielles étant faciles à mettre en œuvre.[31]

Principe de La clé RC4

Permet d'initialiser un tableau de 256 octets en répétant la clef autant de fois que nécessaire pour remplir le tableau. Par la suite, des opérations très simples sont effectuées : les octets sont déplacés dans le tableau, des additions sont effectuées, etc. Le but est de mélanger autant que possible le tableau. Au final, on obtient une suite de bits pseudo-aléatoires qui peuvent être utilisés pour chiffrer le message via un XOR (comme dans le cas d'un masque jetable).[31]



```

RC4 key schedule
for i = 0 to 255 do
  S[i] := i // permutation identité
j := 0
for i = 0 to 255 do
  j := (j + S[i] + K[i mod ℓ]) mod 256
  swap(S[i], S[j])

```

FIGURE 3.8 – programme de clé
[?]

Chiffrement

Pour le chiffrement deux étapes sont nécessaires : l'initialisation à l'aide de la clé et le chiffrement du texte clair.

La première étape génère deux tableaux de 256 octets en fonction de la clé : un tableau K initialisé avec les octets de la clé et un tableau P (appelé table d'états, laquelle sera le flux appliqué sur le texte clair) initialisé avec les nombres de 0 à 255 permutés. pseudo-aléatoirement selon le tableau K.

La deuxième étape consiste en des permutations pour effectuer le chiffrement. À noter que les additions sont toutes exécutées modulo 256. Le chiffrement est relativement simple.

Le RC4 est populaire en étant extrêmement rapide (environ dix fois plus rapide que le DES).

Maintenant, nous allons aborder l'explication du deuxième type, qui est la cryptographie asymétrique, ainsi que ses différents algorithmes.

3.3 La Cryptographie à clé asymétrique (dite publique)

3.3.1 Définition

La cryptographie à clé publique (asymétrique) consiste en l'existence d'une paire de clés de chaque côté (émetteur et récepteur) liées mathématiquement. Chaque paire est composée d'une clé privée (et différente pour chaque utilisateur, qui doit être gardée secrète) et d'une clé publique connue par tous les utilisateurs.[14]

La cryptographie asymétrique se base sur des fonctions à sens unique. Cela veut dire que les données cryptées avec la clé publique ne peuvent être décryptées que s'il on possède la clé secrète. Ça signifie que même si l'on a obtenu la clé publique, on ne pourra pas déchiffrer les informations.[14]

Le principe de ce cryptosystème est basé sur l'usage d'un couple de clés, l'une publique qui est connue par tout le monde et l'autre privée qui doit être confidentielle.

Le fonctionnement de la cryptographie asymétrique est illustré sur la figure suivante :

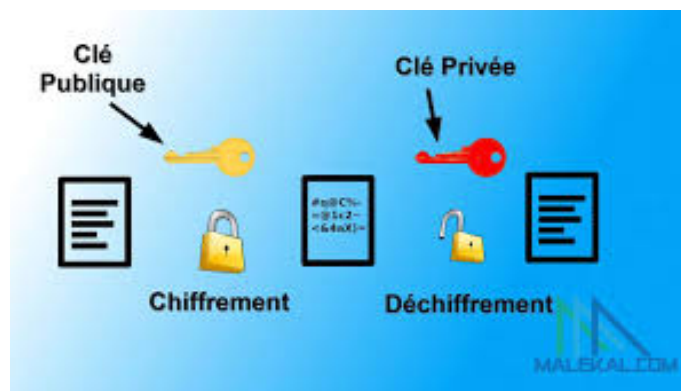


FIGURE 3.9 – Chiffrement Asymétrique [8]

Ses Avantages

- Le problème n'existe plus pour communiquer la clé de déchiffrement, dans la mesure où les clés publiques peuvent être envoyées librement.
- Le chiffrement par clés publiques permet donc à des personnes d'échanger des messages chiffrés sans pour autant posséder de secret en commun, seule la clé secrète à besoin d'être conservée de manière secrète.
- Selon l'usage, une paire de clé (publique/secrète) peut être utilisée plus longtemps qu'une clé symétrique.
- La cryptographie à clé publique permet de réaliser des schémas de signature électronique assurant un service de non répudiation.

Ses Inconvénients

- Tout le challenge consiste à s'assurer que la clé publique que l'on récupère est bien celle de la personne à qui l'on souhaite faire parvenir l'information chiffrée.
- Les performances des systèmes asymétriques sont beaucoup moins bonnes que celles des systèmes symétriques, car ces systèmes nécessitent de pouvoir calculer sur des grands nombres.
- La taille des clés est généralement plus grande pour ces systèmes que pour les systèmes à clé secrète.

Commençons par le Système RSA :

3.3.2 Le système RSA

L'algorithme RSA est utilisé pour la cryptographie à clé publique et est basé sur le fait qu'il est facile de multiplier deux grands nombres premiers, mais difficile de factoriser le produit. C'est l'exemple le plus courant de cryptographie asymétrique, toujours considéré comme sûr, avec la technologie actuelle, pour des clés suffisamment grosses (1024, 2048 voire 4096 bits).[19]

Historique

L'algorithme RSA a été inventé en 1977 par Ronald Rivest, Adi Shamir et Leonard Adleman. C'est la première instance de méthodes de cryptographie à clé publique, dont le principe a été introduit à l'état de concept par Whitfield Diffie et Martin

Hellman. À l'heure actuelle, il s'agit sans doute du cryptosystème asymétrique le plus connu et le plus utilisé à travers le monde. Sa popularité en fait aussi un des algorithmes les plus étudiés, que ce soit d'un point de vue théorique ou pratique.[36]

Voici le principe du fonctionnement de l'algorithme RSA : [36].

Création des clés

- On choisit deux nombres premiers très grands p et q qui serviront à former les clés publiques et privées.
- On calcule N , qui est un constituant de la clé publique et de la clé privée, en faisant : $N = p \times q$.
- Ensuite, on calcule e , qui fait partie de la clé publique avec :

$$\phi(N) = (p - 1)(q - 1) \tag{3.5}$$

et en choisissant e tel que $\text{gcd}(e, \phi(N)) = 1$

- Le couple (N, e) forme ainsi la clé publique de chiffrement.

Chiffrement

Pour le chiffrement, on a un message codé M . Il faut ensuite découper le message en blocs strictement inférieurs à N . On nomme alors chacun des blocs codés le texte crypté C_i .

Afin de crypter, on fait :

$$C_i = M_i^e \pmod{N}. \tag{3.6}$$

Autrement dit, C_i (étant le bloc de texte crypté) équivaut au reste de la division de M_i^e par N .

Déchiffrement

Pour calculer la clé privée d à partir de p et q , il faut satisfaire l'équation $d \cdot e \equiv 1 \pmod{\phi(N)}$. Avec la clé générée d , on peut décrypter bloc par bloc le texte crypté pour retrouver le message original M .

Ainsi, pour décrypter un bloc de texte crypté C_i et retrouver le bloc de texte clair M_i , on utilise la formule :

$$M_i \equiv C_i^d \pmod{N}. \tag{3.7}$$

Cela signifie que M_i (étant le bloc de texte décrypté) est égal au reste de la division de C_i^d par N .

En appliquant cette méthode de déchiffrement, on retrouve les blocs codés du message et il ne reste plus qu'à les convertir en leur équivalent dans l'alphabet défini pour obtenir le message déchiffré.

L'un des principaux avantages de l'algorithme RSA est sa sécurité basée sur la difficulté de factoriser de grands nombres entiers.

3.3.3 Le cryptosystème ELGamal

Le chiffrement ElGamal est une méthode de cryptographie à clé publique inventée par Taher ElGamal en 1985. Sa sécurité repose, comme le protocole de Diffie et

Hellman, sur la difficulté de calculer le logarithme discret.[46]

Ce cryptosystème est composé de trois étapes :

Génération de clés

- Choisir un grand nombre premier p et deux nombres a et g tel que : $a, g < p$;
- Puis on calcule :

$$A = g^a \pmod{p} \quad (3.8)$$

- La clé publique est (A, g, p) et la clé privée est a

Chiffrement

Pour chiffrer un message M , il faut procéder comme suit :

- Choisir un nombre aléatoire b , tel que $b < a$ et le PGCD $(b, p-1) = 1$.
- Puis on calcule $B = g^b \pmod{p}$
- Ensuite, on calcule $C = M \times A^b \pmod{p}$

Le message chiffré est alors (B, C) .

Déchiffrement

Pour décrypter le message, on calcule :

$$M = C \times B^{(p-a-1)} \pmod{p} \quad (3.9)$$

la difficulté de résoudre le problème du logarithme discret est l'un des principaux avantages du cryptosystème ElGamal.

3.3.4 Protocole Diffie-Hellman

Parallèlement à leur découverte de l'aspect asymétrique de la cryptographie, Diffie et Hellman ont proposé en 1976 un protocole d'échange de clés totalement sécurisé.[45] Le problème est le suivant :

Deux entités A et B veulent s'échanger un message crypté en utilisant un algorithme nécessitant une clé K . Ils veulent s'échanger cette clé K , mais ils ne disposent pas de canal sécurisé. Pour cela, le protocole d'échange de clés de Diffie et Hellman répond à ce problème lorsque K est un nombre entier. Il repose sur l'arithmétique modulaire.

Principe du protocole

Étant donné des entiers g, n, x , avec n premier et $1 \leq x \leq n - 1$.

Il est facile de calculer l'entier

$$Y = g^x \pmod{n} \quad (3.10)$$

Si on connaît $Y = g^x \pmod{n}$, g et n , il est très difficile de retrouver x , pourvu que n ne soit pas grand.

Retrouver x connaissant $g^x \pmod{n}$, g et n , s'appelle résoudre le problème du logarithme discret. Comme pour la factorisation d'entiers, c'est un problème dans lequel on ne dispose pas d'algorithme efficace.

Fonctionnement du protocole

Supposant que les entités A et B veulent s'échanger une clé avec le protocole Diffie Hellman, les étapes du protocole sont comme suit :

- **Étape 1** : A et B choisissent ensemble :
 - Un grand nombre premier n
 - Un entier g tel que $1 \leq g \leq n - 1$
 Cet échange n'a pas besoin d'être sécurisé.
- **Étape 2** : A et B choisissent secrètement un entier chacun, respectivement x et y .
- **Étape 3** : A calcule $X = g^x \pmod n$ et B calcule $Y = g^y \pmod n$.
- **Étape 4** : A et B s'échangent les valeurs de X et Y . Cet échange n'a pas besoin d'être sécurisé.
- **Étape 5** : A calcule $Y^x = (g^y)^x \pmod n = g^{yx} \pmod n$ et appelle ce nombre K , qui est la clé partagée avec B.
 De même, B calcule $X^y = (g^x)^y \pmod n = g^{xy} \pmod n$ et appelle ce nombre K , qui représente la clé partagée avec A.

Le protocole Diffie-Hellman permet l'échange sécurisé de clés cryptographiques sur un canal de communication non sécurisé. Le protocole Diffie-Hellman est vulnérable aux attaques de l'homme du milieu. Si un attaquant parvient à s'interposer entre les deux parties qui échangent les clés, il peut alors établir deux connexions sécurisées distinctes, une avec chaque partie, et ainsi déchiffrer tout le trafic.

3.3.5 ECC (pour Elliptic Curve Cryptographie)

Pour utiliser les courbes elliptiques en cryptographie, il faut trouver un problème difficile (tel que la factorisation d'un produit en ses facteurs premiers dans le cas du RSA). Considérons l'équation :

$$Q = kP \tag{3.11}$$

Où :

- Q et P sont des points appartenant à la courbe elliptique $E_p(a, b)$
- k est un entier inférieur à p , le nombre premier définissant le corps fini \mathbb{F}_p sur lequel la courbe elliptique est définie.

Il est facile de calculer Q connaissant k et P , mais il est difficile de déterminer k si on connaît Q et P . Il s'agit du problème du logarithme discret pour les courbes elliptiques : $\log_P(Q)$. Dans une utilisation réelle, le k est très grand, rendant l'attaque par force brute inutilisable (rappelons qu'a priori, l'attaque par force brute est toujours possible.[22])

Principe de chiffrement

Même si la cryptographie par courbes elliptiques est souvent employée pour l'échange d'une clé symétrique, elle est aussi utilisée pour chiffrer directement les données. Voici un exemple de Cryptosystème les utilisant.

Pour chiffrer le message, A détermine aléatoirement un nombre entier positif k et produit C_m comme un couple de points tel que :

$$C_m = \{kG, P_m + kP_B\} \tag{3.12}$$

Où : - G est un point public générant un groupe elliptique $E_q(a, b)$ - P_B est la clé publique de B, obtenue à partir de sa clé privée - P_m est le point représentant le message clair m - k est un nombre entier positif choisi aléatoirement par A.

Le chiffrement consiste donc à multiplier le point générateur G par k pour obtenir le premier élément du couple, et à additionner kP_B au point P_m représentant le message pour obtenir le second élément[45].

Cette opération permet de chiffrer le message m en utilisant la clé publique de B, sans avoir besoin de connaître sa clé privée. Seul B, détenteur de sa clé privée, pourra déchiffrer le message en utilisant des opérations sur les points de la courbe elliptique.

On remarquera l'utilisation de la clé publique de B.

Principe de déchiffrement

Pour déchiffrer, B devra multiplier le premier point par sa clé privée, et soustraire le résultat au second point reçu :

$$P_m + kP_B - nB(kG) = P_m + k(nB)G - nB(kG) = P_m \quad (3.13)$$

Après avoir étudié les systèmes de chiffrement symétrique et asymétrique, nous présentons ci-dessous un tableau comparatif mettant en évidence leurs principales caractéristiques.

Cryptographies	Avantages	Inconvénients
Symétrique	<ul style="list-style-type: none"> -Système rapide de chiffrement et de déchiffrement. -Cles relativement courtes (128 ou 256). -Bonnes performances et sécurité bien étudiée. 	<ul style="list-style-type: none"> -Gestion de clés difficiles (nombreuses clés). -l'échange de la clé secrète. -Dans un réseau de N entités susceptibles de communiquer il faut distribuer $N*(N-1)/2$ clés.
Asymétrique	<ul style="list-style-type: none"> -Pas de secret à transmettre. -Nombre de clés à distribuer est réduit par rapport aux clés symétriques. Tres utile pour échanger les clés. Permet de signer des messages facilement. 	<ul style="list-style-type: none"> -La relation clé publique/clé privée impose : des clés plus longues(1024 à 4096bits). -Gestion de certificats publics. -Lenteur de calcul. -Pas d'authentification de la source.

TABLE 3.1 – les avantages et les Inconvénients symétrique et asymétriques

3.4 la cryptographie Hybride

3.4.1 Définition

La cryptographie hybride est un système de cryptographie faisant appel aux deux grandes familles des systèmes cryptographiques : la cryptographie symétrique et la cryptographie asymétrique. Un cryptosystème hybride consiste à utiliser les avantages des chiffrements symétrique et asymétrique tels que la rapidité d'un cryptosys-

tème symétrique et la possibilité de transmettre la clé secrète par un cryptosystème asymétrique.[?]

3.4.2 PGP : description et fonctionnement

PGP, ou Pretty Good Privacy, est un système de cryptographie hybride qui combine à la fois des techniques de chiffrement symétrique et asymétrique pour assurer la confidentialité et l'authenticité des données. Développé et diffusé aux États-Unis par Philip Zimmerman en 1991.[1]

Fonctionnement

Pour le fonctionnement de PGP, une clé secrète est générée (nommée clé de session, valable pour un seul fichier ou un seul message). Le message ou le fichier est chiffré au moyen de cette clé de session avec un algorithme de cryptographie symétrique. Puis cette clé secrète est chiffrée au moyen de la clé publique du destinataire et ajoutée au début du message ou du fichier.[?]

Pour comprendre bien le fonctionnement de PGP, un graphique est peut-être plus parlant :

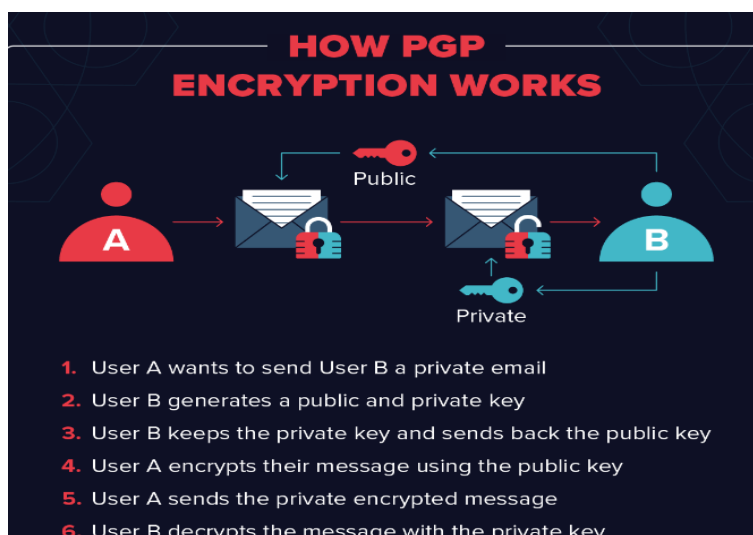


FIGURE 3.10 – fonctionnement PGP [11]

1.Création d'une clé publique et privée : L'utilisateur crée une paire de clés composée d'une clé publique et d'une clé privée.

2.Chiffrement du message : L'émetteur utilise la clé publique du destinataire pour chiffrer le message. De cette manière, seul le destinataire, qui possède la clé privée correspondante, peut déchiffrer le message. Une clé de session symétrique est générée aléatoirement et utilisée pour chiffrer le message compressé. La clé de session est ensuite chiffrée avec la clé publique du destinataire.

3.Envoi du message chiffré : Le message chiffré avec la clé de session et la clé de session chiffrée avec la clé publique sont envoyés au destinataire.

5.Déchiffrement du message : Le destinataire utilise sa clé privée pour dé-

chiffrer la clé de session.

La clé de session est ensuite utilisée pour déchiffrer le message.

Le message est enfin décompressé pour obtenir le message original.

Parmi les travaux existants sur la cryptographie hybride :

Modèle de chiffrement hybride des systèmes de cryptographie AES et ElGamal pour les systèmes de contrôle de vol : Les objectifs de cet article sont les suivants : Introduire un nouveau modèle d'algorithme cryptographique hybride en utilisant une combinaison de deux algorithmes cryptographiques, AES et ElGamal; Fournir une comparaison entre deux algorithmes symétriques, asymétriques et le nouveau modèle hybride; Montrer l'efficacité et la sécurité du nouveau modèle hybride, qui rend l'algorithme robuste contre les vulnérabilités. Actuellement, de nombreux algorithmes de chiffrement sont disponibles pour sécuriser les données, mais certains algorithmes consomment beaucoup de ressources informatiques telles que la mémoire et le temps CPU. Cet article présente une analyse comparative des résultats expérimentaux sur ces algorithmes de chiffrement. L'objectif de cette recherche est d'évaluer la performance des algorithmes de cryptographie AES, ElGamal et de l'algorithme de cryptographie hybride AES et ElGamal.[33]

Nouvel algorithme de chiffrement hybride base sur AES, RSA et TWO-FISH pour le chiffrement bluetooth : Cet article propose un nouvel algorithme basé sur AES, RSA et TWOFISH pour le chiffrement bluetooth afin d'améliorer davantage la sécurité du Bluetooth, qui utilise actuellement l'AES 128 bits pour le chiffrement dans ses dernières versions (Bluetooth 4.0 - 5.0).le principe de chiffrement de cette methode est de chiffré le message en utilisant AES avec une clé de 128 bits, puis l'avons chiffré à nouveau en utilisant TwoFish avec la même clé de 128 bits. Enfin, la clé de 128 bits générée au début sera chiffrée en utilisant RSA avec une clé de 1024 bits pour protéger son transfert par voie aérienne.Donc cet nouvel algorithme améliore la sécurité du chiffrement Bluetooth en éliminant toutes les faiblesses connues, rendant ainsi l'échange de données entre les appareils Bluetooth sécurisé.[38]

Application de l'algorithme hybride AES et RSA dans les e-mails : cet article analyse de manière exhaustive la vitesse de chiffrement, la sécurité, la gestion des clés et les directions d'application des algorithmes AES et RSA, et propose l'utilisation d'un algorithme hybride de chiffrement AES et RSA pour la communication par courrier électronique, avec une simulation réussie en langage Java.L'algorithme de chiffrement hybride combine les avantages de la vitesse rapide de chiffrement de l'algorithme AES, de la gestion facile des clés de l'algorithme RSA et de la signature numérique pour garantir la transmission sécurisée de documents confidentiels.[39] Voici les principaux avantages de la cryptographie hybride :

- Elle combine les avantages de la cryptographie symétrique (rapidité) et asymétrique (transmission sécurisée des clés) pour offrir un système de chiffrement efficace et sécurisé.
- Cela permet d'éviter les inconvénients de chaque approche utilisée seule : lenteur de la cryptographie asymétrique et nécessité de transmettre une clé secrète de manière sûre pour la cryptographie symétrique.
- C'est une approche très utilisée dans de nombreuses applications de sécurité, comme les connexions HTTPS qui utilisent un chiffrement hybride pour une navigation web sécurisée.

3.5 La Cryptographie quantique

Il existe une nouvelle technique assez récente d'échange de clés cryptographiques appelée : « la cryptographie quantique ».

L'informatique quantique, née de la rencontre de physiciens et de théoriciens de l'informatique, commence à jeter les bases d'un nouvel espace technique, malgré qu'on soit encore loin de réalisations de taille industrielle, mais ce domaine est suffisamment prometteur pour que l'on doive dès aujourd'hui s'y intéresser, car cela risque bien de bouleverser le paysage informatique d'ici 10 à 20 ans et avec lui quelques-unes de nos certitudes[34]. C'est ce que Jacqueline Dousson a écrit dans son article qui date de 1999.

3.5.1 Définition

La cryptographie quantique, plus correctement nommée distribution quantique de clés, désigne un ensemble de protocoles permettant de distribuer une clé de chiffrement secrète entre deux interlocuteurs distants, tout en assurant la sécurité de la transmission grâce aux lois et aux propriétés de la mécanique quantique.[27]

3.5.2 Historique

L'idée d'utiliser des lois de la mécanique quantique dans le domaine de la sécurité remonte au début des années 1970, lorsque Stephen Wiesner a écrit un rapport présentant des idées tout à fait nouvelles. Ainsi, il a proposé d'utiliser la mécanique quantique pour coder des billets de banque dont l'infalsifiabilité serait garantie par le principe d'incertitude d'Heisenberg. Bien que les idées de Wiesner n'aient été publiées qu'en 1983 dans la revue *Sigact News*, elles ont incité et ont été la source d'inspiration de deux autres chercheurs en essayant de les concrétiser. C'était vers le milieu des années quatre-vingts, où les deux chercheurs américains Charles et Brassard, de l'université de Montréal, qui se sont rencontrés en 1979 sur une plage de Porto Rico, ont décidé d'aborder ce domaine. Ainsi, ils ont construit le premier prototype opérationnel, à travers lequel ils ont montré qu'il était possible, grâce au canal quantique, hautement sécurisé, de transmettre des clefs secrètes de plusieurs centaines de bits à une vitesse de 10 bit/s entre deux points distants de 32 cm.[27]

3.5.3 Notion D'un Qubit

L'unité de mesure de base de l'information en informatique quantique est le qubit (qubit ou bit quantique), qui exploite les caractéristiques physiques de la matière subatomique. Son atout majeur est de pouvoir stocker plusieurs informations simultanément.[7]

Le qubit n'est pas limité à 0 et 1, car il peut prendre les deux valeurs simultanément. C'est ce que l'on appelle le principe de superposition. Cependant, deux qubits peuvent également interagir, créant une intrigue. Dans la cryptographie quantique, les directions de polarisation sont limitées à deux directions : rectiligne et diagonale, avec des photons polarisés orthogonalement.[16]

3.5.4 Nature du photon

Les photons ont été originellement appelés par Albert Einstein « quanta de lumière ». Leur existence a été prédite par ce dernier, mais c'est Arthur Compton qui fit leur découverte En 1923. En physique des particules, le photon est souvent symbolisé par la lettre (γ).

C'est une particule élémentaire de masse nulle et de spin 1. Du fait de son spin, le photon Transporte également un moment cinétique intrinsèque dont la projection sur l'axe de Propagation est $-\hbar$ ou $+\hbar$. [16]

3.5.5 Protocole BB84

BB84 est le protocole de distribution de clé le plus connu. Il utilise quatre états différents qui font une paire des états de base. BB84 est un protocole non déterministe. Cela signifie qu'il distribue une suite aléatoire des bits.

Le but est de générer une clé partagée entre Alice et Bob n'autorisant aucun tiers à acquérir une information pertinente sur cette clé. Cette clé doit pouvoir servir à un chiffre de Vernam, et conduire ainsi à une transmission d'informations inconditionnellement sûres.[16]

Avec le schéma ci-dessous représentant les quatre états non orthogonaux utilisés dans le protocole BB84. Telle que le bit classique, il est codé par des états quantiques. Chaque état quantique peut représenter les deux bits classiques, le 1 ou le 0, et inversement, chaque 0 ou 1 correspond à un mélange de deux états quantiques égaux, probablement non orthogonaux.

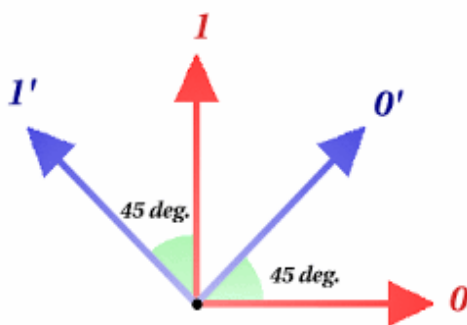


FIGURE 3.11 – polarisation d'un photon [8]

L'information transmise dans le canal quantique est souvent sous la forme de photons polarisés. Le codage des bits classiques est fait en utilisant la direction de la polarisation. Dans le schéma de codage de BB84, le bit classique 0 est représenté par un photon polarisé à 0° et 45° de l'axe horizontal, et les deux directions orthogonales correspondantes, 90° et 135° , sont employées pour le bit 1.

3.6 Limitations et vulnérabilités

- **Limites de la cryptographie asymétrique :** La cryptographie asymétrique, telle que la cryptographie à clé publique, est souvent plus lente que la cryptographie symétrique. Cependant, cela peut être un problème pour les applications nécessitant une rapidité élevée.
- **Rapidité et sécurité :** Les algorithmes de cryptographie post-quantique sont conçus pour résister aux attaques quantiques, mais peuvent être plus lents que les algorithmes conventionnels, ce qui peut être un problème pour les applications nécessitant une rapidité élevée.
- **Vulnérabilités aux attaques de force brute :** Les algorithmes de cryptographie asymétrique, tels que RSA et la cryptographie de courbe elliptique (ECC), peuvent être vulnérables aux attaques de force brute.
- **Puissance de calcul :** Les algorithmes de cryptographie asymétrique sont généralement plus lents et nécessitent plus de puissance de calcul que les algorithmes symétriques.

3.7 Conclusion

Pour conclure ce chapitre, nous avons exploré la cryptographie moderne comprenant ses deux types principaux, symétrique et asymétrique, ainsi que la nouvelle approche qui est la cryptographie quantique. Notre prochain chapitre consistera à examiner la cryptographie hybride, une combinaison de deux méthodes symétrique et asymétrique.

CHAPITRE 4

PROPOSITION D'UN SYSTÈME DE CRYPTOGRAPHIE HYBRIDE

4.1 Introduction

Dans de nombreuses techniques de cryptographie modernes, les chercheurs ont contribué à identifier les meilleurs mécanismes de cryptographie en fonction de leur performance. La sélection d'une technique de cryptographie en fonction d'un contexte spécifique reste une question. Pour répondre à cette question de choix, de nombreuses études ont affirmé que ce choix dépendrait des attributs de qualité souhaités, tels que l'efficacité et la sécurité. Il a été constaté que les études existantes se concentrent généralement sur les types de chiffrement symétrique ou asymétrique.[13]

4.2 Évaluation des techniques de cryptographie moderne

1.Importance de l'évaluation

L'évaluation des systèmes de cryptographie permet de déterminer leur efficacité, leur résistance aux attaques et leur conformité aux normes de sécurité établies. Cela implique de tester les algorithmes de chiffrement sous divers scénarios et conditions, d'analyser leur performance en termes de vitesse de chiffrement et de déchiffrement, d'évaluer la taille et la gestion des clés.[13]

2.Méthodologie d'évaluation

il est possible de fournir une évaluation des systèmes de cryptographie modernes, ce qui est crucial pour garantir la sécurité des données. Lors de l'évaluation des performances des systèmes de cryptographie, plusieurs critères doivent être pris en compte pour garantir la sécurité et la vitesse. Voici quelques-uns des critères de performance clés à considérer :

1. La vitesse de chiffrement et de déchiffrement : Mesurez le temps nécessaire pour chiffrer et déchiffrer un ensemble de données de taille standard (temps d'exécution).
2. La taille de la clé : examinez la taille de la clé utilisée dans l'algorithme de chiffrement. Une clé plus longue peut améliorer la sécurité, mais elle peut également ralentir les opérations de chiffrement et de déchiffrement.
3. Utilisation de la mémoire : Il est important de considérer l'utilisation de la mémoire vive (RAM) pendant les opérations de chiffrement et de déchiffrement, car cela peut influencer les performances globales du système.
4. Flexibilité : capacité à s'adapter à différents contextes d'utilisation.
5. La complexité est une métrique importante pour évaluer les algorithmes de cryptographie moderne. Elle mesure la difficulté d'implémentation, de maintenance et de mise à jour d'un algorithme, le nombre d'instruction.
6. La résistance aux attaques : capacité à résister aux différentes techniques d'attaque.

3.Choix des algorithmes

Après avoir étudié et comparé les algorithmes de cryptographie modernes, nous avons décidé de combiner l'AES (Advanced Encryption Standard) avec l'ECC (Elliptic Curve Cryptography) pour assurer une sécurité meilleur lors du chiffrement des messages. Notre algorithme consiste à d'abord chiffrer le message avec AES pour garantir sa confidentialité, puis à chiffrer la clé AES générée avec ECC pour assurer sa protection pendant le transfert. Cette combinaison permet d'allier l'efficacité de l'AES dans le chiffrement de volumes de données avec la robustesse de l'ECC dans la gestion des clés. Ce que nous allons bien détailler maintenant.

AES (Advanced Encryption Standard)

AES est l'un des algorithmes de chiffrement par blocs symétriques les plus courants et les plus répandus. Il est extrêmement difficile pour les pirates d'obtenir les données réelles lorsqu'elles sont cryptées par l'algorithme AES. Jusqu'à présent, il n'y a pas aucune preuve de l'échec de cet algorithme.

Processus de chiffrement

AES s'appuie sur un certain nombre de tours et, à l'intérieur de chaque tour, comprend quatre sous-processus. Chaque tour se compose des quatre étapes suivantes pour crypter un bloc de 128 bits.[43]

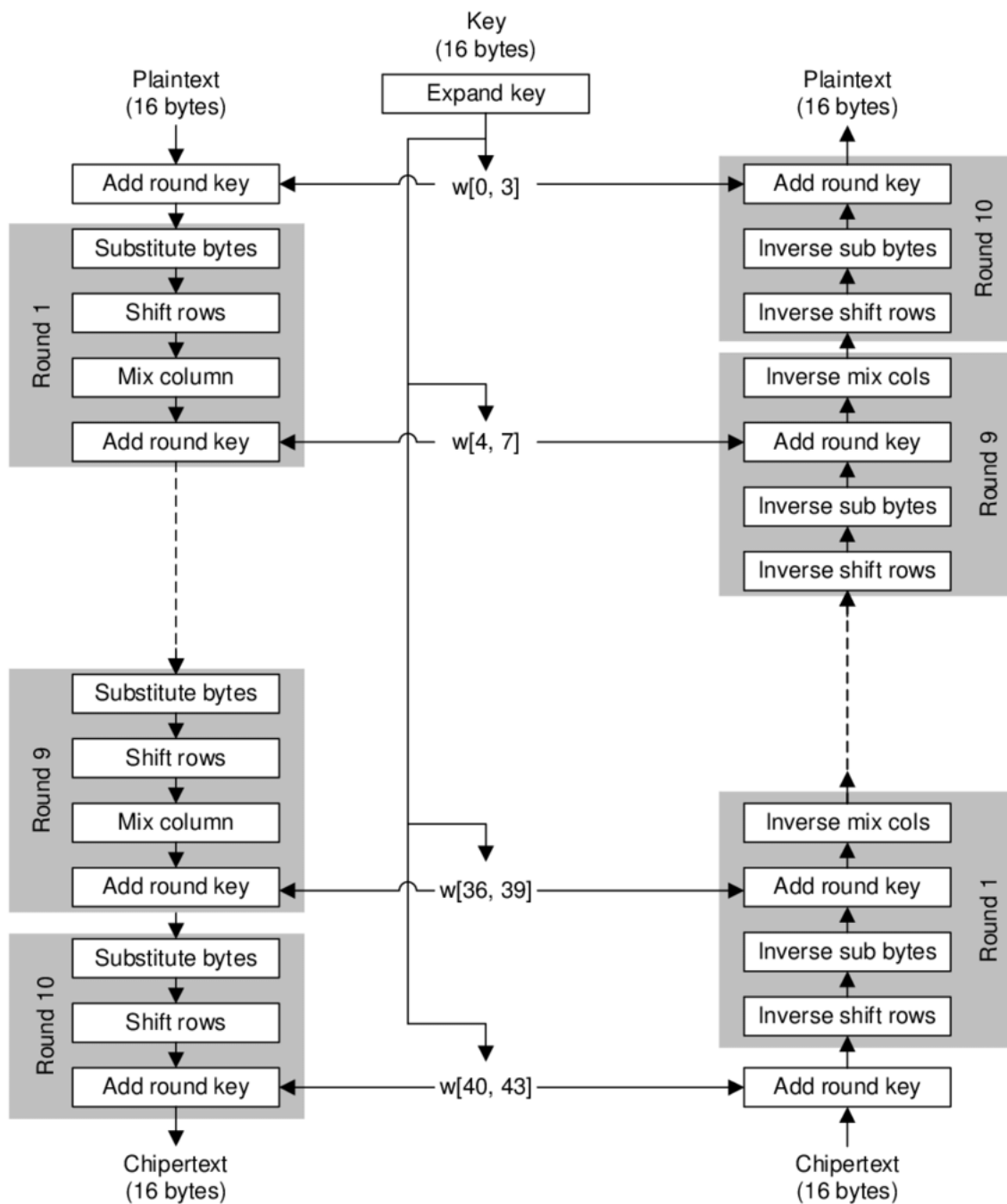


FIGURE 4.1 – Chiffrement AES [10]

A. Transformation des octets de substitution

La première étape de chaque tour commence par la transformation des sous-octets. Cette étape dépend de la boîte en S non linéaire pour remplacer un octet dans l'état par un autre octet. Selon les principes de diffusion et de confusion principes de Shannon pour la conception d'algorithmes cryptographiques, elle joue un rôle important dans l'obtention d'une sécurité. Par exemple, dans l'algorithme AES, si nous avons un hexa 53 dans l'état, il doit être remplacé par l'hexa ED. ED est créé à partir de l'intersection de 5 et de 3. Pour les octets restants de l'état, effectuer

cette opération :

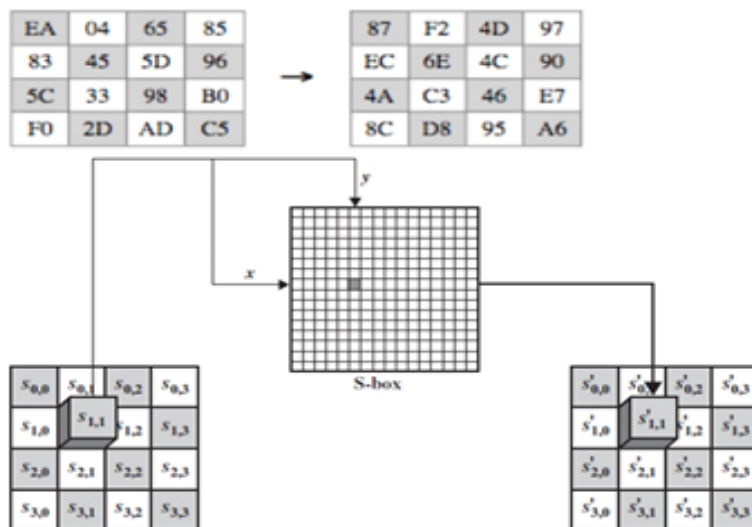


FIGURE 4.2 – Transformation de substitution

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
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	85	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 4.3 – Table de boîte de substitution AES [10]

B. Transformation ShiftRows

L'idée principale de cette étape (Décalage des lignes) est de décaler cycliquement les octets de l'état vers la gauche dans chaque ligne plutôt que la ligne numéro zéro. Dans ce processus, les octets de la ligne numéro zéro restent et n'effectuent aucune permutation. Dans la première ligne, seul un octet est décalé circulairement vers la gauche. La deuxième rangée est décalée de deux octets vers la gauche. La dernière ligne est décalée de trois octets vers la gauche. La taille du nouvel état n'est pas modifiée, elle reste la même taille originale de 16 octets, mais la position des octets dans l'état est décalée, comme illustré à la suivante :

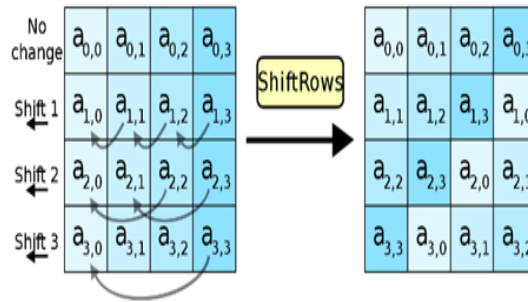


FIGURE 4.4 – La transformation ShiftRows
[10]

La figure 4.4 illustre le décalage des lignes dans l'algorithme de chiffrement AES

C. Transformation MixColumns

Une autre étape (Mélange des colonnes) se produit dans l'état MixColumns. La multiplication est effectuée par l'état. Chaque octet d'une ligne de la matrice par chaque valeur (octet) de la colonne de l'état. En d'autres termes, chaque ligne de la transformation matricielle doit être multipliée par chaque colonne de l'état. Les résultats de ces multiplications sont utilisés avec XOR pour produire un nouveau quatre octets pour l'état suivant. Dans cette étape, la taille de l'état n'est pas modifiée.[40]

$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \times \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

FIGURE 4.5 – La transformation Mix-colonne
[10]

Cette figure monte la multiplication de cette matrice par chaque colonne de l'état, donc, par exemple, pour calculer b_0 , on fait :

$$b_0 = (2 \cdot a_0) \oplus (3 \cdot a_1) \oplus (1 \cdot a_2) \oplus (1 \cdot a_3) \quad (4.1)$$

Ces valeurs donneraient la nouvelle colonne transformée de l'état après l'application de MixColumns.

D. Transformation AddRoundKey

AddRoundKey (Addition de la clé de tour) est l'étape la plus importante de l'algorithme AES. La clé et les données d'entrée (également appelées l'état) sont structurées dans une matrice 4 x 4 d'octets. La figure suivante montre comment la clé de 128 bits et les données d'entrée sont réparties dans la matrice de 4 x 4 octets.

AddRoundKey a la capacité de fournir une sécurité beaucoup plus de sécurité lors du cryptage des données. Cette opération est basée sur la création d'une relation entre la clé et le texte chiffré. Le texte chiffré provient de l'étape précédente. La sortie AddRoundKey s'appuie exactement sur la clé indiquée par les utilisateurs. En outre,

la sous-clé est également utilisée et combinée avec l'état.[40]

La clé principale est utilisée pour dériver la sous-clé à chaque tour en utilisant le calendrier des clés de Rijndael. La taille de la sous-clé et de l'état est la même. La sous-clé est ajoutée en combinant chaque octet de l'état avec l'octet correspondant de la sous-clé à l'aide d'un XOR.

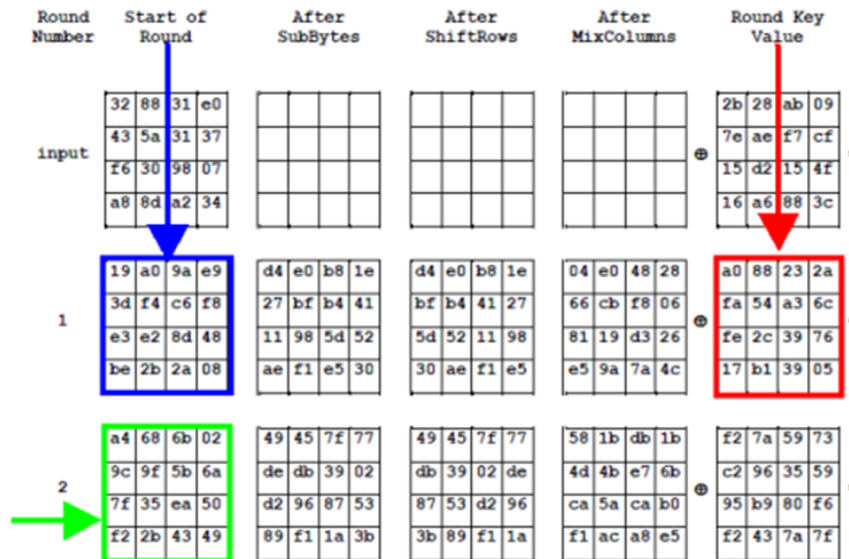


Fig. 6 Add Round Key

FIGURE 4.6 – La transformation Addroundkey [10]

La figure 4.6 montre. Le tableau de la progression du bloc de texte chiffré à travers les différentes étapes d'un tour AES, illustrant comment chaque transformation (SubBytes, ShiftRows, MixColumns, AddRoundKey) contribue à la sécurité et à la complexité du chiffrement final.

Expansion de la clé AES

L'algorithme AES est basé sur l'expansion de clé AES pour crypter et décrypter les données, qui est le processus par lequel la clé de chiffrement initiale est étendue pour générer une série de sous-clés utilisées dans chaque tour de l'algorithme de chiffrement. Cette opération prend la clé d'origine, qui peut être de 128, 192 ou 256 bits, et produit plusieurs sous-clés de 128 bits pour chaque tour du processus de chiffrement. Il s'agit d'une autre étape importante de la structure AES. Chaque tour a une nouvelle clé. Cette section se concentre sur la technique d'expansion de clé AES.

La routine d'expansion de clé crée les clés de chaque tour mot par mot, un mot étant un tableau de quatre octets. La routine crée $4 \times (N_r + 1)$ mots, où N_r est le nombre de tours[14].

Le processus est le suivant :

La clé de chiffrement (clé initiale) est utilisée pour créer les quatre premiers mots.

La taille de la clé est de 16 octets (k_0 à k_{15}), comme le montre la figure 8 qui la représente dans un tableau. Les quatre premiers octets (k_0 à k_3) représentent w_0 , les quatre octets suivants (k_4 à k_7) dans la première colonne représentent w_1 , et ainsi de suite.

Nous pouvons utiliser l'équation particulière pour calculer et trouver les clés à chaque tour comme suit :

$$K[n] : W[i] = K[n - 1] : w[i] \oplus k[n] : w[i] \quad (4.2)$$

Cette équation permet de trouver une clé pour chaque tour plutôt que w_0 . Pour w_0 , nous devons utiliser une équation particulière qui est différente de l'équation ci-dessus :

$$K[n] : w_0 = k[n - 1] : w_0 \oplus \text{SubBytes}(k[n - 1] : w_3 \ggg 8) \oplus \text{Rcon}[i] \quad (4.3)$$

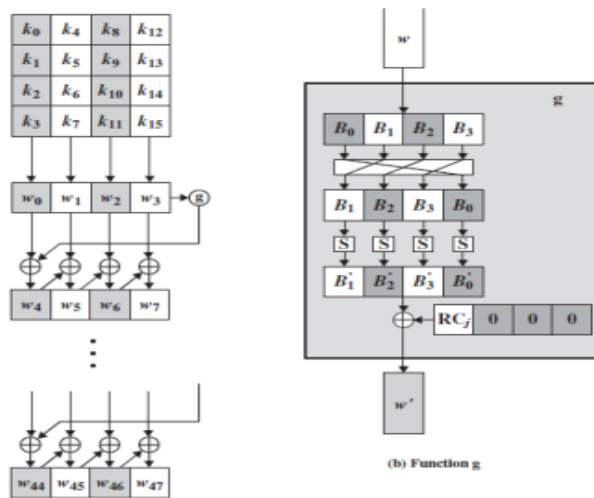
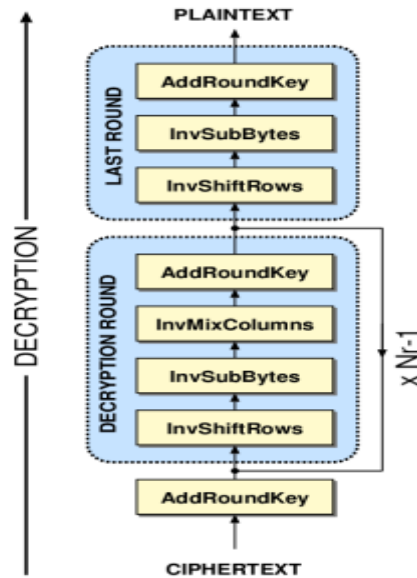


FIGURE 4.7 – Expansion de la clé AES [10]

Processus de Décryptage

Ce processus est basé sur la clé qui a été reçue de l'expéditeur des données. Le processus de décryptage d'un AES est similaire au processus de cryptage dans l'ordre inverse, et l'expéditeur et le destinataire ont la même clé pour crypter et décrypter les données.[40]

le dernier tour d'une étape de décryptage se compose de trois étapes telles que InvShiftRows, InvSubBytes et AddRoundKey comme le montre la figure suivante.


 FIGURE 4.8 – Déchiffrement AES
 [10]

Caractéristiques et Points forts de AES

AES est un algorithme de chiffrement qui répond très bien aux besoins en termes de sécurité et aux contraintes (temps réel, taille des données et ressources limitées). En effet, AES offre un niveau de sécurité suffisant pour tout type d'application commerciale. Il se distingue par sa facilité de calcul, sa rapidité de traitement, ses faibles besoins en ressources et en mémoire, ainsi que sa simplicité.

ECC (Elliptic Curve Cryptographie)

Le principe

Le principe des courbes elliptiques (EC) repose sur l'utilisation de points sur une courbe définie par une équation spécifique pour effectuer des opérations cryptographiques, offrant ainsi une sécurité équivalente à des clés de plus grande longueur utilisées dans d'autres systèmes comme RSA. Les opérations cryptographiques sur ces courbes, notamment la multiplication de points, sont computationally complexes, ce qui les rend particulièrement efficaces pour des applications nécessitant une sécurité élevée avec des clés plus courtes.[26]

D'une manière générale, sur \mathbb{R} , les courbes elliptiques seront considérées comme l'ensemble des couples (x, y) tels que :

$$y^2 = x^3 + ax + b \quad (4.4)$$

Dont le discriminant :

$$\Delta = -(4a^3 + 27b^2) \quad (4.5)$$

doit être non nul.

Pour tracer la courbe, pour des valeurs fixées de a et b , on calcule y tel que :

$$y = \sqrt{x^3 + ax + b} \quad (4.6)$$

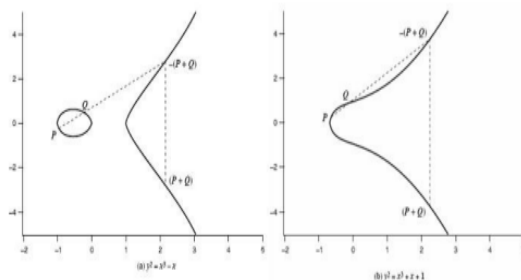


FIGURE 4.9 – Exemple d'une courbe ECC [8]

Définition géométrique

Soit $E(a, b)$ une courbe elliptique définie par l'équation $y^2 = x^3 + ax + b$, où a et b répondent à la condition du discriminant non nul.

Si 3 points sur une courbe elliptique sont alignés, leur somme vaut O (point à l'infini).

- O est l'élément neutre pour l'addition : $O + P = P$ pour tout point P de la courbe.
- L'opposé d'un point $P(x, y)$ est $-P(x, -y)$.
- Pour additionner deux points P et Q , on trace la droite les reliant. Elle intercepte la courbe en un troisième point R .
- On définit alors $P + Q$ comme étant le symétrique de R par rapport à l'axe des abscisses.

En conséquence, on définit l'addition de points sur une courbe elliptique de manière géométrique, en utilisant les propriétés des droites et des courbes[45].

Les courbes elliptiques sur $\mathbb{Z} - p$

Les variables et coefficients prennent des valeurs dans l'ensemble $[0, p - 1]$ pour un certain nombre premier p , où toutes les opérations sont calculées modulo p . L'équation devient :

$$y^2 \pmod p = (x^3 + ax + b) \pmod p \quad (4.7)$$

Cette équation est par exemple satisfaite pour $a = 1, b = 1, x = 9, y = 7$ et $p = 23$:

$$7^2 \pmod{23} = (9^3 + 9 + 1) \pmod{23}$$

$$49 \pmod{23} = 739 \pmod{23}$$

$$3 = 3$$

On note $E_p(a, b)$ l'ensemble des couples d'entiers (x, y) qui satisfont cette équation. On parle de groupe elliptique.

La cryptographie sur ECC

Pour utiliser les courbes elliptiques en cryptographie, il faut trouver un problème

difficile (tel que la factorisation d'un produit en ses facteurs premiers dans le cas du RSA). Considérons l'équation :

$$Q = kP \tag{4.8}$$

ECC pour l'échange de clés

Soit un grand entier premier q et les paramètres a et b satisfaisant l'équation :

$$y^2 \pmod q = (x^3 + ax + b) \pmod q \tag{4.9}$$

Cela nous permet de définir $E_q(a, b)$.

Prenons ensuite un point de départ $G(x_1, y_1)$ dans $E_q(a, b)$ dont l'ordre n est élevé. L'ordre n d'un point sur une EC est le plus petit entier positif tel que $nG = O$.

$E_q(a, b)$ et G sont rendus publiques[30].

L'échange d'une clé par ECC entre deux entités A et B se déroule comme suit :

- A choisit un n_A inférieur à n qui sera sa clé privée.
- A génère alors sa clé publique $P_A = n_A \times G$.
- B choisit un n_B inférieur à n qui sera sa clé privée.
- B génère alors sa clé publique $P_B = n_B \times G$.
- A génère la clé secrète $K = n_A \times P_B$ et B génère la clé secrète $K = n_B \times P_A$.

Principe de chiffrement

Pour chiffrer un message, on encode le texte clair m comme un point P_m de coordonnées x et y . C'est ce point qui sera chiffré. Il est nécessaire de rendre publique un point G et un groupe elliptique $E_q(a, b)$. Les utilisateurs doivent également choisir une clé privée et générer la clé publique correspondante.

Pour chiffrer le message, l'entité A détermine aléatoirement un nombre entier positif k et produit C_m comme un couple de points tel que :

$$C_m = \{kG, P_m + kP_B\} \tag{4.10}$$

Principe de déchiffrement

Pour déchiffrer, B devra multiplier le premier point par sa clé privée, et soustraire le résultat au second point reçu :

$$P_m + kP_B - n_B(kG) = P_m + k(n_BG) - n_B(kG) = P_m \tag{4.11}$$

Ce principe permet à l'entité B de retrouver le message clair P_m en utilisant sa clé privée pour décrypter les données chiffrées reçues.

les Points forts de ECC

Les courbes elliptiques sont largement utilisées en cryptographie en raison de leurs propriétés spéciales qui les rendent idéales pour divers protocoles cryptographiques. En résumé, voici quelques points :

- Sécurité avec de petites clés : les courbes elliptiques permettent d'obtenir un niveau de sécurité comparable à celui des autres schémas cryptographiques (comme RSA ou Diffie-Hellman) avec des clés beaucoup plus petites.
- Efficacité en termes de calculs : Les opérations arithmétiques sur les courbes elliptiques sont plus rapides que celles sur les grands entiers utilisés dans d'autres schémas cryptographiques. Cela se traduit par des opérations plus rapides pour le chiffrement, la signature et d'autres opérations cryptographiques.

4.3 Structuration de Notre cryptosystème HAE

1. Génération de la clé partagée

Nous Utilisons l'algorithme ECC pour générer une paire de clés publique/privée. La clé privée restera secrète et sera utilisée pour le déchiffrement, tandis que la clé publique sera partagée avec les destinataires.

2. Chiffrement hybride

Lorsque nous souhaitons envoyer un message à un destinataire, on génère une clé de chiffrement symétrique aléatoire (par exemple, une clé AES) :

1. Nous chiffons le message avec la clé AES.
2. Ensuite, nous chiffons la clé AES avec la clé publique du destinataire en utilisant l'algorithme ECC.
3. Nous envoyons à la fois le message chiffré avec AES et la clé AES chiffrée avec ECC au destinataire.

3. Déchiffrement hybride

1. Le destinataire reçoit à la fois le message chiffré et la clé AES chiffrée.
2. Il utilise la clé privée ECC pour déchiffrer la clé AES.
3. Ensuite, il utilise la clé AES déchiffrée pour déchiffrer le message.

Cette approche hybride profite des avantages de chaque type d'algorithme :

- Sécurité renforcée L'utilisation de la cryptographie à courbes elliptiques pour chiffrer la clé de session AES offre une couche de sécurité supplémentaire. Les algorithmes de chiffrement asymétrique comme ECC, sont généralement considérés comme très sécurisés lorsqu'ils sont correctement mis en œuvre, ce qui renforce la sécurité de l'échange de clés.
- Échange de clés sécurisé Avec l'utilisation d'ECC pour l'échange de clés, les clés de session peuvent être échangées de manière sécurisée entre les parties sans avoir besoin de partager de clés secrètes à l'avance. Cela simplifie le processus d'échange de clés et réduit les risques liés à la sécurité des clés.
- Efficacité AES est un algorithme de chiffrement symétrique rapide et efficace qui est bien adapté au chiffrement de grandes quantités de données.

Voici l'explication de notre approche : Afin de comprendre bien ces étapes, on a mentionné comme un scénario entre Alice et Bob comme suit : Alice et Bob conviennent d'une courbe elliptique E spécifique à utiliser, ainsi que d'un point P générateur sur cette courbe. Ce point doit être d'ordre suffisamment élevé pour offrir une sécurité adéquate.

Étape 1 : Génération de la clé secrète partagée S

- a. Choisir un grand nombre entier a .
- b. Calculer $Q_a = a \times P$ et l'envoyer à l'entité B.

- c. Recevoir $Q_b = b \times P$ de l'entité B.
- d. Calculer $K = a \times Q_b = a \times b \times P$ qui représente la clé secrète établie entre l'entité A et B.

Étape 2 : Génération d'une clé de session AES

Alice génère une clé de session aléatoire K pour le chiffrement AES, il utilise un générateur de nombres aléatoires cryptographiquement sécurisé pour créer une séquence de bits aléatoire de longueur appropriée. Une fois générée, K est prête à être utilisée pour chiffrer le message M .

Étape 3 : Chiffrement du message avec AES

L'algorithme AES divise le message en blocs de taille fixe (par exemple, 128 bits pour AES-128, 192 bits pour AES-192, ou 256 bits pour AES-256).

- b. Chaque bloc de message est chiffré séparément en utilisant la clé de session K .
- c. L'algorithme AES utilise une série de transformations (SubBytes, ShiftRows, MixColumns et AddRoundKey) répétées sur chaque bloc de données pour produire le texte chiffré C_{aes} .

Étape 4 : Chiffrement de la clé de session avec ECC

Une fois que la clé de session K est préparée, Alice chiffre cette valeur en utilisant l'algorithme de chiffrement asymétrique ECC choisi comme suit :

1. Alice choisit un entier aléatoire r et calcule :

$$C_{ecc} = r \times P \quad (4.12)$$

(où P est le point générateur).

2. ensuite, elle calcule :

$$C' = K + r \times B. \quad (4.13)$$

Étape 5 : Envoi du texte et de clé chiffrés

Alice envoie (C_{aes}, C_{ecc}, C') à Bob.

Étape 6 : Réception et déchiffrement par Bob

1. Bob les reçoit et extrait C_{aes}, C_{ecc} . en utilisant sa clé privé.
2. il calcule :

$$K = C' - b \times C_{ecc} \quad (4.14)$$

3. ensuite, il déchiffre C_{aes} avec la clé K pour récupérer le Message M .

4.4 Implementations et Résultats

Dans cette section, nous avons présenté les résultats obtenus lors de l'implémentation des algorithmes de cryptographie, qu'ils soient symétriques ou asymétriques. Nous avons réalisé une comparaison des temps nécessaires au chiffrement, au déchiffrement et à la génération de clés, ainsi qu'au rapport à l'espace mémoire. Notre objectif était de trouver celui qui offre le meilleur compromis entre l'efficacité et la vitesse de traitement, puis les résultats obtenus avec notre approche proposée.

4.4.1 L'environnement de développement

Le Jupyter Notebook est une application Web open source permettant de créer et de partager des documents. Anciennement appelé IPython Notebooks, il s'agit d'un environnement de calcul interactif basé sur le Web permettant aux utilisateurs de Python de créer des documents notebooks.

Python est un excellent choix pour implémenter des algorithmes de cryptographie en raison de sa simplicité, de sa lisibilité et de sa richesse en bibliothèques.



FIGURE 4.10 – jupyter notebook
[2]

Voici quelques bibliothèques populaires en Python pour la cryptographie, avec une brève explication de leur principe :

- Matplotlib est une bibliothèque Python essentielle pour la création de visualisations statiques et interactives, offrant un large éventail de types de graphiques et de personnalisations.
- PyCryptodome : est une version autonome de la bibliothèque PyCrypto, qui est maintenant obsolète. Elle offre une implémentation de nombreux algorithmes de chiffrement, de hachage et d'autres primitives cryptographiques.
- PyNaCl : est une bibliothèque Python qui met en œuvre un ensemble de primitives cryptographiques modernes et sécurisées, y compris le chiffrement à clé publique et à clé secrète, les signatures numériques, et plus encore.
- Eciespy : est une implémentation d'un schéma de chiffrement intégré basé sur des courbes elliptiques.

Notre analyse comparative se base sur une variété d'algorithmes de cryptographie, incluant à la fois des méthodes symétriques telles que RSA, ELGamal et ECC, ainsi que des méthodes asymétriques telles que AES, DES et RC4.

4.4.2 Tests et Résultats

4.4.2.1 Résultats pour les algorithmes Asymétrique

Comparaison entre RSA et ELGamal par rapport au temps de chiffrement et déchiffrement

Selon les résultats des mises en œuvre de ces deux algorithmes, les valeurs obtenues (en ms) ont été obtenues pour différentes tailles de fichiers avec une clé de taille assez lente. Voici le rapprochement :

taille du texte(ko)	RSA		ELGamal	
	C	D	C	D
20	817	619	1235	1112
40	2638	1834	2901	2580
50	2704	2012	4410	3100

TABLE 4.1 – comparaison entre RSA et ElGamal

Voici les résultats sous forme d'un diagramme :

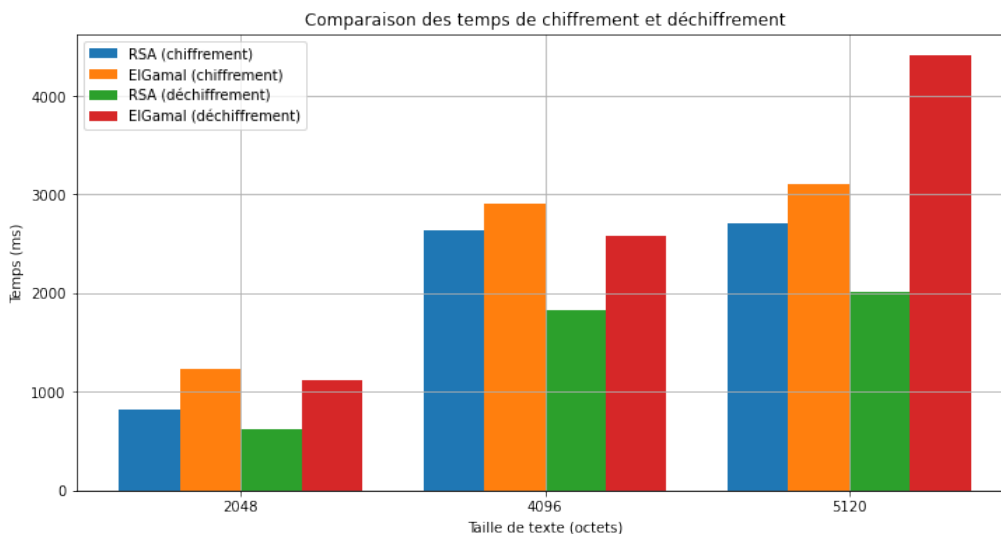


FIGURE 4.11 – comparaison entre RSA et ELGamal par rapport au temps de chiffrement et déchiffrement

Selon ces résultats, l'algorithme RSA a le temps de chiffrement et déchiffrement moins que ElGamal dans ces tâches. Selon les résultats, le temps de traitement augmente en fonction de la taille des données. En outre, il est observé que la vitesse de déchiffrement est généralement supérieure à celle du chiffrement. Cela peut être dû à la disparité de taille des clés utilisées pour chaque opération, où la clé privée est généralement plus petite que la clé publique, ce qui peut entraîner une plus grande vitesse dans les opérations de déchiffrement.

Comparaison entre RSA et ElGamal par rapport au temps de génération de clés

taille des clés	Elgamal		RSA	
	C	D	C	D
1024bits	14	22	60	34
2048bits	47	61	112	98

TABLE 4.2 – comparaison entre RSA et ElGamal en terme du temps de génération de clés

voici les résultats sous forme graphique :

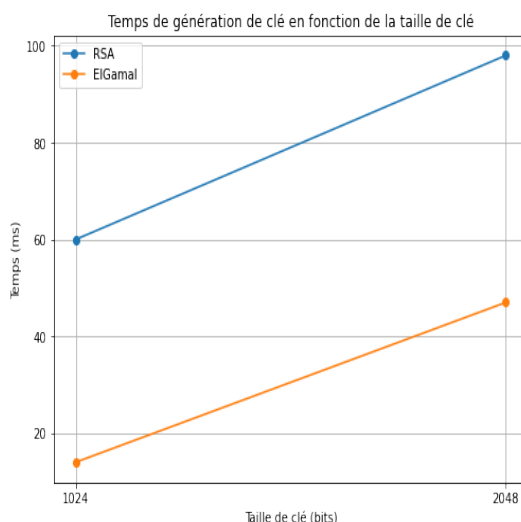


FIGURE 4.12 – comparaison entre RSA et ElGamal en temps de génération de clé

En analysant les délais de génération de clés, il est évident que RSA nécessite davantage de temps que ElGamal. Il s'agit du procédé de création de clés propre à chaque algorithme. Pour le RSA, il s'agit de trouver deux nombres premiers différents et de les multiplier afin d'obtenir le produit n . La durée de ce processus peut varier. D'autre part, pour ElGamal, la clé publique est calculée à partir de la clé privée, ce qui peut être plus rapide.

Comparaison entre RSA et ElGamal par rapport à l'espace mémoire
 l'espace mémoire est déterminé de taille de clés et la taille des opérations de chaque algorithme.

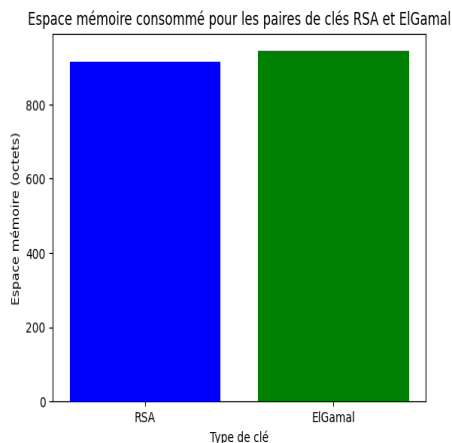


FIGURE 4.13 – comparaison entre RSA et ELGamal en terme d'espace mémoire

On constate que l'algorithme RSA nécessite moins d'espace mémoire que ELGamal pour chiffrer et déchiffrer les données. Et à mesure que le message à chiffrer devient plus grand, l'espace mémoire utilisé par l'algorithme de chiffrement augmente.

Comparaison entre RSA et ECC par rapport au temps de chiffrement, déchiffrement et génération de clés

En utilisant deux clés de 192 et 256 bits pour chaque algorithme, ce tableau montre les résultats des temps de chiffrement, de déchiffrement et de génération de clés (moyenne) obtenus lors de l'implémentation de RSA et ECC pour diverses tailles de fichiers :

	RSA		ECC	
taille de clé(bits)	192	256	192	256
génération de clé	6	8	3	5
chiffrement(ms)	1566	1701	1044	1324
déchiffrement(ms)	1401	1492	981	988

TABLE 4.3 – comparaison entre RSA et ECC

D'après les résultats, ECC est plus rapide que RSA en termes de temps d'exécution, car il n'est pas nécessaire d'utiliser des exponentiations modulaires qui nécessitent un certain temps, ECC utilise des multiplications simples.

Pour la génération de clés, remarquons également que ECC est plus rapide. Ça, c'est logique d'après notre étude dans les chapitres précédents, car ECC utilise des tailles de clés plus petites par rapport à RSA qui donne le même niveau de sécurité.

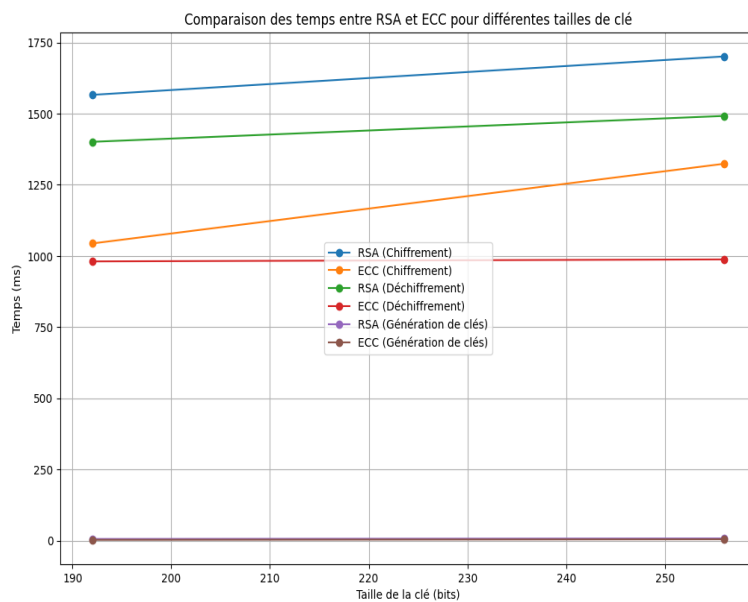


FIGURE 4.14 – comparaison entre RSA et ECC en terme de temps

Comparaison entre RSA et ECC par rapport a l'espace mémoire
voici la proportion de l'espace mémoire consommé :

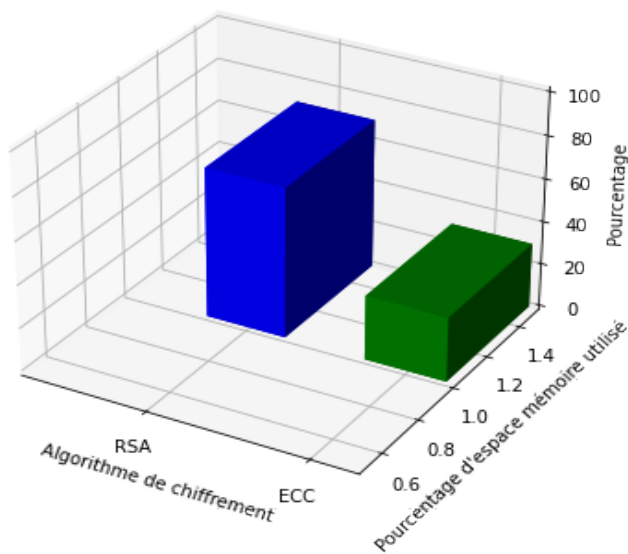


FIGURE 4.15 – comparaison entre RSA et ECC en terme d'espace mémoire

Étant donné la taille réduite des clés ECC, elles demandent moins d'espace de stockage sur les appareils et dans les bases de données. Cela revêt une grande importance dans les situations où l'espace mémoire est restreint.

4.4.2.2 Résultats pour les algorithmes symétrique

Comparaison entre AES et DES par rapport au temps de chiffrement et déchiffrement

Pour le DES, nous avons employé une clé de 64 bits et pour l'AES, une clé de 128 bits. Les résultats obtenus (moyenne) des deux algorithmes ont été étudiés sur différentes tailles de fichiers assez grandes.

taille en ko	DES		AES	
	chiffrement	déchiffrement	chiffrement	déchiffrement
10ko	70	41	40	27
25ko	86	49	52	31

TABLE 4.4 – comparaison entre DES et AES

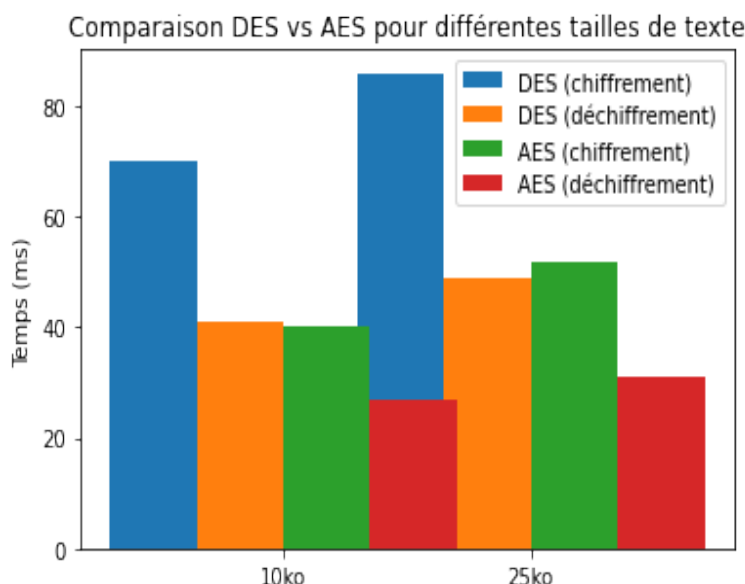


FIGURE 4.16 – comparaison entre DES et AES par rapport au temps de chiffrement et déchiffrement

Les résultats montrent clairement que :

AES est plus rapide que DES par rapport au temps de chiffrement.

AES est plus rapide que DES par rapport au temps de déchiffrement.

donc AES offre des performances de chiffrement et déchiffrement meilleur par rapport à DES, parce que AES effectue un nombre fixe de tours de chiffrement, qui dépend de la taille de la clé (10 tours pour 128 bits, 12 tours pour 192 bits, 14 tours pour 256 bits) par contre DES, utilise un schéma de chiffrement à 16 tours. Avec moins de tours, AES effectue donc moins d'opérations que DES, ce qui le rend plus rapide dans ses opérations de chiffrement et de déchiffrement.

Comparaison entre AES et DES par rapport à l'espace mémoire

Comparaison de l'espace mémoire entre DES et AES

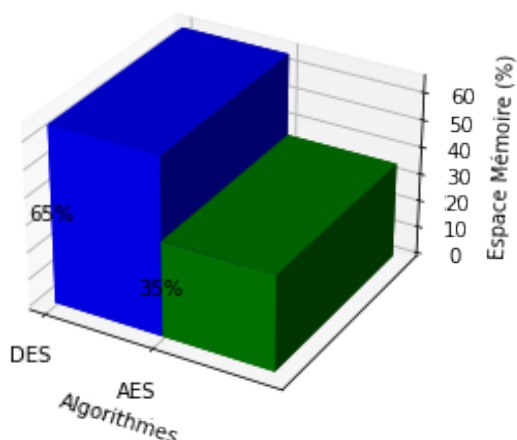


FIGURE 4.17 – comparaison entre DES et AES en terme d'espace mémoire

AES traite les données par bloc de 128 bits, tandis que DES traite les données par blocs de 64 bits. Cela signifie qu'avec AES, nous pouvons chiffrer et déchiffrer plus de données avec AES.

Comparaison entre AES et RC4 par rapport au temps de chiffrement et déchiffrement

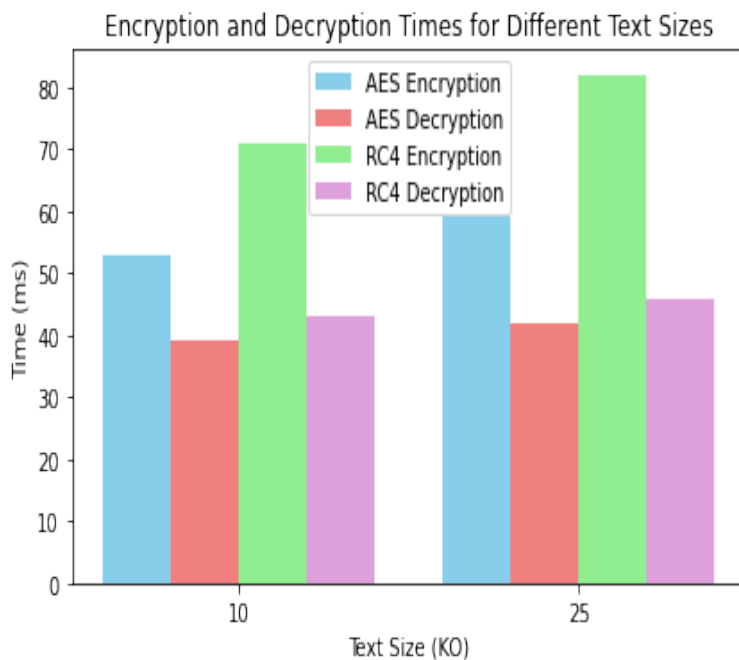


FIGURE 4.18 – comparaison entre AES et RC4 en terme de temps de chiffrement et déchiffrement

Pour les deux tailles de texte (10 KO et 25 KO), le temps de chiffrement avec AES est plus rapide que celui avec RC4.

Comparaison entre AES et RC4 par rapport au temps de génération de clés

On a utilisé une clé de 128 bits pour chacun, on a trouvé que le temps de génération de cette clé en utilisant l'algorithme RC4 est plus rapide que AES, parce que : La clé de RC4 est générée à l'aide d'un algorithme de génération de flux de clés. Par contre, pour AES, la clé est générée à partir de la clé principale.

Comparaison entre AES et RC4 par rapport a l'espace mémoire

Comparaison de l'espace mémoire entre AES et RC4

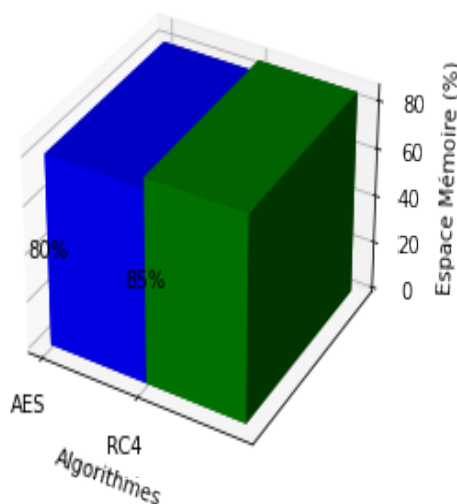


FIGURE 4.19 – comparaison entre AES et RC4 en terme d'espace mémoire

Nous remarquons que les deux algorithmes prend presque le même espace mémoire.

4.4.3 Résultats pour la proposition hybride

L'objectif de cette méthode hybride est de bénéficier des avantages des deux types de chiffrement afin de garantir à la fois la sécurité et l'efficacité du processus de chiffrement.

Paramètres de Sécurité Utilisés

- Pour la partie à clé publique de notre approche hybride, nous avons opté pour la courbe elliptique SECP256R1, également connue sous le nom de P-256. Cette courbe est largement utilisée dans les protocoles de sécurité TLS en raison de sa standardisation et de sa robustesse. On a choisi de cette courbe parce que ces paramètres tels que : le nombre premier p , Le discriminant, les coordonnées de point x, y et Les coefficients de

la courbe elliptique a, b sont choisis avec soin pour éviter d'utiliser une courbe faible, et qui peut résister à toutes les attaques connues. Il peut aussi y avoir d'autres contraintes pour des raisons de sécurité ou de mise en œuvre.[26]

- Pour la partie à clé symétrique, nous avons généré une clé de session aléatoire de 128 bits pour l'algorithme AES, parce que le processus de génération d'une clé de 128bits est plus rapide par rapport à une clé de 256 bits.

Nous avons effectué des tests en chiffrant un fichier de test spécifique à l'aide des deux approches : l'algorithme AES seul et notre approche hybride.

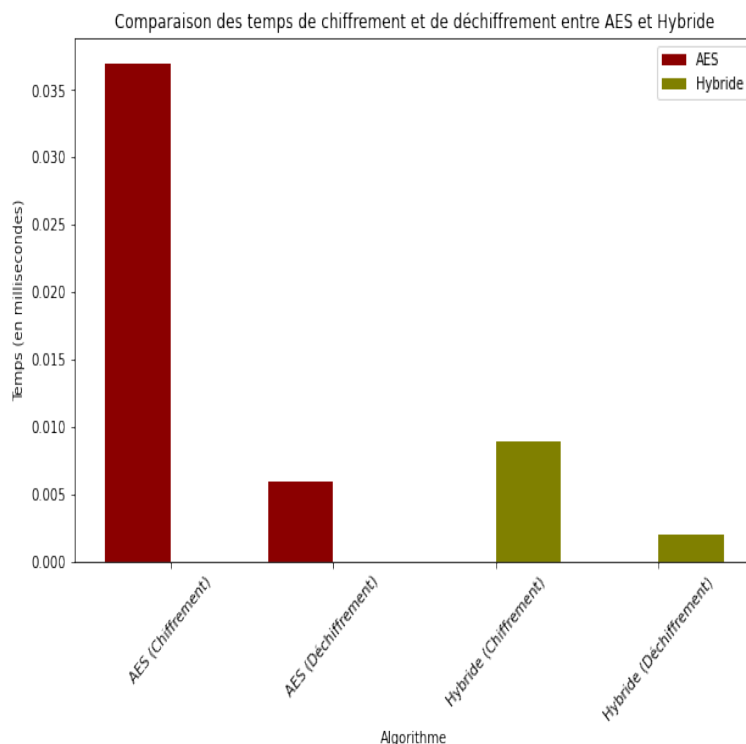


FIGURE 4.20 – comparaison entre l'approche proposée et AES

Analyse et Évaluation de l'approche Hybride

Les tests effectués, comparant l'algorithme AES seul et l'approche hybride, ont montré que cette dernière offre de meilleurs résultats en termes de temps de chiffrement et de déchiffrement. Cela souligne plusieurs points forts de l'approche hybride :

1.Rapidité : L'amélioration des performances dans ce domaine démontre que l'approche hybride est non seulement sécurisée mais aussi efficace en termes de temps de chiffrement et déchiffrement.

2.Sécurité : L'utilisation de la cryptographie à courbes elliptiques (ECC) pour chiffrer la clé de session AES ajoute une couche de sécurité supplémentaire. ECC est réputée pour offrir une sécurité équivalente à celle des autres algorithmes de clé publique, mais avec des tailles de clés beaucoup plus petites, ce qui se traduit par des économies de ressources et une meilleure performance.

3.Échange de Clés Sécurisé : En utilisant ECC pour l'échange de clés, l'approche permet aux parties de partager des clés de session de manière sécurisée

sans avoir besoin de partager de clés secrètes à l'avance. Cela simplifie le processus d'échange de clés et réduit les risques associés à la sécurité des clés, notamment le problème de la distribution sécurisée des clés.

Contribution de l'approche Hybride

L'approche hybride combinant AES et ECC présente plusieurs contributions notables :

- Performance : L'amélioration des temps de chiffrement et de déchiffrement démontre une efficacité accrue, ce qui est essentiel pour les applications en temps réel et les environnements contraints en ressources.
- Sécurité de l'échange de Clés : En éliminant la nécessité de partager des clés secrètes à l'avance, l'approche hybride simplifie et sécurise le processus d'échange de clés, minimisant ainsi les risques de compromission.
- Robustesse : En s'appuyant sur des algorithmes standardisés et robustes comme AES, l'approche garantit une sécurité éprouvée et reconnue.

4.5 Conclusion

Grâce aux résultats obtenus lors de la comparaison des algorithmes symétriques et asymétriques, où nous avons examiné les principales fonctionnalités de chaque algorithme en temps réel, nous avons pu mettre au point une approche qui garantit des délais d'exécution en temps de chiffrement et déchiffrement réduits tout en assurant une protection adéquate des informations confidentielles.

CONCLUSION GÉNÉRALE

A travers notre étude, on a proposé une nouvelle approche cryptographique hybride, que nous avons nommée HAE. Cette approche a été développée à la suite d'une étude détaillée et d'une comparaison approfondie des algorithmes de cryptographie symétrique et asymétrique existants.

Nous avons dressé un panorama de la cryptographie depuis l'antiquité, en examinant à la fois la cryptographie classique avec ses méthodes de chiffrement et la cryptographie moderne avec ses différents types à savoir la cryptographie symétrique et asymétrique. Nous avons comparé ces types selon diverses métriques de performances qui sont : le temps de chiffrement, déchiffrement, temps de génération de clés et l'espace mémoire afin d'identifier le meilleur algorithme en termes de sécurité et d'efficacité. Cette analyse nous a conduit à adopter une approche hybride, combinant le meilleur des deux systèmes pour développer un algorithme efficace, optimisant à la fois la sécurité et l'efficacité dans la protection des données.

En effet, nous avons rencontré divers obstacles tout au long de notre expérience, tels que le manque de temps et les difficultés liées à la mise en œuvre et la compréhension des divers algorithmes. En optant pour le langage de programmation le plus adapté à ce domaine, a été un défi majeur. Malgré ces difficultés, ce travail nous a apporté de nombreux avantages. Nous avons enrichi notre savoir-faire et de nouvelles compétences et techniques dans le domaine de la cryptographie à chaque étape, approfondi notre expertise en analyse et étendu nos connaissances en matière de sécurité. Cette expérience a été très bénéfique pour nous et nous a donné l'opportunité d'acquérir des connaissances importantes dans différents aspects du domaine.

Cependant, pour évaluer pleinement ses performances, il est essentiel de la comparer avec d'autres approches hybrides existantes. Cette comparaison ne devrait pas se limiter uniquement au temps de chiffrement et de déchiffrement, mais aussi inclure des métriques de résistance aux nouvelles attaques potentielles. De plus, il est important d'examiner la capacité de cette approche à s'adapter à différents contextes d'utilisation. Une avenue de recherche future serait d'évaluer l'efficacité de l'approche hybride sur divers types de fichiers, tels que des images, des fichiers audio et d'autres formats de données. Cette diversification permettrait de mieux comprendre l'universalité et la robustesse de l'algorithme dans des scénarios d'application réels.

En outre, la mise en œuvre de cette approche dans le domaine médical, un secteur à la fois essentiel et sensible, représente une perspective particulièrement intéressante. Les données médicales sont d'une importance cruciale et nécessitent des niveaux de sécurité élevés pour protéger la confidentialité des patients.

**“Un ordinateur en sécurité est un ordinateur éteint. Et
encore...”
Bill Gates**

BIBLIOGRAPHIE

- [1]
- [2] itnext.io. <https://itnext.io/jupyternotebooks-a-beginners-guide-8484f6>. consulte le 6 mai 2024.
- [3] mathweb. <http://mathweb.free.fr/crypto/poly/vigenere.php3>. consulte le 24 decembre 2023.
- [4] medium. <https://medium.com/kobalt-si/chiffrement-et-signature-num%C3%A9rique-5798b1e1f8cf>. consulte le 24 decembre 2023.
- [5] nopb. <http://nopb.chez.com/crypto2.html>. consulte le 19 decembre 2023.
- [6] Pretty good privacy.
- [7] Pretty good privacy.
- [8] Researchgate. https://www.researchgate.net/figure/Lalgorithme-AES-Lalgorithme-de-chiffrement-AES-a-ete-developpe_fig1_255666241. consulte le 20 juin 2024.
- [9] researchgate. https://www.researchgate.net/figure/Taxonomy-of-cryptography-techniques_fig4_318200344. consulte le 25 decembre 2023.
- [10] scribd. <https://www.scribd.com/document/489578125/AESAlgorithmpaper2017AKOMAbdullah-docx>. consulte le 17 mars 2024.
- [11] varonis. <https://www.varonis.com/blog/pgp-encryption>, note= consulte le 1 mai 2024 .
- [12] web.maths. <https://web.maths.unsw.edu.au/~lafaye/CCM/crypto/des.htm>. consulte le 25 decembre 2023.
- [13] Shahzad Ahmed and Tauseef Ahmed. Comparative analysis of cryptographic algorithms in context of communication : A systematic review. *International Journal of Scientific and Research Publications*, 12 :161–173, 2022.
- [14] Yacine Mehdi Ait Ameer. Sécurisation des communications dans les réseaux d'ordinateurs (couche ssl). Mémoire de master, Université Mohamed Khider Biskra, Algérie, 06 2014.
- [15] A. Alaoui Ismaili and A. Moussa. Self-partial and dynamic reconfiguration implementation for aes using fpga. *International Journal of Computer Science Issues (IJCSI)*, 1 :33–40, 08 2009.

- [16] Zoé Amblard. *Cryptographie quantique et applications spatiales*. Thèse de doctorat, Université de Limoges, 2016.
- [17] ASJP. Signature numérique de l'information. *ASJP*, 03 2024. <https://www.asjp.cerist.dz/en/downArticle/42/3/1/232884>.
- [18] Frederic Bayart. La saga du des. <http://www.bibmath.net/crypto>, 2023.
- [19] Rabab Beniani. *Sécurité des Images Numériques Compressées JPEG*. Thèse de doctorat, Université Djillali Liabès de Sidi Bel Abbes, 2019.
- [20] Umang Bhargava, Aparna Sharma, Raghav Chawla, and Prateek Thakral. A new algorithm combining substitution transposition cipher techniques for secure communication. In *2017 International Conference on Trends in Electronics and Informatics (ICEI)*, pages 619–624, 2017.
- [21] Umang Bhargava, Priyanka Sharma, Shubham Sharma, and Shubham Sharma. A new algorithm combining substitution, transposition cipher techniques for secure communication. In *2017 International Conference on Trends in Electronics and Informatics (ICEI)*, pages 1023–1027. IEEE, 2017.
- [22] Chaker BOUAKKAZ. Application des techniques de cryptage pour la transmission sécurisée des images. Mémoire de master, Université Larbi Tebessi, Tébessa, 2020.
- [23] L. Bouchenafa, F. Z. Haroun, and A. Krobba. Amélioration de la sécurité du système de vérification du locuteur dans un environnement mobile. Mémoire de master, Université Yahia Fares - Medea, 2021.
- [24] F. Carraro, D. Casajus, S. Houdart, A. Herrou, and I. Rivoal. Cryptographies. In *Introduction*. 2021.
- [25] Ayoub Chanane. *Implantation sur circuit SoC-FPGA d'un système de chiffrement/déchiffrement AES-128 bits en utilisant deux approches de différents niveaux d'abstraction*. Thèse de doctorat, Université du Québec à Rimouski, 2022.
- [26] Thierno Amadou Diallo. Nombre de points rationnels sur une courbe elliptique dans un corps fini. <https://rivieresdusud.uasz.sn/handle/123456789/1742>, 2023.
- [27] Jacqueline Dousson. 2021, l'odyssée quantique. *FI-3-99*. Publié le 13 avril 1999.
- [28] Renaud Dumont. *Cryptographie et Sécurité informatique*. Faculté des Sciences Appliquées, 2010.
- [29] Pr Mohamed Essaidi. *CONTRIBUTIONS À LA SÉCURITÉ DES SYSTÈMES BASÉS SUR LA CRYPTOGRAPHIE CLASSIQUE ET POST-QUANTIQUE*. PhD thesis, Université Hassan II, Casablanca, 2022.
- [30] Pierre-Alain Fouque. Le partage de clés cryptographiques : Théorie et pratique, 2023.
- [31] Arthur Gontier. *Utilisation de solveurs génériques pour la cryptanalyse de chiffrements symétriques*. PhD thesis, Université de Rennes, 2023. Thèse présentée et soutenue à Rennes en novembre 2023.
- [32] M Mahdi Hamza. *Étude et comparaison des principaux systèmes de cryptage et les techniques y afférentes*. Mémoire de master, Université Mohamed Boudiaf - M'Sila, Faculté des Mathématiques et de l'Informatique, 2016.
- [33] Maksim Iavich, Sergiy Gnatyuk, Elza Jintcharadze, Yuliia Polishchuk, and Roman Odarchenko. Hybrid encryption model of aes and elgamal cryptosystems

- for flight control systems. In *2018 IEEE 5th International Conference on Methods and Systems of Navigation and Motion Control (MSNMC)*, pages 229–233, 2018.
- [34] G. Jagadeesh and S.M. Ali. Hybrid aes-modified ecc algorithm for improved data security over cloud storage. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, 32(1) :46–56, 2021.
- [35] David Kahn. *The Story of Secret Writing*. Weidenfeld Nicolson, 1968.
- [36] Mélanie Langlois. Cryptographie quantique - solution au problème de distribution de clés secrètes. *Université d'Ottawa*, 12 1999.
- [37] Nacer Lebsir. Charm : Cryptosystème hybride avancé pour les réseaux mobiles application pour la gestion des comptes bancaires. Mémoire de master, Université Mohammed Seddik Ben Yahia de Jijel, 2019.
- [38] Ye Liu, Wei Gong, and Wenqing Fan. Application of aes and rsa hybrid algorithm in e-mail. 2018.
- [39] Ye Liu, Wei Gong, and Wenqing Fan. Application of aes and rsa hybrid algorithm in e-mail. In *2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS)*. IEEE, 2018.
- [40] Tran Thi Luong. Strengthening aes security through key-dependent shiftrow and addroundkey transformations utilizing permutation. *International Journal of Advanced Computer Science Applications*, 14, 2023.
- [41] Lina MOUSSAOUI and Safa CHOUAICHIA. *Etude et simulation d'un cryptosystème basé sur l'algorithme AES-GCM : Application au cryptage des images médicales*. Mémoire de fin d'étude, Université 8Mai 1945 – Guelma, 06 2023.
- [42] Matar Niane. Conventional hill algorithm : From classical cryptography to modern cryptography. 2022.
- [43] Komal Raesheq, Naveen Kumar, Sandeep Kumar, Ashok Kumar Kashyap, and Ritesh Rana. Performance evaluation of cryptography algorithms : Aes, des, rsa, and ecc. *Journal of Emerging Technologies and Innovative Research (JETIR)*, 10(1) :a217–a224, 01 2023.
- [44] Jérôme Randimbiarison. *Signature numérique d'un document basée sur FIDO2*. PhD thesis, Université Laval, 2020.
- [45] Ismahane SOUCI. *Cryptographie Nouvel Algorithme de Chiffrement Evolutive basé Occurrences (ACEO)*. Diss., 2008.
- [46] G. Tesleanu. *Cryptographic Protocols*. PhD thesis, Simion Stoilow Institute of Mathematics of the Romanian Academy (Roumanie), 2021.
- [47] Ankit Thakkar and Ritika Lohiya. A review on machine learning and deep learning perspectives of ids for iot : Recent updates, security issues, and challenges. *Archives of Computational Methods in Engineering*, 28 :3211–3243, 2020.
- [48] Céline Thuillet. *Implantations cryptographiques sécurisées et outils d'aide à la validation des contremesures contre les attaques par canaux cachés*. Thèse de doctorat, Université BORDEAUX I, France, 03 2012.
- [49] Zhiwei Xu and Jie Ni. Research on network security of vpn technology. In *2020 International Conference on Information Science and Education (ICISE-IE)*, pages 539–542. IEEE, 2020.
- [50] Ouelhadj Yasmina, Soumia Hellal, and Fatiha Benbekhti. Sécurité des réseaux sans fil : Conception et simulation, 2020.

RESUMÉ

La cryptographie est une promesse de sécurité dans un monde où les informations à transmettre sont exposées à divers risques. Dans ce mémoire, nous avons entrepris une étude pour découvrir les secrets de la cryptographie, en commençant par la cryptographie classique et en explorant les nouvelles approches au fil du temps. Nous présenterons plusieurs systèmes de chiffrement, en analysant leurs points forts et leurs faiblesses, et en testant toutes les implémentations dans un environnement de développement. Cette recherche nous conduira à proposer une nouvelle approche hybride combinant deux algorithmes, l'AES (symétrique) et l'ECC (asymétrique).

Mots clés : métrique, comparaison, sécurité, évaluation, algorithme, symétrique, asymétrique, AES, ECC, HAE.

ABSTRACT

Cryptography is a promise of security in a world where transmitted information is exposed to various risks. In this thesis, we have undertaken a study to uncover the secrets of cryptography, starting with classical cryptography and exploring new approaches over time. We will present several encryption systems, analyzing their strengths and weaknesses, and testing all necessary implementations in development environments. This research will lead us to propose a new hybrid approach combining two algorithms : AES (symmetric) and ECC (asymmetric).

KeyWords : metric, comparison, security, evaluation, algorithm, symmetric, asymmetric, AES, ECC, HAE.