REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA
FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE

DEPARTEMENT D'INFORMATIQUE

MEMOIRE de fin d'étude

Présenté pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et Informatique

Filière : Informatique

Spécialité : Technologie de l'Information et de Communication

Par : Mahdi Hamza

SUJET

étude et comparaison des principaux systèmes de cryptage et les technique y afférentes

Soutenu publiquement le : / /2016 devant le jury composé de :

M. El khier DEHMECHE	**Université de M'sila**	**Encadreur**
……	**Université de M'sila**	**Président**
……	**Université de M'sila**	**Rapporteur**
……	**Université de M'sila**	**Examinateur**

Promotion : 2015/20 16

Table des matières

CHAPITRE 3 LA CRYPTOGRAPHIE MODERNE

CHAPITRE 4 IMPLÉMENTATION ET RÉALISATION

Liste des Figures :

Liste des tableaux

INTRODUCTION GENERALE

Tout au long l'histoire, l'être humain a essayé d'envoyer où bien de transmettre des informations de manière sécurisée, c'est à dire confidentielle. le chiffrement d'information a été utilisé comme instrument de sécurisation pour des stratégies militaires et des changes de données secrètes au cours de la seconde guerre mondiale, ou encore de nos jours où les banques recherchent des techniques fiables pour assurer à leurs clients des moyens de régler leurs achats sans risque de fraude, la cryptographie s'est imposée comme passage incontournable dans le transit des informations sensibles.

Utilisée sous diverses formes, sans cesse en progrès, la cryptographie reste une science peu connue quant à son fonctionnement. Si ses formes primitives (remplacement d'un alphabet par un autre, usage de pseudo-langues, de dessins...) sont connues du grand public, les applications modernes, avec usage des mathématiques, de la physique quantique, et des technologies les plus avancées, le sont beaucoup moins. Les utilisateurs de logiciels de cryptage destinés au grand public comme RSA, AES , PGP..., sont certes nombreux, mais ignorent le principe de fonctionnement du programme qu'ils utilisent. Le but de cette étude n'est pas d'être exhaustive quant aux différents algorithmes de cryptage existant, bien trop nombreux pour être seulement cités dans le cadre d'un mémoire de master, mais bien de servir d'introduction à la découverte d'un univers en plein essor, riche en innovations et problèmes irrésolus, en défis humains et informatiques.

Un survol de la méthode dis classiques ou à algorithme secret sera effectué dans la première partie du mémoire. La base de la cryptographie et aussi sa plus ancienne application à savoir la cryptographie à clé secrète ainsi que la cryptographie à clé publique, plus récente, et sur laquelle sont fondés beaucoup d'espoirs, seront toutes les deux abordées dans la deuxième partie. Les activités annexes telles que les fonctions de hachage, la signature électronique. Le point qui sera abordé en dernier concerne le côté pratique du mémoire où sera menée une étude comparative de plusieurs algorithmes cryptographique pour ce qui concerne le chiffrement, le déchiffrement et le hachage. Les plans détaillés de chaque partie ainsi qu'une brève introduction seront présentés au début de chacune.

L’objectif de ce mémoire :

Dans ce mémoire, nous allons répondre aux questions souvent posées par ceux qui veulent avoir un aperçu sur la cryptographie, ses méthodes, ses forces et ses faiblesses ainsi que sur son utilisation pratique :

- Quelles sont les différentes formes qu'a prises la cryptographie à travers le temps ?
- Quel est l'intérêt de la cryptographie moderne et de son utilisation pratique ?
- Quels sont les avantages de la cryptographie à clé secrète et celle à clé publique ? Et quel est le moyen pratique pour combiner les deux méthodes afin de tirer profit des avantages de chacune des deux méthodes ?
- Quels sont les algorithmes cryptographiques fiables par rapports aux autres et quels sont les ordres de grandeur de vitesse, de tailles des clés ?

À travers la réponse sue ces questions on a :

- Comprendre comment la cryptographie permet d’atteindre les différents objectifs (confidentialité, intégrité, authentification, etc.) reliés à la sécurité informatique.

• Comprendre comment la cryptographie est mise en œuvre dans certaines applications réelles (courrier électronique, commerce électronique, etc…).

- Réaliser un logiciel regroupant plusieurs algorithmes de chiffrement, déchiffrement et de hachage puis l’exécuter afin d’obtenir quelques résultats qui vont servir à mener une étude comparative entre ces algorithmes.
- Développer des simulateurs en applets Java afin de vulgariser l'utilisation pratique des différentes méthodes citées à travers ce mémoire.

L’organisation de mémoire :

Pour le plan méthodologique de ce mémoire, nous allons diviser en quatre chapitres :

- Le premier chapitre présente une vue générale sur la sécurité informatique
- Le deuxième chapitre présenté la cryptographie classique tel que l’histoire de la cryptographie et sa l’objectifs et l’étude de quelque algorithmes classique.
- La troisième chapitre étude la cryptographie moderne (La cryptographie à clé secrète et à clé publique, la signature numérique, fonctions de hachage) et l’analyse de quelque algorithmes.
- Le quatrième chapitre traite l’implémentation d’un certain algorithme (classique, symétrique, asymétrique) sur différents plateforme (Windows, exploiteur), et ensuite tester la performance de chaque algorithme, et à la fin la comparaison des résultats avec une discutassions.

CHAPITRE 1

GENERALITES SUR LA SECURITE INFORMATIQUE

1 Introduction

Ce premier chapitre présente une vue générale sur la sécurité tel que : L'objectif de la sécurité, Les attaques, Domaines d'application de la sécurité, Aspects techniques de la sécurité informatique, et Mécanismes de sécurité.

2 Introduction à la sécurité

La sécurité du transport de l'information est une préoccupation primordiale dans le domaine des réseaux. Elle nécessite plusieurs mécanismes pour le sécurisé, il faut l'authentifier. Puisqu'on ne sait pas par où passent les données, il faut les chiffrer. Puisqu'on ne sait pas si quelqu'un ne va pas modifier les informations émises, il faut vérifier leur intégrité.

Aujourd'hui, les réseaux sont toujours devants des menaces. Il y a de plus en plus de techniques pour les protéger, mais il y a aussi de plus en plus de techniques pour les attaquer.

En effet, il est pratiquement impossible d'avoir un réseau complètement sûr et de le protéger contre toutes les attaques possibles.

Pour évaluer et assurer les besoins de sécurité d'une entreprise, il faut considérer trois aspects :

- les services de sécurité.
- les attaques de sécurité.
- les mécanismes de sécurité pour la prévention, la détection des attaques et la réponse à ces attaques.

2.1 Terminologie de la sécurité informatique

La sécurité informatique utilise un vocabulaire bien définie, nous en définirons quelques termes.

- **Sécurité réseau :** [17] définit les moyens mis en œuvre pour protéger les données durant leur transmission, ainsi que pour garantir que les données transmises sont authentiques.
- **Les vulnérabilités :** défaut ou faiblesse dans la conception d'un système, son implémentation, fonctionnement ou administration et qui pourrait être exploité pour violer la politique de sécurité.
- **Les attaques (exploits) :** tentative d'assauts sous forme d'une action ou combinaisons d'actions intelligentes et délibérées dont l'objectif est la violation de la politique de sécurité.
- **Les intrusions :** événement ou combinaisons d'événements permettant d'avoir indûment accès (sans autorisation) à un système et ses ressources.

- **Les contre-mesures :** ce sont les procédures ou techniques permettant de résoudre une vulnérabilité ou de contrer une attaque spécifique (auquel cas il peut exister d'autres attaques sur la même vulnérabilité).
- **Les menaces :** danger potentiel pouvant exploiter une vulnérabilité pour violer la politique de sécurité causant éventuellement des dégâts. Une menace peut être réelle ou non fondée.
- **Risque :** la probabilité qu'une menace exploitera une vulnérabilité du système ; couple (menace, vulnérabilité).
- **Détection d'intrusions :** analyse des événements ayant lieu dans un système dans le but de trouver en temps réel, en quasi temps réel ou en différé des tentatives d'accès non autorisé et aussitôt de notifier ces tentatives à l'administrateur du système.
- **Faux-positif :** détection en absence d'attaque, alarme générée par un IDS pour un événement légal.
- **Faux-négatif :** absence de détection en présence d'attaque, non génération d'alarme par un IDS pou un événement illégal.

2.2 L'objectif de la sécurité

Le système d'information est généralement défini par l'ensemble des données et des ressources matérielles et logicielles de l'entreprise permettant de les stocker ou de les faire circuler. Le système d'information représente un patrimoine essentiel de l'entreprise, qu'il convient de protéger. La sécurité informatique, d'une manière générale, consiste à assurer que les ressources matérielles ou logicielles d'une organisation sont uniquement utilisées dans le cadre prévu.

Il existe six services de sécurité génériques :

- **l'a*uthentification* :** L'authentification est un mécanisme permettant d'identifier des personnes ou des entités et de certifier leur identité.
- **la *confidentialité* :** protège contre la révélation d'informations à des entités non autorisées.
- **l'*intégrité des données* :** protège contre la modification, la destruction et la substitution de données sans autorisation.
- **le *non répudiation* :** au cours d'une communication, il peut arriver que l'un des deux interlocuteurs nie avoir participé à l'échange d'information. Ce service permet de protéger l'autre interlocuteur.
- **la *disponibilité* :** protège les utilisateurs légitimes contre une indisponibilité des ressources, services et informations.

3 Les attaques

Une attaque est l'exploitation d'une faille d'un système informatique connecté à un réseau.

Etant donné le nombre important d'attaques possibles, nous allons d'abord commencer par les différentes étapes d'une attaque, classer puis nous en présenterons quelques-unes.

3.1 Types d'attaques

Les attaques peuvent à première vue être classées en 2 grandes catégories :

- les attaques passives : consistent à écouter sans modifier les données ou le fonctionnement du réseau. Elles sont généralement indétectables mais une prévention est possible.
- les attaques actives : consistent à modifier des données ou des messages, à s'introduire dans des équipements réseau ou à perturber le bon fonctionnement de ce réseau. Noter qu'une attaque active peut être exécutée sans la capacité d'écoute. De plus, il n'y a généralement pas de prévention possible pour ces attaques, bien qu'elles soient détectables (permettant ainsi une réponse adéquate).

3.2 Les différentes étapes d'une attaque

La plupart des attaques, de la plus simple à la plus complexe fonctionnent suivant le même schéma :

- Identification de la cible
- Le scanning
- L'exploitation
- La progression

4 Domaines d'application de la sécurité

- Sécurité physique
- Sécurité de l'exploitation
- Sécurité logique
- Sécurité applicative
- Sécurité des télécommunications.

5 Aspects techniques de la sécurité informatique

Les problèmes techniques actuels de sécurité informatique peuvent, au moins provisoirement, être classés en deux grandes catégories :

- ceux qui concernent la sécurité de l'ordinateur proprement dit, serveur ou poste de travail, de son système d'exploitation et des données qu'il abrite;
- ceux qui découlent directement ou indirectement de l'essor des réseaux, qui multiplie la quantité et la gravité des menaces. Si les problèmes de la première catégorie citée ici existent depuis la naissance de l'informatique, il est clair que l'essor des réseaux, puis de l'Internet, en a démultiplié l'impact potentiel en permettant leur combinaison avec ceux de la seconde catégorie.

6 Mécanismes de sécurité

La sécurité vise à garantir la confidentialité, l'intégrité et la disponibilité des services. Il faut mettre en place des mécanismes pour s'assurer que seules les personnes autorisées ont accès à l'information et que le service est rendu correctement. Parmi ces mécanismes, on peut citer :

6.1 Cryptage

Cryptographie est une science mathématique dans laquelle on fait les études des méthodes permettant de transmettre des données de manière confidentielle.

Afin de protéger un message, on lui applique une transformation qui le rend incompréhensible ; c'est ce qu'on appelle le chiffrement, qui, à partir d'un texte clair, donne un texte chiffré ou cryptogramme. Inversement, le déchiffrement est l'action qui permet de reconstruire le texte en clair à partir du texte chiffré. Dans la cryptographie moderne, les transformations en question sont des fonctions mathématiques, appelées algorithmes cryptographiques, qui dépendent d'un paramètre appelé clef.

Dans les réseaux, pour contrer les vols d'informations dans la voie de transmission, on utilise les techniques de cryptographie pour chiffrer et déchiffrer les messages transmis. Il existe à l'heure actuelle deux grands principes de cryptage : le cryptage symétrique basé sur l'utilisation d'une clef privée et le cryptage asymétrique qui repose sur un codage à deux clefs, une privée et l'autre publique. La Figure I.1 montre le fonctionnement de chiffrement.

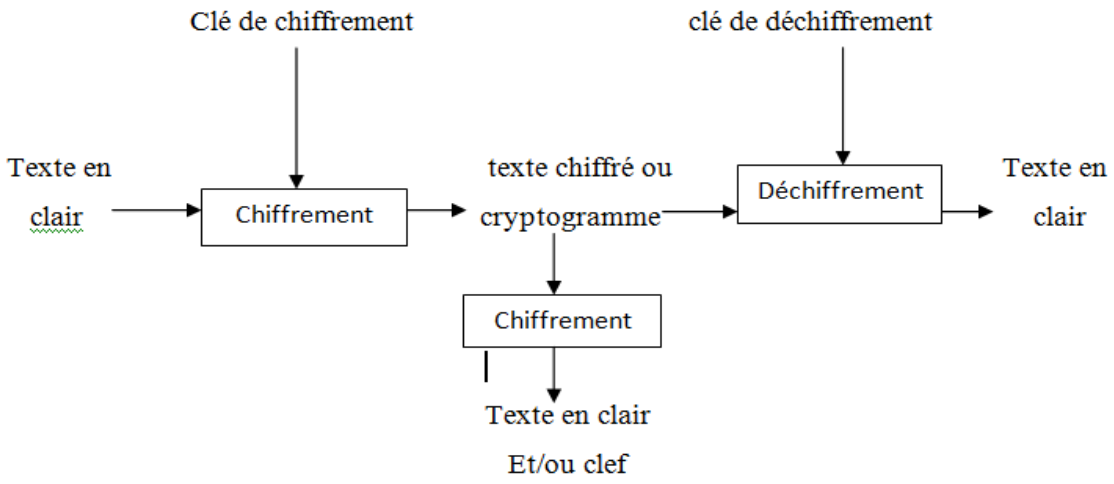

Figure 1.1 Fonctionnement de chiffrement

6.2 Pare-feu

Un pare-feu (firewall) est une solution matérielle ou logicielle mise en place au sein de l'infrastructure du réseau afin de filtrer l'accès à des ressources réseau définies. Il ne laisse entrer que les utilisateurs autorisés, disposant d'une clef ou d'un badge, et crée une couche protectrice entre le réseau et le monde extérieur. Il est doté de filtres intégrés qui peuvent empêcher des documents non autorisés ou potentiellement dangereux d'accéder au système. Il enregistre également les tentatives d'intrusions dans un journal transmis aux administrateurs du réseau.

Il permet également de contrôler l'accès aux applications et d'empêcher le détournement d'usage.

Le pare-feu permet à laisser passer tout ou partie des paquets qu'ils sont autorisés, et à bloquer et journaliser les échanges qui sont interdits.

Le pare-feu est un IDS, mais il détecte seulement les attaques de l'extérieur. Pour Intranet, les pare-feu sont nécessaires, mais pas suffisants, pour commencer à implémenter une politique de sécurité.

Certains pare-feu laissent uniquement passer le courrier électronique. De cette manière, ils interdisent toute autre attaque qu'une attaque basée sur le service de courrier. D'autres pare-feu, moins strictes, bloquent uniquement les services reconnus comme étant des services dangereux.Généralement, les pare-feu sont configurés pour protéger contre les accès non authentifiés du réseau externe. La figure I.2 schématise le fonctionnement d'un pare-feu.

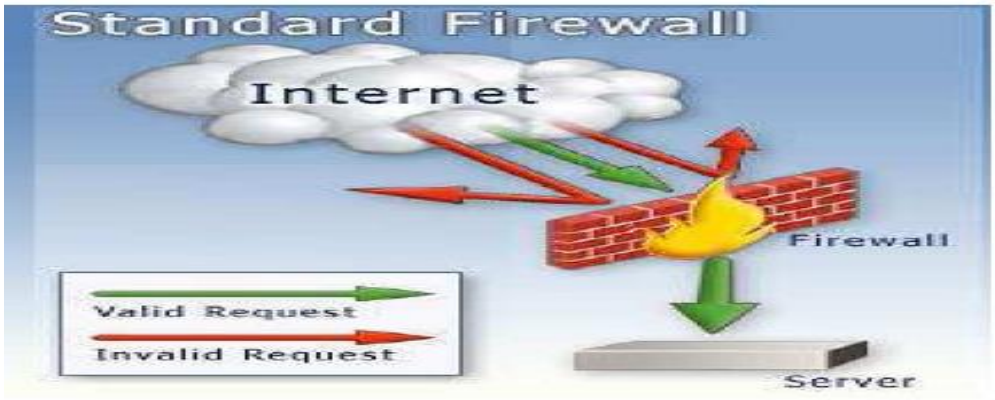

Figure 1.2 Fonctionnement d'un pare-feu

6.3 Antivirus

Un antivirus est un logiciel qui protège une machine contre les virus. Les antivirus se fondent sur des fichiers de signatures et comparent alors les signatures génétiques du virus aux codes à vérifier. Certains programmes appliquent également la méthode heuristique tendant à découvrir un code malveillant par son comportement.

Les antivirus peuvent scanner le contenu d'un disque dur, mais également la mémoire de l'ordinateur. Pour les plus modernes, ils agissent en amont de la machine en scrutant les échanges de fichiers avec l'extérieur, aussi bien en flux montant que descendant. Ainsi, les courriers sont examinés, mais aussi les fichiers copiés sur ou à partir de supports amovibles tels que cédéroms, disquettes, connexions réseau,...

Aujourd'hui, il y a beaucoup d'antivirus comme Norton Antivirus, McAfee Antivirus, Kaspersky Antivirus...

7 Détection et prévention d'intrusions

Un système de détection d'intrusions (IDS) permet de surveiller constamment le réseau. Il analyse les flux de paquets de données transitant par le réseau, à la recherche d'activités non autorisées (telles que les attaques de pirates) et permet aux utilisateurs de traiter les failles de sécurité avant que les systèmes ne soient compromis.

Aujourd'hui, les systèmes IDS évoluent vers ce que l'on appelle des systèmes de, prévention d'intrusions (IPS) qui en complément de la détection apporte une protection active. Un système

IPS peut ainsi décider, suite à des remontées d'alertes, de fermer des ports et de rejeter des paquets en fonctions du paramétrage qui en a été fait.

8 Conclusion

Dans ce chapitre nous avons présenté un aperçu général des services de sécurité, description de quelques attaques ainsi que les mécanismes de sécurité, pour pouvoir la une petite introduction sur la cryptographie dans le chapitre suivant.

CHAITRE 2

INTRODUCTION A LA CRYPTOLOGIE

1 Introduction

Dans ce chapitre, nous allons fait Un survol de la cryptologie tel que définition et mots clés, la différence entre chiffrement et codage, l'objectifs de la cryptographie, histoire de la cryptologie, quelques algorithmes classiques (Chiffrement de César, Le chiffre de Hill…), et la machine ENIGMA.

2 Définition et mots clés

La cryptographie utilise des concepts issus de nombreux domaines (informatique, mathématique, électrique).Toutes fois, les techniques évoluent et trouvent aujourd'hui régulièrement racine dans d'autres branches (biologie, physique, etc.)

2.1 Vocabulaire de base

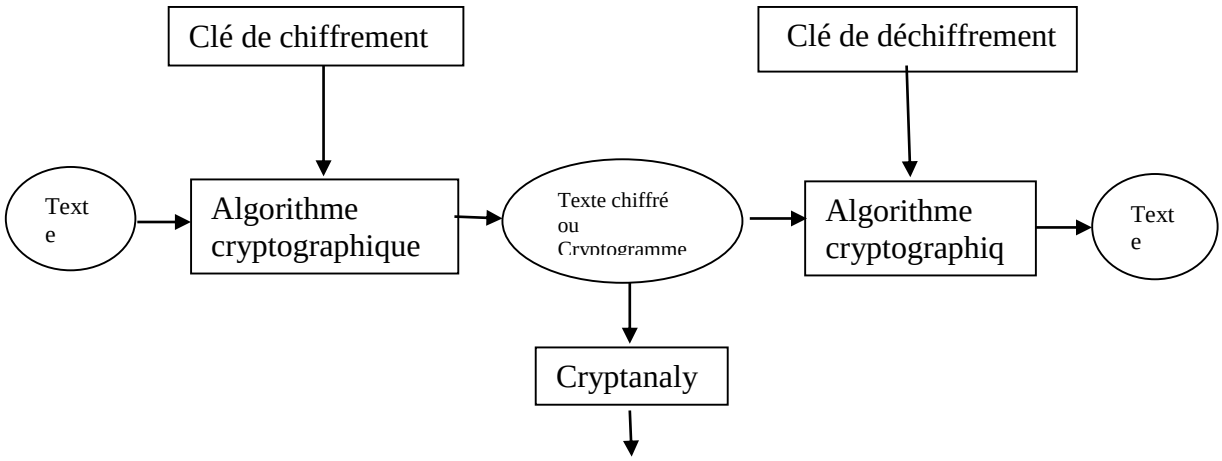

Figure 2.1 : Protocole de chiffrement

- **Cryptologie** : Il s'agit d'une science mathématique comportant deux branches : englobe à la fois la cryptographie et la cryptanalyse.
- **Cryptographie :** (du grec κρυπτος: caché et γραφειν: écrire) est l'étude des méthodes des principes et techniques mathématiques donnant la possibilité d'envoyer des données de manière confidentielle sur un support donné.

Elle aussi définie comme étant " la science du secret '' [01].

- **Chiffrement :** Le chiffrement consiste à transformer une donnée (texte, message, ...) afin de la rendre incompréhensible par une personne autre que celui qui a créé le message et celui

qui en est le destinataire. La fonction permettant de retrouver le texte clair à partir du texte chiffré porte le nom de déchiffrement.

- **Le déchiffrement :** opération mathématique, effectuée à l'aide de la clé privée, qui consiste à décoder le message crypté à l'aide de la clé publique associée. Cette opération ne peut être effectuée que par le détenteur de la clé privée.
- **Texte chiffré :** Appelé également cryptogramme, le texte chiffré est le résultat de l'application d'un chiffrement à un texte clair.
- **Cryptanalyse :** étudie la sécurité des procédés de chiffrement utilisés en cryptographie. Elle consiste alors à casser des fonctions cryptographiques existantes, c'est-à-dire à démontrer leur sécurité. La cryptanalyse mêle une intéressante combinaison de raisonnement analytique, d'application d'outils mathématiques, de découverte de redondances, de patience, de détermination et de chance.
- **Cryptosystème :** Il est défini comme l'ensemble des clés possibles (espace de clés), des textes clairs et chiffrés possibles associés à un algorithme donné.
- **Clef :** Il s'agit du paramètre impliqué et autorisant des opérations de chiffrement et/ou déchiffrement. Dans le cas d'un algorithme symétrique, la clef est identique lors des deux opérations. Dans le cas d'algorithmes asymétriques, elle diffère pour les deux opérations.
- **Le code :** ensemble de procédés et ensemble de symboles (lettres, nombres, signes, etc.) employés pour remplacer les mots du message à coder.
- **Le protocole :** un protocole est une série de pas, impliquant deux parties ou plus conçues pour accomplir une tâche. C'est une définition importante. "Une série de pas" signifie que le protocole a un ordre, du début à la fin. Chaque pas doit être exécuté à son tour et aucun pas ne peut être entamé avant que le pas précédent ne soit fini. "L'implication de deux parties ou plus" signifie qu'on exige au moins que deux personnes achèvent le protocole ; une personne seule ne fait pas de protocole. Une personne seule peut exécuter une série de pas pour accomplir une tâche, mais ce n'est pas un protocole. Finalement, "conçu pour accomplir une tâche" signifie que le protocole doit réaliser quelque chose. Quelque chose qui ressemble à un protocole, mais qui n'accomplit aucune tâche n'est pas un protocole. [02]
- **Le cryptographe :** a comme travail de fournir des outils pour éliminer les risques, afin de rendre les échanges d'information confidentiels, infalsifiables, authentiques et inaltérables.

L'information est chaque jour échangée d'un point à un autre et se trouve susceptible d'être lue, copiée, supprimée, altérée ou falsifiée.

- **La stéganographie :** (du grec στεγανος: couvert et γραφειν: écrire) est une branche particulière de la cryptographie qui consiste non pas à rendre le message inintelligible, mais à le camoufler dans un support (texte, image, séquence vidéo,…etc.) de manière à masquer sa présence. [11]
- **La signature :** partie chiffrée d'un message générée dans le but d'authentifier le message (intégrité) et l'expéditeur (identité).
- **Le certificat :** document informatique délivré sous la forme d'un fichier normalisé (X509) qui permet la signature, la vérification et le cryptage de messages. Le contenu du certificat lie les données, signées ou cryptées, au détenteur du certificat : l'utilisateur autorisé. Un certificat se compose d'une clé privée, d'une clé publique et d'une signature émanant de l'autorité de certification.
- **La clé privée :** partie du certificat, sous la responsabilité de l'utilisateur autorisé. Elle est destinée à la signature et au décryptage des messages. C'est la partie sensible de la clé qui doit rester secrète. La clé privée est utilisée pour déchiffrer les messages reçus et chiffrer la signature. [11]
- **La clé publique :** partie du certificat qui sera diffusée. Elle est destinée à la vérification de signature et au cryptage des messages. C'est la partie qui est transmise aux interlocuteurs ou placée sur des serveurs de clés. La clé publique est utilisée pour chiffrer les messages à envoyer et déchiffrer la signature des messages reçus. [11]
- L'autorité de certification : l'organisme qui assure la délivrance et le renouvellement des certificats, la diffusion des listes de révocation et des clés publiques.
- Le condensé : le condensé est le résultat de la transformation numérique d'une information en une suite de caractères dont le contenu garantit l'intégrité du message transféré. Cette intégrité est vérifiée à la réception du message, lors de la vérification de signature, par comparaison du condensé reçu et du condensé calculé. Le caractère mathématiquement irréversible de la transformation garantit que le contenu du message n'a pas été falsifié.

2.2 Les objectifs de la cryptographie [03]

Vie privée ou confidentialité	Tenir l'information secrète pour tous sauf pour ceux qui sont autorisés à la voir.
Intégrité de données	l'assurance que l'information n'a pas été changée par des moyens non autorises ou inconnus
Authentification d'entité ou identification	Confirmation de l'identité d'une entité (par exemple, une personne, un terminal d'ordinateur, une carte de crédit, etc.).
Authentification de message	Corroboration de la source d'information ; aussi connue comme authentification d'origine de données
Signature	Moyen de lier l'information a une entité
Autorisation	Transfert, à une autre entité, de sanction officielle pour faire quelque chose ou de l'être
Validation	De fournir opportunité d'autorisation d'employer ou manipuler information ou ressources.
Contrôle d'accès	Limitation d'accès a ressources de favorise entités
Certification	Endossement d'information par une entité crue
Horodatage	Enregistrement du temps de création ou existence d'information
Témoignage	Vérification de la création ou de l'existence de l'information par une entité autre que le créateur
Non reniement	Prévention du démenti d'obligations précédentes ou actions
Révocation	Rétraction de certification ou autorisation
Propriété	Moyen de fournir une entité au droit légal d'employer ou transférer une ressource a d'autres

Table 1.1 :l'objectif de la cryptographie

2.3 La différence entre chiffrement et codage

Les opérations de chiffrement et de codage font partie de la théorie de l'information. La différence essentielle réside dans la volonté de protéger les informations et d'empêcher des tierces personnes d'accéder aux données dans le cas du chiffrement. Le codage consiste à transformer de l'information (des données) vers un ensemble de mots. Chacun de ces mots est constitué de symboles. La compression est un codage : on transforme les données vers un ensemble de mots

adéquats destinés à réduire la taille mais il n'y a pas de volonté de dissimuler (bien que cela se fasse implicitement en rendant plus difficile d'accès le contenu).

Le "code" dans le sens cryptographique du terme travaille au niveau de la sémantique (les mots ou les phrases). Par exemple, un code pourra remplacer le mot "avion" par un numéro. Le chiffrement travaille sur des composantes plus élémentaires du message, les lettres ou les bits, sans s'intéresser à la signification du contenu. Un code nécessite une table de conversion, aussi appelée "dictionnaire" (codebook en anglais). Ceci étant, "code" et "chiffrement" sont souvent employés de manière synonyme malgré cette différence. On peut aussi considérer que le chiffrement doit résister à un adversaire "intelligent" qui peut attaquer de plusieurs manières alors que le codage est destiné à une transmission sur un canal qui peut être potentiellement bruité. Ce bruit est un phénomène aléatoire qui n'a pas d'"intelligence" intrinsèque mais peut toutefois être décrit mathématiquement.

2.4 Critères d’évaluation des algorithmes de chiffrement

Les algorithmes de chiffrement peuvent être évalués à l'aide de plusieurs mesures, et on peut retenir essentiellement cinq critères principaux qui sont :

- **Le niveau de sécurité :** C'est d'habitude difficile de l'évaluer quantitativement. Souvent on le donne en termes du nombre d'opérations nécessaires (en utilisant les meilleures méthodes actuellement connues) pour défaire l'objectif destiné. Typiquement le niveau de sécurité est défini par un maximum de travail nécessaire afin de défaire l'objectif. C'est parfois appelé le facteur de travail
- **La fonctionnalité :** Des primitives devront être combinées pour répondre à divers objectifs de sécurité de l'information. Quelles primitives sont les plus efficaces pour un objectif donné, elles seront déterminées par les propriétés de base de ces primitives.
- **La performance :** Cela se réfère à l'efficacité d'une primitive dans un mode de fonctionnement particulier. (Par exemple, un algorithme de chiffrement peut être évalué par le nombre de bits par seconde qu'il peut chiffrer.)
- **La facilité d'implémentation** : Cela se réfère à la difficulté de réaliser la primitive dans une instanciation pratique. Cela pourrait inclure la complexité de mettre en œuvre la primitive aussi bien dans l'environnement matériel que logiciel.

3 Histoire de la cryptologie

Les origines de la cryptologie remontent à l'antiquité ; son utilisation était alors très rudimentaire. Cependant la cryptologie a rapidement révélé sa double nature, celle de l'étude combinée de deux arts opposés mais complémentaires : la cryptographie, qui consiste à protéger des informations, et la cryptanalyse, qui est l'art de les retrouver ou de les exploiter. Le célèbre ouvrage de Kahn, « The Codebreakers » [4], retrace la longue histoire de la cryptologie à travers les âges.

La cryptologie a été utilisée initialement dans le but de protéger les communications sensibles. Elle était alors considérée davantage comme un art que comme une science ; ainsi les procédés des Spartiates ou de César assuraient la confidentialité. Ce n'est que plus tard qu'ont été définis d'autres besoins, comme l'authentification. le chemin si dessue présente la structure de l'histoire de la cryptologie.

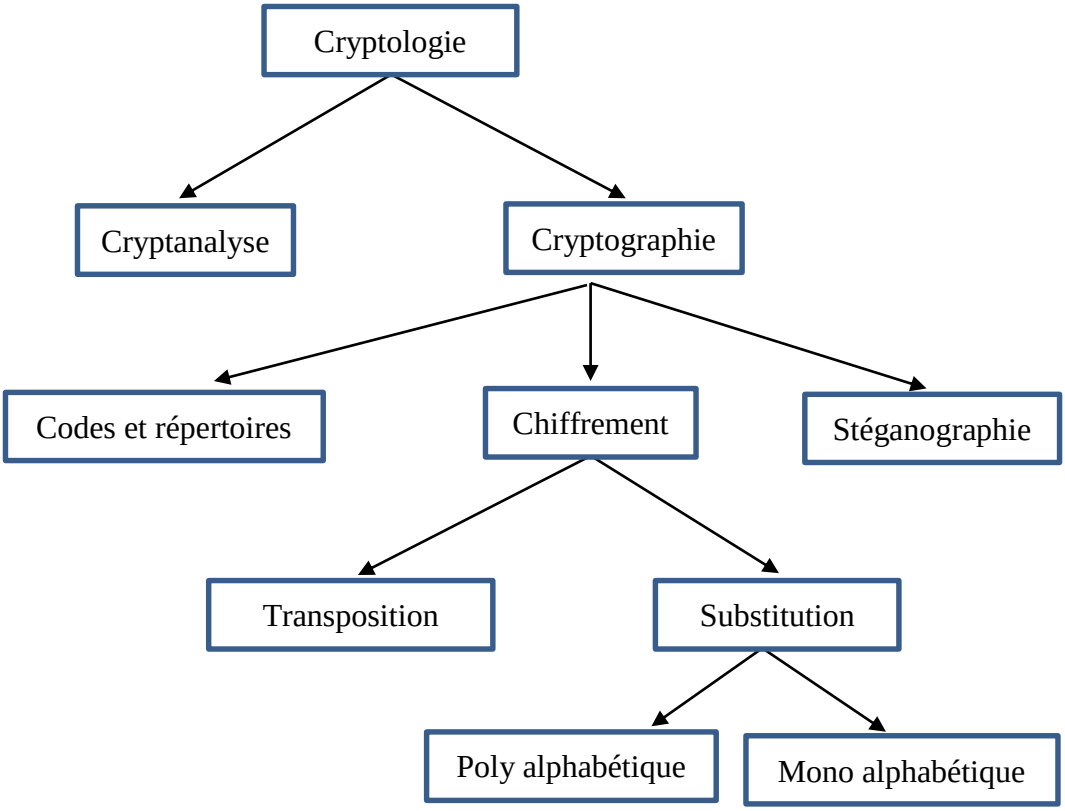

Figure 2.2 : histoire de la cryptologie

4 Techniques classiques de chiffrement

4.1 Historique de cryptographie

Ancien âge :

1900 av. J-C	Un scribe égyptien utilise des hiéroglyphes afin d'éterniser la civilisation des Pharaons.
487 av. J-C	Des grecs utilisent "skytale", un grand bâton de chiffrage auquel on enroulait une longue et mince bande de cuir sur laquelle se trouvaient des informations.
50 av. J-C	Julius Caesar utilise une simple substitution dans l'alphabet pour les communications gouvernementales.
Entre 0 et 400	Le Kama Sutra de Vatsayana liste la cryptographie comme étant le 44ème et 45ème art, dans une liste totalisant 64 arts (yogas).
200	Le papyrus de Leyde utilise un algorithme de chiffrement pour cacher les parties importantes de certaines recettes (magiques).
1918	Gilbert Vernam, mathématicien américain, invente le one-time cipher (one-time pad), encore aujourd'hui l'algorithme de chiffrement le plus sécuritaire, mais impraticable.
1919	L'American Black Chamber, un organisme installé à New York, a comme rôle de déchiffrer les codes venant du Japon.

Nouvel âge :

1923	L'allemand Dr Arthur Scherbius met au point une machine nommée Enigma qui sert à encoder des messages. Le prix très élevé de la machine en fait un échec.
1925	La marine de guerre allemande reprend le projet d' Enigma en le confiant au Chiffrier stelle, le service de chiffrement de l'armée allemande.

1937	Enigma M3 est adopté par la Wehrmacht, l'armée allemande.
1939	Début de la seconde guerre mondiale, où 12 000 scientifiques et mathématiciens anglais, polonais et français travaillent à solutionner Enigma et les milliers de messages chiffrés. L'équipe d'Alan Turing réussit finalement à le résoudre.
1970	IBM développe Lucifer
1976	IBM publie un algorithme basé sur Lucifer. Il devient le DES (Data Encryption Standard).
1976	Whitfield Diffie et Martin Hellman introduisent l'idée d'un système à clé publique.
1978	L'algorithme de chiffrement à clé publique RSA est publié par Ronald L. Rivest, Adi Shamir et Leonard M. Adleman.
1984	L'algorithme ROT13 est utilisé dans le USENET et devient le premier exemple de l'utilisation des clés publiques.
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 anonymement dans une liste de distribution de Cypherpunks.
1990	Première publication des résultats expérimentaux de la cryptographie quantique par Charles H. Bennett et Gilles Brassard.
1991	Phil Zimmermann rend disponible sa première version de PGP.
1992	L'IDEA est inventé en Suisse par Xuejia Lai et James Massey. MD5 est développé par Ronald L. Rivest.
1993	Bruce Schneier conçoit Blowfish. Don Coppersmith crée SEAL.

1994	Ron Rivest, déjà auteur de RC2 et RC4, publie RC5. Un standard régissant les signatures numériques voit le jour : le DSA (DSS).
1995	Le NIST développe le Secure Hash Algorithm (SHA)
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.
2000	**Rijndael devient l'AES, le standard du chiffrement avancé.**

Tableau 2.1 : Historique de cryptographie

4.2 La cryptographie classique : [11]

Elle décrit la période avant les ordinateurs. Elle traite des systèmes reposant sur les lettres et les caractères d'une langue naturelle (allemand, anglais, français, etc.). Les principaux outils utilisés remplacent des caractères par des autres et les transposent dans des ordres différents. Les meilleurs systèmes de cette classe d'algorithmes répètent ces deux opérations de base plusieurs fois. Cela suppose que les procédures (de chiffrement ou déchiffrement) soient gardées secrètes ; et sans cela comme nous l'avons déjà dit le système est complètement inefficace (n'importe qui peut déchiffrer le message codé). On appelle généralement cette classe de méthodes : le chiffrement à usage restreint.

Le chiffrement classique : il existe des centaines de façon de chiffrer des données représentées par l'alphabet classique, tout en gardant les opérations réalisées secrètes. Ici on va présenter en premier toutes ces méthodes, mais plutôt les concepts mathématiques (connus depuis très longtemps) qui sont à la source de celles-ci. On va ainsi voir que finalement il n'y en a pas tant que l'on pouvait le penser, et surtout qu'elles sont extrêmement simples.

a) Substitution

- Substitution simple ou substitution monoalphabétique : chaque caractère du texte en clair est remplacé par un caractère correspondant dans le texte chiffré.
- Substitution homophonique : comme pour le principe précédent, sauf qu'à un caractère du texte en clair on fait correspondre plusieurs caractères dans le texte chiffré.
- Substitution polyalphabétique : le principe consiste à remplacer chaque lettre du message en clair par une nouvelle lettre prise dans un ou plusieurs alphabets aléatoires associés.

- Substitution par polygrammes : les caractères du texte en clair sont chiffrés par blocs.

b) Transposition :

Avec le principe de la transposition toutes les lettres du message sont présentes, mais dans un ordre différent. Il utilise le principe mathématique des permutations. Plusieurs types différents de transpositions existent :

- Transposition simple par colonnes
- Transposition complexe par colonnes
- Transposition par carré polybiqu

5 Quelques algorithmes classiques

5.1 Les scytales des Spartiates

Un procédé de chiffrement avait été imaginé par les Spartiates, dans le souci de protéger la confidentialité de leurs informations. Le principe consistait à enrouler une lanière de cuir ou de papyrus autour d'un bâton de bois de diamètre fixé, puis à écrire le message en travers des spires, c'est-à-dire parallèlement à l'axe du bâton. Une fois déroulée, la lanière contenait donc un texte illisible, sauf pour le correspondant connaissant le diamètre adéquat. Ce mécanisme porte le nom de permutation, puisqu'il consiste à mélanger les lettres du message, sans modifier ces lettres. Des historiens comme Thucydide ou Plutarque signalent l'utilisation de ce procédé au Vème siècle avant notre ère.

Figure 2.3 : Le Scytale Spartiate

5.2 chiffrement de César [5]

Le chiffrement de César est la méthode de cryptographie la plus ancienne communément admise par l'histoire. Il consiste en une substitution mono-alphabétique : chaque lettre est remplacée ("substitution") par une seule autre ("mono-alphabétique"), selon un certain décalage dans l'alphabet ou de façon arbitraire. D'après Suétone, César avait coutume d'utiliser un décalage de 3 lettres : A devient D, B devient E, C devient F, etc. Il écrivait donc son message normalement, puis

remplaçait chaque lettre par celle qui lui correspondait Dans les formules ci-dessous, p est l'indice de la lettre de l'alphabet, k est le décalage.

Pour le chiffrement, on aura la formule : C = E(p) = (p + k) mod 26

Pour le déchiffrement, il viendra : p = D(C) = (C − k) mod 26

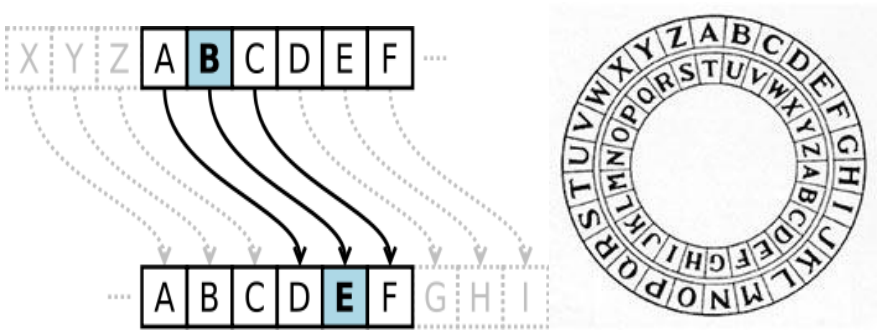

Figure 2.4 : Chiffrement de César

5.3 Le chiffre de Hill [6]

Le chiffre publié en 1929 par Lester S. Hill (1891-1961) est un chiffre polygraphique), c'est-à-dire qu'on ne (dé) chiffre pas les lettres les unes après les autres, mais par paquets. Nous étudierons un seul cas qui est le cas de groupement bigraphique du chiffre de Hill, puisque nous grouperons les lettres deux par deux, mais on peut imaginer des paquets plus grands, par exemple des paquets de trois lettres.

Chiffrement

Les lettres sont d'abord remplacées par leur rang dans l'alphabet. Les lettres et P_{k+1} du texte clair seront chiffrées C_k et C_{k+1} avec la formule ci-dessous:

$$\begin{pmatrix} C_k \\ C_{k+1} \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} Pk \\ P_{k+1} \end{pmatrix} mod\ (26)$$

Ce qui signifie, pour fixer les idées, que les deux premières lettres du message clair (P_1 et P_2) seront chiffrées (C_1 et C_2) selon les deux équations suivantes:

$C_1 \equiv aP_1 + bP_2\ (mod\ 26)$

$C_2 \equiv cP_2 + dP_2\ (mod\ 26)$

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.

$$\begin{pmatrix} p1 \\ P2 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} \begin{pmatrix} C1 \\ C2 \end{pmatrix}$$

Cette matrice doit être l'inverse de$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ matrice de chiffrement (modulo 26). Ordinairement l'inverse de la matrice est :

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ab - bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

5.4 Chiffrement de VIGENERE (1523-1596)

Le Chiffre de Vigenère est un système de chiffrement, élaboré par Blaise de Vigenère (1523-1596), diplomate français du XVIe siècle. C'est un système de substitution polyalphabétique. Cela signifie qu'il permet de remplacer une lettre par une autre qui n'est pas toujours la même, contrairement aux chiffres vu précédemment qui se contentaient d'utiliser la même lettre de substitution. C'est donc un système relativement plus « solide ».

L'outil indispensable du chiffrement de Vigenère est : « La table de Vigenère » ou « carré de Vigenère ». On l'obtient en écrivant 26 fois l'alphabet, et en décalant chaque ligne d'une lettre. Pour la 1er ligne : ABCDE ... XYZ

Pour la 2eme : BCDEF... YZA

La 3eme : CDEFG...ZAB …Etc.

Chiffrement

Pour chaque lettre en clair, on sélectionne la colonne correspondante et pour une lettre de la clé on sélectionne la ligne adéquate, puis au croisement de la ligne et de la colonne on trouve la lettre codée. La lettre de la clé est à prendre dans l'ordre dans laquelle elle se présente et on répète la clé en boucle autant que nécessaire.

Le tableau de Vigenère est une matrice de 26 x 26 éléments

	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 2.5 : La table de Vigenère

6 chiffrement mécanisé (La machine ENIGMA)

Ces méthodes de chiffrement utilisent principalement des machines capables de chiffrer les messages clairs instantanément et se caractérisent par la rapidité d'exécution par rapport aux méthodes manuelles.

L'histoire de la machine Enigma commence en 1919, quand un ingénieur hollandais, Hugo Alexander Koch, dépose un brevet de machine à chiffrer électromécanique. Ses idées sont reprises par le Dr Arthur Scherbius, qui crée à Berlin une société destinée à fabriquer et à commercialiser une machine à crypter civile : l'Enigma. Cette société fait un fiasco vu les coûts financiers de fabrication donc de

commercialisation, mais la machine Enigma a attiré l'attention des militaires.

Le fonctionnement d'Enigma. Le codage effectué par la machine Enigma est à la fois simple et astucieux. 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 :

- Le tableau de connexions : il permet d'échanger des paires de l'alphabet, deux à deux, au moyen de fiches. Il y a 6 fiches qui permettent donc d'échanger 12 lettres. Un tableau de connexions est donc une permutation très particulière où on a échangé au plus 6 paires. Par exemple, dans le tableau suivant (avec simplement 6 lettres), on a échangé A et C, D et F, tandis que B et E restent invariants.
- les rotors : un rotor est également une permutation, mais cette fois quelconque. À chaque lettre en entrée correspond une autre lettre.

On peut composer les rotors, c'est-à-dire les mettre les uns à la suite des autres. La machine Enigma disposera, au gré de ses évolutions successives, de 3 à 6 rotors. Parmi ces rotors, seuls 3 sont utilisés pour le codage, et on a le choix de les placer dans l'ordre que l'on souhaite (ce qui constituera une partie de la clé). Surtout, les rotors sont cylindriques, et ils peuvent tourner autour de leur axe. Ainsi, à chaque fois qu'on a tapé une lettre, le premier rotor tourne d'un cran, et la permutation qu'il engendre est changée. Observons ce changement sur la figure suivante : le rotor transforme initialement D en B. Lorsqu'il tourne d'un cran, cette liaison électrique D--->B se retrouve remontée en C--->A.

Chaque rotor possède donc 26 positions. A chaque fois qu'une lettre est tapée, le premier rotor tourne d'un cran. Après 26 lettres, il est revenu à sa position initiale, et le second rotor tourne alors d'un cran. On recommence à tourner le premier rotor, et ainsi de suite... Quand le second rotor a retrouvé sa position initiale, c'est le troisième rotor qui tourne d'un cran.

Le réflecteur : Au bout des 3 rotors se situe une dernière permutation qui permet de revenir en arrière. On permute une dernière fois les lettres 2 par 2, et on le fait retraverser les rotors, et le tableau de connexion.

Résumons sur la machine simplifiée suivante (6 lettres, 2 rotors) comment est codée la lettre A :

- on traverse le tableau de connexions : on obtient C
- on traverse les 2 rotors : on obtient successivement A et F

- on traverse le réflecteur où on obtient E, puis on renvoie dans les rotors pour obtenir F, A et finalement C après le tableau de connexions.

Remarquons que si on avait tapé C, le courant aurait circulé dans l'autre sens et on aurait obtenu A.

Nombre de clés possibles. Il y a trois éléments à connaître pour pouvoir coder un message avec la machine Enigma :

- la position des 6 fiches du tableau de connexion : d'abord, il faut choisir 12 lettres parmi 26. C'est donc le nombre de combinaisons de 12 parmi 26, soit 26!/(12!14!). Maintenant, il faut choisir 6 paires de lettres parmi 12, soit 12 ! /6!, et comme la paire (A,D) donne la même connexion que la paire (B, A), il faut encore diviser par 26. On trouve enfin 100 391 791 500.
- l'ordre des rotors : il y a autant d'ordres que de façons d'ordonner 3 éléments : 3!=6.
- la position initiale des rotors : chaque rotor ayant 26 éléments, il y a 26*26*26=17 576 choix.

On multiplie tout cela, et on obtient plus de 1016 possibilités, ce qui est énorme pour l'époque !

Il est important de remarquer que les permutations employées dans les rotors et les réflecteurs ne peuvent pas être considérées comme faisant partie du secret. En effet, toutes les machines utilisent les mêmes, et il suffit donc d'en avoir une à disposition. Les Anglais, par exemple, en ont récupéré une pendant la guerre dans un sous-marin coulé. Ceci est une illustration d'un principe général en cryptographie, dit principe de Kerckhoffs, qui veut que tout le secret doit résider dans la clé secrète de chiffrement et de déchiffrement, et pas dans une quelconque confidentialité de l'algorithme (ici de la machine) qui ne peut être raisonnablement garantie.

7 Conclusion

L’origine de la cryptographie remonte sans doute aux origines de l’homme, dès que ceux-ci apprirent à communiquer. Alors, ils durent trouver des moyens d’assurer la confidentialité d’une partie de leurs communications.

Dans ce chapitre, nous avons présenté l’histoire decryptograhie ainsi quelque algorithme classique.

Dans le chapitre suivent nous avons présenté la cryptographie moderne.

CHAPITRE 3

CRYPTOGRAPHIE MODERNE

1 Introduction

Ce chapitre présente une vue générale sur la cryptographie moderne à travers L'étude de la cryptographie à clé secrète et à clé publique et leur algorithmes, ainsi La signature numérique, Les fonctions de hachage et Code d'authentification de message MAC.

2 Techniques de cryptographie

Pour assurer les objectifs de la cryptographie nous pouvons utiliser des algorithmes basés sur des clés. Ces algorithmes sont définis par plusieurs types de cryptographie ou crypto systèmes :

2.1 La cryptographie à clé secrète (symétrique)

La cryptographie à clé secrète utilise la même clé pour les processus de chiffrement et de déchiffrement ; cette clé est le plus souvent appelée "secrète" (en opposition à "privée") car toute la sécurité de l'ensemble est directement liée au fait que cette clé n'est connue que par l'expéditeur et le destinataire.

On retrouve la cryptographie à clé secrète également sous les termes cryptographie asymétrique ou à clé privée. Les termes sont adéquats car c'est en effet la même clé qui est utilisée pour crypter le message par son auteur et le décrypter par son destinataire. C'est l'approche la plus authentique du chiffrement de données et mathématiquement la moins problématique.

Caractéristiques :

- Les clés sont identiques : KE = KD = K
- La clé doit rester secrète,
- Les algorithmes les plus répandus sont le DES, AES, RC4, 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.

- Avantages de la cryptographie à clé secrète (symétrique)
  - Système rapide du chiffrement/déchiffrement
  - Clés relativement courtes (128 ou 256 bits)
  - Bonnes performances et sécurité bien étudié
- Inconvénients de la cryptographie à clé secrète (symétrique)
  - Gestion des clés difficiles (nombreux clés) ;

- Point faible = l'échange de la clé secrète ;
- Dans un réseau de N entités susceptibles de communiquer secrètement il faut distribuer N*(N-1)/2 clés.

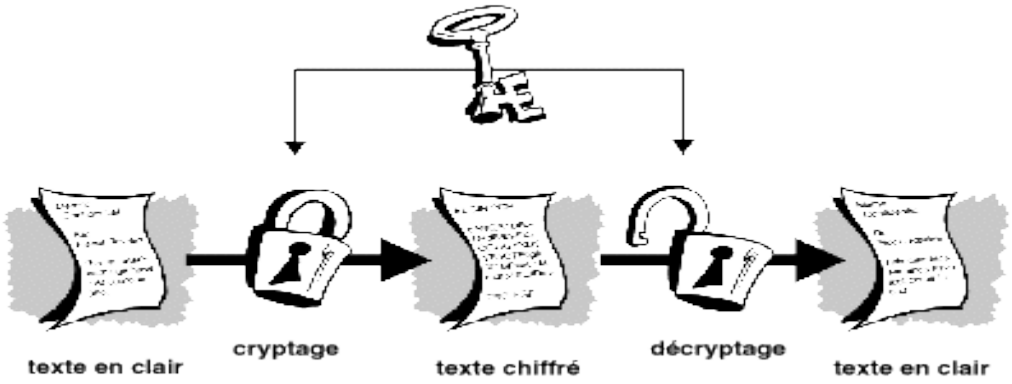

Figure 3.1 La cryptographie à clé secrète

Le cryptage symétrique fonctionne selon deux procédés différents :

2.1.1 Le chiffrement par bloc (block cipher)

La cryptographie symétrique utilise la même clé pour les processus de chiffrement et de déchiffrement ; cette clé est le plus souvent appelée "secrète" (en opposition à "privée") car toute la sécurité de l'ensemble est directement liée au fait que cette clé n'est connue que par l'expéditeur et le destinataire. La cryptographie symétrique est très utilisée et se caractérise par une grande rapidité (opérations simples, chiffrement à la volée) et par des implémentations aussi bien software que hardware ce qui accélère nettement les débits et autorise son utilisation massive.

- L'idée générale du chiffrement par blocs est la suivante :
  - Remplacer les caractères par un code binaire
  - Découper cette chaîne en blocs de longueur donnée
  - Chiffrer un bloc en l'"additionnant" bit par bit à une clef.
  - Déplacer certains bits du bloc.
  - Recommencer éventuellement un certain nombre de fois l'opération 3.
  - Passer au bloc suivant et retourner au point 3 jusqu'à ce que tout le message soit chiffré.

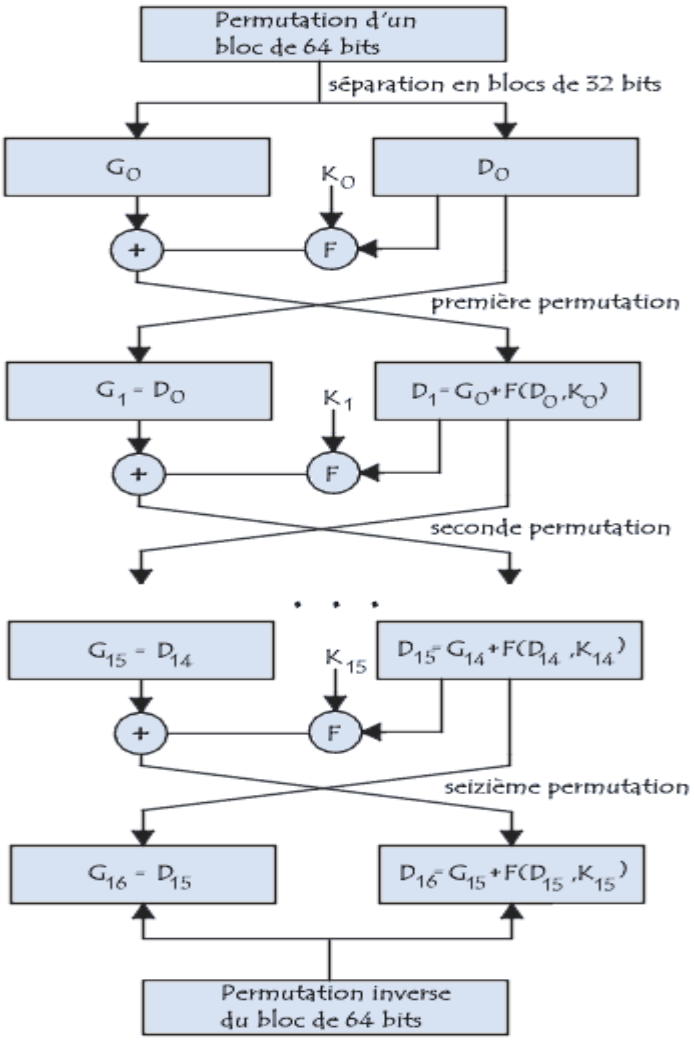

Figure3.2 Le chiffrement par bloc

Un exemple de taille de bloc et de clé utilisés par les algorithmes les plus connus:

- DES : blocs de 64 bits, clé de 56 bits.
- IDEA : blocs de 64 bits, clé de 128 bits.
- AES : blocs de 128 bits, clé de 128 à 256 bits

2.1.2 Chiffrements de flot (Stream Cipher)

Le principe du chiffrement par flot est de chiffrer une suite de caractères un `à la fois, à l'aide d'une transformation qui varie au fur et à mesure du texte. Au contraire, le chiffrement par bloc utilise une transformation fixe, sur des blocs plus gros, typiquement 64 ou 128 bits.

- L'avantage du chiffrement par flot est qu'il
  - très rapides (en matériel et en logiciel)
  - I implémentation matérielle avec peu de portes
  - adaptés aux applications temps rée
  - ne n´nécessite pas de zone tampon (buffer).

- Inconvénients du chiffrement par flot
  - I propagation d'erreurs (prob. de synchronisation)
  - I s´sécurité difficile à atteindre (pas de preuve)

On distingue deux classes de chiffrement par flot :

2.1.2.1 **le chiffrement par flot synchrone**

Avec un chiffrement de flux synchronous, un flux de nombre pseudo-aléatoire est généré indépendamment du texte de base et du texte chiffré. Ce flux est utilisé pour chiffrer le texte de base, ou pour déchiffrer le texte chiffré.

En général, le flux est composé de chiffres binaires, et le chiffrement se fait par une opération XOR entre le keystream et les données de base.

Les deux communicants doivent être exactement à la même étape (synchrone) pour que le déchiffrement se passe correctement. Si des ajouts ou des suppressions sont réalisés pendant le transfert du message, on perd la synchronisation. Cependant, si un bit est altéré pendant le transfert, cela n'affecte pas le reste du message et donc la synchronisation est toujours présente.

2.1.2.2 le chiffrement par flot auto-synchronisant

Les algorithmes de chiffrement de flux appelés self-synchronizing, calcule le keystream à partir du message lui-même.

Le plus simple algorithme de chiffrement self-synchronizing est l'autokey, il utilise le message lui-même comme clé. Lorsqu'un bit est ajouté ou supprimé pendant le transfert des données, l'algorithme l'identifiera puisque la clé est issue du message d'origine lui-même, contrairement aux algorithmes synchrones qui génèrent des keystream aléatoires indépendamment du message d'origine. Les algorithmes de chiffrement de flux self-synchronizing sont beaucoup moins répandus que les synchronous.

2.1.3 Quelque algorithme de La cryptographie à clé secrète (symétrique)

- **DES (Data Encryption Sandard)** [13]

Le D.E.S. (Data Encryption Standard, c'est-à-dire Standard de Chiffrement de Données) est un standard mondial depuis la fin des années 1970.

Le D.E.S. est un crypto système agissant par blocs. Cela signifie que D.E.S. ne chiffre pas les données à la volée quand les caractères arrivent, mais il découpe virtuellement le texte clair en blocs de 64 bits qu'il code séparément, puis qu'il concatène. Un bloc de 64 bits du texte clair entre par un coté de l'algorithme et un bloc de 64 bits de texte chiffré sort de l'autre côté. L'algorithme est assez simple puisqu'il ne combine en fait que des permutations et des substitutions.

C'est un algorithme de chiffrement à clé secrète. La clé sert donc à la fois à chiffrer et à déchiffrer le message. Cette clé a ici une longueur de 64 bits, c'est-à-dire 8 caractères, mais dont seulement 56 bits sont utilisés. On peut donc éventuellement imaginer un programme testant l'intégrité de la clé en exploitant ces bits inutilisés comme bits de contrôle de parité.

L'entière sécurité de l'algorithme repose sur les clés puisque l'algorithme est parfaitement connu de tous. La clé de 64 bits est utilisée pour générer 16 autres clefs de 48 bits chacune qu'on utilisera lors de chacune des 16 itérations du D.E.S.. Ces clés sont les mêmes quel que soit le bloc qu'on code dans un message.

Cet algorithme est relativement facile à réaliser matériellement et certaines puces chiffrent jusqu'à 1 Go de données par seconde. Pour les industriels, c'est un point important notamment face à des algorithmes asymétriques, plus lents, tels que l'algorithme R.S.A.

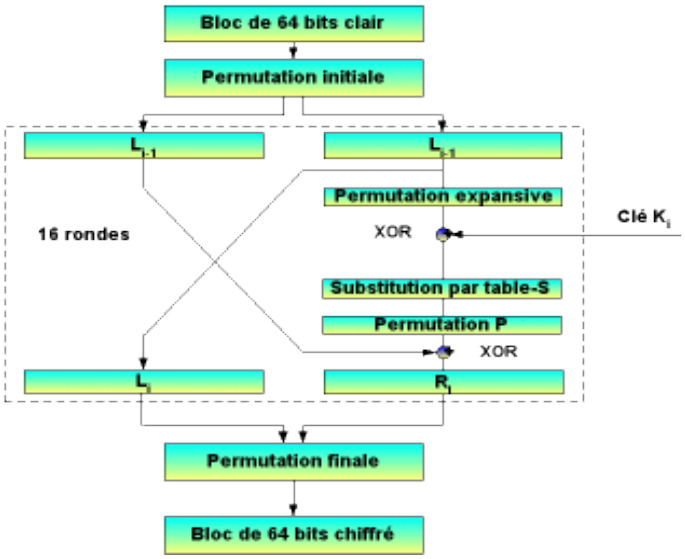

Figure 3.3 algorithme DES

L'algorithme repose principalement sur 3 étapes, en plus de la gestion spécifique de la clé :

- ✓ Permutation initiale
- ✓ Calcul médian (16 fois) : application d'un algorithme complexe appliqué en fonction de la clé
- ✓ Permutation finale

- **AES (Advanced Encryption Standard)**

En 1997, le NIST (National Institue of Standards and Technology) a lancé un appel d'offre pour un algorithme de chiffrement symétrique avec un bloc de taille 128 bits et supporte des clés de 128, 192 et 256 bits. Les critères d'évaluation de l'offre comprenaient la sécurité, la puissance de calcul, les contraintes de la mémoire des petits dispositifs (comme la carte à puce),

la plateforme logicielle et matérielle et la flexibilité. Un total de 15 algorithmes ont été envoyés et 5 ont été sélectionné parmi ces 15 algorithmes [12].

L'algorithme de Rijndael a été sélectionné pour devenir l'AES en 2001. Rijndael a été conçu par Joan Daemen et Vincent Rijmen, deux chercheurs de la Belgique. Les autres algorithmes sont : Serpent, Twofish, RC6, et MARS. L'AES n'utilise pas le schéma de Feistel

Mais il utilise des opérations mathématiques comme des substitutions, des permutations et des XORs. Il a plusieurs rounds identiques (de 10 à 14) et leur nombre dépend de la taille de la clé. L'AES opère au niveau octet ce qui permet une implémentation efficace au niveau matérielle et logicielle. L'AES est un standard, donc libre d'utilisation, sans restriction d'usage ni brevet. NIST spécifie actuellement AES dans le document FIPS 197, comme le nouveau standard de chiffrement symétrique. AES est approuvé pour utilisation par les organisations du gouvernement U.S pour protéger l'information sensible non classifiée.

AES a trois niveaux forts de sécurité : 128 bits, 192 bits et 256 bits. La sécurité 128 bits fournira au moins 30 ans de protection. AES ne fournit pas seulement une sécurité supérieure à TDES mais il délivre aussi une meilleure performance. Une meilleure sécurité et une meilleure performance rendent l'AES une alternative plus attractive que TDES et un bon choix pour un algorithme de chiffrement symétrique [12].

AES est également un candidat particulièrement approprié pour les dispositifs mobiles limités en ressources de calcul et de stockage. Le monde de la 3G (3ème génération de dispositifs mobiles) a adopté l'algorithme AES pour son schéma d'authentification.

AES est un algorithme de Chiffrement par bloc (Block Cipher).

- **RC4**

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.

➢ **Principe La clef 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).

2.2 La cryptographie à clé publique

A l'ère de l'Internet, la cryptographie à clé publique s'est discrètement immiscée dans notre vie quotidienne. Elle permet notamment d'assurer la sécurité des cartes à puce ou du commerce électronique. La cryptographie à clef publique permet d'éliminer le problème de la distribution des clefs. En effet, une clef publique diffusée sur un annuaire permet à n'importe qui de chiffrer un message, en revanche, il n'est pas possible de déchiffrer celui-ci par cette seule clef, il en faut une autre qui est gardée secrètement. [11]

Dans un cryptosystème asymétrique (aussi appelé cryptosystème à clés publiques), les clés existent par paires (on parle souvent de bi-clés) :

- Une clé publique pour le chiffrement
- Une clé secrète pour le déchiffrement

Ainsi, dans un système de chiffrement à clé publique, les utilisateurs choisissent une clé aléatoire dont ils sont seuls connaisseurs (il s'agit de la clé privée). A partir de cette clé, ils déduisent chacun automatiquement un algorithme (il s'agit de la clé publique). Les utilisateurs s'échangent cette clé publique au travers d'un canal non sécurisé. [11]

Lorsqu'un utilisateur désire envoyer un message à un autre utilisateur, il lui suffit de chiffrer le message à envoyer au moyen de la clé publique du destinataire (qu'il trouvera par exemple dans un serveur de clés tel qu'un annuaire). Ce dernier sera en mesure de déchiffrer le message à l'aide de sa clé privée (qu'il est seul à connaître).

2.2.1 Les avantages de la cryptographie asymétrique sont :

- Gestion de la secrète facilitée
- Pas de secrète à transmettre
- Nombre de clés à distribuer est réduit par rapport aux clés symétriques
- Très utile pour échanger les clés
- Permet de signer des messages facilement

2.2.2 Les inconvénients de la cryptographie asymétrique sont :

- Des clés plus longues (1024 à 4096 bits)
- Gestion de certificats de clés publiques
- Lenteur de calcul
- Pas d'authentification de la source

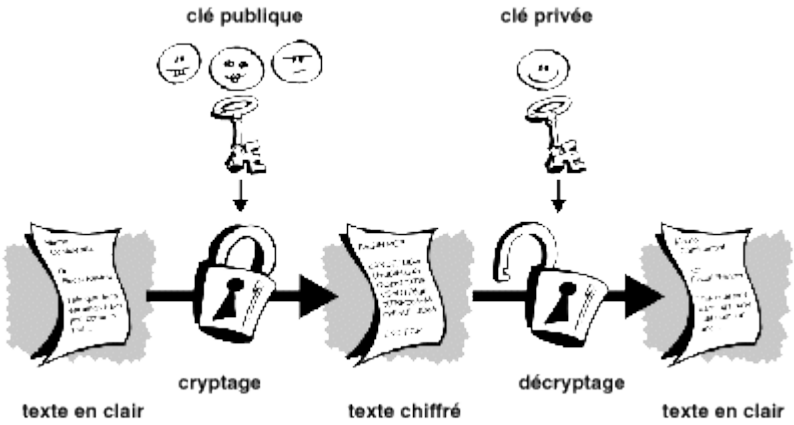

Figure3.4 La cryptographie à clé publique

2.2.3 Quelque algorithme de La cryptographie à clé publique

- **RSA**

Le premier système à clé publique solide à avoir été inventé, et le plus utilisé actuellement, est le système RSA. Publié en 1977 par Ron Rivest, Adi Shamir et Leonard Adleman de l'Institut de technologie du Massachusetts (MIT), le RSA est fondé sur la difficulté de factoriser des grands nombres, et la fonction à sens unique utilisée est une fonction "puissance". [HSB14]

Principes :

On possède une paire de clés, l'une publique (e,n) et une privée (d,n). La première étape revient à choisir n. Il doit s'agir d'une valeur assez élevée, produit de 2 nombres premiers très grands p et q. En pratique, si p et q ont 100 chiffres décimaux, n possèdera 200 chiffres. Selon le niveau de sécurité souhaité, la taille de n peut varier : 512 bits, 768, 1024 ou 20483.

Dans un second temps, on choisira un très grand entier e, relativement premier à (p-1)*(q-1). La clé publique sera formée par (e,n). On choisira ensuite un d tel que

$$e*d \equiv 1 \bmod (\Phi(n)).$$

La clé privée sera donnée par (d,n).

- **El Gamal** [10]

El Gamal C'est un algorithme à clef publique présent à la base de la norme U.S. de signature électronique. Il fut inventé par Taher ElGamal en 1984. Il est basé sur la difficulté de calculer des logarithmes discrets. Le problème du logarithme discret consiste à retrouver un entier λ tel que

$$h = g^{\lambda} \bmod p.$$

Principe du chiffrement :

Soit un entier premier p très grand et $p-1$ doit avoir un grand facteur premier. On produit :

– une clé secrète s, telle que $s \in (1...p-2)$,

– une clé publique reposant sur l'entier p, un entier a premier avec *p*, et l'entier *P* telque $P = a^s \bmod p$ Le nombre a est pris tel que $a \in (0...p-1)$ et $\forall\ k \in (1...p-2)$:

$$a^k\ 6 \neq 1 \bmod p$$

Soit un message M, avec M < p. On détermine un nombre aléatoire k qui n'est connu que de celui qui chiffre et différent à chaque message. On calcule alors

$$C_1 = a^k \bmod p$$

$$C_2 = M.P^k \bmod p$$

On obtient alors le message chiffré $C = (C_1, C_2)$. Le message chiffré est alors deux fois plus long que le message original.

Principe du déchiffrement :

A la réception, on calcule

$$R_1 = (C_1)s^* \bmod p$$

$$= a^{sk} \bmod p$$

$$= P^k \bmod p$$

Le destinataire possède la clé privée (s). Ayant P^k, on divise C_2 par cette valeur :

$$D_K(C) = C_2/(R_1)$$

$$= M.P^k \bmod p / P^k \bmod p$$

$$= M$$

3 La signature numérique (ou digitale)

La cryptographie à clé publique permet de s'affranchir du problème de l'échange de la clé, facilitant le travail de l'expéditeur. Mais comment s'assurer de l'authenticité de l'envoi ? Comment être sûr que personne n'usurpe l'identité d'Alice pour vous envoyer un message ? Comment être sûr qu'Alice ne va pas nier vous avoir envoyé ce message ?

La norme [ISO 7498-2] définit la signature numérique comme des "données ajoutées à une unité de données, ou transformation cryptographique d'une unité de données, permettant à un destinataire de prouver la source et l'intégrité de l'unité de données et protégeant contre la contrefaçon (par le destinataire, par exemple)". La mention "protégeant contre la contrefaçon" implique que seul l'expéditeur doit être capable de générer la signature. [11]

Une signature numérique fournit donc les services d'authentification de l'origine des données, d'intégrité des données et de non répudiation. Ce dernier point la différencie des codes d'authentification de message, et a pour conséquence que la plupart des algorithmes de signature utilisent la cryptographie à clef publique**.** [11]

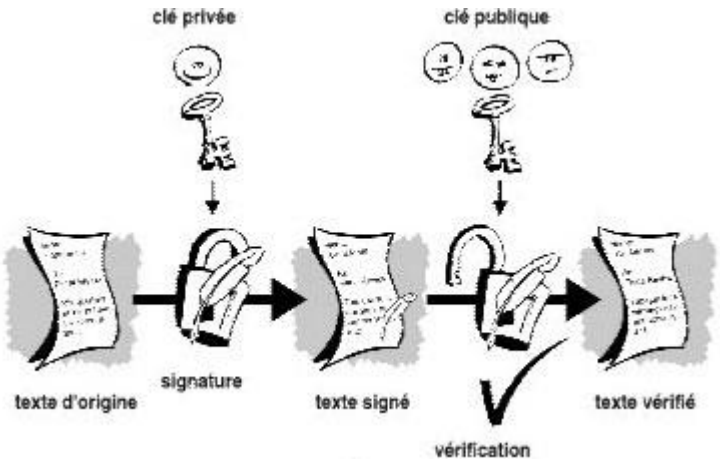

Figure 3.5 La signature numérique [SCAI99]

3.1 Les principaux algorithmes de signature

3.1.1 Digital Signature Standard (DSS)

Proposé par le NIST en 1991, puis finalement adopté en 1994 [18], DSS (Digital Signature Standard) est la norme de signature numérique du gouvernement Américain. Elle se base sur l'algorithme DSA (Digital Signature Algorithm), qui utilise SHA comme fonction de hachage à sens unique et Elgamal pour la génération et la vérification de la signature.

Le DSS, plus connu sous le sigle DSA (Digital Signature Algorithm) fait partie de la spécification DSS pour Digital Signature Standard adoptée en 1993 [18]. Une révision mineure a été publiée en 1996 (FIPS 186-1) et le standard a été amélioré en 2002 dans FIPS 186-2. Il est couvert par le patent n° 5231668 aux USA (26 juin 1991), attribuée à David Kravitz, ancien employé de la NSA.

Le DSA est similaire à un autre type de signature développée par Claus Schnorr en 1989. Il a aussi des points communs avec la signature ElGamal. Le processus se fait en trois étapes :

- génération des clés
- signature du document
- vérification du document signé

3.1.2 La signature RSA

Si DSS est la norme officielle aux U.S.A., RSA est en revanche une norme de fait, qui est en pratique beaucoup plus utilisé pour la signature que DSA.

4 Fonctions de hachage

Une fonction de hachage est une fonction qui fait subir une succession de traitements à une donnée quelconque fournie en entrée pour en produire une « empreinte » servant à identifier la donnée initiale sans que l'opération inverse de décryptage soit possible. Le terme hachage évite l'emploi de l'anglicisme hash. Le résultat de cette fonction est par ailleurs aussi appelé somme de contrôle, empreinte, résumé de message, condensé, condensat ou encore empreinte cryptographique.

Les fonctions de hachage sont conçues pour effectuer un traitement de données rapide : calculer l'empreinte d'une donnée ne doit couter qu'un temps négligeable. Une fonction de hachage doit aussi éviter le plus possible les collisions (deux empreintes identiques alors que les données différaient) Selon l'emploi de la fonction de hachage, il peut être souhaitable qu'un infime changement de la donnée en entrée (un seul bit, par exemple) entraine une perturbation conséquente de l'empreinte correspondante, rendant une recherche inverse par approximations successives impossible : on parlera d'effet avalanche.

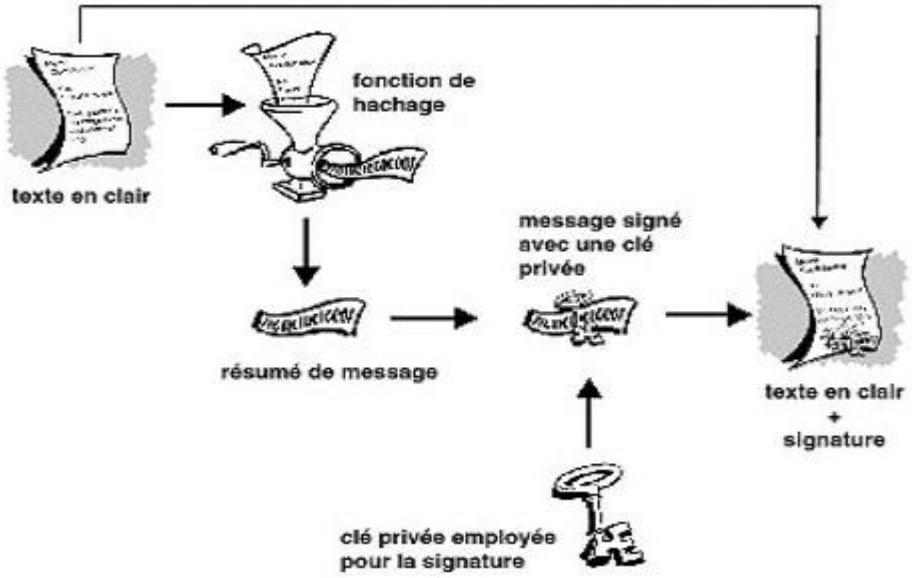

Figure 3.6 Fonctions de hachage

4.1 Algorithmes de hachage :

4.1.1 Algorithme de hachage sécurisé SHA : [07]

L'algorithme SHA a été conçu par NSA, standardisé par NIST et publié comme FIPS PUB 180 en 1993. Une version révisée a été délivrée en 1995 comme FIPS PUB 180-1.

La version actuelle est FIPS PUB 180-2 avec une notice de changement pour SHA-224 en février 2004.[08]

La liste des algorithmes SHA est : SHA-1, SHA-256, SHA-384 et SHA-512. A cause des

Calculs intensifs dans SHA-384 et SHA-512, ces deux algorithmes ne sont pas recommandés pour être utilisés sur les dispositifs mobiles.

SHA-1 est l'algorithme le plus utilisé dans la famille des algorithmes SHA. SHA-1 prend en entrée un message de taille inférieure à 2st bits et produit une empreinte de 160 bits.

SHA-256 prend aussi en entrée un message de taille inférieure à 2^{64} bits mais produit une empreinte de 256 bits.

Il y a une nouvelle attaque sur SHA-1 sans utiliser la force brute, qui trouve des collisions à l'intérieur de 269 opérations hash.

A cause de l'avancement technologique et les attaques, NIST a planifié d'abandonner SHA-1 la fin 2010, et recommande des fonctions HASH plus fortes comme SHA-256.

Les algorithmes SHA-256, SHA-384 et SHA-512 sont relativement nouveaux et sont aussi standardisés par NIST mais ils sont plus intensifs en calcul et crypto analyses

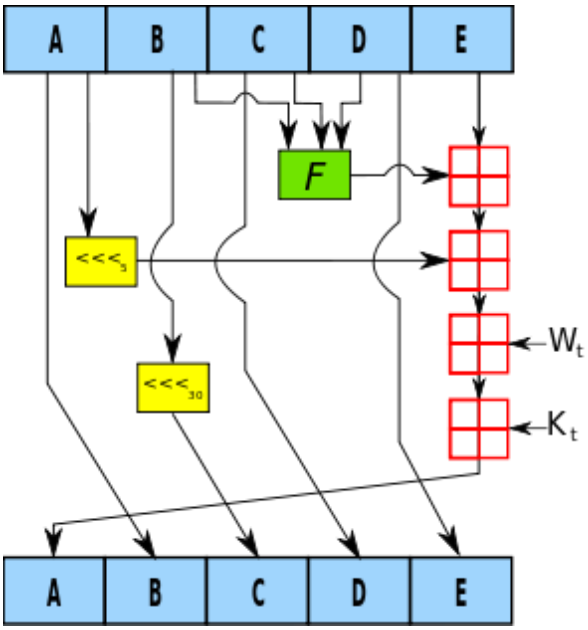

Figure 3.7 Algorithme SHA

4.1.2 L'algorithme MD5 (Message Digest version 5)

En 1991, Ronald Rivest améliore l'architecture de MD4 et crée MD5 (Message Digest 5). C'est une fonction de hachage cryptographique qui permet d'obtenir pour chaque message une chaine de 32 caractères (soit 128 bits) hexadécimaux avec une probabilité très forte que, pour deux messages différents, leurs empreintes soient différentes.

Ce quelle que soit la taille de l'information en entrée (de 0 octets à plusieurs gigas). Cette transformation est donc irréversible (Dans le sens où on ne peut pas trouver l'information en entrée à partir d'une somme MD5).

En 1996, une faille grave (possibilité de créer des collisions à la demande) est découverte. En 2004, une équipe chinoise découvre des collisions complètes. MD5 n'est donc plus considéré comme sûr au sens cryptographique. MD5 reste encore utilisé comme outil de vérification lors des téléchargements (par exemple, en FTP). Les sites affichent encore souvent la signature en MD5 de leurs fichiers.

Le programme John the ripper permet de casser les MD5 triviaux par force brute. Des serveurs de "tables inverses" (à accès direct, et qui font parfois plusieurs gigaoctets) permettent de les craquer souvent en moins d'une seconde.

Aujourd'hui, il est par exemple possible de créer des pages HTML aux contenus très différents et ayant pourtant le même MD5. La présence de codes de "bourrage" placés en commentaires, visibles seulement dans la source de la page web, trahit toutefois les pages modifiées pour usurper le MD5 d'une autre.

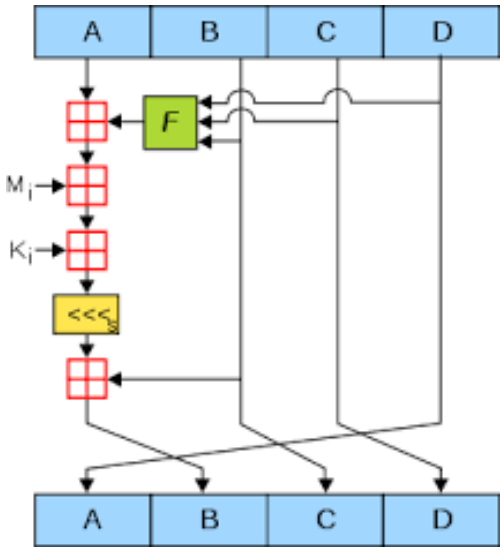

Figure 3.8 L'algorithme MD5

5 Code d'authentification de message MAC : [07]

MAC-Message Authentification Code partage les 5 premières propriétés des fonctions de hachage à l'exception qu'il implique l'utilisation d'une clé secrète comme une entrée avec le message à authentifier. Avec MAC nous avons la fonction : u(v,p)=: avec u est la fonction MAC qui génère d à partir d'un secret s et un message x.

Une fonction de hachage ne peut pas être utilisée comme MAC parce qu'elle n'exige pas une clé secrète.

Une fonction HASH peut opérer d'une manière identique à un MAC comme o (v||p), o(p||v) Yx o(v || p || v). Il est donc fondamental d'éviter l'utilisation de fonction de hachage comme un MAC. [09]

Il y avait plusieurs approches pour incorporer une clé secrète dans un algorithme de HASH existant. L'approche la plus connue est HMAC (keyed-Hash Message Authentification Code). HMAC a été choisi pour implémenter MAC pour le protocole IPSec. Il est aussi utilisé dans d'autres protocoles comme TLS et SET (Secure Electronic
Transaction).

6 Conclusion

Dans ce chapitre, nous avons présenté les deux différents types de chiffrement symétrique et asymétrique, puis on a focalisé sur ses techniques et en particulier les algorithmes les plus connus, qui sont devenus des standards comme DES, AES, RSA et autres, pour mieux comprendre l'objective de tels algorithmes ainsi que leur faiblesses

Dans le chapitre suivant on va implémenter ses algorithmes pour mieux comprendre.

CHAPITRE 4

IMPLEMENTATION ET REALISATION

2. Introduction

Nous avons présenté dans ce chapitre, les résultats obtenus de l'implémentation des algorithmes cryptographie (classique, symétrique, asymétrique) sur les défirent plateformes (desktop, page web) et faire une comparaison du temps de chiffrement, déchiffrement et la consommation du l'espace mémoire entre ces algorithmes.

3. L'environnement de développement

Figure 4.1 NetBeans

NetBeans est un environnement de développement intégré (EDI), placé en open source par Sun en juin. En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, JavaScript, XML, Ruby, PHP et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web). Conçu en Java, NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Développement Kit JDK est requis pour les développements en Java. [14]

4. Les applets de simulation

Définition d'une Applet :

Une applet est un programme Java qui s'exécute dans un logiciel de navigation supportant java ou dans l'appletviewer du JDK (Java Developpment Kit). Le mécanisme d'initialisation d'une applet se fait en deux temps :

1. la machine virtuelle java instancie l'objet Applet en utilisant le constructeur par défaut
2. la machine virtuelle java envoie le message init à l'objet Applet

Les Applets programmées :

Afin de vulgariser le fonctionnement de certains algorithmes, des Applets ont été réalisée pour simuler les étapes de fonctionnement et obtenir les résultats de traitements instantanément.

Les algorithmes dont les Applets ont été programmées sont :

- Algorithme d'RSA
- Algorithme d'ECC

5. L'architecture de notre système

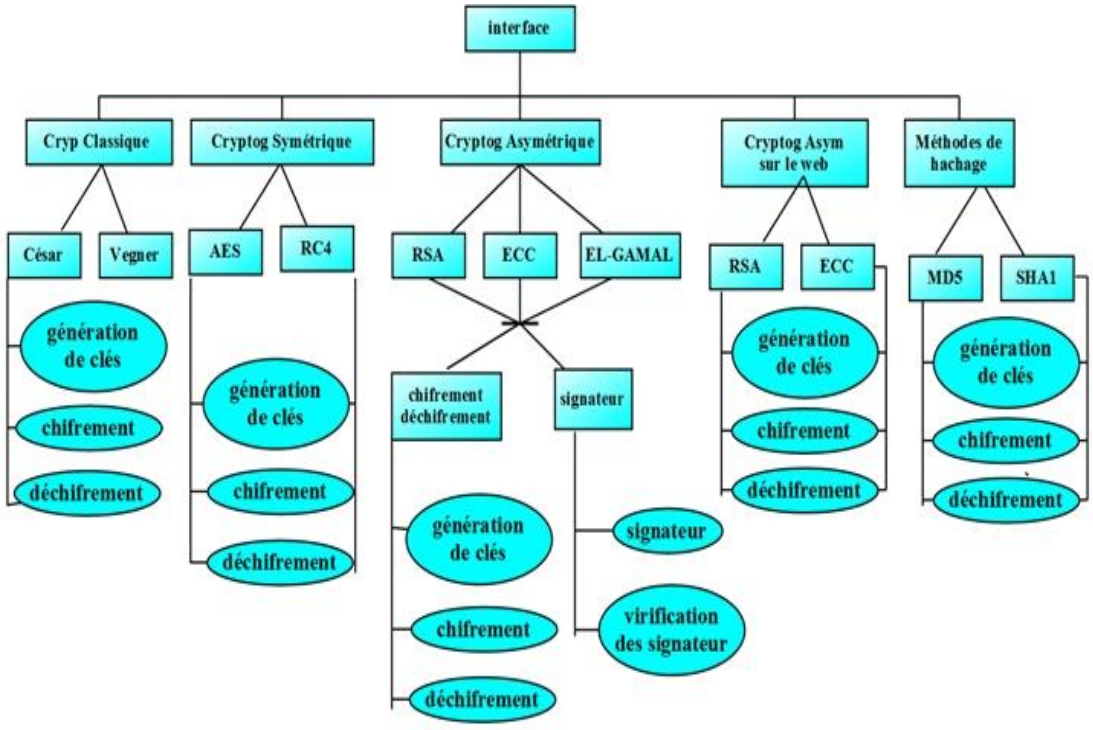

Figure 4.2 l'architecture de système

6. Implémentation et résultat :

paramètre	sécurité	Longueur (bits)	algorithme
P1024	80	1024	RSA
P2048	112	2048	
P15360	256	15360	
P512	80	512	EL-Gamal
P2048	112	2048	
P15360	256	15360	
secp160r1	80	160	ECC
Secp224r1	112	224	
Secp521r1	256	521	

Table 4.1 Sélection des paramètres [15] [16]

6.1.Calcule de l'exécution de temps et l'espace mémoire

- Résultat pour les algorithmes asymétrique : le message chiffré (tic2016)

Le temps d'exécution et L'espace mémoire du RSA

	Le temps d'exécution (ms)			L'espace mémoire(ko)		
paramètre	512	1024	2048	512	1024	2048
Générations de la clé	*352*	*250*	*637*	*74821*	*1800*	*1344*
chiffrement	*646*	*63*	*47*	*3098288*	*1256*	*3400*
déchiffrement	*78*	*31*	*62*	*1938752*	*1232*	*880*
RSA - Signature	*63*	*31*	*63*	*27608*	*1632*	*488*
RSA - Vérification	*78*	*63*	*62*	*704*	*8480*	*8888*

Table 4.2 RSA

Le temps d'exécution et L'espace mémoire de l'EL-Gamal :

	Le temps d'exécution (ms)			L'espace mémoire		
paramètre	512	1024	15360	512	1024	15360
Générations de la clé	*11073*	*4247688*	*>1h*	*636872*	*1878856*	*>1h*
chiffrement	*653*	*94*	*>1h*	*3198104*	*5144*	*>1h*
déchiffrement	*64*	*47*	*>1h*	*1940840*	*968*	*>1h*
RSA - Signature	*>1h*	*>1h*	*>1h*	*>1h*	*>1h*	*>1h*
RSA - Vérification	*>1h*	*>1h*	*>1h*	*>1h*	*>1h*	*>1h*

Table 4.3 EL-Gamal

Le temps d'exécution et L'espace mémoire d'ECC :

	Le temps d'exécution(ms)			L'espace mémoire		
paramètre	160	256	239	160	256	239
Générations de la clé	*453*	*63*	*32*	*729880*	*376*	*1104*
chiffrement	*656*	*63*	*47*	*4150784*	*4456*	*1232*
déchiffrement	*93*	*31*	*46*	*1933080*	*4120*	*3744*
RSA - Signature	*47*	*31*	*31*	*47200*	*408*	*136*
RSA - Vérification	*78*	*31*	*46*	*5936*	*6272*	*6272*

Table 4.4 ECC

- Résultat pour les algorithmes symétrique : le message chiffré (tic2016)

Le temps d'exécution et L'espace mémoire d'AES

	Le temps d'exécution(ms)			L'espace mémoire		
Taille de clé	*128*	*192*	*256*	*128*	*192*	*256*
Générations de la clé	*62*	*32*	*63*	*30472*	*12776*	*13336*
chiffrement	*31*	*70*	*109*	*48*	*784*	*128512*
déchiffrement	*31*	*42*	*47*	*2936*	*12554*	*2808*

Table 4.5 AES

Le temps d'exécution et L'espace mémoire d'RC4

	Le temps d'exécution(ms)			L'espace mémoire		
Taille de clé	64	96	128	64	15	128
Générations de la clé	*31*	*15*	*47*	*1592*	*112*	*48*
chiffrement	*31*	*16*	*109*	*648*	*1320*	*1282*
déchiffrement	*31*	*15*	*16*	*168*	*744*	*672*

Table 4.6 RC4

- Résultat pour les algorithmes asymétrique dans le web (applet) :

Le message chiffré (tic2016)

Le temps d'exécution et L'espace mémoire d'RSA dans le web

	Le temps d'exécution(ms)	L'espace mémoire
Générations de la clé	*63*	*136*
chiffrement	*31*	*1080*
déchiffrement	*16*	*32*
RSA - Signature	*56*	*7522*
RSA - Vérification	*31*	*4896*

Table 4.7 RSA dans le web

Le temps d'exécution et L'espace mémoire d'ECC dans le web

	Le temps d'exécution(ms)	L'espace mémoire
Générations de la clé	*453*	*10652*
chiffrement	*47*	*1937608*
déchiffrement	*31*	*48*
RSA - Signature	*47*	*4546*
RSA - Vérification	*15*	*288*

Table 4.8 ECC dans le web

➢ Résultat pour l'implémentation des techniques de hachage

Le message chiffré (tic2016)

		Le temps d'exécution(ms)	L'espace mémoire
SHA	Hash	63	21632
MD5	hash	47	2936

Table 4.9 techniques de hachage

5.2 Comparaison et performance des tests des algorithmes

5.2.1 Le temps d'exécution

➢ Asymétrique

RSA vs EL-GAMMAL vs ECC

Fonction	unité	RSA			EL-GAMMAL			ECC		
Paramètres		512	1024	2048	512	1024	15360	160	256	239
Chiffrement	Bloc/ms	*646*	*63*	*47*	*653*	*94*	*>1h*	*656*	*63*	*47*
Déchiffrement	Bloc/ms	*78*	*31*	*62*	*64*	*47*	*>1h*	*93*	*31*	*46*

Table 4.10 Comparaison RSA vs EL-GAMMAL et ECC

Figure 4.3 Comparaison RSA vs EL-GAMMAL et ECC

Les résultats obtenus à partir (tableau 3.10) montrent clairement que

- Aux niveaux de bas sécurité (80), Le temps d'exécution de chiffrement est plus grand que le déchiffrement, et l'ECC est le plus rapide
- Aux niveaux de moyenne sécurité (112), Le temps d'exécution de chiffrement est plus grand que le déchiffrement et EL-GAMMAL est le plus rapide que les autres algorithmes.
- Aux niveaux de haute sécurité (256), Le temps d'exécution est plus moins par rapport à les autres niveaux, sauf l'algorithme EL-GAMMAL, Sa vitesse de chiffrement/déchiffrement est très lente

➢ Symétrique

AES vs RC4

fonction	unité	AES			RC4		
Paramètres		128	192	256	64	96	128
chiffrement	Bloc/ms	*31*	*70*	*109*	*31*	*16*	*109*
Déchiffrement	Bloc/ms	*31*	*42*	*47*	*31*	*15*	*16*

Table 4.11 Comparaison d'AES vs RC4

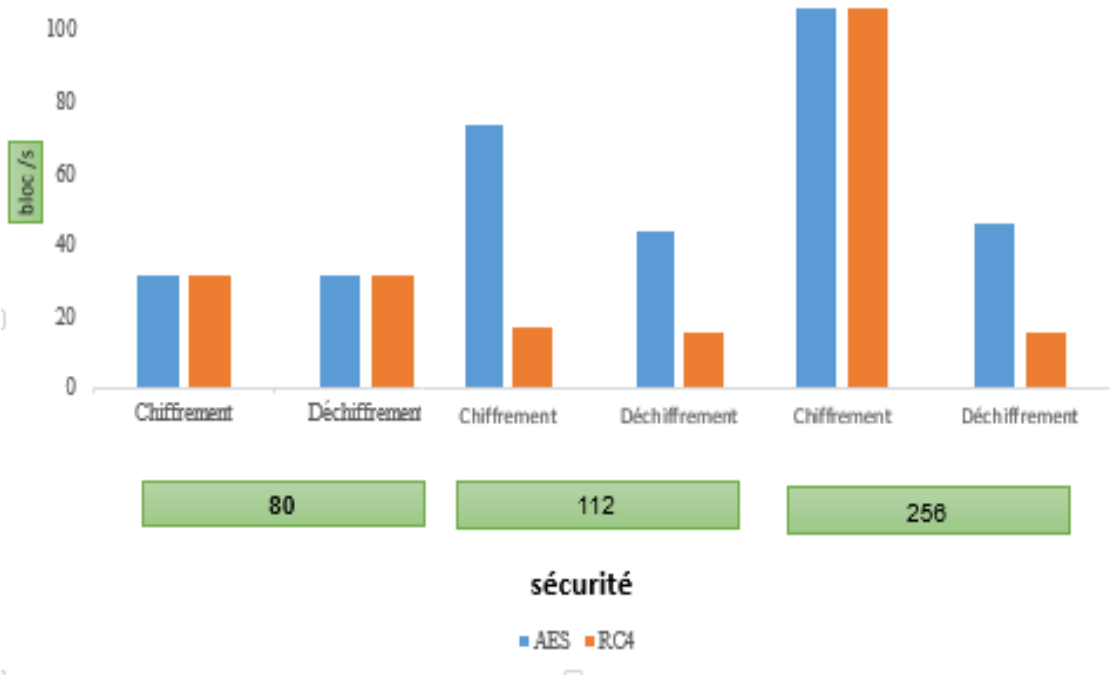

Figure 4.4 Comparaison d'AES vs RC4

Les résultats obtenus à partir (tableau 3.11) montrent clairement que

- Aux niveaux de bas sécurité (80), Le temps d'exécution de chiffrement est le même que le déchiffrement pour l'AES et RC4
- Aux niveaux de moyenne sécurité (112), Le temps d'exécution de chiffrement est plus grand que le déchiffrement et RC4 est le plus rapide que l'AES.
- Aux niveaux de haute sécurité (256), Le temps d'exécution de chiffrement est plus grand que le déchiffrement et l'RC4 est plus rapide que, l'AES.

➢ dans le web (applet)

RSA vs ECC

fonction	unité	RSA	ECC
Chiffrement(ms)	Bloc/s	*31*	*47*
Déchiffrement(ms)	Bloc/s	*16*	*31*

Table 4.12 Comparaison de RSA vs ECC

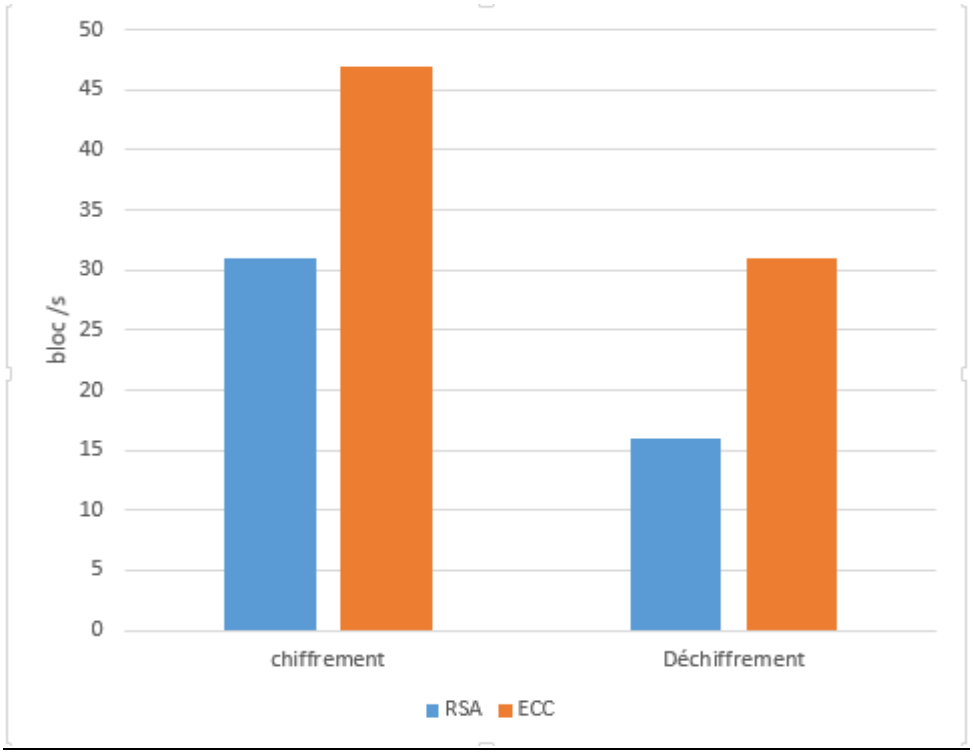

Figure 4.5 Comparaison de RSA vs ECC

Les résultats obtenus à partir (tableau 12) montrent clairement que

- Pour l'RSA Le temps d'exécution de chiffrement est plus grand que le déchiffrement
- Pour l'ECC Le temps d'exécution de chiffrement est plus grand que le déchiffrement.

Donc dans ce cas Le temps d'exécution de chiffrement est plus grand que le déchiffrement.

6. Quelque interface de notre projet

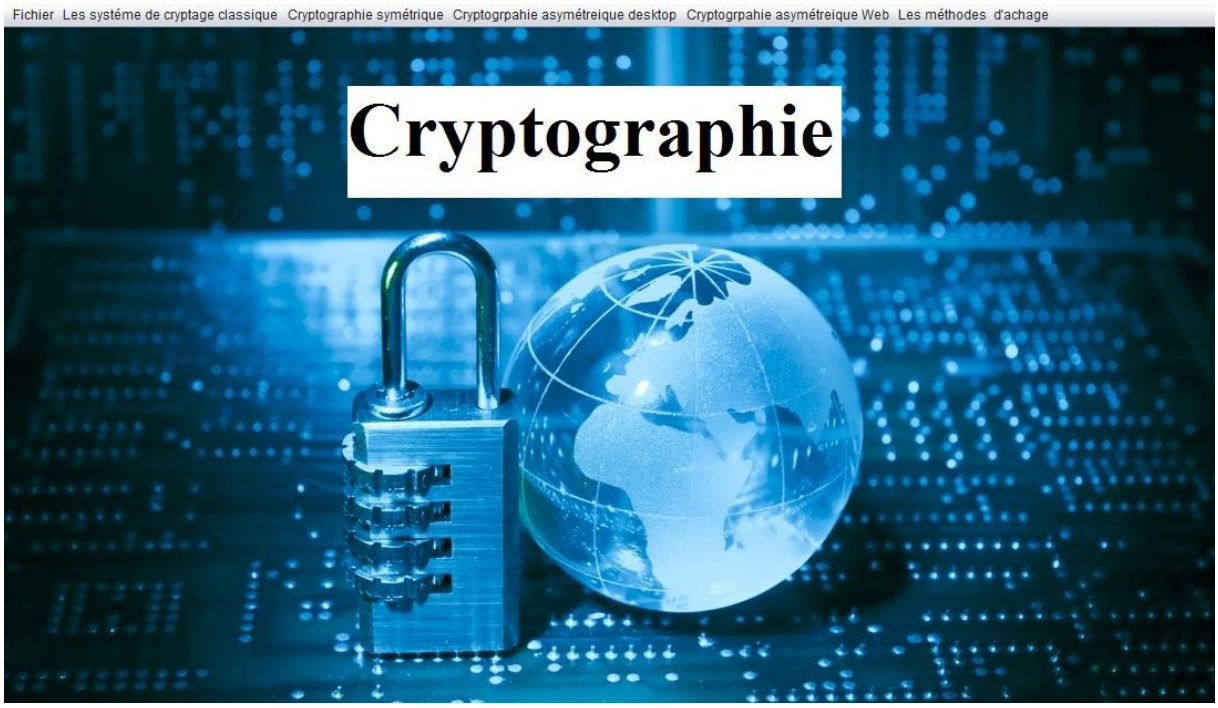

Figure 4.6 interface générale

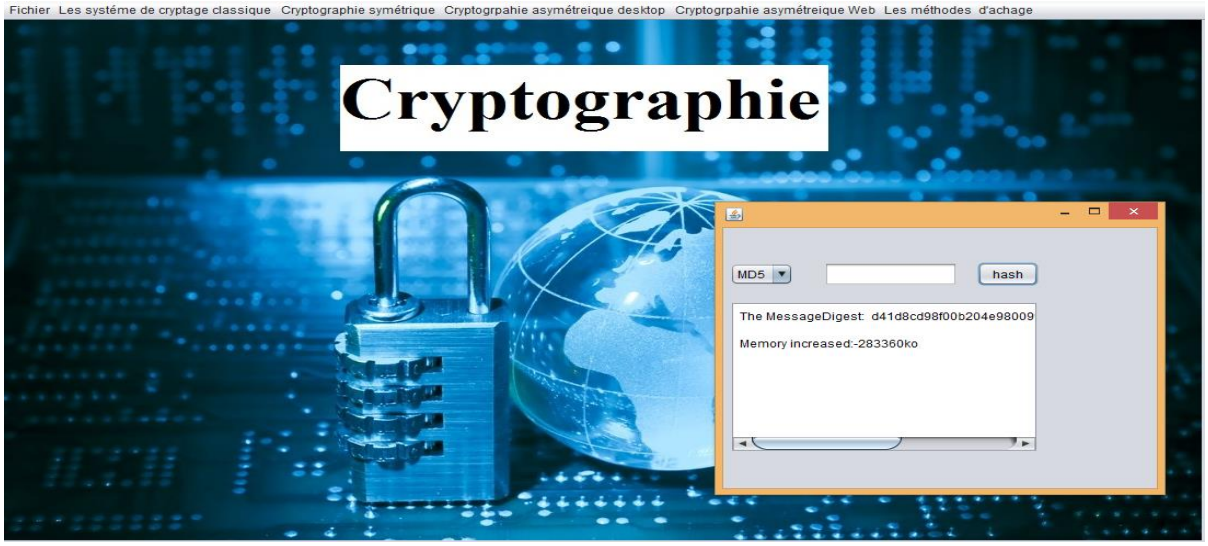

Figure 4.7 méthode de hachage

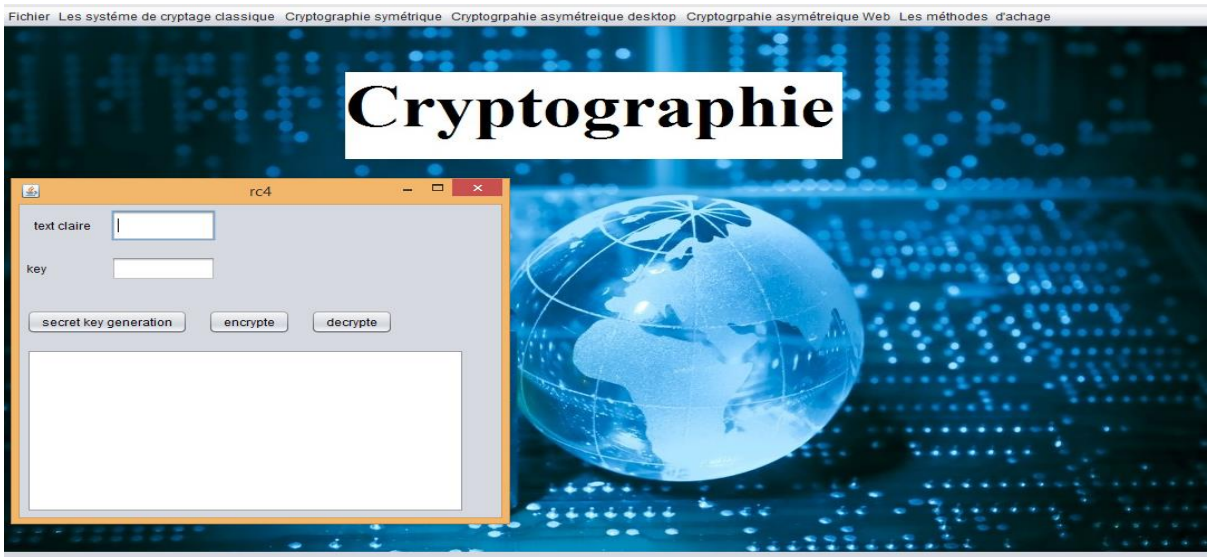

Figure 4.8 chiffrement symétrique

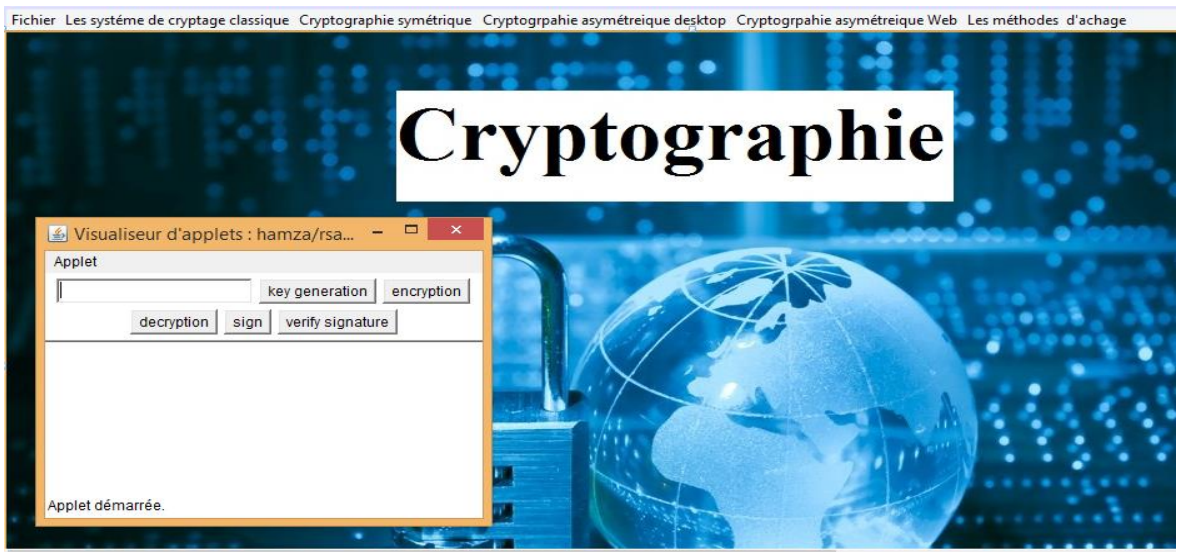

Figure 4.9

7 **conclusion**

En pratique, la cryptographie est utilisée pour permettre des échanges sécurisés d'informations et des vérifications d'authenticité de messages. Comme on va le voir dans le chapitre comment chiffré et déchiffré l'information est quel est le meilleur algorithme utilisé.

CONCLUSION GÉNERALE

Ce projet a été réalisé dans le cadre de projet de fin d'étude niveaux master au sein de département des MI (mathématique et informatique) d'université de M'SILA. Dans lequel nous avons appliqué toutes les compétences et connaissances acquises durant les années d'étude.

Notre travail se résumé en étude et comparaison des principaux systèmes de cryptage et les technique y afférentes.

A travers notre étude, nous avons vu un panel de méthodes de chiffrement de l'antiquité à nos jours, les attaques existantes sur les cryptosystèmes actuels les plus utilisées et les moyens inventés pour s'assurer de l'intégrité, de l'authentification de l'expéditeur et du destinataire d'un message.

Ainsi, la cryptographie est une science en perpétuelle évolution, la cryptanalyse l'aidant à trouver les failles d'un système pour toujours avancer. Cette évolution est importante car la cryptographie joue un grand rôle dans la sécurité internationale, tout étant aujourd'hui informatisé.

Dans notre projet, nous avons utilisé le JAVA-NetBeanse comme environnement de programmation.

En effet, malgré les grosses difficultés qu'on a rencontré, des quelles la non organisation et moins de l'expérience, le peu de temps consacré à cette étude, l'implémentation de notre projet fut l'une des majeures difficultés que nous avons rencontré, choix du langage de programmation plus efficace. Nous pouvons dire qu'on a très bien tiré profit de ce travail et qu'en toutes circonstances, on a appris beaucoup de choses : le travail en groupe, les technique et les compétences acquises dans le domaine du la cryptographie, la maîtrise et le savoir-faire dans les domaines de l'analyse et la conception, ainsi que beaucoup d'autres connaissances concernant le milieu de la sécurité.

BIBLIOGRAPHIE

[01]: STERN JACQUES, "La science du secret", Editions Edile Jacob, 1998

[02] BRUCE SCHNEIER: "Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C", John Wiley & Sons, Inc., 01/01/96

[03] A. MENEZES, P. VAN OORSCHOT, AND S. VANSTONE: "Handbook of Applied Cryptography", CRC Press, 1996 .

[04]: DAVID KAHN, "The Codebreakers: The Story of Secret Writing", New York: Macmillan Publishing Co., 1967.

[05]: Hacini Souleyman Boumedyen, "Implémentation d'algorithmes de Cryptographie", Université Abou Bakr Belkaid– Tlemcen Faculté des Sciences Département d'Informatique, 2013-2014

[06]: HILL LESTER S.,"Cryptography in an Algebraic Alphabet", American Mathematical Monthly-N36-P306à312, 1929

[07]: Mahammedi Nadjiba, Mahdadi Houda, "Implémentation de benchmark d'opérations crypto basées ECC pour l'étude et comparaison de courbes elliptiques, Mémoire MASTER ACADEMIQUE, UNIVERSITE KASDI MERBAH OUARGLA, 2012 /2013

[08] National Institute of Standards and Technology (NIST), Secure Hash Standard. Federal Information Processing Standards Publication 180-2, 2002, en ligne **http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf.**

[09] Niels Ferguson, Bruce Schneier. Practical Cryptography. Indiana: Wiley Publishing, Inc. : s.n., 2003

[10] T. El Gamal, A public key cryptosystem and signature scheme based on discrete logarithms. IEEE Transactions on Information Theory IT-31, 496-473,1976.

[11] El Khier DEHMECHE, ETUDE ET COMPARAISON DES PRINCIPAUX SYSTEMES DE CRYPTAGE, diplome de magister en informatique, université M'sila, 2006.

[12] SecuriteInfo.com,L'AES : Advanced Encryption Standard http://www.securiteinfo.com/cryptographie/aes.shtml.

[13] Bayart, Frederic. La saga du DES. http://www.bibmath.net/crypto.

[14] Jason Wexbridge , Walter Nyland,"NetBeans Platform for Beginners", Lean Publishing, 31-08-2014

[15] STANDARDS FOR EFFICIENT CRYPTOGRAPHY, SEC 2: Recommended Elliptic Curve Domain Parameters, Certicom Research http://www.secg.org/collateral/sec2_final.pdf.

[16] JeF Hoffstein, Nicholas Howgrave-Graham, Jill Pipher, Joseph H. Silverman, William Whyte, Performance Improvements and a Baseline Parameter Generation Algorithm for NTRUSign, 5 Burlington Woods, MA 01803.
https://www.securityinnovation.com/uploads/Crypto/NTRUSignParams-2005-08.pdf.
[17] Sitegénéralistesur lesréseaux. http://www.orbytes.fr
[18]: FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, "Digital Signature Standard (DSS)", FIPS PUB 186, 19 Mai 1994
[19]: Santa Clara, An Introduction to Cryptography, Copyright © 1990-1999 Network Associates, Inc. and its Affiliated Companies, pages 11, 15.

الملخص:

ان الهدف من هذه المذكرة هو دراسة علم التشفير وتطوره مع مرور الزمن، و إيجاد افضل أنظمة التشفير الفعالة في مجال تكنولوجيا المعلومات، من حيث، السرعة واستهلاك سعة الذاكرة ودرجة الأمان مع الاخذ بعين الاعتبار بيئة التطوير، وذلك بإجراء دراسة تحليلية لمختلف خوارزميات التشفير، المتماثل مثل (AES, RC4…) والغير المتماثل (RSA, ECC…) .

كلمات دلالية: تشفير، خوارزمية، سرعة التشفير AES, RSA, hachage, ECC,

Abstract:

The purpose of this research paper is stady cryptography science, and Evolved with the times. as well find a better and efficient cryptography algorithemn that use on information techology, frome memory usage and safty degrees, whitout ingrowibg Development environment during act analy stady a defferent symtric crytography algorithms as RC4 , AES, and asymtric cryptography as RSA, ECC .

Keys words: cryptography, algorithms, hachage, ECC, performance.

Résume:

L'objectif de ce mémoire est L'étude de la cryptographie et son développement avec le temps de trouver des systèmes de cryptage plus efficaces (en fonction de la vitesse et l'aspect de sécurité), dans le domaine des technologies informatique et adaptés pour les différents environnements de programmation, en fait une étude et comparaison chiffrement/déchiffrement : soit symétrique (AES, RC4...), asymétrique (RSA, ECC...).

Mottes clés : cryptage, algorithme, hachage, ECC, performance.