République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université Mouloud MAMMERI de TIZI-OUZOU

Faculté de Génie Electrique et d'Informatique

Département d'Informatique

MEMOIRE

De fin d'études

En vue de l'obtention du diplôme de master en Informatique option Réseau, Mobilité et Systèmes Embarqués

Thème

Conception et réalisation d'une application de cryptage

Basée sur l'algorithme DES

Dirigé par :

- **M^me^ Hadaoui**

Réalisé par :

- M^elle^ Ait tayeb Souad
- M^elle^ Djender Lilia

2012/2013

Remerciements

Nous tenons à exprimer notre profonde gratitude à notre promotrice, madame Hadaoui pour nous avoir encadrés durant cette année, ainsi que pour ses conseils judicieux.

Que les membres du jury trouvent ici nos plus vifs remerciements pour avoir accepter d'honorer par leur jugement notre travail.

Un grand merci aussi à toute personne qui de prés ou de loin a contribué à ce que ce modeste travail voit le jour.

Dédicaces

A Mes chers parents,

A mes frères ainsi que toute ma famille,

A tous mes amis,

Lilia.

A Mes chers parents,

A mes frères ainsi que toute ma famille,

A tous mes amis,

Souad.

Liste des figures

Liste des tableaux

Introduction générale

L’évolution rapide des réseaux informatiques, privés ou publics engendre un volume toujours plus important de données sauvegardées et transmises, générant ainsi de nouveaux besoins en matière de sécurité. Dans un mode où l’entreprise dépend de plus en plus de son système informatique, la sécurité est donc devenue une préoccupation primordiale.

L’apparition de l’informatique et des télécommunications a contribué à une complexité accrue des problèmes et des solutions de sécurité en amenant des notions telles que virus informatiques, accès non autorisé aux données, fausses informations, etc. Mais avec ces nouveaux moyens de communication est arrivée la nécessité de protéger le contenu de certains messages des inévitables curieux. Dans ce contexte la cryptographie est l’un des mécanismes de sécurité de la transmission d’information, elle consiste à transformer un message clair en un message indéchiffrable pour tous, sauf les destinataires du message. On utilise pour cela un algorithme cryptographique et une ou plusieurs clés, secrètes ou publiques. L’émetteur crypte son message à l’aide d’un algorithme, le transmet crypté, et le récepteur peut alors le décrypter à l'aide d'une même clé.

La cryptographie est la science qui utilise les mathématiques pour le cryptage et le décryptage de données, elle nous permet ainsi de stocker des informations confidentielles ou de les transmettre sur des réseaux non sécurisés (tel que Internet), afin qu'aucune personne autre que le destinataire ne puisse les lire.

.

Divers algorithmes ont été proposés afin de protéger des informations personnelles contre des attaques, nous avons basé principalement sur l’algorithme DES (Data Encryption Standard) qui est un algorithme de chiffrement symétrique qui permet de sécuriser les informations à l’aide d’une clé secrète.

Objectif

Notre but consiste à concevoir un crypto-système pour le chiffrement et le déchiffrement de messages et de données sous forme de fichiers (texte, image, vidéo, …etc.) en utilisant l'algorithme DES (Data Encryption Standard).

Pour mieux comprendre le sujet, nous allons d'abord parler dans le 1er chapitre des différentes notions de la sécurité informatique, ainsi que les outils et les mécanismes de la sécurité ,on détaillera dans le 2éme chapitre les fondements et le fonctionnement de l'un de ses mécanismes qui est la cryptographie, et dans le 3éme chapitre on parlera de l'algorithme de cryptographie symétrique Data Encryption Standard (DES),et dans le chapitre qui suit on présentera la conception de notre application.

Enfin, dans le dernier chapitre, on illustrera les différentes fonctionnalités de notre application avec des captures d'écran.

Chapitre I

Généralités sur la sécurité des réseaux informatiques

I.1 Introduction

.

Les réseaux informatiques sont nés d'un besoin d'échanger des informations de manière simple et rapide entre les machines, la sécurité de ces derniers est de nos jours devenue un problème majeur dans la gestion des réseaux d'entreprise ainsi que pour les particuliers toujours plus nombreux à se connecter a Internet. La transmission d'informations sensibles et le désir d'assurer la confidentialité de celles-ci est devenue un point primordial dans la mise en place de réseaux informatiques.

Il est possible d'assurer la sécurité des données en protégeant physiquement l'accès au matériel, mais ceci devient impossible dés que le réseau devient physiquement étendu, d'où la nécessité de mettre en place une politique de sécurité efficace qui répond aux besoins des utilisateurs.

I.2 Généralités sur les réseaux :

I.2.1 Définition

Un réseau (Network) est un ensemble d'ordinateurs et périphériques interconnectés. Il permet de faire circuler des données informatiques et ainsi d'échanger du texte, des images, de la vidéo ou du son entre chaque équipement selon des règles et protocoles bien définis.

I.2.2 Objectif d'un réseau [28]

Les réseaux ont été et sont toujours développés pour un certain nombre de raisons. Il y en a en fait quatre principales.

- **Partage des ressources**

  Les réseaux permettent de rendre accessible un certain nombre de ressources (logiciels, bases de données, imprimantes...) indépendamment de la localisation géographique des utilisateurs.

- **Augmentation de la fiabilité et des performances**

  Les réseaux permettent par exemple de dupliquer en plusieurs endroits les fichiers vitaux d'un projet, d'une entreprise ; en cas de problème, la copie de sauvegarde est immédiatement disponible. L'augmentation des performances vient également

du fait qu'il est relativement facile d'augmenter les performances d'un système en réseau en ajoutant tout simplement un ou deux autres ordinateurs supplémentaires.

- **Réduction des coûts**

  En effet, les ordinateurs individuels coûtent bien moins cher que les gros systèmes centralisés (1000 fois moins environ), et ce pour une baisse des performances à peine un facteur 10. Les réseaux permettent également de réagir plus vite à certains événements (un appel d'offres par exemple), et donc faire gagner (ou économiser) de l'argent.

- **Accès à l'information**

  Avec les réseaux et en particulier Internet, il est très facile de s'informer sur toute sorte de sujets très rapidement. Ce dernier objectif joue en fait un rôle capital dans l'utilisation que les gens ont des réseaux. C'est peut-être même l'utilisation principale aujourd'hui.

- **Autres utilisations**

  Au delà de ces quatre points, il existe quelques autres objectifs aux réseaux, mais ces objectifs sont apparus récemment avec la démocratisation des réseaux et l'émergence d'Internet notamment, et ne correspondent pas véritablement à un besoin des professionnels. Les réseaux vont servir par exemple de support pour des jeux interactifs et autres divertissements, ainsi que de médium de communication. Ces dernières raisons ont des conséquences sociales relativement importantes car elles influencent énormément le comportement des gens.

I.2. 3 Classification des réseaux [3]

On peut classifier les réseaux selon la distance qui sépare les ordinateurs en trois catégories :

- LAN (local area network)
- MAN (metropolitan area network)
- WAN (wide area network)

I.2.3.1 LAN

LAN signifie *Local Area Network* (en français *Réseau Local*). Il s'agit d'un ensemble d'ordinateurs appartenant à une même organisation et reliés entre eux dans une petite aire géographique par un réseau, souvent à l'aide d'une même technologie (la plus répandue étant Ethernet).

I.2.3.2 MAN

Les MAN (*Metropolitan Area Network*) interconnectent plusieurs LAN géographiquement proches (au maximum quelques dizaines de kms) à des débits importants. Ainsi un MAN permet à deux noeuds distants de communiquer comme s'ils faisaient partie d'un même réseau local.

I.2.3.3 WAN

Un WAN (*Wide Area Network ou réseau étendu*) interconnecte plusieurs LAN à travers de grandes distances géographiques. Les WAN fonctionnent grâce à des routeurs qui permettent de choisir le trajet le plus approprié pour atteindre un noeud du réseau. Le plus connu des WAN est Internet.

I.2.4 Fonctionnement d'un réseau

Pour assurer le bon fonctionnement d'un réseau, il faut réunir les supports physiques nécessaires et prévoir une bonne architecture logicielle et une normalisation de celle-ci s'impose. Deux familles d'architectures ont vu le jour : La première s'appelle ***le Modèle OSI***. La seconde est ***l'architecture TCP/IP***. Détaillons chacune d'elles :

I.2.4.1 Le Modèle OSI (Open System Interconnexion)

Ce modèle est une norme définie par *ISO* (International Standarlization Organisation). Fondé sur un principe énoncé par *JULES CESAR* « Diviser pour mieux régner».Ce modèle est composé de sept couches (elles seront détaillées ultérieurement) : ***couche physique, liaison de données, réseau, session, transport, présentation et application.***

Ce modèle permet la communication entre plusieurs réseaux hétérogènes, cette communication passe donc par un ensemble de couches empilées:

Chaque couche a un rôle précis (conversion, routage, découpage, vérification etc.)

- Chaque couche dialogue avec la couche juste au dessus et celle juste au dessous : elle fournit des services à la couche dessus et utilise les servies de la couche dessous.
- Chaque couche encapsule les données venant de la couche dessus en y ajoutant ses propres informations avant de les passer à la couche dessous (opération inverse dans l'autre sens).
- Les données traversent les couches vers le bas quand elles sont envoyées et elles remontent les couches à la réception.

Voyons donc le rôle de chacune de ces couches :

1. Couche physique : C'est le support de transmission lui-même : un fil de suivre, une fibre optique etc.

2. Couche Liaison de données : En charge d'encodes (ou moduler) les données pour qu'elles soient transportables par le couche physique et fournit également la détection d'erreur de transmission et la synchronisation.

3. Couche réseau : En charge du transport, de l'adressage et du routage des paquets.

4. Couche Transport : En charge de la liaison d'un bout à l'autre. Cette couche s'occupe de la fragmentation des données en petits paquets et vérifie éventuellement qu'elles ont été transmises correctement.

5. Couche Session : En charge d'établir et maintenir des sessions (c'est-à-dire débuter le dialogue entre machines, vérifier que l'autre machine est prête à communiquer, s'identifier, etc..).

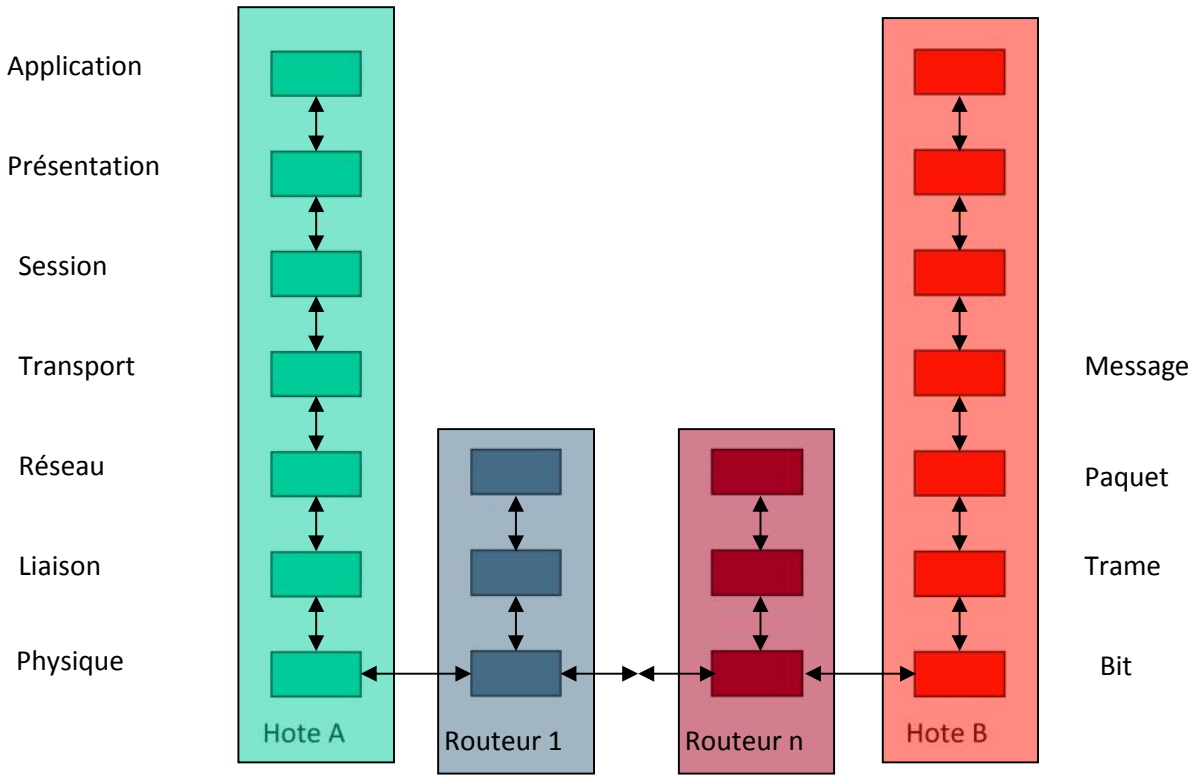


Fig.I.1 : Modèle de référence OSI

6. Couche Présentation : Encharge de la représentation des données (de telle sorte qu'elle soit indépendante de microprocesseur ou du système d'exploitation par exemple) et éventuellement du chiffrement.

7. Couche application : Représente la couche la plus élevée du modèle OSI, elle utilise les services de la couche présentation (indirectement des autres couches) pour exécuter une application spécifique .l'application peut être un échange des courriers, transfert des fichiers ou toute autre application réseau.

Au niveau de chaque couche un ensemble de protocole est intégré. Un protocole réseau est un langage que vont utiliser toutes les machines d'un réseau pour communiquer entre eux.

HTTP, FTP, TCP, IP, ICMP, et la totalité des autres protocoles entrent dans le modèle OSI.

I.2.4.2 Le Modèle TCP/IP [29]

Développé par l'armée américaine. Il désigne deux protocoles étroitement liés : un protocole de transport TCP (Transmission Control Protocol), et un protocole réseau IP (Internet Protocol).Le modèle TCP/IP est en fait une architecture réseau à quatre couches :

couche hôte réseau, Internet, couche transport et ***application*** **.**Détaillons chacune de ces couches :

1. Couche hôte réseau : Cette couche semble regrouper les couches : physique et liaison de données du modèle OSI. Elle permet à un hôte d'envoyer des paquets IP sur le réseau

2. Couche Internet : Cette couche est la clé de voûte de l'architecture IP.Cette couche réalise l'interconnexion des réseaux (hétérogènes). Son rôle est de permettre l'injection des paquets dans n' importe quel réseau et l'acheminement de ces paquets indépendamment les uns des autres jusqu'à destination, les paquets peuvent arriver dans le désordre, le contrôle de l'ordre est la tâche des couches supérieures. L'implémentation officielle de cette couche est le protocole IP**.**

3. Couche transport : Son rôle est le même que celui de la couche transport du modèle OSI. Officiellement, cette couche n'a que deux implémentations : le protocole *TCP* et le protocole *UDP* (User Datagram Protocol)**.**

4. Couche application : Contrairement au modèle OSI, c'est la couche immédiatement supérieure à la couche transport, tout simplement parce que les couches présentation et session sont apparues inutiles.

On s'est en effet aperçu avec l'usage que les logiciels réseaux n'utilisent que très rarement ces deux couches (Présentation et session), et finalement, le modèle OSI dépouillé de ces deux couches ressemble fortement au modèle TCP/IP.

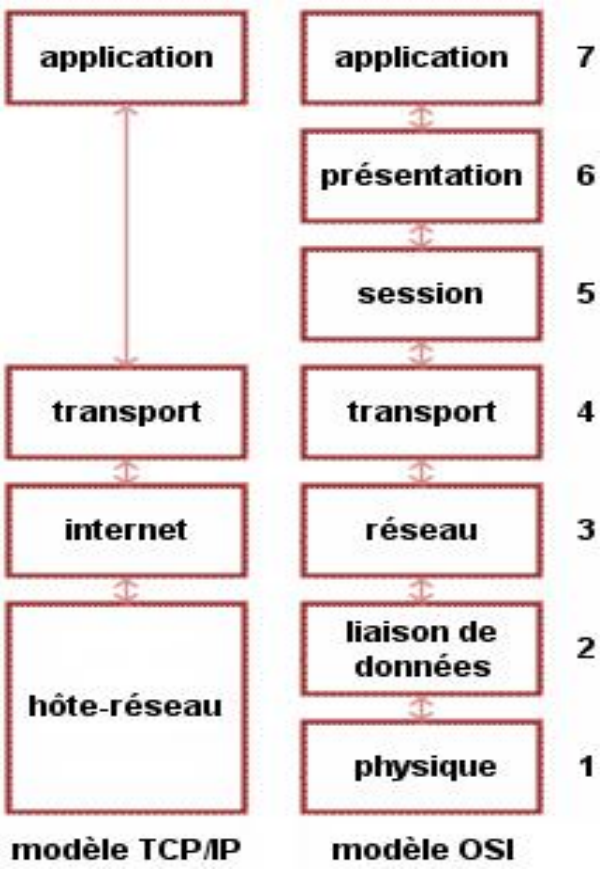


Fig.I .2 : Modèle TCP et OSI

TCP/IP est le protocole utilisé dans le réseau Internet. L'implémentation de ce modèle engendre malheureusement des vulnérabilités dus au failles des langages de programmation utilisés dans l'implémentation (exp : langage C). Ces vulnérabilités peuvent être exportées par les attaquant pour réaliser leurs attaques, d'où le problème de la sécurité réseau.

I.3 Sécurité réseau

I.3.1 Objectifs de la sécurité informatique [18]

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. Néanmoins, les points principaux sont les suivants :

- ✓ empêcher la divulgation non-autorisée de données.
- ✓ empêcher la modification non-autorisée de données.
- ✓ empêcher l'utilisation non-autorisée de ressources réseaux ou informatiques de façon générale.

I.3.2 Terminologie de la sécurité informatique

- ✓ **Intrus :** entité responsable d'une attaque de sécurité qui contourne les mécanismes de sécurité mis en place pour de différentes raisons :
  - Vérification de la sécurité d'un système.
  - Espionnage.
  - L'attirance de l'interdit.
  - Le désir d'argent (voler un système bancaire par exemple).
  - L'envie de nuire.
  - Pour apprendre.
  - Etc.

- ✓ **Vulnérabilité :** est une faille ou bug pouvant être utilisé pour obtenir un niveau d'accès illicite à une ressource ou à des privilèges supérieurs, et qui est exploité par une menace pour engendré une attaque.par exemple :
  - Utilisation des mots de passe non robustes.
  - Présence de comptes non protégés par mot de passe
  - Absence d'antivirus, pare-feu, …etc.

- ✓ **Menace** : ce sont des adversaires déterminés capables de monter une attaque exploitant une vulnérabilité on y retrouve :
  - Menaces passives : consistent à écouter ou copier des informations de manière illicite.

- Menaces actives : consistent à altérer des informations ou le bon fonctionnement d'un service.

✓ **Les attaques :** elles représentent les moyens d'exploiter une vulnérabilité. Il peut y avoir plusieurs attaques pour une même vulnérabilité mais toutes les vulnérabilités ne sont pas exploitables.

I.3.3 Critères fondamentaux de la sécurité [1] [2]

Les solutions de sécurité qui seront mises en place doivent contribuer à satisfaire les critères qui sont présentées dans la figure suivante :

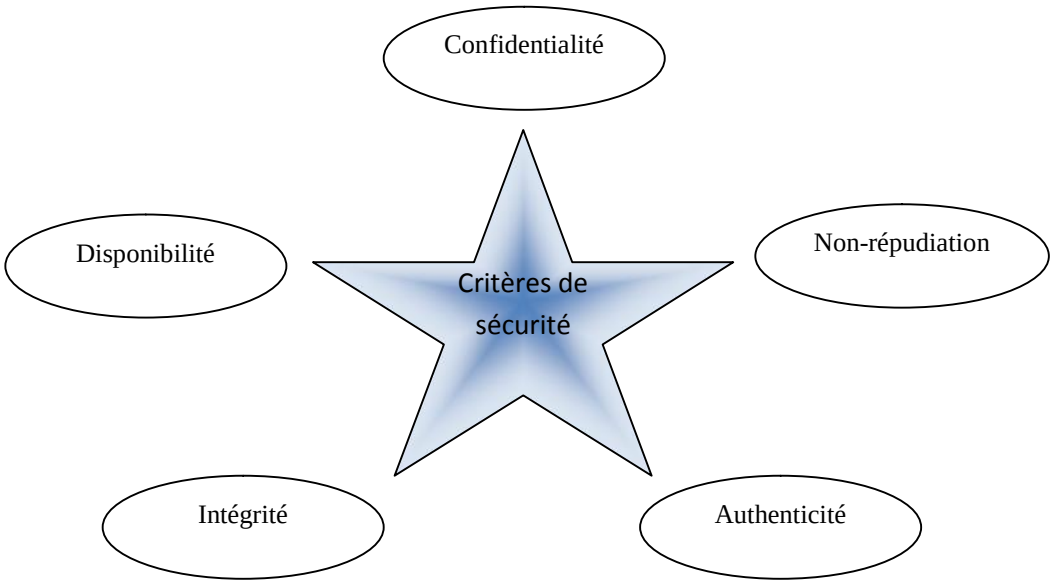


Fig. I.3 Critères de la sécurité

I .3.3.1 Confidentialité

La confidentialité définit l'absence de divulgation non autorisée de l'information. Une attaque contre la confidentialité par une personne malveillante consiste à tenter de récupérer des informations pour lesquelles elle ne possède pas d'autorisation, soit en tentant d y accéder sur le système, soit en écoutant les communications réseaux, soit de toute autre façon possible.

Il existe deux actions complémentaires permettant d'assurer la confidentialité des données :

- ✓ Limiter leurs accès par un mécanisme de contrôle d'accès ;
- ✓ Transformer les données par les procédures de chiffrement afin qu'elles deviennent inintelligible aux personnes ne possédant pas les moyens de les déchiffrer.

I.3.3.2 Authentification

L'authentification permet de vérifier l'identité annoncée et de s'assurer de la non-usurpation de l'identité d'une entité. Pour cela, l'entité devra produire une information spécifique telle que : un mot de passe ,un code, une empreinte biométrique, etc..

I.3.3.3 Intégrité

L'intégrité garantit que les données reçues sont exactement celles qui ont été émises par l'émetteur autorisé, il convient de se prémunir contre l'altération des données en ayant la certitude qu'elles n'ont pas été modifiées lors de leur stockage ou de leur transfert. Une attaque contre l'intégrité vise à introduire de fausses informations, ou à modifier ou détruire l'information existante. Tout comme pour la confidentialité, l'attaquant peut chercher à atteindre l'information directement sur le système ou à l'intercepter durant une communication.

I.3.3.4 Disponibilité

La disponibilité définit le fait que le système soit prêt à délivrer son service, la disponibilité d'une ressource est indissociable de son accessibilité : il ne suffit pas qu'elle soit disponible, elle doit être utilisable avec des temps de réponses acceptables.

Elle est mesurée sur la période de temps pendant laquelle le service offert est opérationnel.
Le volume potentiel de travail susceptible d'être pris en charge durant la période de disponibilité d'un service, détermine la capacité d'une ressource (serveur ou réseau).

Une attaque contre la disponibilité peut avoir deux origines. La première consiste à déjouer les politiques de sécurité et à exploiter une faute pour qu'elle produise une erreur affectant la délivrance du service. La seconde méthode consiste à engorger le système de demandes de

service valides afin d'occuper le système et rendre sa disponibilité faible ou inexistante pour l'utilisateur légitime.

I.3.3.5 Non répudiation

Non répudiation est le fait de ne pouvoir nier ou rejeter qu'un événement (action, transaction) a eu lieu. A ce critère de sécurité sont associées les notions d'imputabilité, de traçabilité et éventuellement d'auditabilité.

- ✓ L'imputabilité se définit par l'affectation certaine d'une entité à une action ou à un événement. L'imputabilité est réalisée par l'ensemble des exigences garantissant l'enregistrement des informations pertinentes sur l'individu agissant.
- ✓ La traçabilité est la fonction de sécurité qui comprend, le cas échéant, bien évidement, l'imputation, mais qui mémorise l'origine d'un message, d'un événement, d'une information ou d'une donnée. Elle permet, par exemple, de retrouver l'adresse à partir de la quelle ces données ont été envoyées.
- ✓ L'auditabilité se définit par la capacité d'un système à garantir la présence des informations nécessaires à une analyse ultérieure d'un événement dans le but de déterminer s'il ya effectivement eu violation de la sécurité.

I.3.4 Objectifs des pirates

Les motivations des hackers (Selon les individus) peuvent être multiples. On y retrouve : **[18]**

- ✓ Vérification de la sécurité d'un système.
- ✓ Espionnage.
- ✓ L'attirance de l'interdit.
- ✓ Le désir d'argent (voler un système bancaire par exemple).
- ✓ Le besoin de renommées (impressionner des amis).
- ✓ L'envie de nuire.
- ✓ Pour apprendre.

I.3.5 Types de pirates [20]

- ✓ **Les hackers :** ce sont "des passionnés des réseaux". Ils veulent comprendre le fonctionnement des systèmes informatiques et tester à la fois les capacités des

outils et leurs connaissances. En général, les hackers s'introduisent dans les systèmes par passion pour l'informatique et pas dans l'objectif de détruire ou de voler des données.

- ✓ **Les Crackers** : Sont des criminels informatiques dont leur but principal est de détruire ou voler des données, de mettre hors service des systèmes informatiques ou de 'kidnapper' un système informatique en vue de demander une rançon.

- ✓ **Les script-kiddies :** ce sont des pirates débutants qui agissent uniquement à l'aide des logiciels prêts à utiliser. Ils sont dans une logique de destruction ou de gain financier. Ils utilisent des outils qu'ils ne maîtrisent pas et dont ils ignorent le fonctionnement. La seule chose qu'ils font est l'exécution du logiciel et attendent le résultat.

I.3.6 Types de piratage

- ✓ **Le Hacking** : C'est l'accès non autorisé à un système ou un réseau informatique. Les pirates de **Hacking** attaquent essentiellement les réseaux informatiques (hackers, crackers et script-kiddies).

- ✓ **Le phreaking** : c'est le détournement de services de télécommunication par divers procédés, dans le but d'éviter les grosses factures de téléphone ou les oreilles indiscrètes. Un autre type de piratage téléphonique est l'utilisation détournée des téléphones cellulaires. Avec ce type de téléphones, aucune connexion physique n'est nécessaire, et il est facile d'écouter les conversations au moyen de scanners GSM et autres. Les téléphones cellulaires sont aussi facilement reprogrammables : les malfaiteurs peuvent ensuite les utiliser sans payer leurs communications, qui seront facturées aux véritables propriétaires.

- ✓ **Le « carding »** : Ces pirates s'attaquent principalement aux systèmes de cartes à puces (en particulier les cartes bancaires) pour en comprendre le fonctionnement et en exploiter les failles.

- ✓ **Les hacktivistes :** Se sont des hackers dont la motivation est principalement idéologique.

I.3.7 Les attaques [1][2][3]

Les informations ou les systèmes d'informations d'une entreprise peuvent subir des dommages de plusieurs façons : certains intentionnels (malveillants), d'autres par accidents. Ces événements seront appelés des « attaques ». Il existe quatre catégories principales d'attaque.

I.3.7.1 Attaque d'accès

Une attaque d'accès est une tentative d'accès à l'information par une personne non autorisée. Ce type d'attaque concerne la confidentialité de l'information.

❖ **Sniffing**

Cette attaque est utilisée par les pirates informatiques pour obtenir des mots de passe, ect. Grâce à un logiciel appelé renifleur de paquets (sniffer), on peut intercepter tout les paquets qui circulent sur un réseau même ceux qui ne nous sont pas destinés comme illustre la figure suivante :

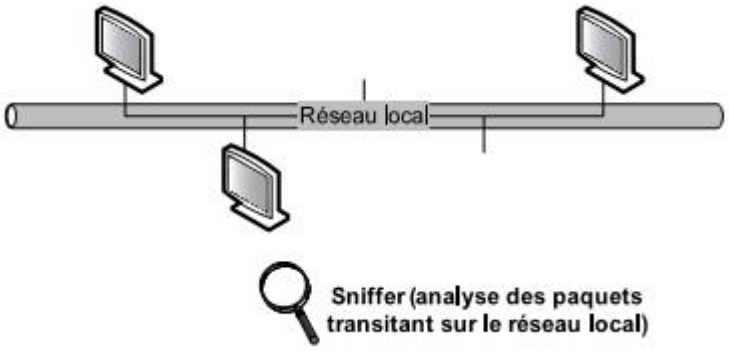


Fig. I.4 Ecoute sur un réseau

Grâce à des outils tels qu'Ethereal ou WinDump/TCPDump, le sniffer peut analyser tous les paquets IP ainsi que les protocoles contenus dans les données du paquet. Par exemple, un sniffer peut analyser un paquet Ethernet susceptible de contenir un paquet IP, qui lui-même pourrait contenir un paquet de type TCP, lequel à son tour pourrait contenir un paquet HTTP renfermant des données HTML.

Si une personne établit une session authentifiée sur un flux réseau non chiffré (Telnet,X11, etc.), son mot de passe transite en clair sur le réseau. De même, il est possible de connaître à tout moment les personnes connectées au réseau, les sessions de routage en cours, etc., par une analyse des paquets qui transitent sur le réseau et qui contiennent toutes les informations nécessaires à cette analyse.

❖ **Attaque de l'homme du milieu [4]**

Une attaque du type homme du milieu (« man-in-the-middle » ou MIM en anglais) est menée par un pirate qui s'arrange pour se placer entre deux hôtes légitimes.
Pour mettre en œuvre l'échange réseau approprié, il faut soit que la machine du pirate se trouve physiquement sur le chemin réseau emprunté par les flux de données, soit que le pirate réussisse à modifier le chemin réseau afin que sa machine devienne un des points de passage.

L'échange se présente sous l'une des trois formes suivantes :

- ✓ **Relais transparent** : La machine du pirate transforme les données à la volée. Elle veut rester la plus transparente possible et se comporte comme un routeur, conservant toutes les caractéristiques des paquets dont elle assure le transit, à l'exception du contenu. En termes d'adresses IP, A et B sont réellement en relation l'une avec l'autre.

Fig. I.5 Machine du pirate en tant que relais transparent

- ✓ **Relais applicatif** : La machine du pirate assure l'échange entre les deux machines A et B. A parle avec la machine du pirate, laquelle parle avec B. A et B n'échangent jamais de données directement. Cette méthode est nécessaire pour les attaques vers SSL.

Fig. I.6 Machine du pirate en tant que relais applicatif

- ✓ **Hijacking** : La machine du pirate utilise la session engagée entre les deux machines A et B à fin qu'elle soit en session avec la machine B. A perd la session avec B, et la machine du pirate continue la session engagée par A sur B.

Fig. I.7 Machine du pirate en tant que hijacker

❖ **Exploitation de la confiance [4]**

Une attaque par exploitation de la confiance a pour but de compromettre un hôte de confiance et de l'utiliser ensuite pour lancer des attaques sur d'autres hôtes du réseau. Dans cette attaque, l'intrus utilise les privilèges d'une autre entité de confiance pour pénétrer dans un système sécurisé.

❖ **Craquage de mots de passe**

Le craquage consiste à faire de nombreux essais jusqu'à trouver le bon mot de passe.
Il existe deux grandes méthodes :

- L'utilisation de dictionnaires : le mot testé est pris dans une liste prédéfinie contenant les mots de passe les plus courants et aussi des variantes de ceux-ci. Les dictionnaires actuels contiennent dans les 50 000 mots et sont capables de faire une grande partie des variantes.
- La méthode brute : toutes les possibilités sont faites dans l'ordre jusqu'à trouver la bonne solution.

I.3.7.2 Attaque de modification

Une attaque de type « modification » consiste, pour un attaquant à tenter de modifier des informations. Ce type d'attaque est dirigé contre l'intégrité de l'information.

❖ **Virus**

Un virus est un segment de programme qui, lorsqu'il s'exécute, sera produit en s'adjoignant à un autre programme (du système ou d'une application), et qui devient ainsi un cheval de Troie. Puis le virus peut en suite se propager à d'autres ordinateurs (via un réseau) à l'aide du programme légitime sur lequel il s'est greffé. Il peut également avoir comme effets de nuire en perturbant plus ou moins gravement le fonctionnement de l'ordinateur infecté. PsybOt, découvert en 2009, est considéré comme étant le seul virus informatique ayant la capacité d'infecter les routeurs et modems haut-débit.

❖ **Vers (wormes)**

Un ver est un programme autonome qui se reproduit et se propage à l'insu des utilisateurs. Contrairement aux virus, un ver n'a pas besoin d'un logiciel hôte pour se dupliquer.
Les différentes phases de l'attaque d'un ver sont les suivantes :

- **Activation de la vulnérabilité** : un ver s'installe en exploitant les vulnérabilités connues d'un système, comme les utilisateurs naïfs qui exécutent sans vérification un fichier exécutable joint à un courriel.
- **Mécanisme de propagation :** l'accès à l'hôte étant acquis, le ver s'y reproduit, puis sélectionne d'autres cibles.
- **Charge** : une fois l'hôte infecté par un ver, l'assaillant peut y accéder, bien souvent en tant qu'utilisateur privilégié. Les assaillants peuvent utiliser une faille locale pour augmenter leur niveau de privilèges jusqu'à celui d'administrateur.

Le ver Blaster avait pour but de lancer une attaque par déni de service sur le serveur de mises à jour de Microsoft.

❖ **Chevaux de Troie**

Les chevaux de Troie sont des programmes informatiques cachés dans d'autres programmes. Ce nom vient de la légende grecque de la prise de Troie à l'aide d'un cheval en bois rempli de soldats qui attaquèrent la ville une fois à l'intérieur.

En général, le but d'un cheval de Troie est de créer une porte dérobée (backdoor) pour qu'un pirate informatique puisse ensuite accéder facilement à l'ordinateur ou à un réseau informatique. Il peut aussi voler des mots de passe, copier des données, exécuter des actions nuisibles.

❖ **Porte dérobée**

Lorsqu'un pirate informatique arrive à accéder à un serveur à l'aide d'une des techniques présentées dans cette section, il souhaiterait y retourner sans avoir tout recommencer. Pour cela, il laisse donc des portes dérobées (backdoor) qui lui permettra de reprendre facilement le contrôle du système informatique.

Il existe différents types de portes dérobées :

- Création d'un nouveau compte administrateur avec un mot de passe choisi par le pirate.
- Création de compte ftp
- Modification des règles du pare-feu pour qu'il accepte des connexions externes.

Dans tous les cas, l'administrateur per le contrôle total du système informatique. Le pirate peut alors récupérer les données qu'il souhaite, voler des mots de passe ou même détruire des données.

I.3.7.3 Attaque par saturation (Denis de service)

Les attaques par saturation sont des attaques informatiques qui consistent à envoyer des milliers de messages depuis des dizaines d'ordinateurs, dans le but de submerger les serveurs.

Cette technique de piratage est assez simple à réaliser, elle est jugée comme de la pure malveillance, et ne fait que bloquer l'accès aux sites, sans en altérer le contenu.

❖ **Le TCP-SYN flooding**

Le TCP-SYN flooding est une variante du flooding qui s'appuie sur une faille du protocole TCP. En effet, on envoie un grand nombre de demande de connexions au serveur (SYN) à partir de plusieurs machines. Le serveur va envoyer un grand nombre de paquet SYN-ACK et attendre en réponse un paquet ACK qui ne viendra jamais. Si on envoie les paquets plus vite que le timeout des « demi-connexions » (connexions autorisées mais non terminé), le serveur sature et finit par se déconnecter.

❖ **Smurf**

Le smurf est une attaque qui s'appuie sur le ping et les serveurs de broadcast. On falsifie d'abord son adresse IP pour se faire passer pour la machine cible. On envoie alors un ping sur un serveur de broadcast. Il le fera suivre à toutes les machines qui sont connectées qui renverront chacune un « pong » au serveur qui fera suivre à la machine cible. Celle-ci sera alors inondée sous les paquets et finira par se déconnecter.

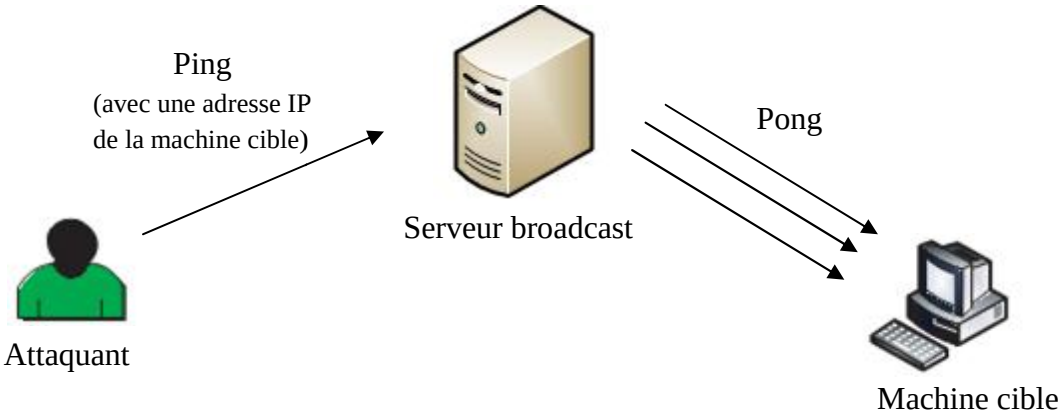


Fig. I.8 Attaque smurf

❖ **Débordement de tampon**

Cette attaque se base sur une faille du protocole IP. On envoie à la machine cible des données d'une taille supérieure à la capacité d'un paquet. Celui-ci sera alors fractionné pour l'envoi et rassemblé par la machine cible. A ce moment, il y aura débordement des variables internes.

Suite à ce débordement la machine se bloque ou se redémarre.

❖ **Ping de la mort**

Le ping de la mort consiste à envoyer une suite de fragments d'une requête de type écho ICMP. Une fois à nouveau assemblés par la pile IP/TCP du système cible, ces fragments forment un paquet d'une taille supérieure à la taille maximale autorisée (65 507 octets) et peuvent faire déborder les variables internes, provoquant un comportement anormal du système. Plus aucun système récent n'est vulnérable à ce type d'attaque.

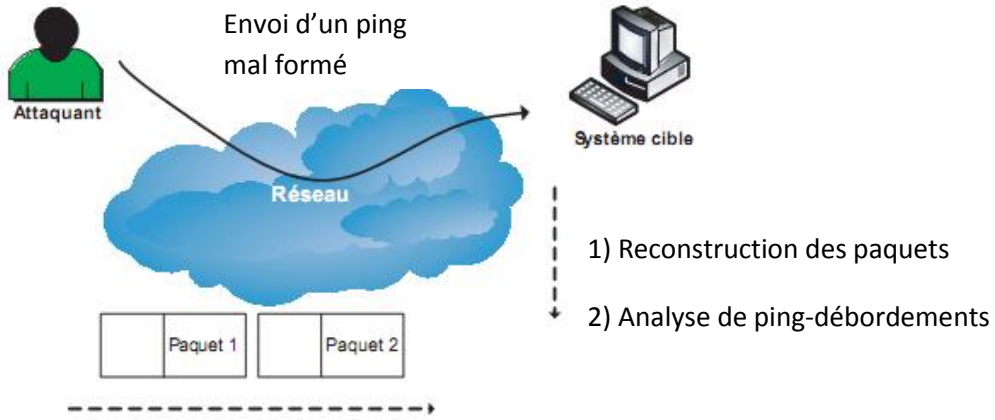


Fig. I .9 L'attaque ping de la mort

I.3.7.4 Attaques de répudiation

La répudiation est une attaque contre la responsabilité. Autrement dit, la répudiation consiste à tenter de donner de fausses informations ou de nier qu'un événement ou une transaction se soient réellement passé.

❖ **IP spoofing**

Cette attaque va se dérouler en plusieurs étapes :

- ✓ Trouver la machine de confiance (son adresse IP)
- ✓ Mettre hors service cette machine de confiance (avec un SYN Flooding par exemple) pour éviter qu'elle ne réponde aux paquets éventuellement envoyés par le serveur cible.
- ✓ Prédire les numéros de séquence TCP du serveur cible (un numéro de séquence initial est généré à chaque nouvelle connexion TCP).

- ✓ Lancer l'attaque qui consiste à créer une connexion TCP sur le serveur cible. Pour cela, l'attaquant va forger un paquet TCP avec le flag SYN et l'adresse IP source de la machine de confiance. Le serveur cible va répondre par un paquet TCP avec les flags SYN-ACK. L'attaquant, qui aura prédis le numéro de séquence TCP, pourra forger un paquet TCP avec le flag ACK et le bon numéro d'acquittement .Une connexion TCP est alors établie au niveau du serveur cible. L'attaquant n'a plus qu'à envoyer un paquet TCP avec le flag PSH et accéder librement au serveur cible.

Cette attaque usurpe l'identité de la victime et prend le contrôle du serveur.

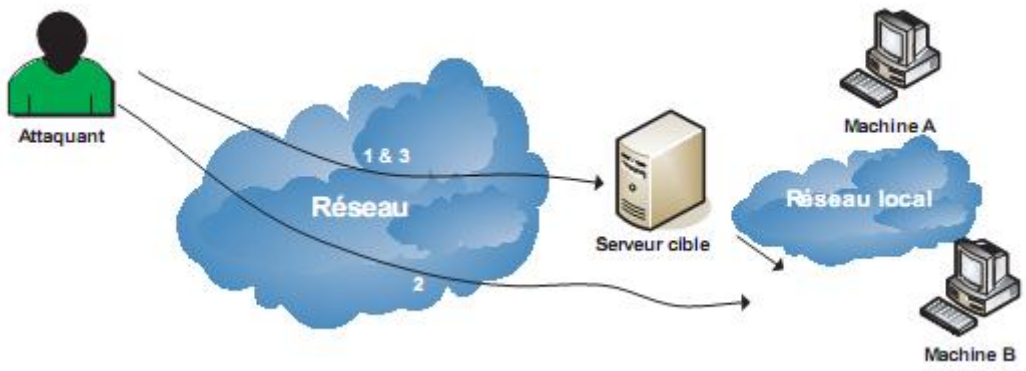


Fig.I.10 Attaque IP spoofing

❖ **Spoofing ARP**

Comme son nom l'indique, l'attaque ARP spoofing s'appuie sur le protocole ARP(Address Resolution Protocol), qui implémente le mécanisme de résolution d'une adresse IP (32 bits) en une adresse MAC (48 bits) pour rediriger le trafic réseau d'un ou plusieurs systèmes vers le système pirate. Lorsqu'un système désire communiquer avec ses voisins sur un même réseau, des messages ARP sont envoyés afin de connaître l'adresse MAC des systèmes voisins et d'établir ainsi une communication avec un système donné. Sachant que chaque système possède localement une table de correspondance entre les adresses IP et MAC des systèmes voisins, la faiblesse d'authentification du protocole ARP permet à un

système pirate d'envoyer des paquets ARP réponse au système cible indiquant que la nouvelle adresse MAC correspondant à l'adresse IP d'une passerelle est la sienne. Le système du pirate reçoit donc tout le trafic à destination de la passerelle. Il lui suffit d'écouter ou de modifier passivement le trafic et de router ensuite les paquets vers leur véritable destination, comme le montre la figure Fig. I.11**.**

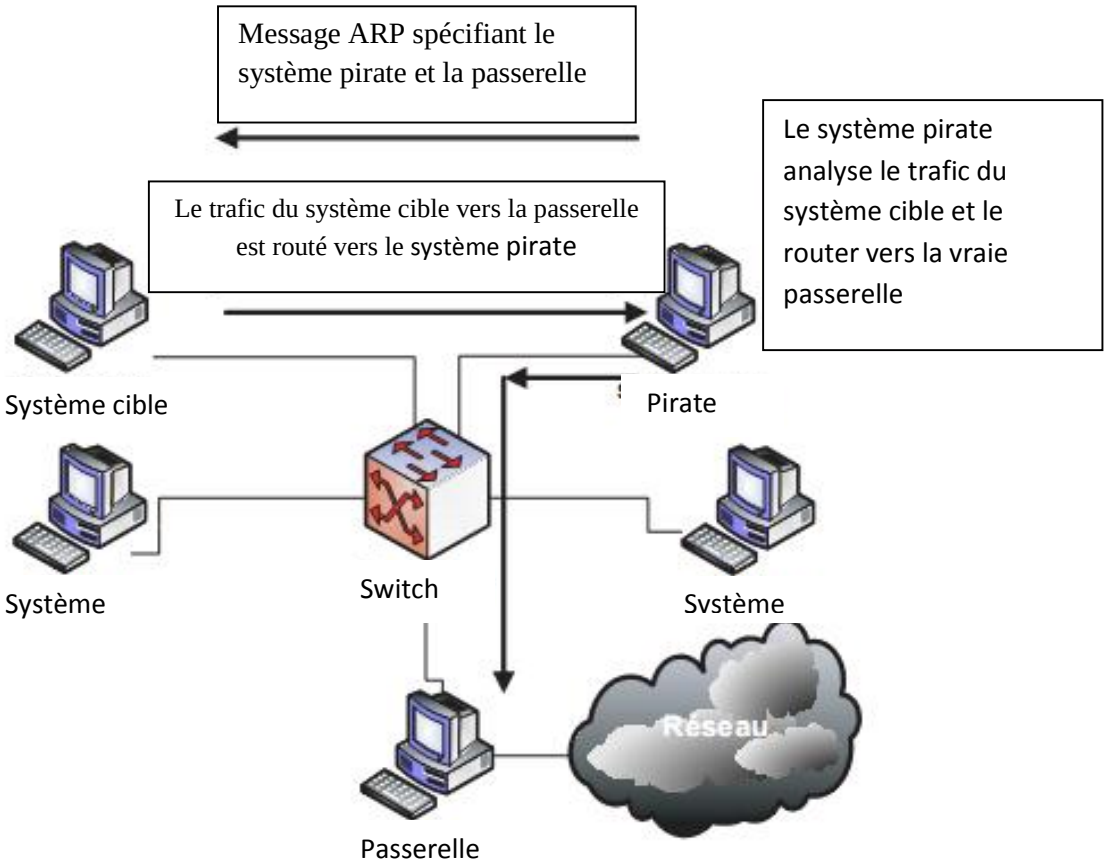


Fig. I.11 Attaque ARP spoofing

I.3.8 Politique de sécurité réseau [19] [3]

La politique de sécurité réseau est un document générique qui définit des règles à suivre pour les accès aux réseaux informatiques, détermine comment les politiques sont appliquées et présente une partie de l'architecture de base de l'environnement de sécurité du réseau. Ces documents à caractère non technique donnent aux responsables de l'entreprise les axes à suivre.

Voici quelques types de documents :

I.3.8.1 Guides

Il s'agit de documents détaillant comment implémenté les politiques de sécurité. Ils sont considérés comme des documents complémentaires aux politiques.

I.3.8.2 Standards

Il s'agit de documents de standardisation de normes et méthodes émanant d'organismes internationaux tels que l'ISO (International Standardization Organization), l'IETF (Internet Engineering Task Force), l'IEEE (Institute of Electrical andElectronics Engineers), etc.

I.3.8.3 Procédures

Il s'agit de documents à caractère opérationnel et technique, qui décrivent de manière claire et précise les étapes à suivre pour atteindre un objectif de sécurité donné. Une politique de sécurité réseau est donc indépendante de tout produit ou technologie. Elle est avant tout constituée d'une suite de règles et de principes répondant aux besoins de sécurité de l'entreprise. Les documents de sécurité peuvent être représentés par une structure pyramidale représentant le positionnement respectif de chaque document.

I.3.9 Mécanismes de Sécurité [21][5][1]

Nous avons constaté que les attaquants disposent de plusieurs moyens pour réussir leurs attaques. La disponibilité des outils d'attaques et la richesse des sources d'informations accentuent le risque des intrusions. Par conséquent, les administrateurs sécurisent de plus en plus leurs systèmes informatiques. Ils s'appuient sur diverses solutions comme :

I.3.9.1 Cryptage, Signature électronique et Certificat

- **Cryptage**

  Le cryptage est un procédé de cryptographie grâce auquel on souhaite rendre la compréhension d'un document impossible à toute personne qui n'a pas la clé de chiffrement ou de déchiffrement.

- **Signature numérique** (**signature électronique**)

La signature est un mécanisme permettant de garantir l'intégrité d'un document électronique et d'en authentifier l'auteur, par analogie avec la signature manuscrite d'un document papier. (Ajout d'informations cryptées à une unité de données afin de prouver la source et l'intégrité de cette unité de données).

- **Certificat [6]**

  Un certificat est un document contenant une affirmation certifiée, un certificat est une sorte de pièce d'identité par exemple le passeport ou l'extrait de naissance. Un certificat numérique est une information attachée à une clé publique, et qui permet de vérifier que cette clé est authentique, ou valide.

  Un certificat numérique comporte trois éléments:

  - ✓ Une clé publique
  - ✓ Une information de certification ("l'identité" de l'utilisateur, comme son nom, son adresse e-mail, etc.).
  - ✓ Une ou plusieurs signatures numériques.

I.3.9.2 Notarisation [22]

La notarisation est la certification des différentes étapes de l'évolution d'un document électronique en vue de :

- ✓ Permettre de garantir le contenu, l'origine, la date et la destination d'un message électronique lors d'un échange entre deux machines.
- ✓ Archiver de façon sécurisée des documents numériques

La notarisation électronique permet la vérification et l'archivage des preuves d'échanges et d'archivages électroniques par un tiers de confiance agréé (à la manière d'un notaire). Cette technique améliore la sécurité des échanges et de l'archivage électronique.

I.3.9.3 Horodatage (timestoping)

L'horodatage protège de toute contestation concernant le contenu d'un fichier et sa date d'émission ou de réception.

L'utilisation de l'horodatage est comme preuve de :

- ✓ Non altération : d'une archive, d'un contrat...

- ✓ Respect de délais légaux : de rétractation, de prise d'effet d'un contrat (assurance, crédit, abonnement, contrat de formation)…
- ✓ Antériorité : dépôt de candidature à un appel d'offre, dépôt de brevet…
- ✓ Accusé de réception opposable : mise en demeure, résiliation/reconduction d'un contrat...
- ✓ Traçabilité des actions : exigences réglementaires type Bâle 2, SOX…
- ✓ Facture électronique : en raison de l'obligation de pouvoir garantir l'authenticité.

I.3.9.4 Parfeu (filtrage) [5]

Le parfeu est un élément du réseau informatique, logiciel et/ou matériel, qui a pour fonction de sécuriser un réseau domestique ou professionnel en définissant les types de communication autorisés ou interdits. Le principe de base de la sécurité d'un réseau, intranet, repose sur l'installation d'un ou plusieurs firewalls qui peuvent être logiciels ou matériels. L'idée principale d'un firewall est la connexion sécurisée du réseau interne avec l'Internet (ou un autre réseau local non sûr).

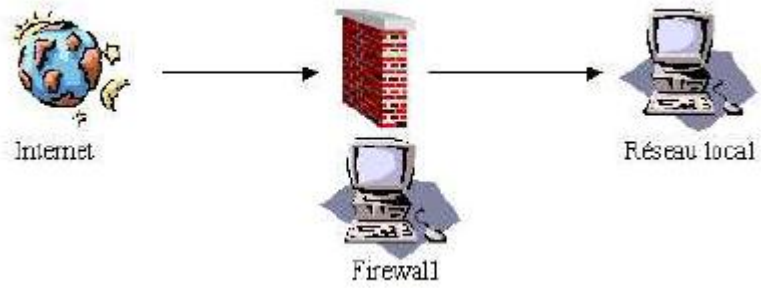


Fig. I.12 Placement d'un firewall [5]

Un firewall permet donc de délimiter les environnements publics et privés afin de protéger son réseau. On arrive à mieux gérer les flux entrants et sortants et ainsi séparer son réseau intranet du réseau internet.

Un pare-feu fonctionne sur des triplets (client/service/ condition). Ainsi, chaque « client » du firewall a accès à certains services sous certaines conditions. Le pare-feu peut bloquer un trafic particulier, ou en laisser passer un autre. On peut donc protéger le réseau d'intrusions non autorisées, tout en permettant aux employés un accès aux services Internet tels que l'e-

mail, le web ou autre. Dans la pratique, on peut configurer un firewall de manière à le rendre plus ou moins stricte.

I.3.9.4.1 Types de firewall

- **Firewalls bridge**

Les firewalls bridge agissent comme de vrais câbles réseau avec la fonction de filtrage en plus, d'où leur appellation de firewall. Ils ne disposent pas d'adresse IP sur leurs interfaces, et ne font que transférer les paquets d'une interface a une autre en leur appliquant les règles prédéfinies. Cette absence d'adresse IP est particulièrement utile, car cela signifie que le firewall est indétectable sur le réseau. En effet, quand une requête ARP est émise sur le câble réseau, le firewall ne pourra pas répondre. Ses adresses Mac ne circuleront jamais sur le réseau, et comme il ne fait que « transmettre » les paquets, il sera totalement invisible sur le réseau. Cela rend impossible toute attaque dirigée directement contre le firewall, étant donné qu'aucun paquet ne sera traité par ce dernier comme étant sa propre destination. Donc, la seule façon de le contourner est de passer à travers ses règles de « drop ». Ces firewalls se trouvent typiquement sur les Switchs.

- **Firewalls logiciels**

Les pare-feux sont présents à la fois dans les serveurs et les machines, on peut les classer en plusieurs catégories

- ✓ **Firewalls personnels** Ils sont pour la plupart commerciaux et ont pour but de sécuriser un ordinateur particulier. Souvent payants, ils peuvent être contraignants et quelque fois très peu sécurisés. En effet, pour rester accessible à l'utilisateur final, ils s'orientent plus vers la simplicité d'utilisation, et donc mettent de côté l'aspect technique.
- ✓ **Firewalls plus « sûre »** Ils se trouvent généralement sous Linux, car ce Système d'Exploitation offre une sécurité réseau plus élevée et aussi un contrôle plus précis. Ils ont généralement le même comportement que les firewalls matériels des routeurs, à la seule différence qu'ils sont configurables à la main.

Ces firewalls logiciels ont néanmoins une grande faille : ils n'utilisent pas la couche bas réseau. Pour récupérer des paquets qui auraient été normalement « droppés », il suffit de passer outre le noyau (en utilisant une librairie particulière).

Néanmoins, cette faille signifie qu'on s'est déjà introduit sur l'ordinateur en question ; ce qui induit une intrusion dans le réseau, où une prise de contrôle physique de l'ordinateur, donc qui est synonyme de faille.

- **Firewalls matériels**

Ils se trouvent souvent sur des routeurs achetés dans le commerce par de grands constructeurs comme Cisco ou Nortel.

Intégrés directement dans la machine, ils font office de « boite noire », et ont une intégration parfaite avec le matériel. Leur configuration est souvent relativement ardue, mais leur avantage est que leur interaction avec les autres fonctionnalités du routeur est simplifiée par leur présence sur le même équipement réseau. Souvent relativement peu flexibles en terme de configuration, ils sont aussi peu vulnérables aux attaques, car présent dans la « boite noire » .

De plus, étant souvent très liés au matériel, l'accès à leur code est assez difficile, et le constructeur a eu toute latitude pour produire des systèmes de codes « signés » afin d'authentifier le logiciel (système RSA ou assimilés). Ce système n'est implanté que dans les firewalls haut de gamme, car cela évite un remplacement du logiciel par un autre non produit par le fabricant, ou toute modification de ce dernier, rendant ainsi le firewall très sûr.

Son administration est souvent plus aisée que les firewalls bridges, les grandes marques de routeurs utilisant cet argument comme argument de vente. Leur niveau de sécurité est de plus très bon.

I.10.4.2 Types de filtrage

- **Firewalls à filtrage de paquets**

Ce type de firewall travaille sur la composition même des paquets réseaux réseau. Ils analysent les paquets entrants/sortants suivant leurs types, leurs adresses source et destination ainsi que les ports utilisés, chaque paquet d'informations entrant ou sortant est accepté ou rejeté selon des règles établies par l'utilisateur.

- **Firewalls Proxy**

Les firewalls proxy (ou firewalls applicatifs) ont un mode de fonctionnement différent des firewalls à filtrage de paquets. Ils masquent les ressources internes du réseau, ce par feu empêche l'extérieur de connaître les adresses internes du réseau.

Chaque application passe alors par le firewall proxy et envoie sa requête non pas au serveur qu'elle désire atteindre mais au firewall qui la retransmettra. Inversement, les communications émises depuis Internet à destination des systèmes internes ne les atteignent pas directement mais sont préalablement traitées par le firewall.

Il est à noter que ces firewalls sont des gros consommateurs de ressources informatiques.

❖ **Les Proxy « SOCKS »**

Ce type de firewall ne travaille pas sur les flux applicatifs mais rétablisse, à chaque connexion, la connexion vers l'extérieur. Ce type de firewall est peu utilisé désormais. Il est à noter que ces firewalls ne réalisent pas d'authentification des utilisateurs même s'ils ont la capacité d'enregistrer les coordonnées de l'utilisateur qui a demandé la connexion.

I.3.9.5 Antivirus [21]

Un antivirus est un logiciel informatique destiné à identifier et à effacer des logiciels malveillants, également appelés virus, Chevaux de Troie ou vers selon les formes.

Fonctionnement d'un antivirus

L'antivirus analyse les fichiers entrants (fichiers téléchargés ou courriers électroniques) et, périodiquement, la mémoire vive de l'ordinateur et les périphériques de stockage comme les disques durs, internes ou externes, les clés USB et les cartes à mémoire Flash.

Le logiciel antivirus peut procéder de deux manières différentes :

- ✓ Il analyse les fichiers, en comparant leur contenu aux virus d'un dictionnaire de virus connus. Les virus détectés sont signalés selon la méthode définie par l'utilisateur.
- ✓ Il surveille les processus suspects sur un hôte susceptible d'être infecté. Cette surveillance comprend la saisie de données, la surveillance des ports et d'autres méthodes.

I.3.9.6 Sonde et Pot de miel [7]

❖ **Sonde**

Une sonde est un point de collecte de données sur Internet. La plupart du temps, elle repose sur la récupération des fichiers de journalisation des routeurs sur Internet ou des pare-feux. L'approche par sonde est très intéressante car elle a mis en évidence le caractère excessivement malin de plusieurs programmes sur Internet.

❖ **Pot de miel**

Un pot de miel se définit comme un système informatique connecté à un réseau, Volontairement vulnérable à une ou plusieurs failles et visant à attirer les attaquants afin d'étudier leur comportement. En théorie, aucune activité en provenance ou à destination de ce système ne devrait être enregistrée. Dans le cas contraire, il s'agit au mieux d'une erreur accidentelle, au pire d'une tentative d'attaque intentionnelle.

I.3.9.7 Mot de passe [18]

Une personne peut être authentifiée par une combinaison d'une identification et d'un mot de passe, (code secret personnel).Le mot de passe doit posséder certaines caractéristiques qui sont : non trivial, difficile à deviner, régulièrement modifié. Cependant si l'attaquant accède au fichier de mot de passe, il pourra s'introduire dans le système sécurisé.

Choix d'un mot de passe [18]

Choisir un bon mot de passe n'est pas si évident que ça en a l'air. Il faut respecter quelques règles :

- ✓ Ne jamais choisir un mot du langage courant. Des logiciels spéciaux de type dictionary cracking sont spécialisés dans ce domaine.
- ✓ Ne jamais prendre un mot qui est proche de vous : Votre prénom, le nom de jeune fille de votre femme, le nom du chien, des enfants, de votre hobby préféré...
- ✓ Ne jamais prendre un mot inférieur à 6 lettres. Des logiciels spéciaux de type brute force cracking sont spécialisés dans ce domaine.

- ✓ Un mot de passe ne doit jamais être écrit quelque part. La première chose que fait un pirate, est de fouiller dans vos affaires : Regarder dans votre agenda, sous l'écran, sous le clavier, dans votre poubelle, rechercher un fichier du type "mdp.txt" dans votre disque dur, etc.

I.3.9.8 Systèmes de détection d'intrusions [18]

Système de détection d'intrusions (IDS) est un équipement permettant de surveiller l'activité d'un réseau ou d'un hôte donné, afin de détecter toute tentative d'intrusion et éventuellement de réagir à cette tentative. Les IDS peuvent se classer selon deux catégories majeures selon qu'ils s'attachent à surveiller le trafic réseau ou l'activité des machines.

I.3.9.8.1 IDS Réseau

Ces outils analysent le trafic réseau ; ils comportent généralement une sonde qui " écoute " sur le segment de réseau à surveiller et un moteur qui réalise l'analyse du trafic afin de détecter les signatures d'attaques ou les divergences face au modèle de référence. Les IDS Réseau à base de signatures sont confrontés actuellement à deux problèmes majeurs qui sont le développement de l'utilisation du cryptage et le développement des réseaux commutés. En effet, il est d'une part plus difficile " d'écouter " sur les réseaux commutés et le cryptage rend l'analyse du contenu des paquets presque impossible. La plupart des IDS sont aussi dits IDS inline car ils analysent le flux en temps réel. Pour cette raison, la question des performances est très importante car de tels IDS doivent être de plus en plus performants afin d'analyser les volumes de plus en plus importants pouvant transiter sur les réseaux.

I.3.9.8.2 IDS Système

Les IDS Systèmes analysent le fonctionnement ou l'état des machines sur lesquelles ils sont installés afin de détecter les attaques. Ils sont très dépendants du système sur lequel ils sont installés. Il faut donc des outils spécifiques en fonction des systèmes déployés. Ces IDS peuvent s'appuyer sur des fonctionnalités d'audit propres au système d'exploitation ou non pour vérifier l'intégrité du système et générer des alertes. Il faut cependant noter qu'ils sont incapables de détecter les attaques affectant les couches réseaux de la machine ; typiquement les Déni de service comme SYN FLOOD ou autre.

I.3.9.9 Fichiers historiques

Fichier historique permet d'enregistrer tout ou une partie des actions effectuées sur le système. L'analyse de ses informations permet de détecter d'éventuelles intrusions. Les systèmes d'exploitation génèrent des fichiers historiques, certaines applications aussi. Les différents évènements du système sont enregistrés dans un journal, qui devra être analysé fréquemment, voire en permanence. Sur les réseaux, il est indispensable de disposer d'une base de temps commune pour estampiller les évènements.

Les types d'informations à collecter sur les systèmes pour permettre la détection d'intrusions : On y trouve les informations sur les accès au système (qui a accédé, quand et comment), les informations sur l'usage fait du système (utilisation du processeur, de la mémoire ou des entrées/sorties) et les informations sur l'usage fait des fichiers. Le fichier historique doit également permettre d'obtenir des informations relatives à chaque application (le lancement ou l'arrêt des différents modules, les variables d'entrée et de sortie et les différentes commandes exécutées), Les informations sur les violations éventuelles de la sécurité (tentatives de commandes non autorisées) ainsi que les informations statistiques sur le système seront elles aussi nécessaires.

I.3.9.10 VPN

Un réseau privé virtuel (VPN) est l'extension d'un réseau privé qui inclut les liaisons avec des réseaux partagés ou publics tels qu'Internet. Avec un réseau VPN, il est possible de transmettre des données entre deux ordinateurs par le biais d'un réseau partagé ou public en émulant une liaison privée point à point.

Ces réseaux offrent deux avantages majeurs :

- ✓ De hautes performances en termes de bande passante, autrement dit des communications à très haut débit et de très grande qualité.
- ✓ La sécurité et la confidentialité des données.

I.3.9.10 .1 Fonctionnalités des VPN

Il existe 3 types standards d'utilisation des Vpn :

❖ **Vpn d'accès [27]**

Le Vpn d'accès est utilisé pour permettre à des utilisateurs itinérants d'accéder au réseau privé. L'utilisateur se sert d'une connexion Internet pour établir la connexion Vpn. Il existe deux cas:

- L'utilisateur demande au fournisseur d'accès de lui établir une connexion cryptée vers le serveur distant : il communique avec le Nas (Network Access Server) du fournisseur d'accès et c'est le Nas qui établit la connexion cryptée.
- L'utilisateur possède son propre logiciel client pour le Vpn auquel cas il établit directement la communication de manière cryptée vers le réseau de l'entreprise.

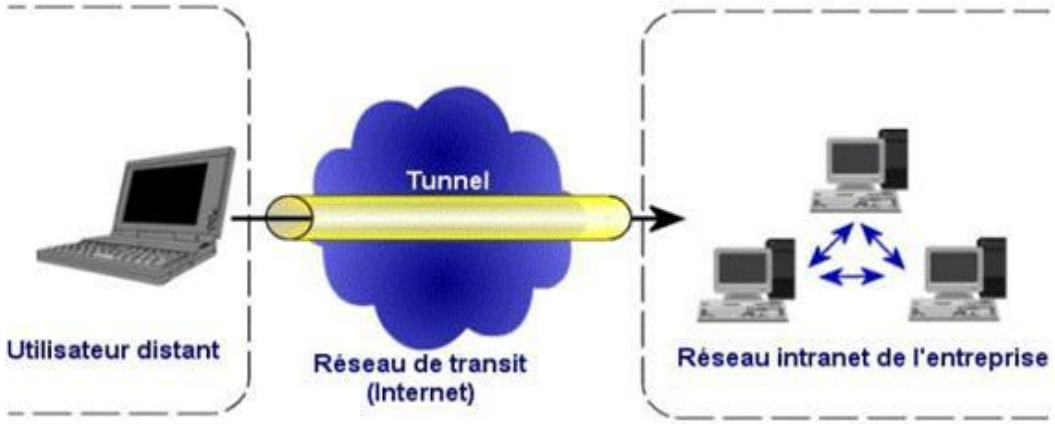


Fig. I.13 VPN d'accès

❖ **Intranet Vpn**

L'intranet Vpn est utilisé pour relier au moins deux intranets entre eux. Ce type de réseau est particulièrement utile au sein d'une entreprise possédant plusieurs sites distants.

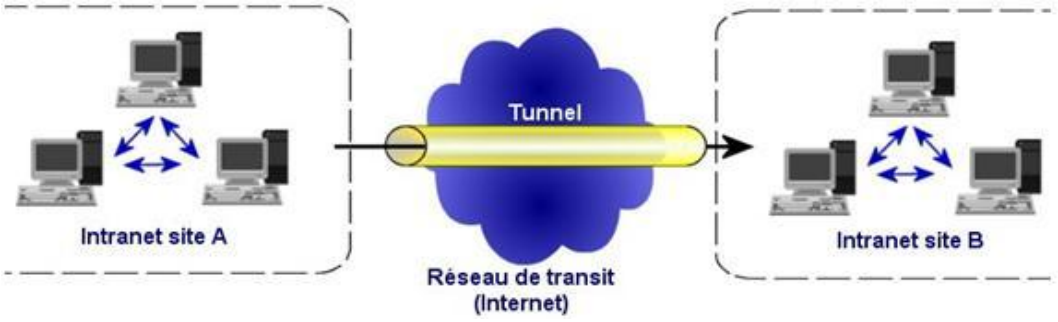


Fig. I.14 VPN intranet

❖ **L'extranet VPN**

Une entreprise peut utiliser le VPN pour communiquer avec ses clients et ses partenaires. Elle ouvre alors son réseau local à ces derniers. Dans Ce cadre, il est fondamental que l'administrateur du VPN puisse tracer les clients sur le réseau et gérer les droits de chacun sur celui-ci.

I.4 Conclusion

Au niveau de ce chapitre, nous avant constater que la sécurité réseau et un point essentielle dans l'environnement informatique à savoir son efficacité de protection contre des attaques malveillantes et ses diverses techniques pour les combattre. Nous détaillons dans le chapitre II l'une des ses technique la plus utiliser dans ce contexte.

Chapitre II

Introduction à la cryptographie

II.1 Introduction

L'idée de coder un message dans le but de le rendre inintelligible à toute tierce personne ne date pas aujourd'hui .les « messages secret » ont joué un rôle important dans tous les conflits depuis que l'homme sait écrire, et sont habituellement associés aux guerres et aux agents secrets.

Le but de ce chapitre, est de présenter les fondements et le fonctionnement de la cryptographie. Il intéressera le lecteur qui connaît peu ou pas le domaine et qui souhaiterait comprendre le fonctionnement et les mécanismes mis en œuvre en cryptographie.

II.2 Cryptologie [8] [9]

La cryptologie est la science du secret, ne peut être vraiment considéré comme une science que depuis peut de temps. Cette science englobe la cryptographie, la cryptanalyse et la stéganographie, comme elle a été décrite dans cette figure en dessous.

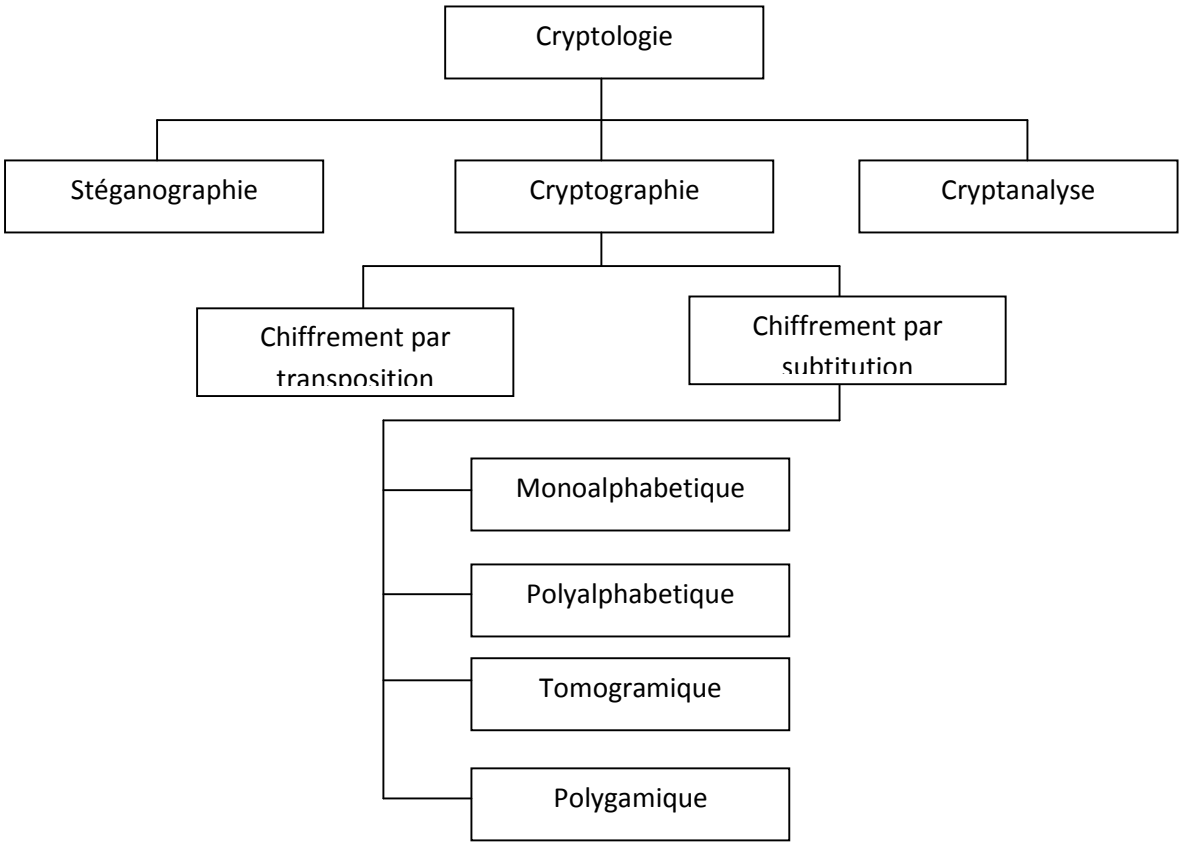


Fig. II.1 les composants de la cryptologie

II.2.1 Cryptographie

La cryptographie est une des disciplines de la cryptologie s'attachant à protéger des messages en s'aidant souvent de secrets ou clef. Le mot cryptographie découle des mots grecques « Krypto » (je cache) et « graphe » (le document).

II.2.2 Cryptanalyse

La cryptanalyse est la science de reconstitution du texte en clair sans connaitre la clé. Une cryptanalyse réussie peut fournir soit le texte en clair, soit la clef. La cryptanalyse peut également mettre en évidence les faiblesses d'un cryptosystème qui peuvent éventuellement faciliter les attaques contre celui-ci. Lorsqu'une méthode de cryptanalyse permet de déchiffrer un message chiffré à l'aide d'un cryptosystème, on dit alors que l'algorithme de chiffrement a été « cassé ». On distingue habituellement quatre méthodes de cryptanalyse :

- ✓ Une attaque sur texte chiffré seulement consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés ;
- ✓ Une attaque sur texte clair connu consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, connaissant le texte en clair correspondant ;
- ✓ Une attaque sur texte clair choisi consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, l'attaquant ayant la possibilité de les générer à partir de textes en clair ;
- ✓ Une attaque sur texte chiffré choisi consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, l'attaquant ayant la possibilité de les générer à partir de textes en clair.

II.2.3 Stéganographie

La stéganographie sert à cacher des messages secrets dans d'autres messages, de sorte que l'existence même du secret est dissimulée. Généralement l'expéditeur écrit un message inoffensif et dissimule un message secret dans la même feuille de papier.

II.2.3.1 Le texte caché

Enfin	voici	le	Printemps
Un	oiseau	de	Passage
Chante	Un	message	Emprunt
D’	Un	doux	Secret

Enfin	**voici**	le	Printemps
Un	oiseau	de	Passage
Chante	Un	**message**	Emprunt
D’	Un	doux	**Secret**

Clé : BACD

Le texte caché est :

Voici un message secret

II.2.3.2 La stéganographie dans les fichiers images :

De la même manière que pour le texte, il est possible de cacher de l'information dans des fichiers images. En modifiant ou altérant quelques bits du fichier, il est ainsi possible de cacher un copyright ou alors un message de son choix sans que cela se voit. En effet, cette altération sera peu ou pas visible car l’œil humain n’est pas capable de discerner des petites aberrations sur une grande image à condition que le ratio taille du message / taille de l'image ne soit pas trop grand.

En conclusion, il faut retenir que la stéganographie est une méthode de cryptographie faible : elle repose uniquement sur le fait que personne ne remarquera le canal caché. Dès lors que ce canal est connu, il n'y a plus aucune protection.

II.3 Cryptographie classique

Ce paragraphe présente plusieurs algorithmes ou méthodes de cryptographie à une époque où les mathématiques ne régnaient pas encore en maîtres sur ce domaine.

II.3.1 Cryptographie par transposition (technique grecque) [23]

Une méthode de chiffrement datée entre le Xème et VIIème siècle avant Jésus Christ repose sur l'utilisation d'un bâton appelé scytale d'un diamètre fixé. Une lanière en cuir était enroulée en hélice autour de ce bâton et le texte en clair était alors écrit sur la lanière. Ensuite, la lanière était déroulée et pouvait être envoyée (sans le bâton) au destinataire du message.

Fig. II.2 Scytale grecque

Pour déchiffrer le texte chiffré, il suffisait d'utiliser un bâton possédant exactement le même diamètre que le précédent, d'y enrouler la lanière de cuir et le texte en clair pouvait alors être relu.

Le procédé utilisé par cette méthode est un chiffrement par transposition, c'est-à-dire que les lettres ne sont pas modifiées mais que seul l'ordre des lettres est changé.

II.3.2 Cryptographie par substitution [1][23]

Les systèmes de cryptographie par substitution sont considérés comme des applications bijectives des lettres de l'alphabet des messages clairs sur des lettres de l'alphabet des cryptogrammes : on remplace des caractères par d'autres. On distingue :

II.3.2.1 Chiffrement monoalphabétique [11] [23]

On parle de chiffrement monoalphabetique lorsque chaque lettre du message clair est toujours remplacée par le même symbole. On obtient ainsi une bijection entre les lettres clairs et les symboles de l'alphabet de chiffrement.

Ce chiffrement était utilisé par les Romains, connue sous le nom de « chiffrement de César », Il suffit de décaler les lettres d'un certain nombre connu aussi bien par celui qui écrit le message que par celui qui le reçoit.

Par exemple si n=4, cela donne :

a=E, b=F, c=G, ... w=A, x=B, y=C, z=D.

Si le texte en clair à chiffrer est "rendons a cesar ce qui est a cesar", avec un décalage de quatre lettres, le texte chiffré est "VIRHSRW E GIWEV GI UYM IWX E GIWEV".

Ce procédé est un procédé de chiffrement par substitution mono-alphabétique. Il est très simple mais il ne résiste pas à une analyse basée sur la fréquence des caractères puisque chaque lettre est toujours remplacée par la même lettre. D'ailleurs dans l'exemple fourni, le

mot cesar est chiffré deux fois de la même manière en GIWEV de même que le a est chiffré deux fois en E.

Un autre algorithme, nommé ROT13, basé sur le même fonctionnement a existé au tout début de l'informatique. Le décalage était de treize lettres, et c'est donc le même algorithme qui était utilisé aussi bien pour le chiffrement que pour le déchiffrement (un premier décalage de treize lettres pour chiffrer suivi d'un autre décalage de treize lettres pour déchiffrer suffisait à redonner le texte en clair).

II.3.2.2 Chiffrement polyalphabetique

Le chiffrement polyalphabétique est inventé en 1968, il utilise simultanément plusieurs alphabets de chiffrement. Bien que de nombreux systèmes existent, nous présentons ici celui qui demeure le plus connu.

II.3.2.2.1 Chiffrement de vigenere

Le chiffrement de vigenere (1523-1596) est un chiffrement à décalage par clé. Au lieu d'effectuer un décalage constant dans l'alphabet du message en clair, comme dans le code de césar, vigenére réalise un décalage lié a une clé (la clé de chiffrement).chaque lettre de la clé sert de valeur de décalage dans l'alphabet du message en clair. Deux lettres identiques de ce message en clair ne seront donc plus obligatoirement codées de la même manière**. [11]**

Il nécessite ensuite un mot clé seulement connu par l'émetteur et le destinataire du message. C'est le premier algorithme à introduire la notion de clé**.** Le chiffrement est effectué à l'aide de la table suivante :

Clé Utilisée \ Lettre en 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	Lettre Codée
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	

Fig. II.3 Table de chiffrement de vigenere

Ce mot clé est répété autant de fois que nécessaire afin d'avoir autant de lettres (ou plus) que le texte en clair à chiffrer. Ensuite l'alphabet utilisé pour chiffrer une lettre est celui correspondant à la lettre du mot clé.

Le croisement de la colonne de la lettre en clair et la ligne de la lettre de la clé donne la lettre codée.

Si le texte à chiffrer est "rendons a vigenere ce qui est a vigenere", avec le mot clé "RAYMOND", le texte chiffré est "IELPCAV R VGSSAHIE AQ EHL VSR M JVJVNCDS".

II.3.2.3 Chiffrement tomogrammique

Dans les systèmes tomogrammique, chaque lettre est tout d'abord représenté par un groupe de plusieurs symboles. Ces symboles sont ensuite chiffrés séparément ou par groupe de taille fixe.

II.3.2.3.1 Chiffrement ADFGVX [11][24]

Le chiffrement ADFGVX est un système utilisé pendant la première guerre mondiale son intérêt est d'utiliser les deux styles de chiffrement : substitution et permutation.
L'algorithme se subdivise en deux étapes :

Chiffrement par substitution

Utilise un tableau secret qui permet de substituer une lettre (26 possibilité) ou un chiffre (10 possibilité) par deux lettres prise parmi les six lettres A, D, F, V, G ou X.
Par exemple, si l'on utilise le tableau de chiffrement suivant :

	A	**D**	**F**	**G**	**V**	**X**
A	8	T	B	W	r	q
D	p	4	c	g	2	9
F	3	o	5	m	x	e
G	d	a	z	j	s	y
V	l	h	7	u	v	0
X	N	1	k	6	i	f

Tab. II.1 Table 6X6 de codage

Le message « lancer assaut » deviendra : AV DG AX FD XF VA DG VG VG DG GV DA.

Chiffrement par transposition

Utilise une permutation et un mot clé secret de taille n, tout d'abord on crée une grille de n colonnes sur laquelle on place ce mot secret en tete et ou l'on inscrit ensuite, ligne après ligne, le cryptogramme intermédiaire obtenu à la première étape. Ensuite, en effectue des permutations de colonnes, de sorte que les lettre du mot-clé secrèt soient réordonnées dans l'ordre alphabétique.

Si on reprend l'exemple précédent avec le mot clé « chat » on obtient le tableau suivant :

Clé original	C	H	a	t
Message Codé	A	V	D	G
	A	X	F	D
	X	F	V	A
	D	G	V	G
	V	G	D	G
	G	V	D	A

Tab. II.2 Table de ADFGVX avec la clé « chat »

Après classement alphabétique des lettres de la clé, le tableau suivant contient le message chiffré final :

Clé classée	A	C	h	t
Message Codé Et transposé	D	A	V	G
	F	A	X	D
	V	X	F	A
	V	D	G	G
	D	V	G	G
	D	G	V	A

Tab. II.3 Table de ADFGVG avec la clé classée « acht »

Le message définitif, obtenu par lecture des colonnes du tableau, est donc : DF VV DD AA XD VG VX FG GV GD AG GA. Le destinataire le déchiffrera en suivant ces mêmes étapes dans l'ordre inverse, à condition de connaître la clé originale de transposition et de disposer de la table 6×6 de codage.

II.3.2.4 Chiffrement polygamique

Dans un chiffrement polygrammique un groupe de n lettres est chiffré par un groupe de n symboles. Parmi les nombreux exemples de tels chiffrements, on citera le chiffrement de playfair.

II.3.2.4.1 Chiffrement de Playfair

Le chiffre de Playfair utilise un tableau de 5x5 lettres, contenant un mot clé ou une phrase. La mémorisation du mot clé et de 4 règles à suivre suffisent pour utiliser ce chiffrement.

Pour chiffrer un message, il faut prendre les lettres 2 par 2 et appliquer les règles suivantes en fonction de la position des lettres dans la table :

- ✓ si les 2 lettres sont identiques (ou s'il n'en reste qu'une) mettre un 'X' après la première lettre. Chiffrer la nouvelle paire ainsi constituée et continuer avec la suivante. Dans certaines variantes, on utilise 'Q' au lieu du 'X', mais n'importe quelle lettre peut faire l'affaire,
- ✓ si les lettres se trouvent sur la même ligne de la table, il faut les remplacer par celles se trouvant immédiatement à leur droite (en bouclant sur la gauche si le bord est atteint),
- ✓ si les lettres apparaissent sur la même colonne, les remplacer par celles qui sont juste en dessous (en bouclant par le haut si le bas de la table est atteint),
- ✓ sinon, remplacer les lettres par celles se trouvant sur la même ligne, mais dans le coin opposé du rectangle défini par la paire originale.

En supposant que la clé soit « exemple Playfair », le tableau doit alors être rempli comme suit :

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

Tab. II.4 Table 5x5 de la clé

Chiffrement du message « Cache l'or dans la souche de l'arbre » :

CA	CH	EL	OR	DA	NS	LA	SO	UC	HE	DE	LA	RB	RE

Tab. II.5 Tableau de message en clair

Le message chiffré est le suivant :

BY	DB	XE	QI	BF	JU	ER	VJ	TD	BL	BM	ER	AH	AL

Tab. II.6 Tableau de message chiffré

II.3.3 Chiffrement de Vernam (masque jetable)

Le chiffrement par la méthode du masque jetable consiste à combiner le message en clair avec une clé présentant les caractéristiques très particulières suivantes :

- ✓ La clé doit avoir la même longueur que le message à chiffrer.
- ✓ Les caractères composant la clé doivent être choisis de façon totalement aléatoire.
- ✓ Chaque clé, ou « masque », ne doit être utilisée qu'une seule fois (d'où le nom de *masque jetable*).

Le principe est alors de combiner clé et message par un XOR bit a bit. Cette technique est inconditionnellement sure sous réserve que la chaine commune K soit parfaitement aléatoire. L'inconvénient majeur de cette technique est qu'il faut partager des chaines aléatoires extrêmement longues.

La cryptographie classique, la sécurité des données est basée sur le secret de l'algorithme de chiffrement, mais cette technique présente de nombreux inconvénients, comme l'obligation de changer l'algorithme de chiffrement si celui-ci est divulgué par un des à l'échange.

II.4 Cryptographie moderne

Avec le temps, les cryptographes ont pris conscience qu'il n'était pas réaliste de faire reposer un système de chiffrement sur l'hypothèse qu'un attaquant n'a pas connaissance de la méthode utilisé. En conséquence, ils ont introduit un nouveau type de cryptographie dans

lequel la sécurité des algorithmes de chiffrement repose uniquement sur le secret de la clé de déchiffrement, il s'agit de la cryptographie moderne.

La cryptographie moderne repose maintenant uniquement sur les mathématiques. De plus, les règles de bases sont :

- ✓ L'algorithme utilisé n'est pas secret. Il peut être diffusé librement, cela ne doit avoir aucun impact sur la facilité ou non à déchiffrer le message.
- ✓ La clé de chiffrage utilisée est secrète.

Un protocole cryptographique basé sur la non-divulgation de l'algorithme mathématique utilisé n'est pas fiable. Tôt ou tard, l'algorithme utilisé sera connu et le protocole deviendra faible. Au contraire, diffuser l'algorithme mathématique utilisé permet à la communauté de valider et de tester la robustesse de cet algorithme.

II.4.1 Historique de la cryptographie moderne [12]

Pendant de nombreuses années, la cryptographie était exclusivement réservée au domaine militaire et diplomatique. La littérature sur le sujet était donc très peu abondante. La première publication fondamentale dans le domaine a été l'article de C laude Shanon 1949/ « the communication thery of secrcy systems ».Dans lequel il jette les bases mathématiques d'un système de communication chiffrée, à partir de la définition d'un nouveau modèle : L a théorie de l'information. Une contribution importante a été ensuite celle de feistel, avec la publication au débit des année 1970 de ses travaux sur les schémas de chiffrement itératifs par blocs qui ont conduit en 1977 ans la proposition de l'algorithme DES comme standard de chiffrement à clé secrète l'accroissement de la sécurité du DES.

L'accroissement de la puissance des ordinateurs ayant remis en cause de la sécurité du DES, il a été remplacé en 2000 par un nouveau standard appelé AES. Cet algorithme est l'aboutissement de recherches récentes notamment dans le domaine de la cryptanalyse. En 1976, après la publication par Diffie et Hellman de l'article : « New Direction in Cryptography », un nouveau concept révolutionnaire de la cryptographie, qu'est la cryptographie à clé publique, a été introduit.

Plus récemment pour faire face aux nouvelles menaces par le développement des réseaux et la numérisation massive des documents, la cryptographie a dû offrir de nouvelles

fonctionnalités : garantie de l'authenticité des messages (provenance et contenu), réalisé par les algorithmes de signature numérique.

Ainsi, la cryptographie moderne offre deux grandes catégories de procédés cryptographiques :

- Algorithmes de chiffrement : servent à protéger la confidentialité des données.
- Algorithmes de signatures : garantissent la provenance et l'intégrité des messages.

II.4.2 Chiffrement [13]

Un algorithme de chiffrement transforme un message, appelé texte clair, en un texte chiffré qui ne sera lisible que par son destinataire légitime. Cette transformation est effectuée par une fonction de chiffrement paramétrée par une clé de chiffrement. Un interlocuteur peut alors déchiffrer le message en utilisant la fonction de déchiffrement s'il connait la clef de déchiffrement correspondant. Un tel système n'est sur que s'il est impossible à un intrus de déduire le texte clair du message chiffré.

II.4.2.1 Différents modes de chiffrements [14] [25]

Le mode de chiffrement correspond à la manière dont on va utiliser un algorithme de chiffrement donné. Ce mode de chiffrement consiste par exemple à rajouter de la contre-réaction entre l'entrée et la sortie de l'algorithme afin de lui rajouter des caractéristiques bien précises. Les différentes modes de chiffrement utilisés sont les suivants :

- ✓ Le mode ECB pour **Electronic Code Book**
- ✓ Le mode CBC pour **Cipher Block Chaining**
- ✓ Le mode **chiffrement en continu**
- ✓ Le mode CTAK pour **Cipher Text Auto Key**
- ✓ Le mode CFB pour **Cipher Feed Back**
- ✓ Le mode KAK pour **Key Auto Key**
- ✓ Le mode OFB pour **Output Feed Back**
- ✓ Le mode CTR pour **CounTeR**
- ✓ Le mode BC pour **Block Chaining**
- ✓ Le mode PCBC pour **Propagating Cipher Block Chaining**
- ✓ Le mode CBCC pour **Ciher Block Chaining with Checksum**
- ✓ Le mode OFBNLF pour **Output Feed Back mode with a Non Linear Function**
- ✓ Le mode PBC pour **Plaintext Block Chaining**

- ✓ Le mode PFB pour **Plaintext Feed Back**
- ✓ Le mode CBCPD pour **Cipher Block Chaining of Plaintext Difference**
- ✓ Le mode CTS pour **Cipher Text Stealing**

Dans les paragraphes suivants nous allons regarder un peu plus en détail les modes ECB, CBC, CFB et OFB. Les autres modes sont nommés à des fins d'exhaustivité mais sont rarement utilisés.

II.4.2.1.1 Mode ECB

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.

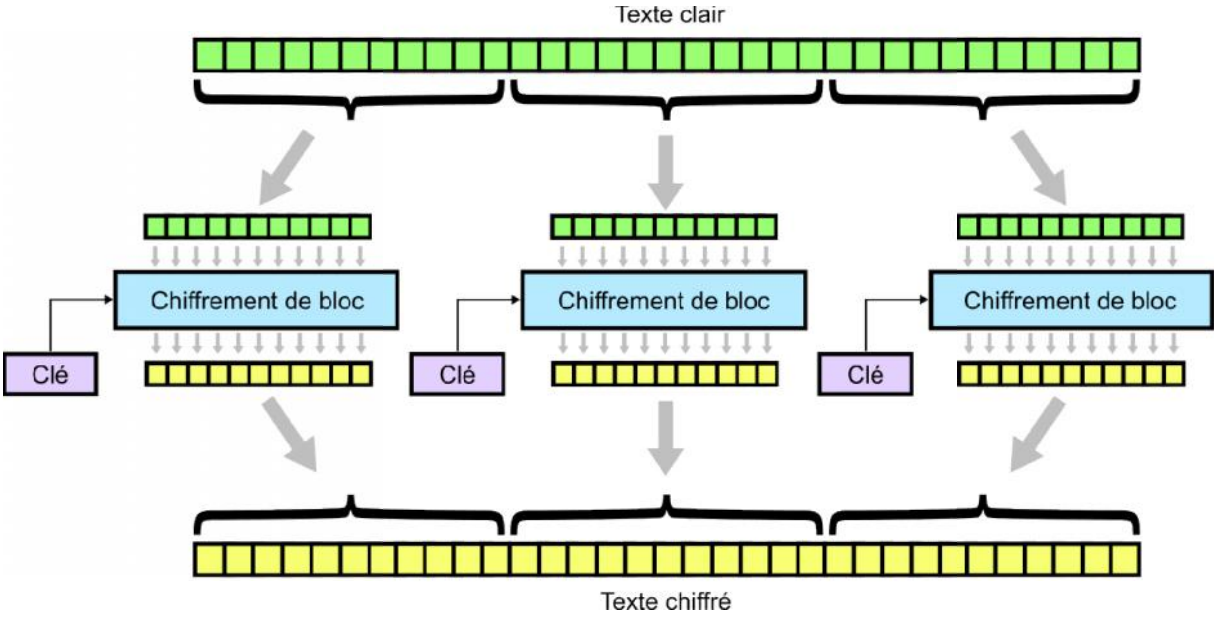


Fig. II.4 Mode ECB

Les avantages de ce mode sont les suivants :

- ✓ Le travail de chiffrement ou de déchiffrement peut être parallélisé. Plusieurs machines ou CPU peuvent travailler simultanément sur des parties différentes du message.
- ✓ Une erreur de transmission d'un bit affecte uniquement le décodage du bloc courant.

Par contre, ce mode a les désavantages suivants :

- ✓ Les répétitions du texte en clair ne sont pas masquées et se retrouvent sous la forme de répétitions de textes chiffrés.
- ✓ Des portions complètes du message peuvent être modifiées, répétées ou remplacées sans difficulté.

II.4.2.1.2 Mode CBC

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 appelée IV (pour Initialisation Vector) qui permet d'initialiser le processus quand aucun bloc n'a encore été chiffré.

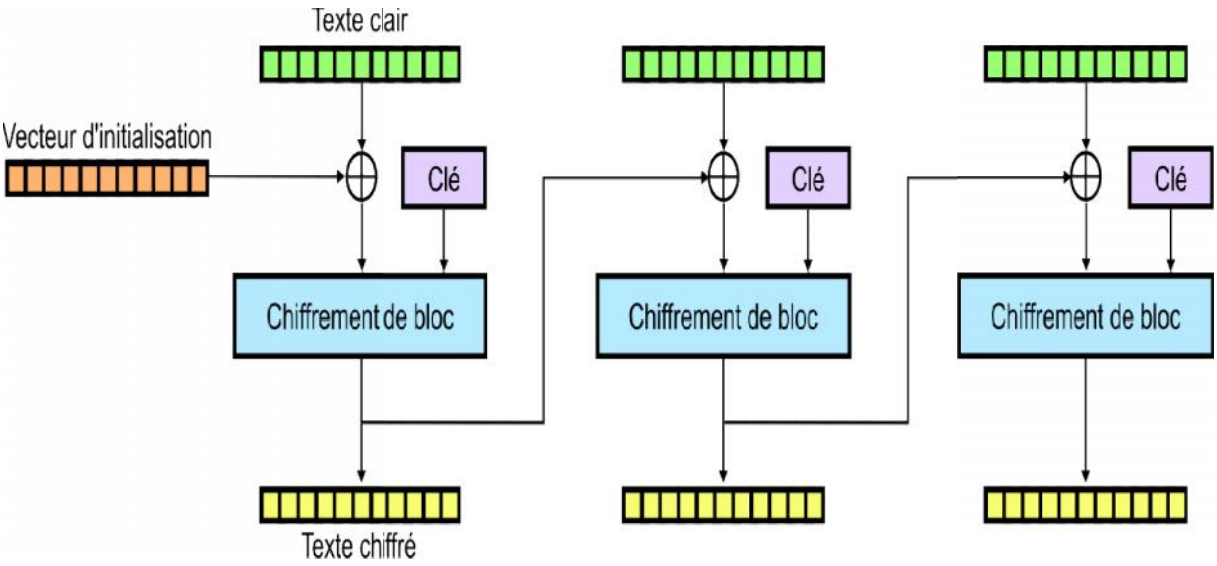


Fig. II.5 Mode CBC

Les avantages de ce mode sont les suivants :

- ✓ Les répétitions de texte en clair sont masquées dans le texte chiffré.
- ✓ La valeur du vecteur d'initialisation IV n'a pas besoin d'être secrète.

Par contre, ce mode a les désavantages suivants :

- ✓ Deux textes en clair commençant pareil auront le même début de texte chiffré.
- ✓ Une erreur de transmission d'un bit affecte uniquement le décodage du bloc courant ainsi que le décodage du même bit dans le bloc suivant.

II.4.2.1.3 Mode CFB

Les modes ECB et CBC travaillent sur des blocs de texte en clair (64 bits par exemple). Ces modes ne sont pas utilisables que lorsqu'un bloc est complet. Sur des applications réseau, cela peut poser des problèmes car les valeurs à chiffrer arrivent sous forme d'octets et doivent être transmises immédiatement.

Ce mode dispose d'un vecteur d'initialisation qui a la même fonctionnalité que celui du CBC. L'octet 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 la fonction de chiffrement.

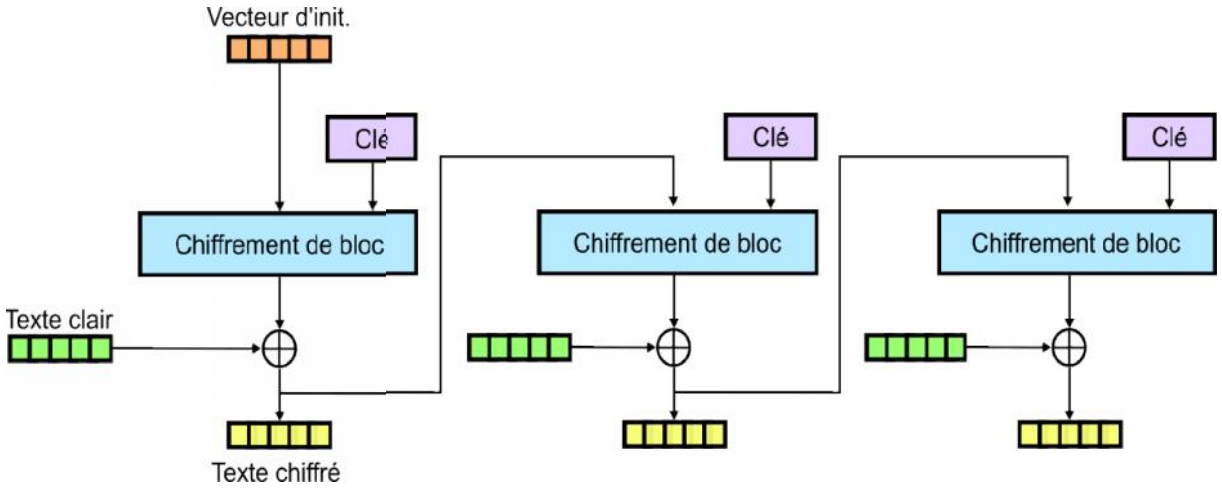


Fig. II.6 Mode CFB

Les avantages de ce mode sont les suivants :

- ✓ Il est possible de chiffrer un flot de valeurs plus petites que la taille standard du bloc géré par l'algorithme.
- ✓ Les répétitions de texte en clair sont masquées dans le texte chiffré.
- ✓ La valeur du vecteur d'initialisation IV n'a pas besoin d'être secrète.

Par contre, ce mode a le désavantage suivant :

- ✓ Une erreur de transmission d'un bit affecte uniquement le décodage du bloc courant ainsi que le décodage du même bit dans le bloc suivant.

II.4.2.1.4 Mode OFB

Le mode OFB ressemble au mode CFB. La seule différence est que l'octet injecté dans la fonction de chiffrement vient du chiffrement successif du vecteur d'initialisation.

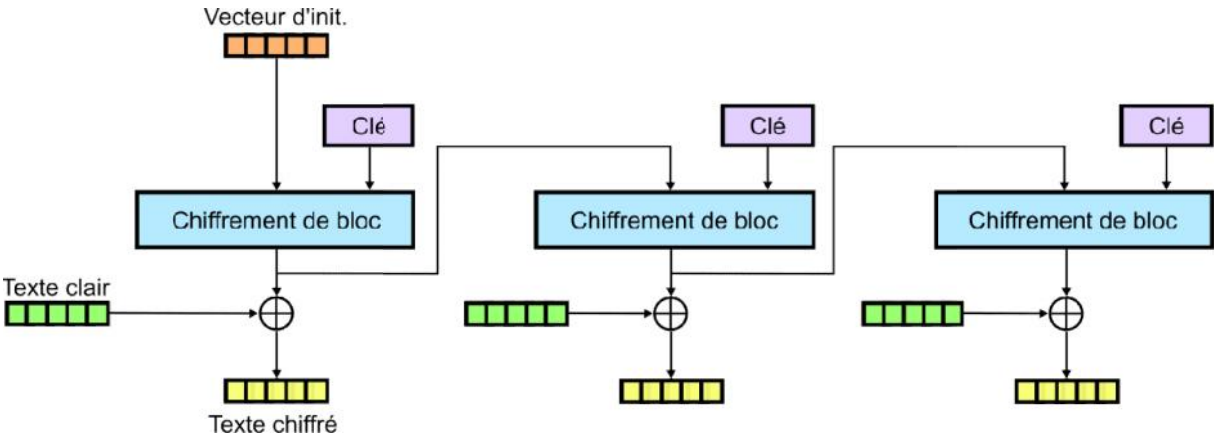


Fig. II.7 Mode OFB

Les avantages de ce mode sont les suivants :

- ✓ Les répétitions de texte en clair sont masquées dans le texte chiffré.
- ✓ La valeur du vecteur d'initialisation IV n'a pas besoin d'être secrète.
- ✓ Ce mode n'amplifie pas les erreurs. Une erreur de transmission d'un bit affecte uniquement ce bit lors du décodage.

Par contre, ce mode a le désavantage suivant :

- ✓ Ce mode est très fragile vis-à-vis d'une attaque au clair. (l'attaquant possède des textes clairs ainsi que leurs versions chiffrées et est libre de les utiliser pour révéler d'autres informations secrètes comme la clé de chiffrement).

II.4.2.2 Cryptographie symétrique et asymétrique [16][15]

II.4.2.2.1 Cryptographie Symétrique

La cryptographie symétrique aussi nommée à clé secrète utilise la même clef pour chiffrer et déchiffrer un message. L'émetteur et le récepteur doivent se mettre d'accord sur une clé à utiliser et gardée secrète.

Les algorithmes à clé secrète étaient utilisés dans le domaine militaire. Ils servaient à protéger la confidentialité des messages entre leurs émetteurs et leurs destinataires. Dans un premier temps, la transformation d'un message clair en un message crypté passait par des procédures secrètes comme le montre la figure Fig. II.8. Un des premiers exemples de cette cryptographie symétrique est le chiffrement de César.

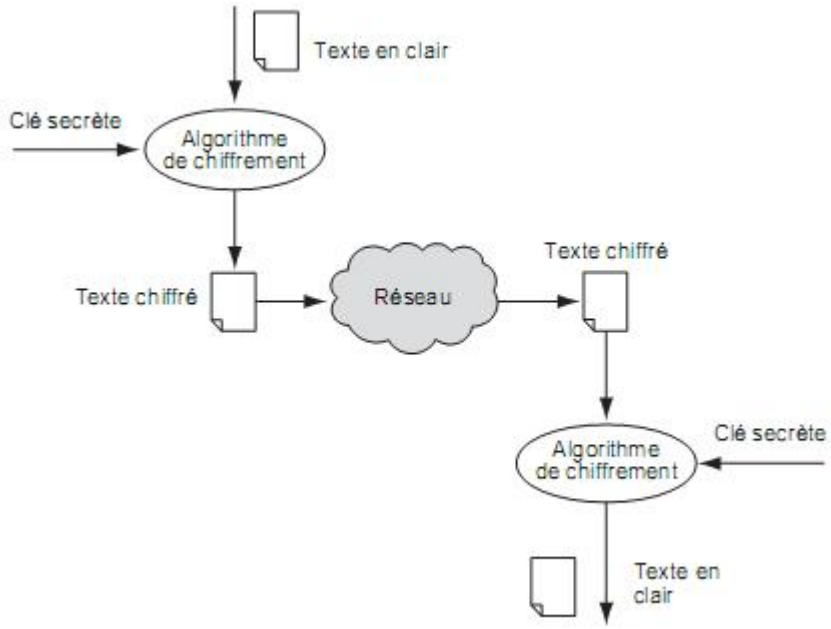


Fig. II.8 Algorithme de chiffrement symétrique

Ces algorithmes fonctionnent habituellement suivant deux procédés différents, le chiffrement par blocs et le chiffrement par flot (en continu).

- ✓ **Chiffrement par flot :** le chiffrement est effectué bit-à-bit sans attendre la réception complète des données à chiffrer. Une technique de chiffrement, du nom de "One-Time Pad"(chiffrement de vernam) est utilisée pour chiffrer les flux (chiffrement de vernam).

- ✓ **Chiffrement par blocs** : opèrent sur le message en clair par groupes de bits (blocs). Il existe plusieurs algorithmes qui fonctionnent sur ce principe par exemple le DES (Data Encryption Standard) qui est l'algorithme à clé symétrique historiquement le plus connu.

II.4.2.2.2 Cryptographie asymétrique

Le principe des algorithmes de chiffrement à clés asymétriques a été introduit en 1976 par Diffie et Hellman. Ils ont été conçus pour utiliser des clés qui possèdent 2 propriétés essentielles :

- Les clés sont crées par couple souvent appelé bi-clé. Ce bi-clé est tel que tout texte chiffré par l'une quelconque des deux clés n'est déchiffrable que par l'autre clé. C'est cette caractéristique qui a donné leur nom aux algorithmes de chiffrement asymétrique
- La connaissance d'une des deux clés ne permet pas de déduire l'autre. En pratique, chaque protagoniste dispose d'un bi-clé (au moins un). On décide arbitrairement, pour chaque bi-clé, que l'une des clés est publique et l'autre privée

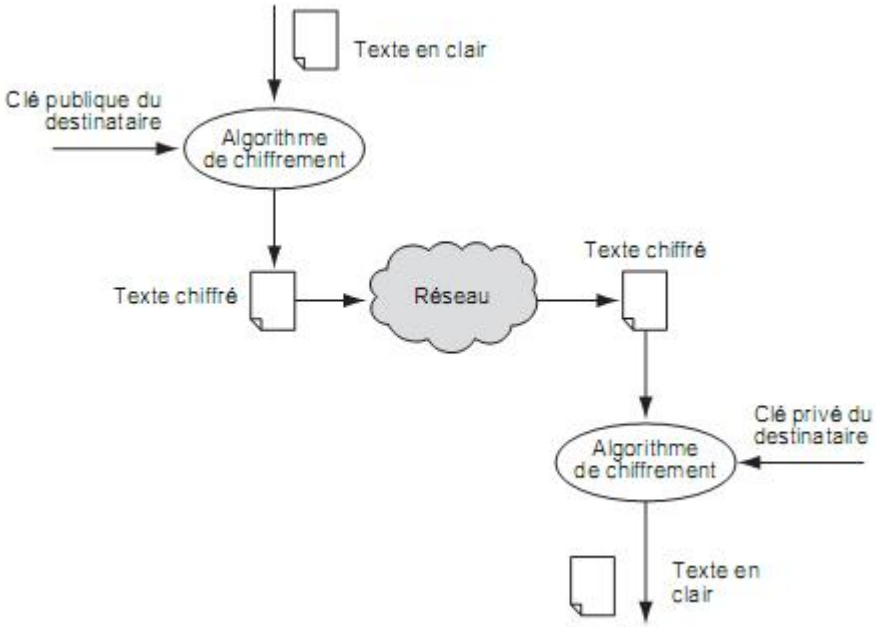


Fig. II.9 Algorithme de chiffrement asymétrique

Algorithme RSA

RSA, du nom de ces inventeurs, est un algorithme de chiffrement appartenant à la grande famille "Cryptographie asymétrique".

RSA peut être utilisé pour assurer :

- ✓ la confidentialité : seul le propriétaire de la clé privée pourra lire le message chiffré avec la clé publique correspondante.
- ✓ la non-altération et la non-répudiation : seul le propriétaire de la clé privée peut signer un message (avec la clé privée). Une signature déchiffrée avec la clé publique prouvera donc l'authenticité du message.

Sa robustesse réside dans la difficulté à factoriser un grand nombre.

Principe de fonctionnement de RSA :

Si Bob souhaite recevoir des messages en utilisant le RSA, il procède de la façon suivante :

- ✓ **Création des clés :** Bob crée 4 nombres p,q, e et d :
  - p et q sont deux grands nombres premiers distincts. Leur génération se fait au hasard, en utilisant un algorithme de test de primalité probabiliste.
  - e est un entier premier avec le produit (p-1)(q-1).
  - d est tel que ed=1 modulo (p-1)(q-1). Autrement dit, ed-1 est un multiple de (p-1)(q-1). On peut fabriquer d à partir de e, p et q, en utilisant l'algorithme d'Euclide.
- ✓ **Distribution des clés :** Le couple (n,e) constitue la clé publique de Bob. Il la rend disponible par exemple en la mettant dans un annuaire. Le couple (n,d) constitue sa clé privée. Il la garde secrète.
- ✓ **Envoi du message codé :** Alice veut envoyer un message codé à Bob. Elle le représente sous la forme d'un ou plusieurs entiers M compris entre 0 et n-1. Alice possède la clé publique (n,e) de Bob. Elle calcule $C=M^e \bmod n$. C'est ce dernier nombre qu'elle envoie à Bob.

- ✓ **Réception du message codé** : Bob reçoit C, et il calcule grâce à sa clé privée $D=C^d \pmod n$. D'après un théorème du mathématicien Euler, $D=M^{de}=M \pmod n$. Il a donc reconstitué le message initial.

II.5 Signature numérique (signature électronique)

Le chiffrement permet de rendre les services de confidentialité. La signature électronique va permettre de garantir l'authentification de l'origine d'un document ou d'un message électronique et son intégrité. Ceci implique un certain nombre de propriétés :

- ✓ une signature ne peut être falsifiée,
- ✓ une signature donnée n'est pas réutilisable par un autre document
- ✓ la modification d'un document signé altère la signature de ce document
- ✓ une signature ne peut être reniée

Pour générer une signature électronique il faut dans un premier temps utiliser une fonction de hachage.

II.6 Fonction de hachage [26] [17]

La fonction de hachage est une fonction permettant d'obtenir un condensé (condensat ou haché) d'un texte, c'est-à-dire une suite de caractères assez courte représentant le texte qu'il condense. Ces fonctions peuvent fournir une assurance de l'intégrité de données. Elle est utilisée pour construire une courte « empreinte numérique» de ces données ; si elles sont modifiées, l'empreinte numérique ne sera plus valide.

Une fonction de hachage H de taille de sortie n est un algorithme faisant correspondre à un message M de taille arbitraire un élément H(M) de taille n bits appelé haché. En pratique, n est de l'ordre de plusieurs centaines de bits, typiquement 128, 160, 256 ou 512 bits.

II.6.1 Propriétés de fonction de hachage

Les fonctions de hachage doivent posséder plusieurs propriétés utiles en cryptographie. Les principales sont la résistance aux attaques recherchant des collisions, des préimages ou des secondes préimages.

- ✓ **collision** : trouver deux messages distincts M1 et M2, tels que H(M1) = H(M2).
- ✓ **seconde préimage** : étant donné un message M1 choisi aléatoirement, trouver un message distinct M2 tel que H(M1) = H(M2).
- ✓ **préimage** : étant donné un haché H1 choisi aléatoirement, trouver un message M1 tel que H(M1) = H1.

Il doit être impossible pour un attaquant de trouver une collision, une préimage ou une seconde préimage.

II.6.2 Présentation des fonctions de la famille MD-SHA [17]

II.6.2.1 MD4 (Message Digest 4)

MD4 est une fonction de hachage conçue par le professeur Ronald Rivest du MIT. La taille de la signature est de 128 bits. L'algorithme a été abandonné au profit du MD5 après la découverte de faiblesses dans sa conception. D'autres attaques encore plus efficaces ont suivi, notamment par le service du chiffre allemand et encore l'équipe chinoise à l'origine de l'attaque sur MD5.

II.6.2.2 MD5 (Message Digest 5)

MD5 est une fonction de hachage cryptographique qui permet d'obtenir pour chaque message une empreinte numérique (en l'occurrence une séquence de 128 bits) avec une probabilité très forte que, pour deux messages différents, leurs empreintes soient différentes. En 1996, une faille grave (possibilité de créer des collisions à la demande) est découverte et indique que MD5 devrait être mis de côté au profit de fonctions plus robustes.

En 2004, une équipe chinoise découvre des collisions complètes. MD5 n'est donc plus considéré comme sûr au sens cryptographique. Leur attaque a permis de découvrir une collision complète sans passer par une méthode de type brute-force. La sécurité du MD5 n'étant plus garantie selon sa définition cryptographique, les spécialistes recommandent d'utiliser des fonctions de hachage plus récentes comme le SHA-256.

MD5 reste encore très utilisé comme outil de vérification lors des téléchargements. Les sites affichent encore souvent la signature en MD5 (128 bits) de leurs fichiers, bien que SHA-1 (160 bits) le remplace de plus en plus.

II.6.2.3 SHA-1 (Secure Hash Algorithm 1)

SHA-1 a été conçu par la National Security Agency des États-Unis (NSA), et publiée par le gouvernement des États-Unis comme un standard. SHA-1 défini en 1995 et produit un condensat de 160 bits. SHA-1 est le successeur du SHA-0 (1993) qui a été rapidement mis de côté par le NIST pour des raisons de sécurité insuffisante.

En février 2005, Bruce Schneier a fait état d'une attaque sur la version complète du SHA-1 par l'équipe chinoise de Wang, Yin et Yu. Leur méthode permet de trouver une collision dans le SHA-1 complet de 128 bits avec 2^{69} opérations. Ayant perdu une longueur d'avance dès l'annonce de l'attaque de Wang, SHA-1 a été retiré progressivement des applications cryptographiques au profit de SHA-256 ou des autres fonctions de hachage.

II.6.2.4 SHA-2 (Secure Hash Algorithm 2)

SHA-2 à été publié récemment et est destiné à remplacer SHA-1. Les différences principales résident dans les tailles de hachés possibles : 256, 384 ou 512 bits. Il sera bientôt la nouvelle référence en termes de fonction de hachage.

En 2005, des problèmes de sécurité de SHA-1 ont été mis en évidence : il existe pour la recherche de collisions une attaque théorique nettement plus rapide que l'attaque générique des anniversaires sur les fonctions de hachage. Bien que l'algorithme de SHA-2 partage des similarités avec celui de SHA-1, ces attaques n'ont actuellement pas pu être étendues à SHA-2. Le NIST a cependant organisé un concours pour sélectionner une nouvelle fonction de hachage, SHA-3. Le concours a débouché fin 2012 sur le choix d'une nouvelle famille de fonctions dont la conception est très différente de SHA-1 et de SHA-2. La nouvelle famille de fonctions est présentée comme un autre choix possible, elle ne remet pas en cause l'utilisation de SHA-2 du moins dans l'immédiat.

II .9 Conclusion

Dans ce chapitre, nous avons étudié les fondements de la cryptographie et son développement de temps classique au temps moderne, et aussi ses différents algorithmes nécessaires pour la protection des informations personnelles ou privés.

Nous détaillerons dans le chapitre qui va suivre le principe et le fonctionnement de l'un ces algorithmes de chiffrement symétrique Data Encryption Standard(DES) .

Chapitre III

Data Encryption Standard

(DES)

III.1 Introduction

Après avoir présentés les notions générales de la cryptographie, à présent nous nous intéressons à l'algorithme de chiffrement symetrique DES,nous soulignons le fonctionnement et les motivations qui nous ont poussées à choisir cet algorithme.

III.2 Histoire de DES

Dans les années 70, seuls les gouvernements et les militaires disposaient de méthodes de chiffrement fiables. Mais devant l'augmentation massive des besoins civils, le gouvernement Américain lança un appel d'offre en 1973 pour la conception d'un tel algorithme. Celui-ci devait répondre au cahier des charges suivant :

- ✓ Il devait utiliser une clé de petite taille qui devait servir à chiffrer et à déchiffrer les messages ;
- ✓ La sécurité du système ne devait pas reposer sur la confidentialité de la méthode, mais bien sur la clé, pour que le système puisse devenir un standard ;
- ✓ Il devait être rapide et facile à mettre en place, tant au niveau logiciel que matériel.

A l'époque, les ingénieurs d'IBM disposent déjà d'un tel algorithme, baptisé **Lucifer**, mis au point deux ans plus tôt. C'est cet algorithme qui aurait dû, en toute logique, être sélectionné par le Bureau des Standards. Data Encryption Standard (standard de l'encryption de données), approuvé comme standard en 1976.

III.3 Principe de base de DES

L'algorithme DES combine deux techniques de base du chiffrement qui ont pour but de compliquer la cryptanalyse mathématique :

- ✓ **Confusion** : gomme les relations entre le texte clair et le texte chiffré pour éviter les attaques par analyse statistique, réalisé à l'aide de boitiers de substitution. Autrement dit un changement d'un seul bit dans le texte clair doit affecter un grand nombre de bits (tous) du texte chiffré.

- ✓ **Diffusion** : disperse la redondance du texte clair dans le texte chiffré par exemple deux lettres doublées ne doivent pas rester côte à côte après cryptage, réalisé à l'aide de la permutation.

III.4 Description du DES

III.4.1 Chiffrement

- ➢ **Permutation initiale :**

L'algorithme fait subir à chaque bloc de 64 bits du texte en clair M permutations initiale selon la fonction IP (Tab.III.1) donnant le message M'.

58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Tab. III.1 Permutation initiale

Pour faire cette permutation on a proposé la méthode suivante :

```
public static char [][] permutation_initial(char [][] tab_bloc){ //permuter chaques blocs selon la table de permutation initial

char []bloc=new char[taille_bloc]; //bloc est un tableau de 64 bits
    for (int i=0;i<tab_bloc.length;i++){
   System.arraycopy(tab_bloc[i], 0, bloc, 0, tab_bloc[i].length); //copier une ligne de tab-bloc dans bloc
        for (int j=0;j<tab_bloc[i].length;j++){
tab_bloc[i][j]=(bloc[table_perm_initial[j]-1
                }
               return tab_bloc;
           }
```

Programme III.1 : Méthode ***permutation_initial()***

- ➢ **Séparation en deux blocs de 32 bits**

M'est ensuite coupé en une partie droite R_0 (nombres pairs) et une partie gauche L_0 (nombres impairs), de 32 bits chacune.

La méthode suivante permet de diviser le blocs de 64 bits en d/eux blocs de 32 bits chacun

```
private void diviser_bloc(char[] bloc) {

 System.arraycopy(bloc, 0, left, 0, bloc.length/2); //copier les premier 32  bits (pairs) dans la partie gauche'indice de 0 a 31'
        for(int i=bloc.length/2;i<bloc.length;i++)
                right[i-(bloc.length/2)]=bloc[i];    //copier les 32 autres bits(impairs) dans la partie droite 'indice de 0 a 31'

                    }
```

Programme III.2 :Methode *diviser_bloc()*

DES exécute ensuite 16 itérations de la fonction f, qui combine substitution et permutation.

- A chaque itération i ($1 \leq i \leq 16$), la fonction prend en entrée les 32 bits de la partie droite des données et 48 bits de la clé, et fournit 32 bits en sortie.

Le schéma suivant résume ce qui précède :

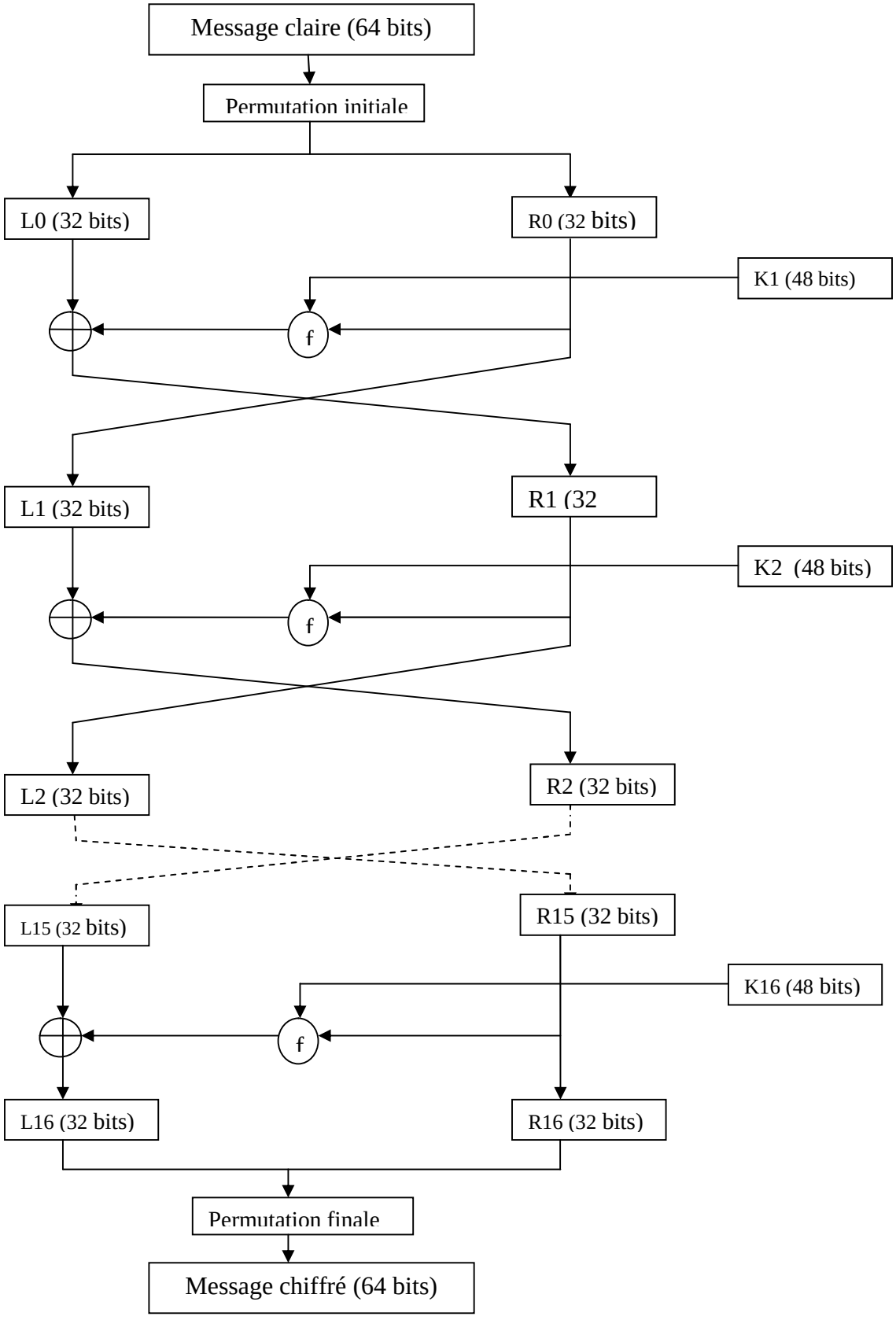


Fig. III.1 Architecture de DES

- **La fonction f**

La fonction f est constituée de 4 opérations comme l'illustre la figure suivante :

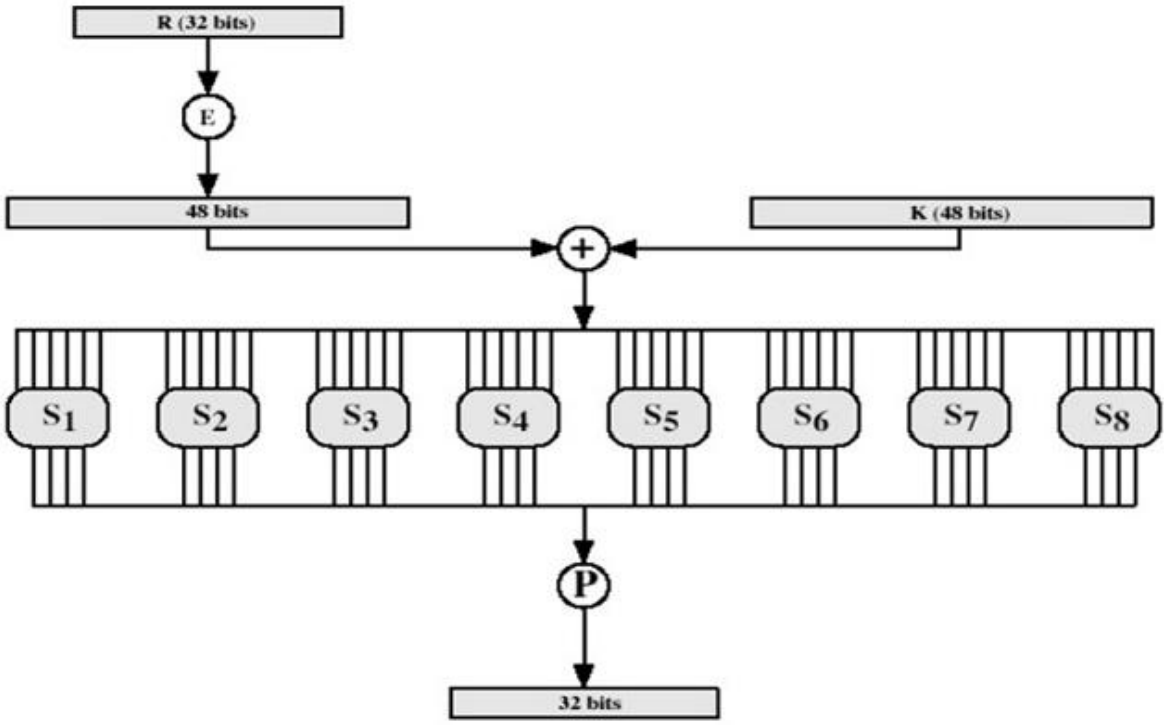


Fig.III.2 La fonction f de l'algorithme

- ✓ **Permutation expansive**

Cette permutation étend la partie droite des données R_i de 32 bits à 48 bits en changeant l'ordre et en répétant certains bits (1^{er} et 4^{eme} bit représentent chacun 2 bits en sortie ; 2^{eme} et 3^{eme} bit représentent 2 bits en sortie)

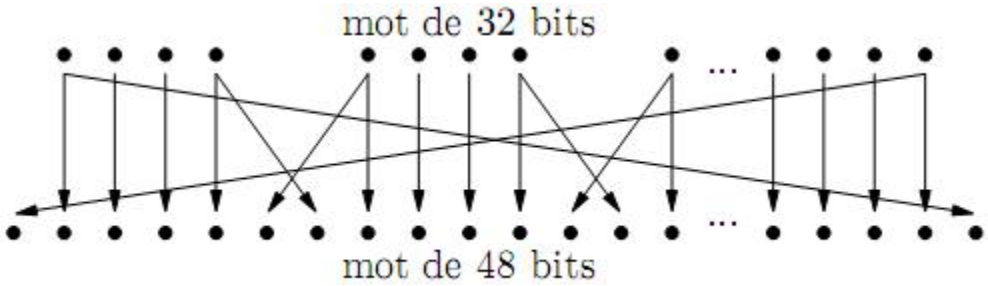


Fig. III.3 Schéma de la permutation expansive

Ces bits sont permutés selon la table suivante :

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

Tab. III .2 Permutation expansive

- ✓ Combinaison des 48 bits de données avec les 48 bits de la clé à l'aide d'une opération de OU exclusif, le résultat est ensuite scindé en 8 blocs de 6 bits.

  On a réaliser combinaison est faite a l'aide la méthode suivante :

```
private char[] ou_exclusif(char[] bloc1, char[] bloc2) {
      char[] bloc_exc=new char[bloc1.length];
for(int i=0;i<bloc_exc.length;i++)
                if(bloc1[i]=='0'){
                      if(bloc2[i]=='0')
                            bloc_exc[i]='0';
                            else bloc_exc[i]='1';
                    }else
                       if(bloc2[i]=='0')
                             bloc_exc[i]='1';
                          else bloc_exc[i]='0';
          return bloc_exc;
        }
```

Programme III.3 : la méthode *ou_exclusif()*

- ✓ Substitution : Chaque bloc de 6 bits résultant de la combinaison donnée/clé, constitue l'entrée d'une table.

Les 8 tables de substitutions sont présentées dans le programme ci-dessous :

```
static byte [][]table_S1={{14, 4,13, 1, 2,15,11, 8, 3,10, 6,12, 5, 9, 0,7},

                          { 0,15, 7, 4,14, 2,13, 1,10, 6,12,11, 9, 5, 3,8},
                          { 4, 1,14, 8,13, 6, 2,11,15,12, 9, 7, 3,10, 5, 0},
                         {15,12, 8, 2, 4, 9, 1, 7, 5,11, 3,14,10, 0, 6,13}};
```

Programme III.4 : Déclaration des 8 tables de substitution

Un boitier de substitution est un matrice à 4 rangs et 16 colonnes, il y a huit S-Box, Si recevant chacun en entrée un bloc de 6 bits B_j, un bloc de 4bits, de manière suivante :

- L'entier représenté par b1b6, sélectionne une ligne de S_i ;
- L'entier représenté par b2b3b4b5 sélectionne une colonne de S_i ;

Chaque S-Box retourne en sortie on aura la représentation binaire de l'entier à cette position.

Pour se faire on aura besoin d'une méthode qui convertit l'entier en binaire.

```
char[] IntToBinary(int entier){ //convertir l'entier en binaire
        String bin = null;
        bin=Integer.toBinaryString(entier);
        int ajout_0=(4-bin.length()%4)%4; //les entiers de sortie de la table de
substitution doivent etre de 4 bit(ajouter des zeros)

        for (int i=0;i<ajout_0;i++) bin="0"+bin;
         char[]bin2=bin.toCharArray();
         return bin2;
       }
```

Programme III.5 : Méthode *IntToBinary()*.

- ✓ Permutation : les huit blocs de 4 bits résultants de l'étape de substitution sont concaténés en un blocs de 32 bits qui subit une permutation pure P (tab.III.4) .

La table montre vers quelles positions migre chaque bit.

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

Tab. III.3 Permutation – P

Le résultat fournit a chaque itération (i) de la fonction f est combinée à chaque fois avec la partie gauche Li. Les parties gauches et droites sont modifiées comme suit :

$$\begin{cases} L_i = R_{i-1} \\ R_i = L_{i-1} \end{cases}$$

- ✓ Permutation finale : Inverse de la permutation initiale (IP^{-1}).c'est le résultat de la 16^{eme} itération, R_{16} L_{16} qui est utilisé comme entrée de la fonction IP^{-1}(Tab.III.5)

40	8	48	16	56	24	64	32	39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30	37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28	35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26	33	1	41	9	49	17	57	25

Tab. III.4 Permutation finale

- ➢ **Génération des clés**

A chacune des itérations de la fonction f de l'algorithme, on utilise une clé différente K_i ($1 \leq i \leq 16$) de 48 bits.

La clé initiale est composé de 64 bits, réduit ensuite à 56 bits en éliminant les bits de parité (8, 16,… ,64),et cela a l'aide de la méthode suivante

```
private char[] permutation_compressive(char[] cle_tour) { //selectionner
48 bits parmi les 56 et les permuter selon la table permute compressive
            char []cle_compre
sse =new char[taille_cle_compresse];

             for (int i=0;i<taille_cle_compresse;i++)

cle_compresse[i]=cle_tour[(table_perm_compressive[i]-1)];//ordonner les
48 bits selon la table compresse
          return cle_compresse;
            }
```

Programme III.6 : Méthode *permutation_compressive()*

Les 56 bits résultants subissent une permutation à l'aide de la fonction PC-1, comme décrit dans la table suivante :

57	49	41	33	25	17	9	1	58	50	42	34	26	18

10	2	59	51	43	35	27	19	11	3	60	52	44	36
63	55	47	39	31	23	15	7	62	54	46	38	30	22
14	6	61	53	45	37	29	21	13	5	28	20	12	4

Tab. III.5 Table PC-1

Le résultat de PC-1(K) est ensuite séparé en deux moitiés égales de 28 bits : C0 et D_0 ; Les blocs C_i et D_i subissent un décalage circulaire vers la gauche comme suit :

$$\begin{cases} C_i = LS_i\,(C_i\text{-}1),\ i=1..16 \\ D_i = LS\,(D_{i-1}) \end{cases}$$

Avec LS_i la fonction de décalage vers la gauche de la clé, selon la ronde, elle est ainsi définie :

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Tab. III.6 Décalage de la clé par ronde

Après décalage, 48 parmi 56 bits sont sélectionnées à l'aide de la fonction de permutation compressive PC-2. La valeur finale de la sous-clé est donnée par : $K_i = PC - 2\,(C_i\,D_i)$;

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Tab. III.7 Table PC-2

III.4.2 Déchiffrement

Le même algorithme est également utilisé pour le déchiffrement. Les seuls différences sont que les sous-clés doivent êtres utilisées dans l'ordre inverse ($K_{16}, K_{15}, K_{14}, \ldots\ldots, K_1$).Dans l'algorithme de génération des sous-clés, le décalage se fait cette fois-ci vers la droite et devient à chaque ronde : 0,1, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2,1.

La méthode suivante nous permet d'effectuer le décalage de la clé

```
char []Decalage_circulaire_droit (int tour,char[] demi){

        char []demi_media=new char[taille_cle_reduite/2];
        System.arraycopy(demi, 0, demi_media, 0, demi.length);


        for(int i=0;i<deca_cir_droit[tour];i++)demi[i]=demi_media[demi.length-
deca_cir_droit[tour]+i];
        for(int i=deca_cir_droit[tour];i<demi.length;i++)
        demi[i]=demi_media[-deca_cir_droit[tour]+i];

         return demi;
```

Programme III.7 : ***Decalage_circulaire_droit()***

III.5 Sécurité du DES

Le DES a été le sujet de nombreuses controverses. Toute la sécurité du système repose sur deux éléments : la fonction de confusion f, et les S-Boxes à l'intérieur de celle-ci. Ces S-Boxes ("Boîtes S") sont huit tables de substitution de 4 par 16 entiers compris entre 0 et 15. Une rumeur affirmant que la NSA (qui, rappelons-le, a finalisé l'algorithme) aurait mis en place des trappes, faiblesses permettant de décrypter n'importe quel texte sans en connaître la clé, tout en assurant que l'algorithme était parfaitement sûr.

En 1994, Don Coppersmith, un ingénieur d'IBM, dévoila un secret gardé depuis l'époque de la conception de DES. L'attaque-T (pour *Tickling attack*) est une variante de la cryptanalyse différentielle. Pendant une vingtaine d'années, IBM a gardé le silence sur cette découverte, révélant pourtant une faiblesse conséquente dans le système. Les tests révélèrent cependant que les tables de substitution avaient été renforcées par la NSA (et non pas affaiblies comme le disait la rumeur).

De nombreuses variantes du DES existent en vue d'améliorer la sécurité, parmi elle le triple DES et L'AES **(**Advanced Encryption Standard).

III.6 Triple DES

Le Triple DES (aussi appelé 3DES) est un algorithme de chiffrement symétrique par bloc, enchaînant 3 applications successives de l'algorithme DES sur le même bloc de données de 64 bits, avec 2 ou 3 clés DES différentes.

Le Triple DES est généralement utilisé avec seulement deux clés différentes. Le mode d'usage standard est de l'utiliser en mode EDE (Encryption, Decryption, Encryption, c'est-à-dire Chiffrement, Déchiffrement, Chiffrement) ce qui le rend compatible avec DES quand on utilise trois fois la même clé.

En écriture mathématique :

$C= E_{k3} (D_{k2} (E_{k1} (M)))$

$M= D_{k1} (E_{k2} (D_{k3}(C)))$

Avec : M : texte en clair ;

C : texte chiffré ;

E : fonction de chiffrement DES ;

D : fonction de déchiffrement DES ;

III.6.1 Modes d'exploitation de DES triple

Selon le mode d'exploitation du DES triple distingue :

- ✓ **TECB :** chiffrement du texte en clair 5M) en entier 3 fois avec l'algorithme DES : $C=E_{K3} (D_{K2} (E_{K15}(M)))$; $M= (D_{K1} (E_{K2} (D_{K3} (C)))$
- ✓ **TCBC :** On peut chiffrer le texte en clair entier en mode CBC et ceci de deux manière :
  - CBC externe : utilise un seul IV (initial value) C0 : $Ci = E_{K3} (D_{K2} (E_{K1} (M \oplus Ci\text{-}1)))$
  - CBC interne : utilise 3 IV différentes : T0, S0, C0 $Ti = E_{K1} (M \oplus T_{i-1})$

$Si=D_{K2}$ (Ti Si_{-1})

$Ci=E_{K3}$(Si Ci-1)

- ✓ **TCFB :** xi texte en clair ;

Initialement, K0=y0=IV(de 64 bits) ;
On chiffre IV avec DES K1 (IV) =res1 ;
On déchiffre le res1 avec DES^{-1} (res1) = res2 ;
On chiffre le res_2 avec :DESK3(res2)= res3 ;
Ensuite :
Xi ⊕ res3= yi (le résultat du 1^{er} bloc chiffré) ;
Yi constitue ensuite le IV du 2^{eme} chiffrement

- ✓ **TCF-P**

Dans ce mode, on utilise 3 vecteurs d'initiation, 3 clés, 64 bits texte en clair. Le chiffrement et le déchiffrement se font en parallèle.

III.7 Advanced Encryption Standard (AES)

Comme son nom l'indique, un standard de cryptage symétrique algorithmes destiné à remplacer le DES (Data Encryption Standard) qui est devenu trop faible au regard des attaques actuelles.

L'AES opère sur des blocs de 128 bits (plaintext P) qu'il transforme en blocs cryptés de 128 bits (C) par une séquence de Nr opérations ou "rounds", à partir d'une clé de 128, 192 ou 256 bits. Suivant la taille de celle-ci, le nombre de rounds diffère : respectivement 10, 12 et 14 rounds.

III.7.1 points forts de l'AES

- ✓ Sécurité ou l'effort requis pour une éventuelle cryptanalyse.
- ✓ facilité de calcul : cela entraine une grande rapidité de traitement
- ✓ besoins en ressources et mémoire très faibles

- ✓ flexibilité d'implémentation: cela inclut une grande variété de platformes et d'applications ainsi que des tailles de clés et de blocs supplémentaires (c.f. ci-dessus).
- ✓ hardware et software : il est possible d'implémenter l'AES aussi bien sous forme logicielle que matérielle (cablé)
- ✓ simplicité : le design de l'AES est relativement simple

III.8 Conclusion

Dans ce chapitre on a présenté le fonctionnement détaillé de l'algorithme DES,ainsi que ses points forts et faibles ,aussi les améliorations apporté sur ce dernier.

Nous implémenterons dans le chapitre qui va suivre la conception de notre application qui est basé sur l'algorithme DES.

Chapitre IV

Analyse et conception

IV.1 Introduction

Avant toute réalisation d'une application informatique, il convient de suivre une démarche méthodologique et rigoureuse pour planifier et concevoir l'application, en mettant en évidence tous les objectifs tracés pour la bonne élaboration du projet souhaité.

Dans notre cas nous avons opté pour la modélisation orientée objet avec le langage UML car il permet de modéliser de manière claire et précise la structure et le comportement d'un système indépendamment de tout langage de programmation.

IV.2 Généralités sur l'UML

IV.2.1 Historique

UML ("**U**nified**M**odeling **L**anguage") est né de la fusion des trois méthodes qui ont le plus influencées la modélisation objet au milieu des années 90 : ***OMT***, ***Booch***et ***OOSE***.

UML a été inventé par des experts reconnus. En quelques années, UML est devenu un langage incontournable.

L'approche objet est pourtant loin d'être une idée récente. Les premiers compilateurs ***C++*** datent du début des années80 et de nombreux langages orientés objets "académiques" ont étayés les concepts objets (***Eiffel***, ***Objective C***,***Loops***...).

L'unification et la normalisation des méthodes objet dominantes (OMT, Booch et OOSE) ne datent que de1995. UML est le fruit de cette fusion. UML, ainsi que les méthodes dont il est issu, s'accordent sur un point : une analyse objet passe par une modélisation objet. UML permet donc de modéliser une application selon une vision objet.

Fin 1997, UML est devenu une norme ***OMG*** (**O**bject **M**anagement **G**roup).L'OMG est un organisme créé en 1989 à l'initiative de grandes sociétés (***HP***, ***Sun***, ***Unisys***,***American Airlines***, ***Philips***...).

Pour sortir les technologies objet de cette impasse fatale, l'OMG propose UML.

UML comble une lacune importante des technologies objet. Il permet d'exprimer et d'élaborer des modèles objet, indépendamment de tout langage de programmation. Mais UML offre bien plus encore ! C'est un langage formel, qui décrit de manière très précise tous les éléments de modélisation et la sémantique de ces éléments. En d'autres termes : UML normalise les concepts objet.

UML est avant tout un support de communication performant, qui facilite la représentation et la compréhension de solutions objet. Son indépendance par rapport aux langages de programmation, aux domaines d'application et aux processus, en font un langage universel. UML est donc bien plus qu'un simple outil qui permet de "dessiner" des représentations mentales... Il permet de parler un langage commun, normalisé mais accessible, car visuel. Il représente un juste milieu entre langage mathématique et naturel, pas trop complexe.

IV.2.2 Diagrammes UML

Un diagramme est la représentation graphique d'un ensemble d'éléments qui constituent le système**.**

Les **13** diagrammes UML sont dépendants hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie :

- **Diagrammes structurels ou statiques**
  - **Diagramme de classes** (Class diagram) : il représente les classes intervenant dans le système
  - **Diagramme d'objets** (Object diagram) : il sert à représenter les instances de classes (objets) utilisées dans le système.
  - **Diagramme de composants** (Component diagram) : il permet de montrer les composants du système d'un point de vue physique, tels qu'ils sont mis en œuvre (fichiers, bibliothèques, bases de données...)
  - **Diagramme de déploiement** (Deployment diagram) : il sert à représenter les éléments matériels (ordinateurs, périphériques, réseaux, systèmes de stockage...) et la manière dont les composants du système sont répartis sur ces éléments matériels et interagissent entre eux.
  - **Diagramme des paquetages** (Package diagram) : un paquetage étant un conteneur logique permettant de regrouper et d'organiser les éléments dans le

modèle UML, le Diagramme de paquetage sert à représenter les dépendances entre paquetages, c'est-à-dire les dépendances entre ensembles de définitions.

- **Diagramme de structure composite** (Composite Structure Diagram) : depuis UML 2.x, permet de décrire sous forme de boîte blanche les relations entre composants d'une classe

❖ **Les diagrammes comportementaux** (Behavior Diagram) rassemblent :

- **Diagramme des cas d'utilisation** (use-cases ou Use Case Diagram) : il permet d'identifier les possibilités d'interaction entre le système et les acteurs (intervenants extérieurs au système), c'est-à-dire toutes les fonctionnalités que doit fournir le système.
- **Diagramme états-transitions** (State Machine Diagram) : permet de décrire sous forme de machine à états finis le comportement du système ou de ses composants.
- **Diagramme d'activité** (Activity Diagram) : permet de décrire sous forme de flux ou d'enchaînement d'activités le comportement du système ou de ses composants.

❖ **Les diagrammes d'interaction ou dynamiques** (Interaction Diagram) rassemblent :

- **Diagramme de séquence** (Sequence Diagram) : représentation séquentielle du déroulement des traitements et des interactions entre les éléments du système et/ou de ses acteurs.
- **Diagramme de communication** (Communication Diagram) : depuis UML 2.x, représentation simplifiée d'un diagramme de séquence se concentrant sur les échanges de messages entre les objets.
- **Diagramme global d'interaction** (Interaction Overview Diagram) : depuis UML 2.x, permet de décrire les enchaînements possibles entre les scénarios préalablement identifiés sous forme de diagrammes de séquences (variante du diagramme d'activité).
- **Diagramme de temps** (Timing Diagram) : depuis UML 2.x, permet de décrire les variations d'une donnée au cours du temps.

IV.3 But et contexte de la plate forme

Le système doit offrir une IHM (L'interface homme-machine) simple et facile à l'utilisation.

La fenêtre principale permet à l'utilisateur de crypter des fichiers (par exemple le Word, pdf, image….ect) comme il peut aussi crypter/décrypter des messages et cela en utilisant une clé K pour but de protéger le fichier ou le message contre :

- ✓ une mauvaise réception (confidentialité) : il devrait être impossible d'obtenir le message ou le fichier en clair à partir du message ou du fichier crypté sans connaître la clé K.
- ✓ une mauvaise émission (authentification) : il devrait être impossible de substituer un autre message ou fichier crypté sans connaître la clé K.

IV.4 Diagramme de contexte

Un diagramme de contexte permet de représenter les différentes acteurs, qui interagissent et collaborent dans le système. Dans notre cas on a un seul acteur qui est l'utilisateur ; le diagramme ci-dessus illustre le cas de notre application :

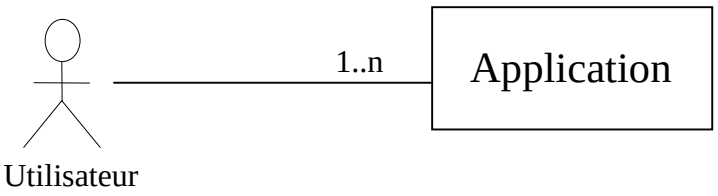


Fig. IV.1 Diagramme de contexte

IV.5 Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation est un graphe d'acteurs, un ensemble de cas d'utilisation englobés par la limite du système, des associations de communication entre les acteurs et les cas d'utilisation, et des généralisations entre cas d'utilisation.

.

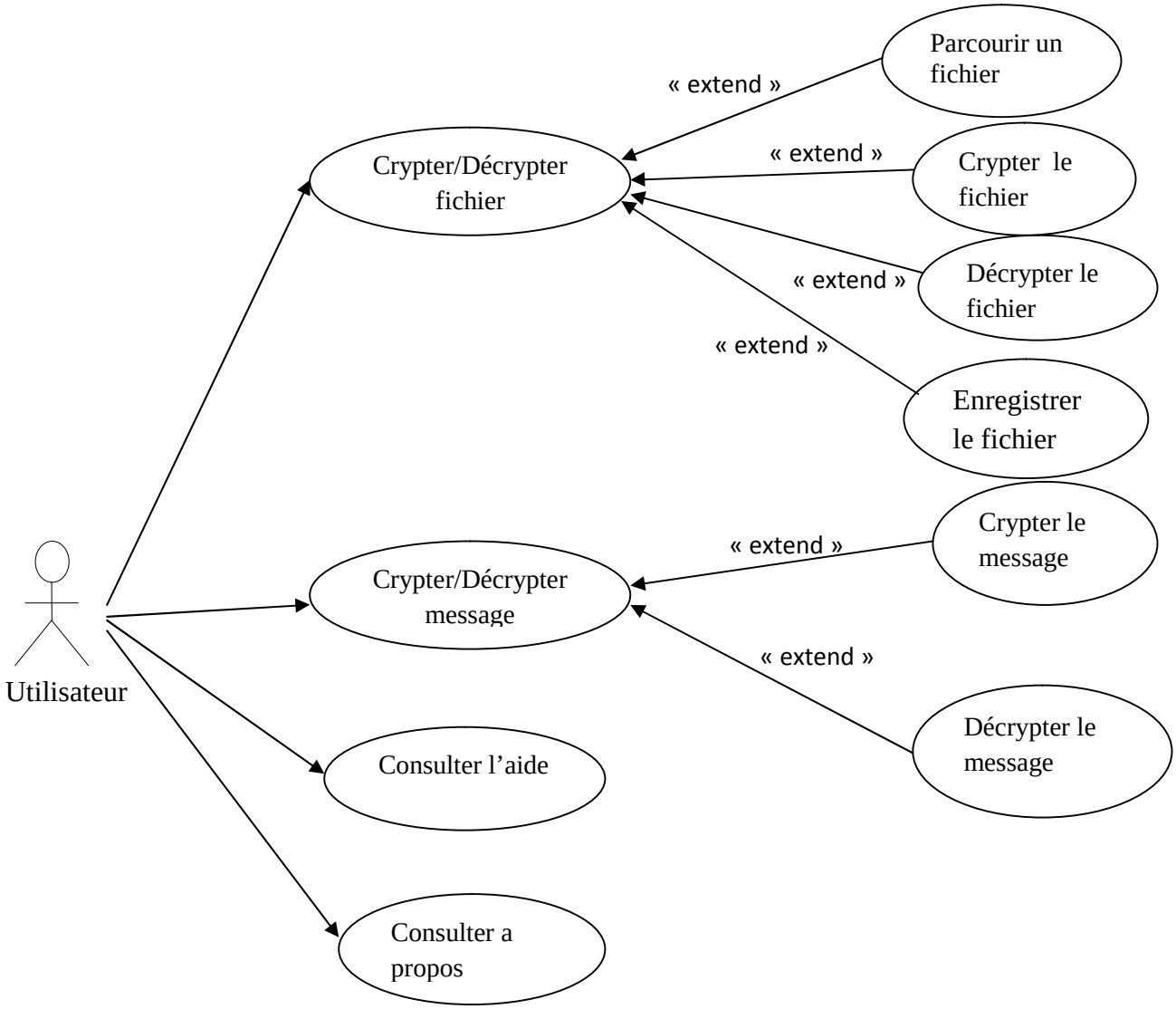


Fig. IV.2 Diagrammes de cas d'utilisation

IV.6 Diagramme de séquence détaillé

Le diagramme de séquence permet de représenté séquentiellement le déroulement des traitements et des interactions entre les éléments du système et ses acteurs.

IV.6.1 Diagramme de séquence détaillé du cas d'utilisation «Crypté un message »

Ce diagramme déroule selon les étapes suivantes :

1. L'utilisateur atteint l'application, le système lui affiche la fenêtre principale de cette dernière.
2. Clique sur le bouton « crypté/décrypté message » et une autre fenêtre s'affiche.
3. L'utilisateur saisit le message a crypté et une clé et clique sur le bouton « crypté » le système lui retourne le message crypté avec une page de confirmation.

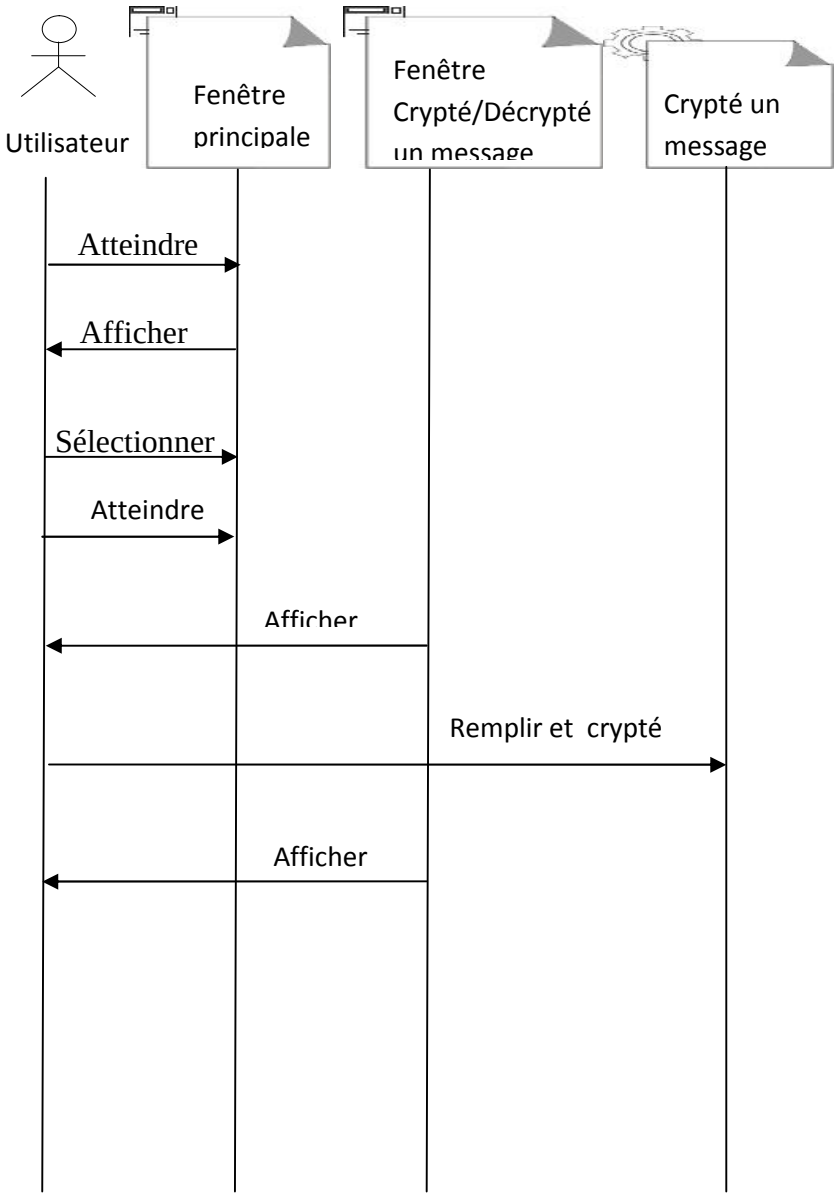


Fig. IV.3 Diagramme de séquence détaillé du cas d'utilisation « Crypté un message »

III.6.2 Diagramme de séquence détaillé du cas d'utilisation «Consulter l'aide»

Ce diagramme déroule selon les étapes suivantes :

1. L'utilisateur atteint l'application, le système lui affiche la fenêtre principale de cette dernière.
2. Clique sur le lien aide dans la barre de menu de la fenêtre principale, une page qui fournit l'aide de l'utilisation de l'application s'affiche.

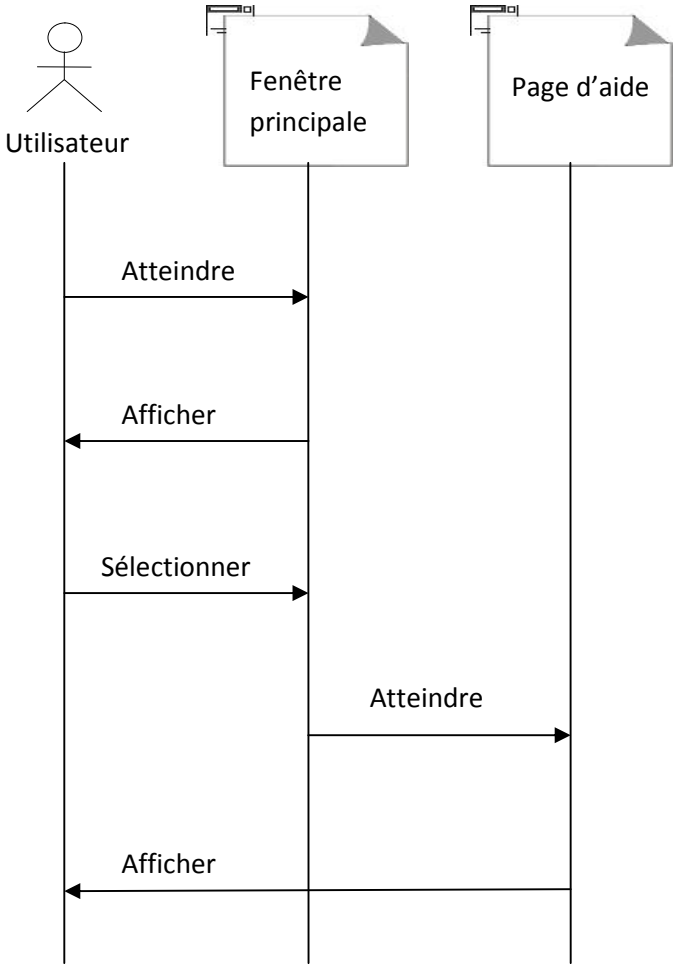


Fig.IV.4 Diagramme de séquence détaillé du cas d'utilisation « Consulter l'aide »

III.6.3 Diagramme de séquence détaillé du cas d'utilisation «Crypté un fichier»

Ce diagramme déroule selon les étapes suivantes :

1. L'utilisateur atteint l'application, le système lui affiche la fenêtre principale de cette dernière.
2. Clique sur le bouton « crypté/décrypté fichier » une autre fenêtre s'affiche.
3. L'utilisateur sélectionne un fichier a crypté en cliquant sur le bouton « parcourir ».
4. Saisit une clé et clique sur le bouton « crypté », quand le système termine le cryptage du fichier, il a affiche une fenêtre qui informe l'utilisateur que le fichier est crypté et qu'il doit cliquer sur enregistrer pour le sauvegarder.
5. L'utilisateur clique sur enregistrer et sélectionne son propre répertoire ou le fichier crypté sera sauvegarder.

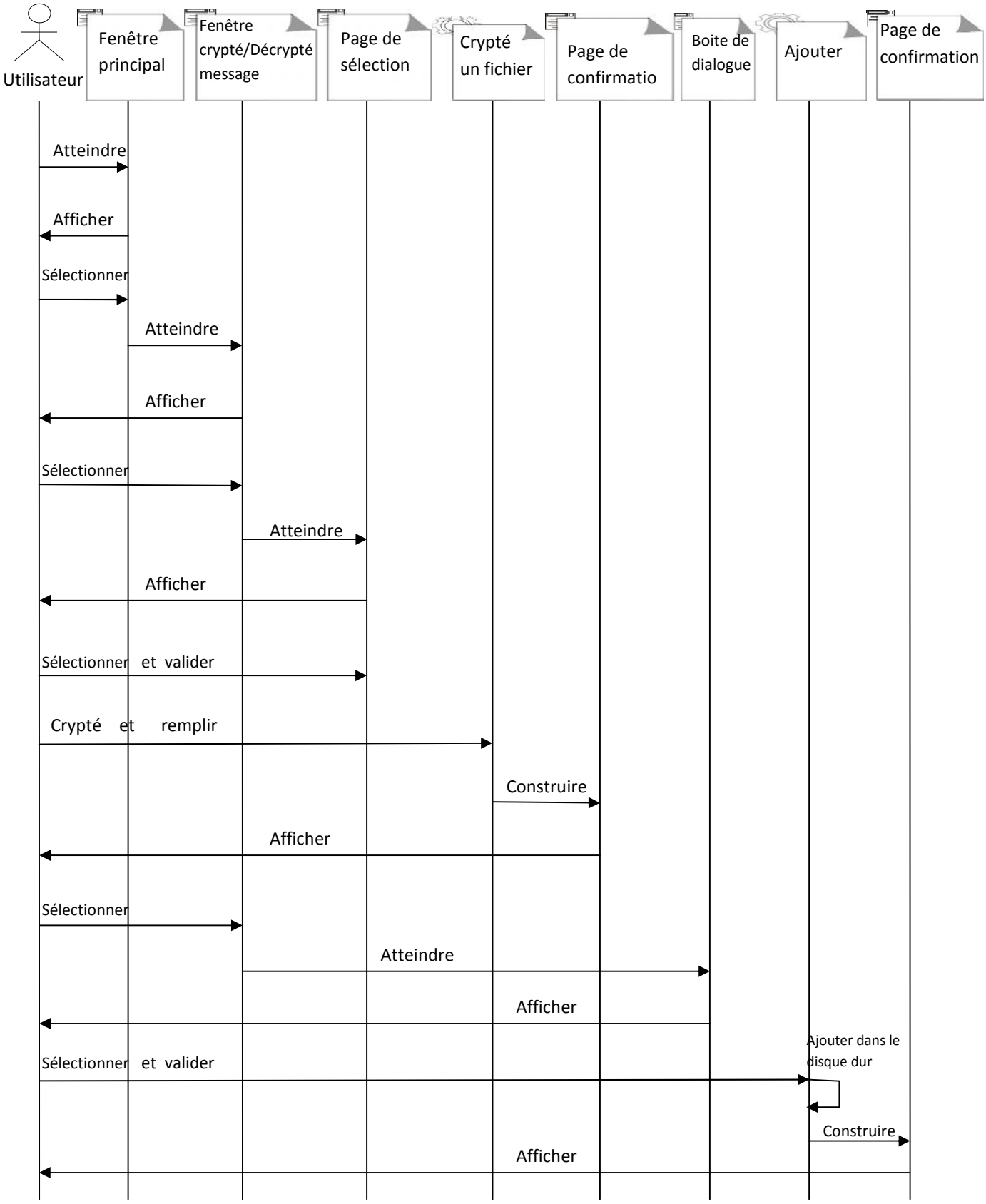


Fig.IV.5 Diagramme de séquence détaillée du cas d'utilisation « Crypté un fichier»

IV.7 Le diagramme de classe de l'application

Le diagramme de classe est une représentation statique des éléments qui composent un système et de leurs relations. L'intérêt majeur des diagrammes de classe est de modéliser les entités d'un système.

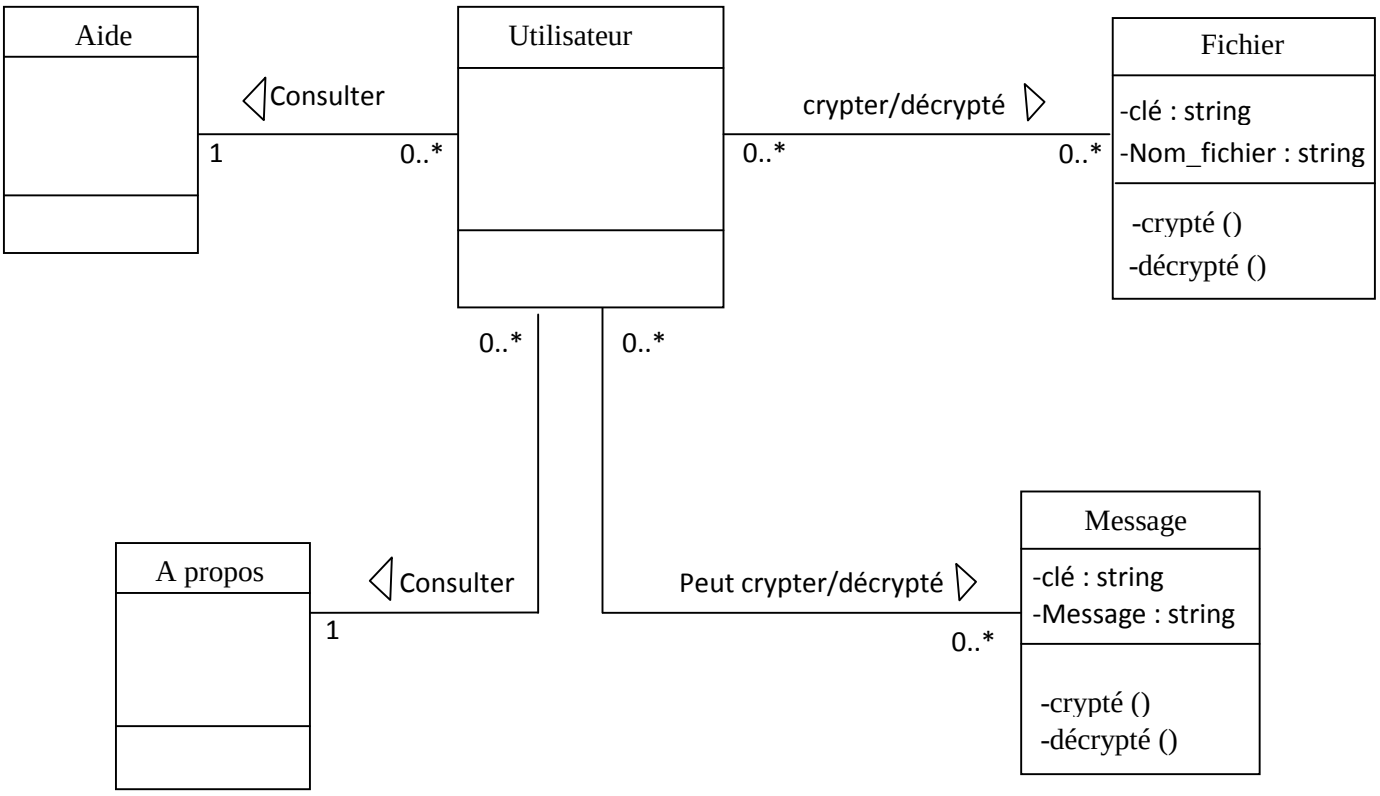


Fig. IV.6 Diagramme de classe

IV.8 Conclusion

Ce quatrième chapitre, nous avons proposé une démarche de modélisation pour développer notre application, en se basant sur la méthode UML qui nous a permis de présenter en détails la solution à implèmenter.il reste à définir l'environnement de développement et la réalisation de notre application, ce qui sera l'objet du chapitre suivant.

Chapitre V

Réalisation

V.1. Introduction

Après avoir présenté dans le chapitre précédent les différentes étapes d'analyse et de conception, nous allons présenter dans ce dernier chapitre l'environnement de développement, les outils qui ont servi à la réalisation de notre application, et nous terminerons par la présentation de ses fonctionnalités à travers ses différentes interfaces.

V.2 Outils de développement

Durant notre réalisation nous avons opté pour l'utilisation de la plate-forme Windows avec son système d'exploitation Windows 7, le logiciel Eclipse.

V.2.1 Le langage Java

Java est un langage de programmation orienté objet issu de la synthèse de plusieurs langages de programmation. En effet il s'inspire de C++ et des techniques approuvées en Small talk et autres langages (organisation en classe, utilisation d'un ramasse-miettes (*garbagecollector*), exécution à l'aide d'une machine virtuelle, gestion d'exception…), tout en supprimant les inconvénients de la plupart de ces langages.

Le principal atout de java réside dans le fait que ce dernier est interprété et indépendant de toute plate-forme. En effet la source est compilée en pseudo code ou bytecode puis exécutée par un interpréteur Java appelé Java Virtual Machine (JVM). Ce concept est à la base du slogan Sun pour Java WORA (Write Once, RunAnywhere). En effet, le bytecode, s'il ne contient pas de code spécifique à une plate-forme particulière peut être exécuté et obtenir les même résultats sur toute machine disposant d'une JVM, qui existe pour la plupart des systèmes d'exploitation.

De plus il n'y a pas de compilation spécifique pour chaque plate-forme. Le code reste indépendant de la machine sur laquelle il s'exécute. Il est donc possible d'exécuter java sur tous les environnements qui possèdent la JVM. Cette indépendance est assurée au niveau du code source grâce à un Unicode et au niveau du bytecode.

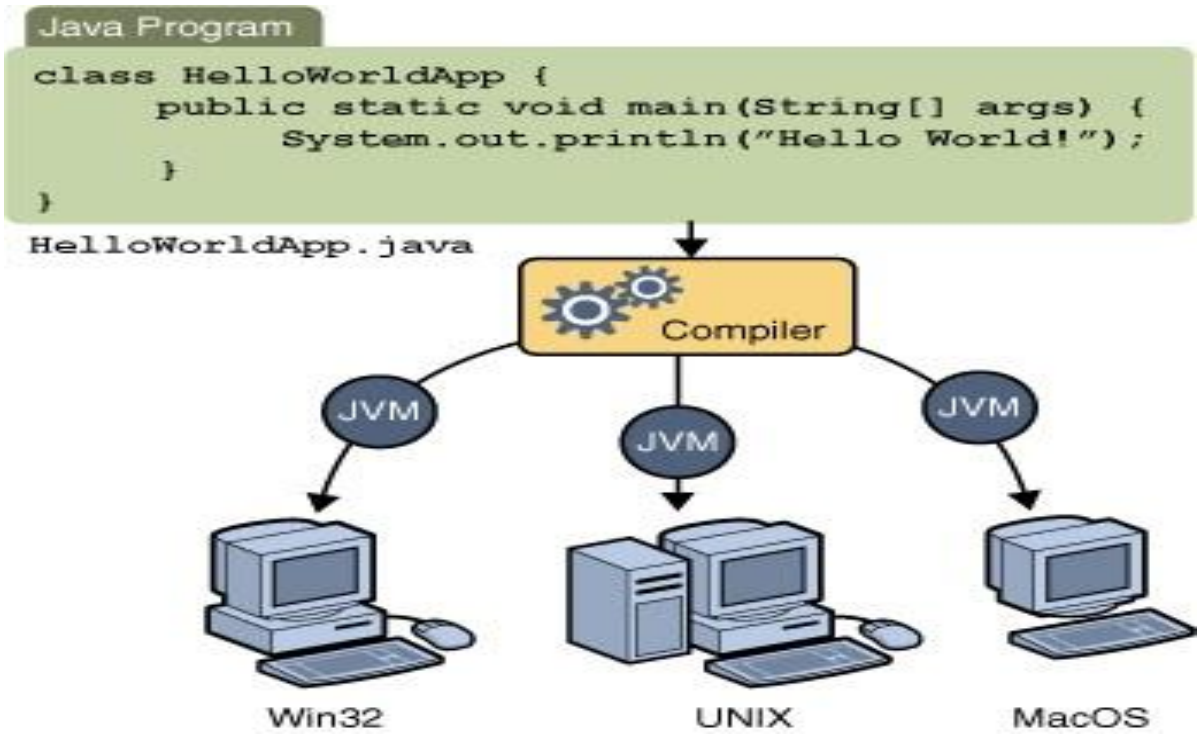


Fig.V.1. Indépendance d'un programme en java de toute plate-forme

V .2.2 Eclipse

Eclipse est un environnement de développement intégré (IDE) dont le but est de fournir une plate-forme modulaire pour permettre de réaliser des développements informatiques.

I.BM. est à l'origine du développement d'Eclipse qui est d'ailleurs toujours au cœur de son outil WebSphere Studio Workbench (WSW), lui-même à la base de la famille des derniers outils de développement en java d'I.BM.

L'image suivante présente l'interface de travail sous Eclipse.

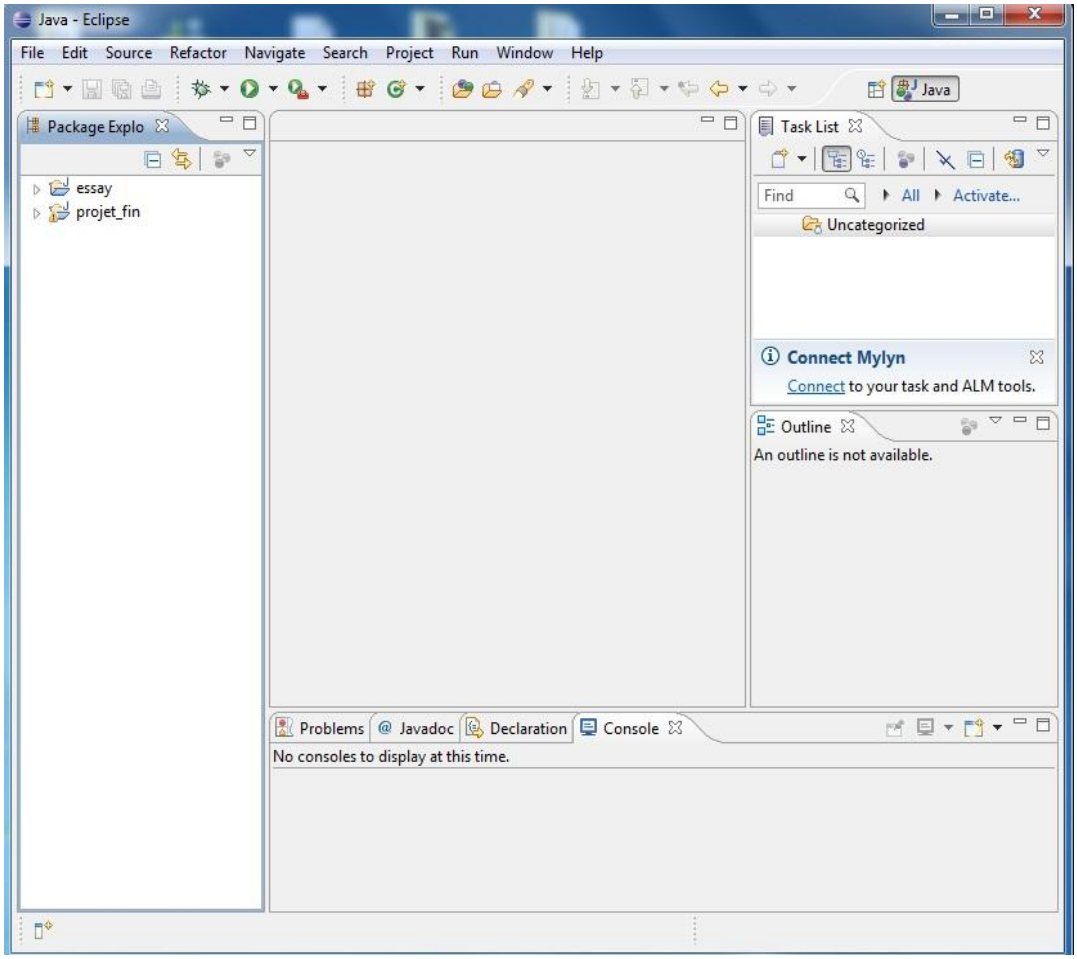


Fig. V.2. Capture d'écran présentant l'interface de développement d'Eclipse.

V.3. Présentation de quelques interfaces de l'application

V.3.1 Menu principal

Au lancement de notre application, un menu principal s'affiche à l'écran, il donne un aperçu des fonctionnalités qu'offre celle-ci.

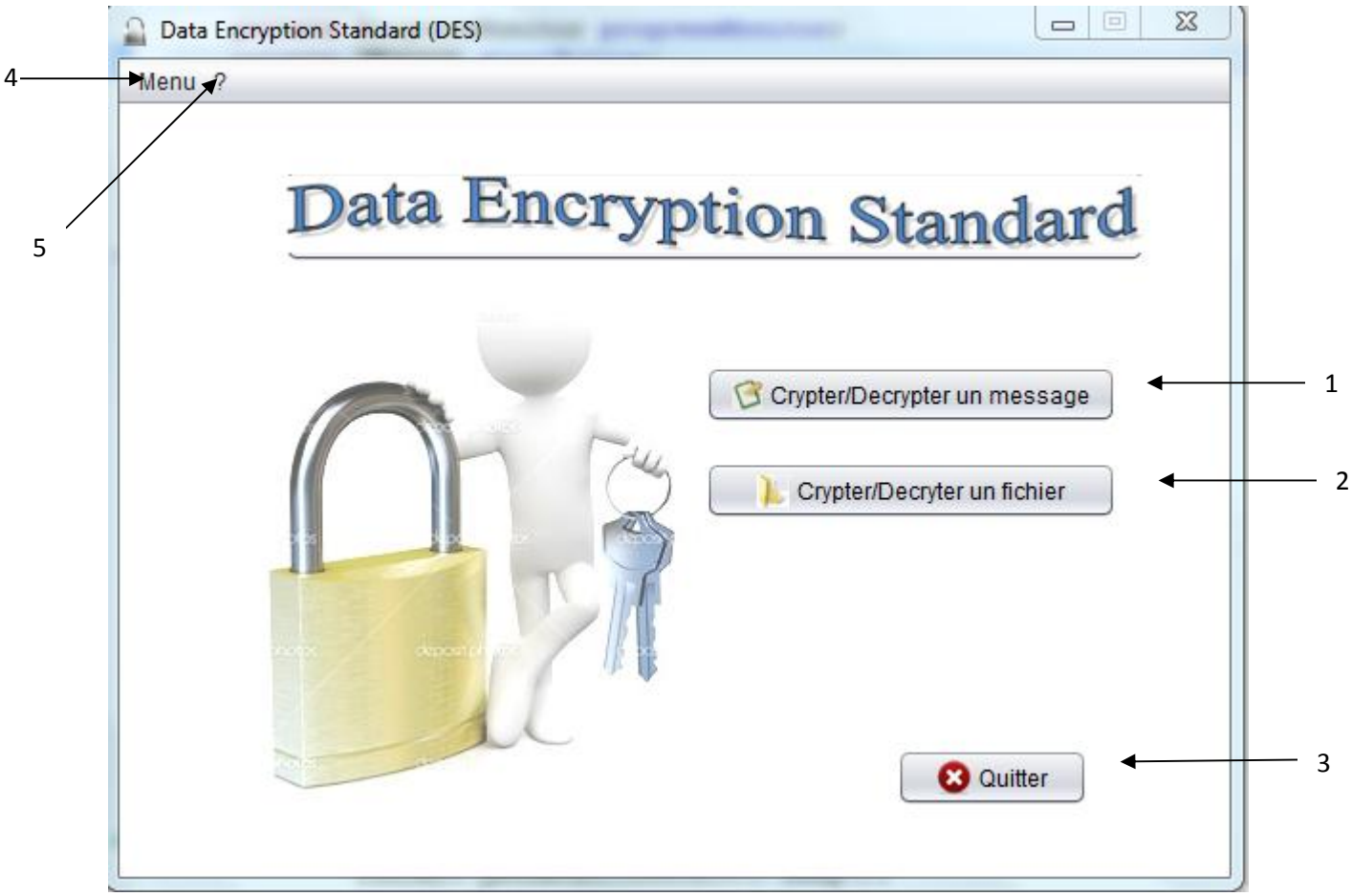


Fig. V. 3. Menu principal

(1) : bouton pour accéder à la fenêtre de Cryptage/Décryptage des messages.
(2) : bouton pour accéder à la fenêtre de Cryptage/Décryptage des fichiers.
(3) : bouton pour quitter l'application.
(4) : Affiche le menu.
(6) : Affiche le menu d'aide (?).

Fig. V. 4. Menu de la fenêtre principale

Ce menu est composé des mêmes boutons vus précédemment.

Fig.V. 5. Menu Aide de la fenêtre principale

(1) : Cette option permet d'accéder à la fenêtre d'aide.

(2) : Cette option permet d'accéder à la fenêtre d'a propos.

V.3.2 Page d'aide de l'application

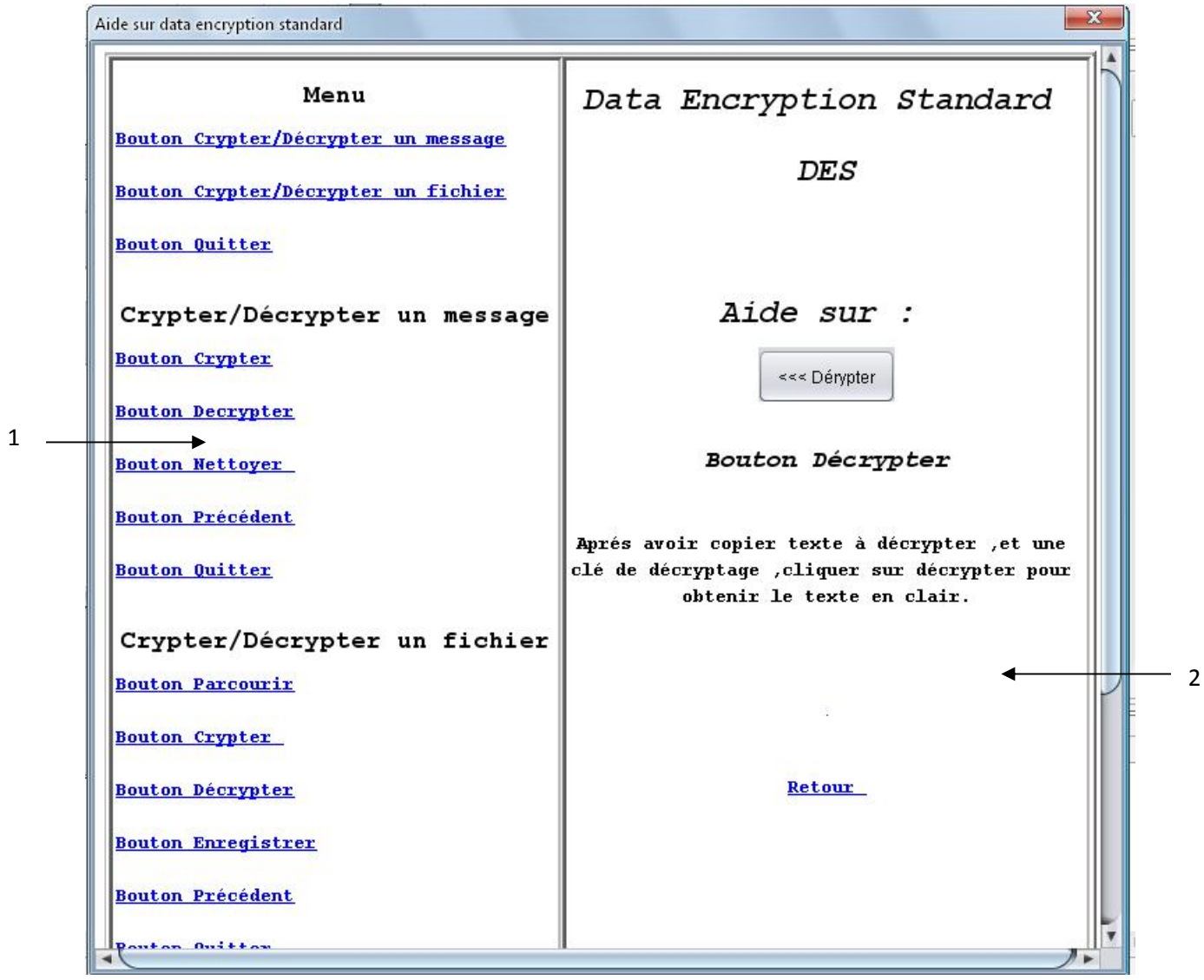


Fig. V.6. Page d'aide de l'application

(1) : Sommaire d'aide.

(2) : Détail de la fonctionnalité sélectionnée dans le sommaire.

V.3.3 Page Crypter/Décrypter un message

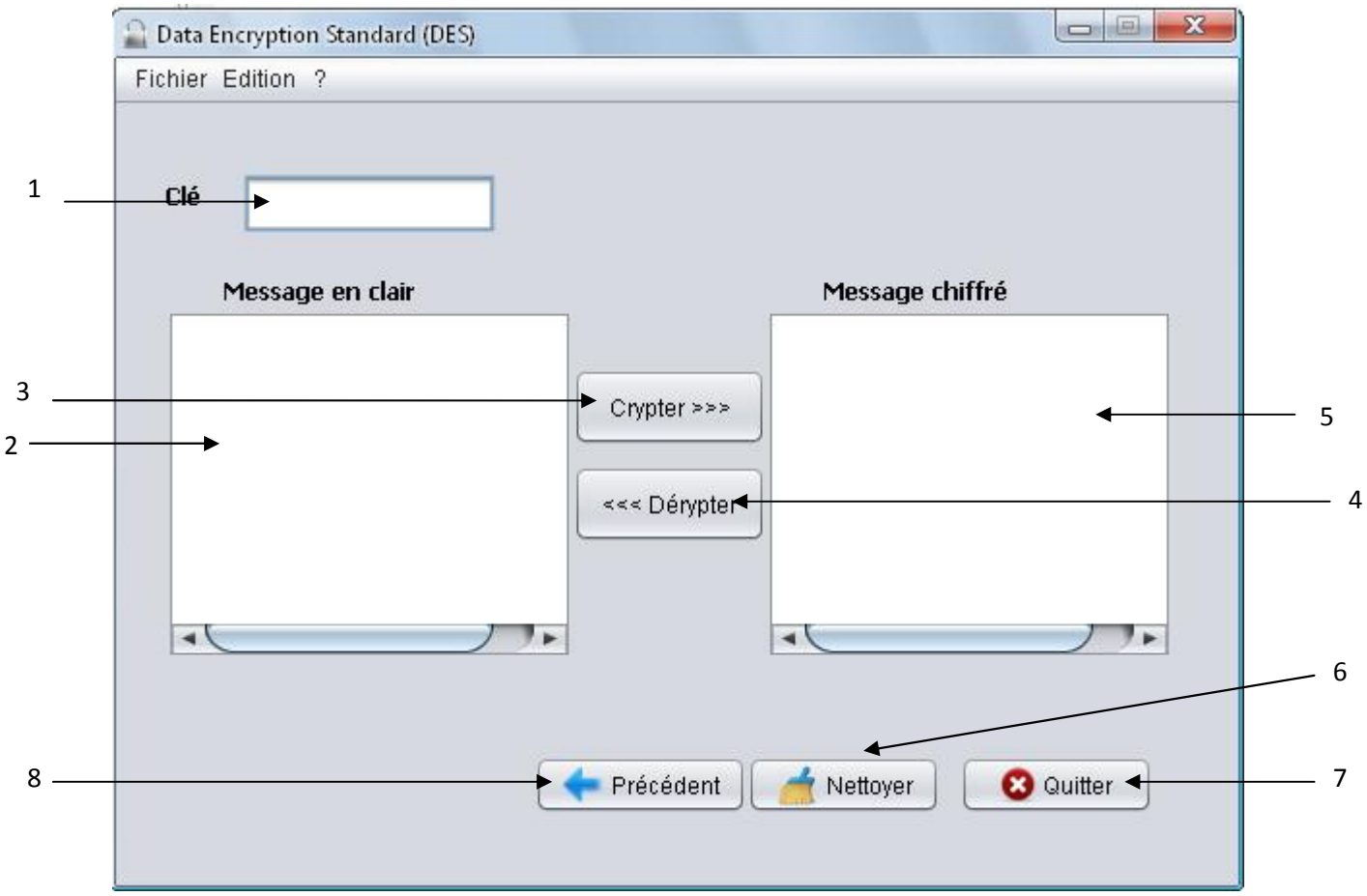


Fig. V. 7. Page Crypter/Décrypter un message.

(1) : Saisir la clé.

(2) : Affiche le message à en clair.

(3) : Crypter le message en clair

(4) : Décrypter le message chiffré.

(5) : Affiche le message crypté.

(6) : Effacer toutes les zones de texte.

(7) : Quitter l'application.

(8) : Revenir au menu principal.

Si l'utilisateur ne saisit pas la clé, alors l'application renvoi le message d'erreur suivant

V.3.4 Page Crypter/Décrypter un fichier

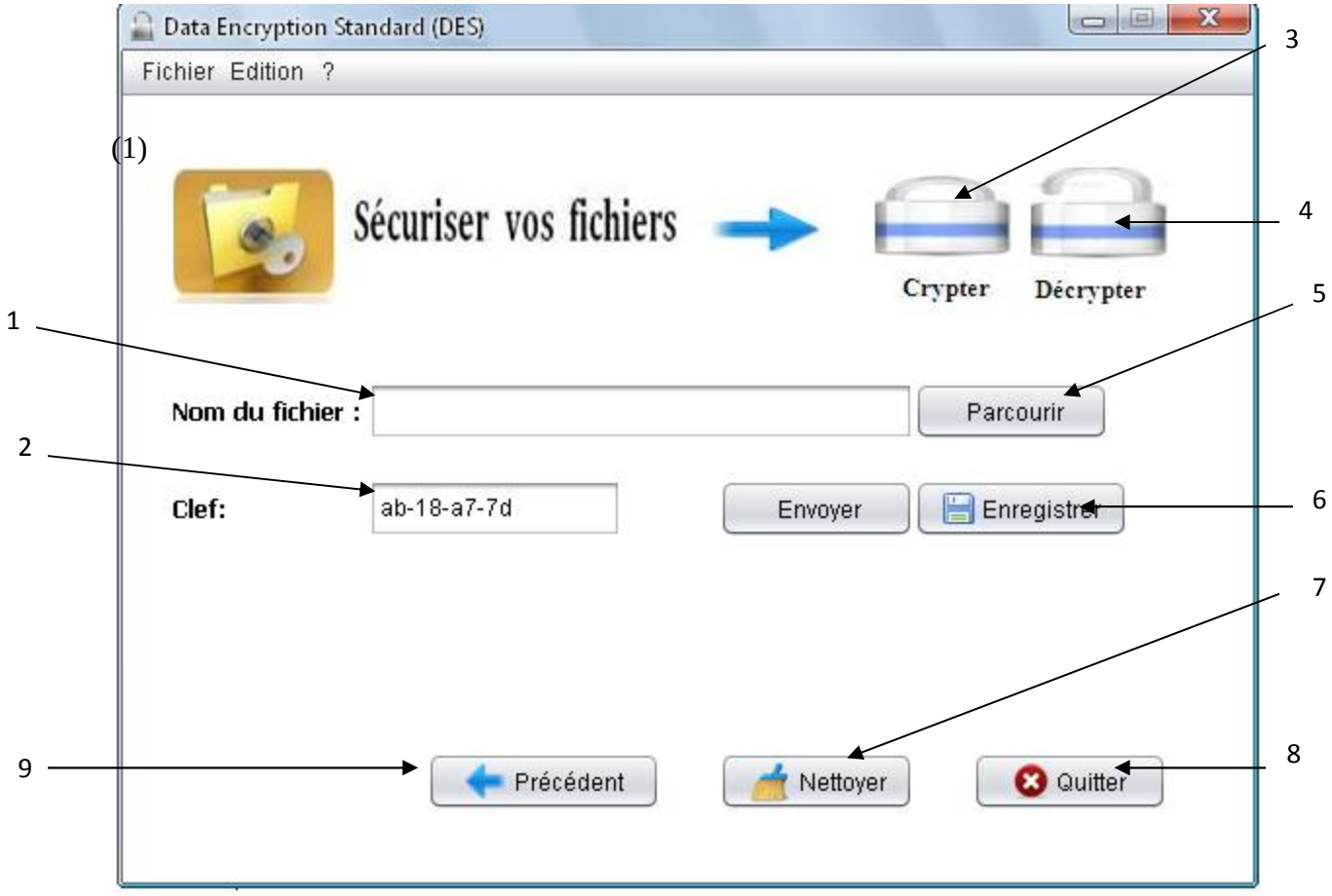


Fig. IV. 8. Page Crypter/Décrypter un fichier.

(1) : Affiche le nom du fichier parcouru.

(2) : Saisir la clé.

(3) : Crypter le fichier.

(4) : Décrypter le fichier.

(5) : Parcourir le fichier à crypter ou à décrypter.

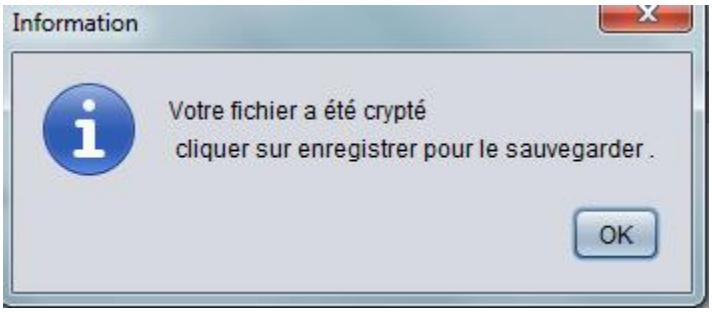
(6) : Enregistrer le fichier crypté ou décrypté.

(7) : Effacer les zones de textes

(8) : Quitter l'application.

(9) : Revenir au menu principal.

Quand le système termine le cryptage d'un fichier, alors l'application renvoi le message informatif suivant



V. 4 Conclusion

Dans cette dernière partie de notre projet, nous avons présenté les différents outils du développement qui nous a permis de réaliser notre application ainsi que ses interfaces essentielles en mettant en évidence le rôle de chaque module pour satisfaire au mieux les objectifs d'un système cryptographique.

Conclusion générale

Depuis toujours la sécurité réseau est un facteur le plus sérieux que connaissent les entreprises dotées d'un réseau informatique. Il ne sera jamais possible de sécuriser totalement un système d'information, car il y'aura toujours des hackers pour découvrir des nouvelles failles dans le système, mais on peut toujours rendre une solution plus difficile en appliquant des nouvelles approches, de ce fait, des améliorations peuvent être apportées à notre travail en intégrant des algorithmes (TDES, AES,…), cela nous permettra d'avoir un haut niveau de sécurité des données.

Le travaille que nous avons mené, nous a permis nous initier dans le domaine de la cryptographie et sur la programmation en java.

Nous tenons à souligner que la cryptographie est large domaine de recherche et un domaine sensible car il concerne la sécurité des données. Tout cryptosystème conçu doit donc être testé plusieurs fois par des cryptographes pour prétendre à une bonne sécurisation de l'information.

Enfin, nous souhaitons que d'autres étudiants puissent tirer profit de ce travail pour concevoir et réaliser des cryptosystèmes assurant une bonne sécurité de l'information.

Bibliographie

Documents écrits

[1] : Michel Riguidel, Sécurité informatique et reseaux, PARIS 2006.

[2] : Olivier Salvatori, Jaques Nozick, Les réseaux, 5eme édition,PARIS 2006.

[3] : C.Llorens, L.Levier,D.Valois , Tableaux de bord de la sécurité réseau2eme edition,PARIS2006.

[4] : CCNA Exploration 4.0,commutation de reseau local et réseau local sans fil

[5] : Amakou Mbata,Olivier persent ,Firewall,parfeu,mur de feu ;PARIS decembre 2006

[6] : une introduction à la cryptographie ; support du cour ,universitées de nante

[7] :Eric ALATA, Observation, caractérisation et modélisation de processus d'attaques sur Internet, thèse de Doctorat , l'Institut National des Sciences Appliquées de Toulouse,2007

[8] : Bruce Schneider, Cryptographie Appliquée. 2eme édition, 2001.Vulbert.

[9]: Philippe PERRET ,Cryptographie .Feverier, 2007.

[10] : Douglas Stinson ,Cryptographie ,théorie et pratique ,2eme edition,PARIS 2003.

[11] : T.ebrahimi , F.Leprévost, B.Warusfel , Cryptographie et securité des systesmes et reseaux ,January 2006.PARIS

[12] : Bruno Martin, Codage, Cryptograhie et Applications.1ere édition, 2004.Presses polytechniques et Universitaires Romandes.

[13] : Françoise levy, Cryptographie moderne,Ecole Nationale Superieure des thecniques avancées.

[14] : A. Oulamara & G.Ouarezki ,Conception et réalisation d'une application de cryptage basé sur les reseaux de feistel. These de Master, Tizi Ouzou,2012.

[15] : Pascal Lafourcade, Vérification de protocoles cryptographiques en présence de théories équationnelles, Thèse de Doctorat ÉCOLE NORMALE SUPÉRIEURE DE CACHAN,Septembre 2006.

[16]: C.Chebli, Signature et chiffrement, These de doctorat ,LIBAN 2003.

[17] : T.Peyrin, Analyse de fonctions de hachage cryptographiques, Thèse de Doctorat ENS, Versailles, novembre 2008.

Sites web

[18] : http://www.securiteinfo.com/conseils/introsecu.shtml

[19] :http://fr.wikipedia.org/wiki/Politique_de_s%C3%A9curit%C3%A9_du_r%C3%A9seauinformatique

[20] : http://securiteinformartique.wordpress.com/2011/11/12/les-types-de-piratage/

[21] : http://www.futura-sciences.com/fr/definition/t/informatique-3/d/antivirus_10999/

[22]: https://fr.wikipedia.org/wiki/Notarisation_%C3%A9lectronique

[23] : http://ram-0000.developpez.com/tutoriels/cryptographie/

[24] : http://fr.wikipedia.org/wiki/Chiffre_ADFGVX

[25] : http://fr.wikipedia.org/wiki/Mode_d%27op%C3%A9ration_%28cryptographie%29

[26] : http://www.commentcamarche.net/contents/212-signature-electronique

[27] : http://www.frameip.com/vpn/#2.2_-_Fonctionnalit%C3%A9s_des_Vpn

[28] : http://www.themanualpage.org

[29]: http://www.frameip.com/tcpip/