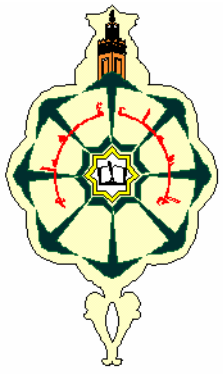
République Algérienne Démocratique et Populaire
Ministre de L'ensiegnement Supérieur et de la Recherche Scientifique

Université Abou Bekr Belkaid - Tlemcen -
Faculté des Sciences
Département de l'informatique



Mémoire

Pour l'obtention du Diplôme de
master en Informatique

.

Spécialité**: réseaux et système distribué**

Titre de Mémoire

Cryptage et sécurité des flux vidéo (modèle client/serveur)

Présenté par :Mr **Laouedj Mounir**

Soutenu le : septembre 2014

Devant le jury composé de :

Président :	**BENAMMAR . A**	Professeur, Université de Tlemcen
Encadreur :	**BENAISSA . Med**	Professeur, Université de Tlemcen
Examinateurs :	**LEHSAINI . M**	Professeur, Université de Tlemcen
	BENZIANE.Y	Professeur, Université de Tlemcen

Année Universitaire *2013- 2014*

DEDICACES

" J'ai voulu , a travers ce mémoire , rendre hommage a tous qui m’ont accompagné dans mon carrière, mes parents, mes coéquipiers mes amis et bien sur, *Mes professeurs* ..."

Remerciements

" Je remercie ALLAH le Tout-puissant de m’avoir donner le courage, la volonté et la patience de mener à terme ce présent travail…"

Laouedj Mounir

Table des matières :

Introduction générale

Les réseaux informatiques sont devenus incontournables aujourd'hui. Ils sont employés dans toutes les entreprises et même chez les particuliers. Ils permettent de mettre en oeuvre des applications très diverses, des plus simples aux plus sophistiquées. La plus connue est la navigation sur le Web, c'est-à-dire le partage d'informations grâce à Internet .

Qu'il s'agisse de réseaux locaux, de réseaux sans fil, de réseaux d'opérateurs ou de petits réseaux privés, ils obéissent tous à des principes de structuration qu'il est indispensable de comprendre. Ils utilisent une architecture en couches, dans laquelle la communication entre ordinateurs obéit à des règles précises définies par des protocoles de communication. Les protocoles les plus connus sont TCP et IP, ils ont donné leur nom à l'architecture TCP/IP.

Dans un système distribué, où les ressources sont réparties entre différents ordinateurs et équipements reliés entre eux par des réseaux, toute cette belle mécanique quasiment invisible au programmeur d'application remonte dans les couches applicatives. Les programmes d'applications doivent alors prendre en charge certains états du réseau qui affectent leurs comportements ; ils doivent se faire notifier des débuts ou des fins de travaux effectués sur d'autres équipements pour pouvoir euxmêmes se déclarer en état de terminaison.

Dans ce cas le domaine réseaux informatique est devenus plus dangereux ,perce que il ya des entreprises echange leur informations secrète dans un réseaux , alors il ya des pirates qui attaque ce système de réseaux pour exploité ces informations

Un des principaux risques sur les réseaux provient de "l'écoute" possible puisque toutes les données transitent, si rien n'est fait, en clair sur les réseaux. C'est à dire qu'elles ne sont pas cryptées.

Le Cryptage permis les meilleurs système pour assuré une information sensible echangé dans un réseaux informatique pour augmenté la confidentialité entre les utilisateurs .

La transmission de vidéo permis les meilleurs solution dans le domaine informatique dans la vie on utilise boucoup ce genre de multimedia dans plusieur domaine comme les études ,les service télévision , téléphone . ce domaine utilisé aussi par les gouvernemant les militaires pour resté on contact directemant .

Nous sommes aussi intéressé dans notre projet la sécurisation de la transmission de flux vidéo sur un réseaux de communication, nous sommes aussi intéressés par une architecture client serveur en utilisant un algorithme de chiffrement d'un clé publique RSA.

L'objectif principal de notre mémoire , est de construire une application qui vas chiffrer les flux vidéo capté par une caméra et envoyer ces information dans un réseau

Notre mémoire est structuré comme suit :

Chapitre 1 : Introduction aux architectures client/serveur.

Chapitre 2 : Les techniques et les algorithmes de chiffrement a clé publique et a clé secrète

Chapitre 3 : Etude l'algorithme RSA

Chapitre 4 : Etude le format des fichiers vidéo

Chapitre 5 : Chiffrement des flux vidéo par algorithme RSA

Chapitre 1

Intorduction aux architecture Client/Serveur

1.1. Introduction

Le développement de réseau informatique a conduit à l'apparition d'une architecture distribué basé sur le modèle Client/serveur.

Architecture Client/serveur est une technologie d'un ensemble d'évolutions survenus dans les 30 dernières années

Sont des deux programmes dans un système réseaux échange information entre eux

Les clients envoient des requêtes et serveur envoie des réponses.

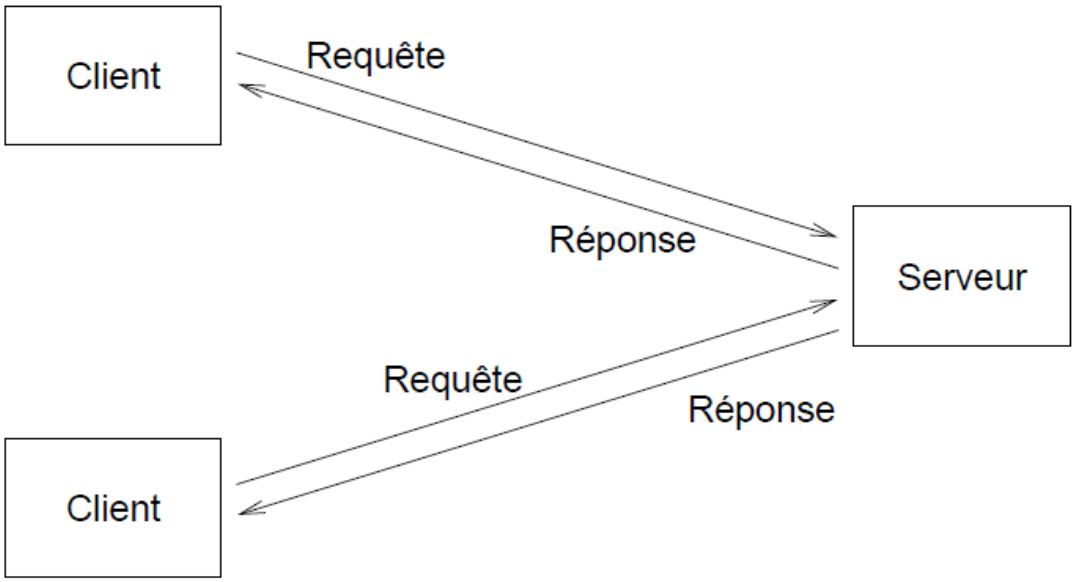


Fig 1.1 : modèle client/serveur

1.2. Le serveur

Le **serveur** est la machine sur laquelle s'exécute le logiciel serveur qui offre un service sur le réseau, le serveur doit être sur un site avec **accès permanent** et s'exécuter en permanence, il accepte des requêtes, les traites et envoie le résultat au demandeur (client).

Un service est fourni sur un Port de communication identifié par un numéro. Certains numéros de Port (internationalement définis) identifient le service quelque soit le site. .[s1]

Exemple :

- Le service **FTP** est offert sur les ports numéros 21 (contrôle) et 20 (données),
- Le service **TELNET** (émulation terminal) sur le port 23,
- Le service **SMTP** (mail) sur le port 25.

Les types de serveurs :

Serveur itératifs : ne gèrent qu'un seul client a la fois

Serveur parallèles : fonctionnent en mode concurrent

1.3. Le client

Le **client** est la machine sur laquelle s'exécute le logiciel client qui utilise le service offert par un serveur, il est raccordé par une liaison temporaire. Il envoie des requêtes et reçoit des réponses de serveur.

la plupart des applications clients ne gèrent pas d'interactions avec plusieurs serveurs,

la plupart des applications clients utilisé pour envoyer les informations .[s2]

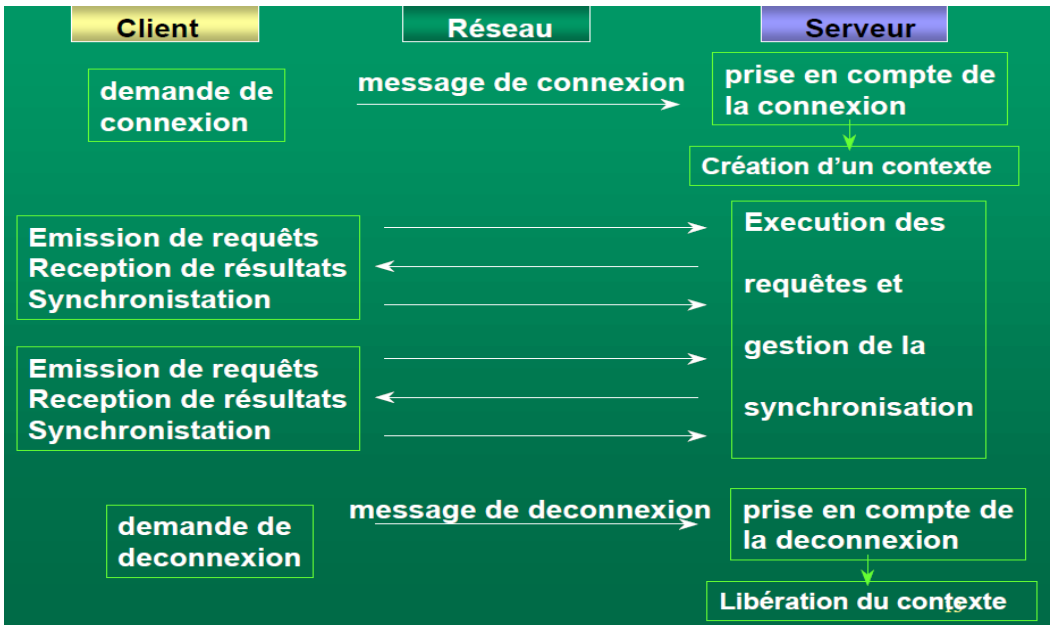


Fig 1.2 : Architecture client/serveur

Les objectifs de l'architecture client/serveur :

Evolution des besoins : production ,informationnel, communication

Evolution des techniques :micro-informatique + réseaux locaux

Evolution des Logiciels : interfaces graphiques, multimédia, interface de communication

1.4 . Avantage de l'architecture client/serveur

- Toutes les données sont centralisées sur un seul serveur, ce qui simplifie les contrôles de sécurité, l'administration, la mise à jour des données et des logiciels.
- Les technologies supportant l'architecture client-serveur sont plus matures que les autres.
- La complexité du traitement et la puissance de calculs sont à la charge du ou des serveurs, les utilisateurs utilisant simplement un client léger sur un ordinateur terminal qui peut être simplifié au maximum.
- Recherche d'information : les serveurs étant centralisés, cette architecture est particulièrement adaptée et véloce pour retrouver et comparer de vaste quantité d'informations (moteur de recherche sur le Web), ce qui semble être rédhibitoire pour le P2P beaucoup plus lent, à l'image de Freenet..[s1]

1.5. Évolution des architectures C\S

1ére génération

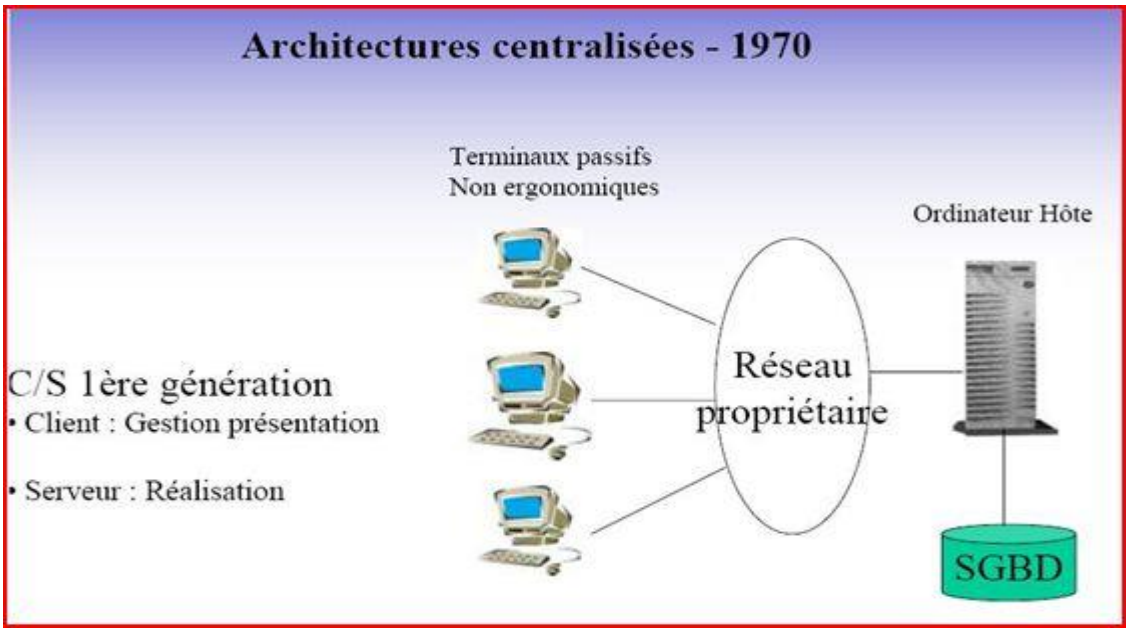


Fig 1.3 : 1ère génération client/serveur

2ième génération

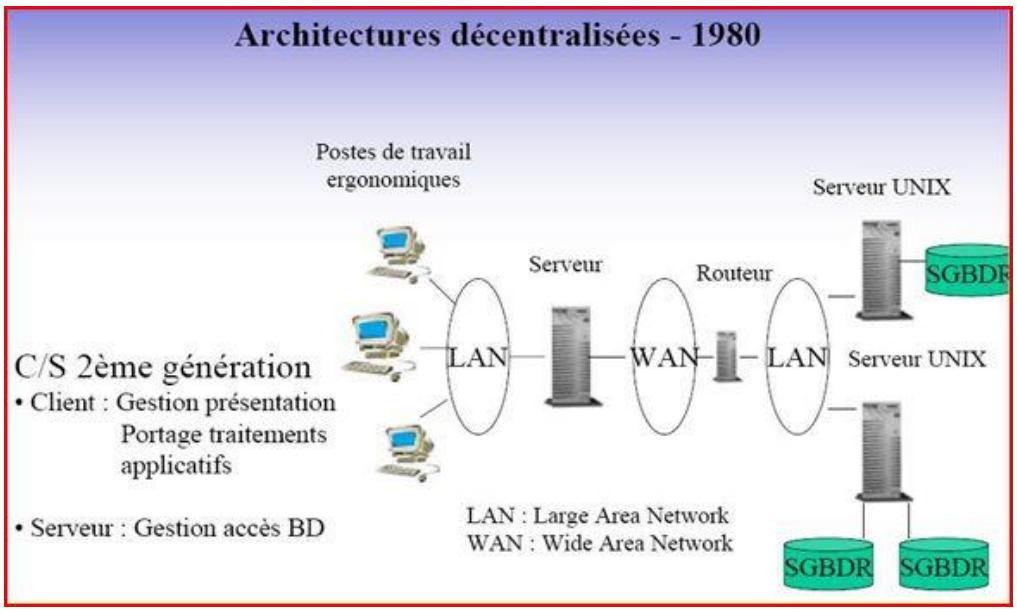


Fig 1.4 : **2ième** génération client/serveur

3éme génération

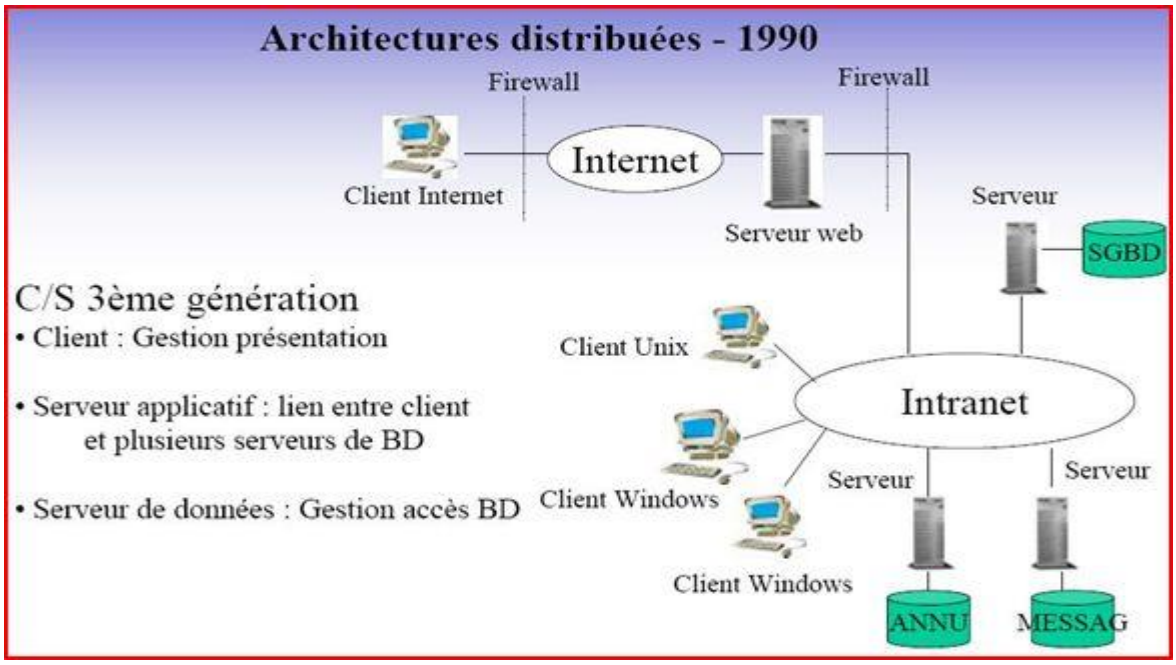


Fig 1.5 : **3ième** génération client/serveur

1.6. définition de protocole et port

Le modèle client/serveur est relié avec Un système dans le réseau a partir de défiction de protocole et port utilisé pour charger la communication .

Les information échangé par le système sant des flux ou bien des packets en mode texte

(ex : ASCII) et peuvent être compris par des humains (ex: SMTP, POP3, HTTP, etc.). D'autres utilisent des échanges de flux binaires (ex : DNS).
Chaque paquet réseau contient :

L'adresse IP de la machine d'origine (le client dans le cas d'une requête),
L'adresse IP de la machine de destination (le serveur dans notre cas),
Numéro de port quit permet de savoir à quel service vas analysée le paquet

Un Port :

Un numéro entier (16 bits) vas identifier a quel service ou un programme vas executé le traitement de paquet reçus

De 0 à 1023 : port reconnus ou réservés.

Un protocole

Un protocole est un langage spécifique a un type de service particulier, le client et le serveur se communique et dialogue par ce langage.

Le but d'un protocole est pour facilité la compréhension entre machines /logiciels

Port	Service ou Application
21	FTP
23	Telnet
25	SMTP
53	DNS
80	HTTP
110	POP3
119	NNTP

Fig 1.6 : notion service et port

1.7 Modes de fonctionnement client/serveur :

Il existe deux grands modes de fonctionnement client/serveur :

Le mode non-connecté : arrivée des données + ordonnancement + duplication possible

Applique généralement sur une connexion synchrone (ex: TFTP)

Le mode Connecté : permet d'abord établir la connexion entre client et serveur pour assurer échange de information, implémentation asynchrone des échanges ce mode plus performante (ex : HTTP, IMAP, FTP...) .[s3]

1.7.2 Caractéristiques du mode non connecté : Le mode « datagramme »

- pas d'établissement préalable d'une connexion
- adapté aux applications pour lesquelles les réponses aux requêtes des clients sont courtes (un message)
- protocole de transport utilisé : UDP
- mode d'échange par messages : le récepteur reçoit les données suivant le même découpage que celui effectué par l'émetteur

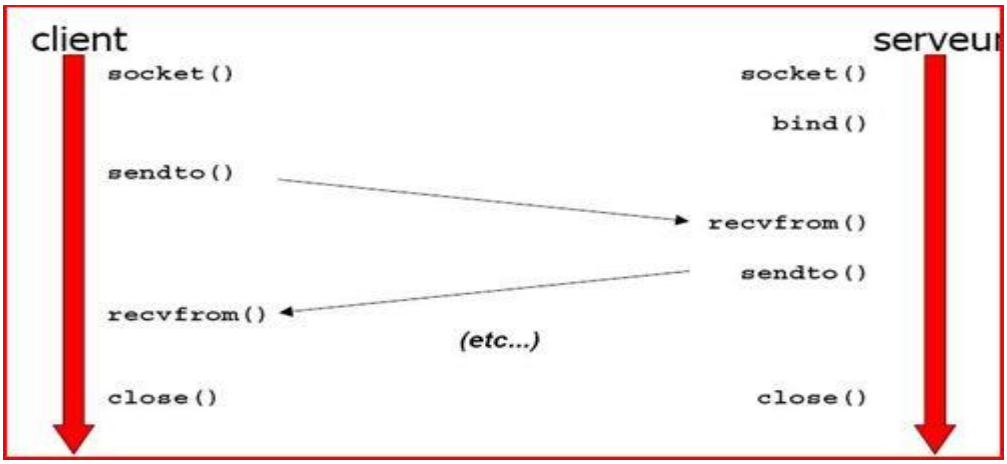


Fig 1.7 : Mode non connecté, UDP (datagrammes)

1.7.3 Caractéristiques du mode connecté

- établissement préalable d'une connexion (circuit virtuel) : le client demande au serveur s'il accepte la connexion
- fiabilité assurée par le protocole de transport utilisé : TCP
- mode d'échange par flots d'octets : le récepteur n'a pas connaissance du découpage des données effectué par l'émetteur
- après initialisation, le serveur est "passif", il est activé lors de l'arrivée d'une demande de

connexion d'un client

- un serveur peut répondre aux demandes de services de plusieurs clients : les requêtes arrivées et non traitées sont stockées dans une file d'attente .[s1]

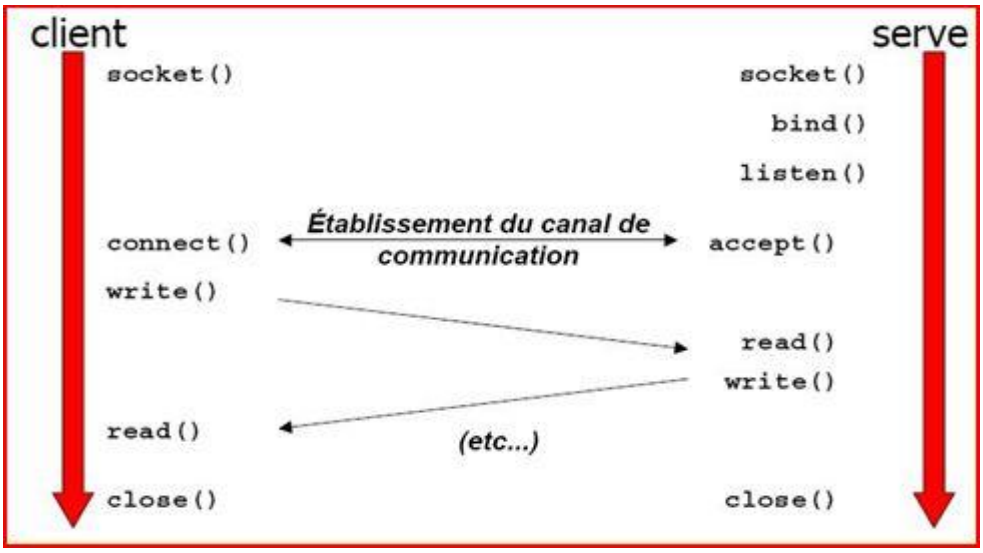


Fig 1.8 Mode connecté, TCP (canal/flux)

1.8 Les Sockets :

Un socket est une abstraction à travers laquelle une application peut envoyer et recevoir des données, en de la même manière comme un fichier ouvert permet une application de lire et écrire des données à stable le stockage. Une prise permet à une application de "Plug-in" sur le réseau et de communiquer avec d'autres applications qui sont également branchés sur le même réseau. Informations écrit à la prise de courant par une application sur une machine peut être lue par une application sur un autre Machine. .[1]

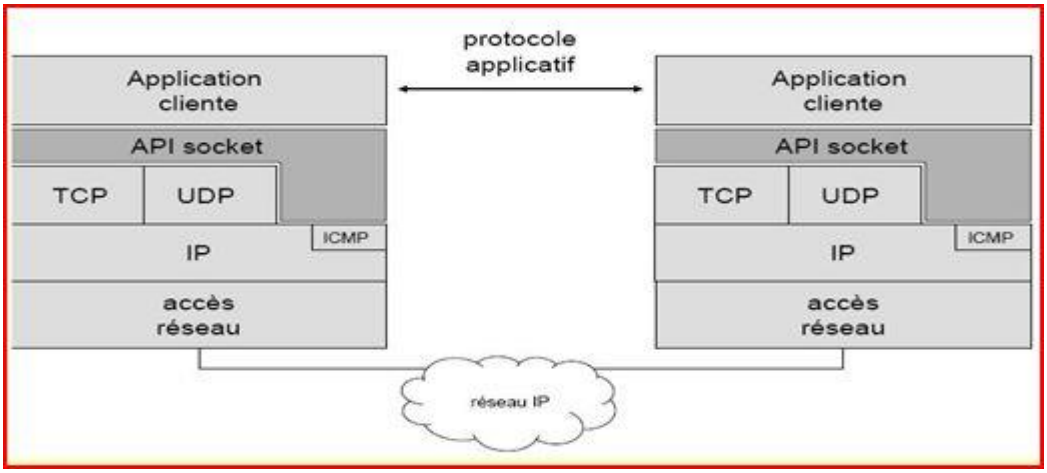


Fig 1.9 sockets

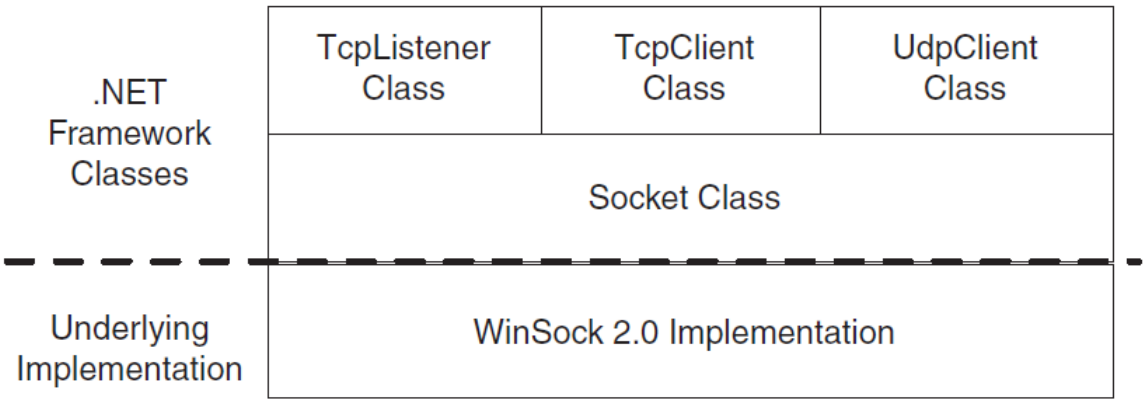


Fig 1.10 sockets sous windows dotnet

1.9. Dialogue entre client et serveur :

C'est logiciel qui assure les dialogues entre clients et serveurs hétérogènes, ou entre 2 applicatifs n'ayant pas les même API. Fait de l'« adaptation de protocole » des couches 5, 6, 7 du modèle OSI

Types de Middleware :

Il ya des middlewares Général :

1. Protocoles de communication, répertoires répartis, services d'authentification ,service de temps ,RPC

2. Services répartis de type NOS (Networked OS) : Service de fichiers, Service D'impression

Il ya des middlewares Spécifique :

Pour BD : ODBC, IDAPI, SQL,etc

Pour groupeware : MAPI, Lotus Notes

Pour D'Object : COBRA, COM/DCOM, .net

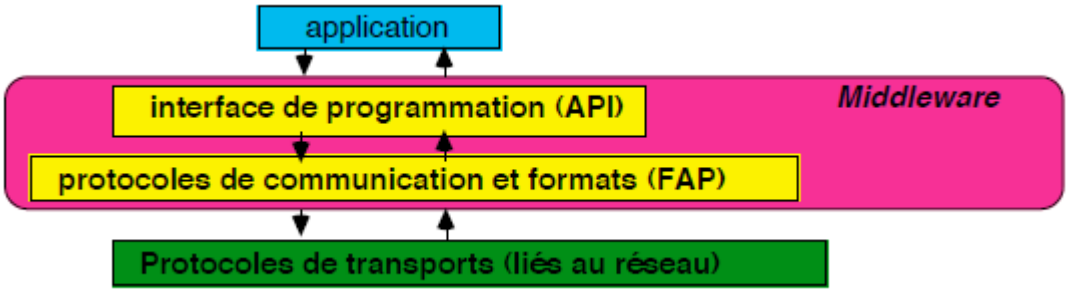


Fig 1.11 Middleware

Rôles des middlewares :

- négociation des connexions,
- conversion des types de données échangées,
- fiabilisation et sécurisation des échanges.
- Permet l'échange de requêtes et des réponses associées entre client et serveur de manière Transparente

Les avantages :

- il offre des services de haut niveau aux applications.
- il rend portable les applications.
- il prend en charge les protocoles de conversion des caractères.
- il établit des sessions entre clients et serveurs

Les objectifs :

- les transports des requêtes et des réponses.
- la simplification de la session utilisateur.
- les performances et la fiabilité.

- Avec le processus client ou serveur, il permet la gestion des appels de fonctions de l'application ou la gestion du renvoi des résultats

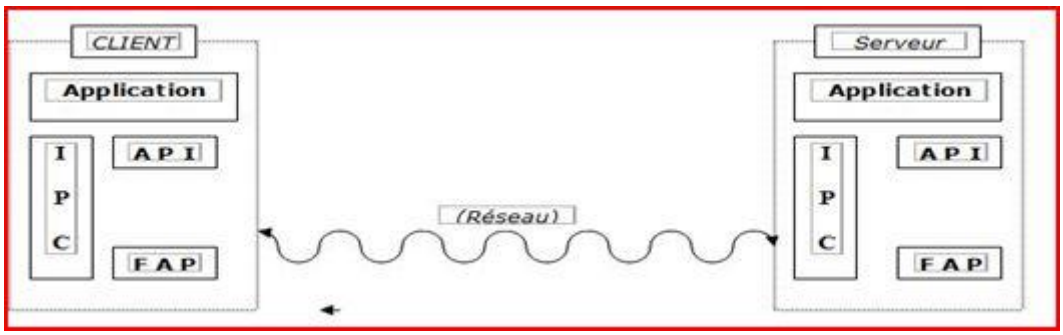


Fig 1.12 : Architecture type du middleware

Architecture type du middleware :

Le Middleware se compose de :

. **API (Application Programming Interfaces) :** sont des fonctions ou des procédures permettent a l'application de faire appel aux services proposés par le serveur

.**FAP (Format and Protocol) :** synchronisation des échanges selon un protocole de communication

Mise en forme des données échangées selon un format connu de part et d'autre

Réalise la synchronisation du dialogue entre client et serveur Définit le format de données échangées Fait le lient avec la couche transport Selon le modèle OSI s'identifie à la couche session et présentation .

Fig 1.8 : modèle ISO et protocole

RPC (Remote Procedure Call) : Technique permettant d'appeler une procédure distante comme une procédure locale en rendant transparent les messages échangés .[s4]

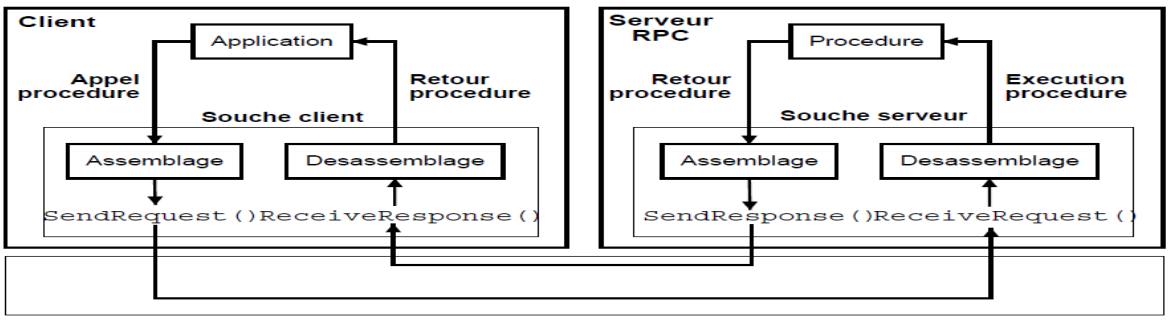


Fig 1.9 : modèle RPC

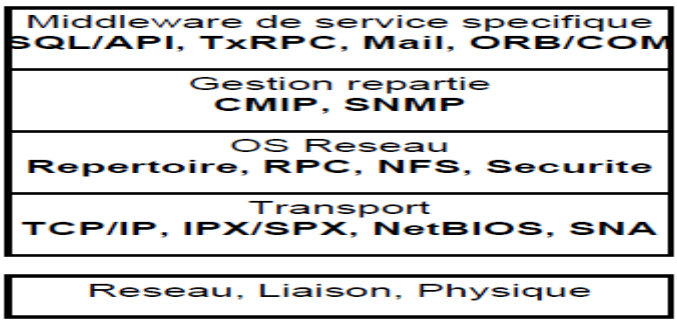


Fig 1.9 : service middleware

1.10 Conclusion

Dans ce chapitre, nous avons présentés une introduction générale du modèle client serveur et leur utilité dans le réseau informatique. Ce modèle client/serveur est la base de touts les applications développer dans le monde des systèmes distribués. Plus, nous avons indiqués l'importance de la notion de protocoles, ports et sockets pour simplifier l'implémentation des applications client/serveur dans l'Internet. Dans le chapitre suivant, nous avons présentés les notions de base de la configuration et administration des interfaces réseau sous linux ou Windows.

Chapitre 2

Chiffrement a clé publique et a clé secrète

2.1 Introduction

L'objectif fondamental de la cryptographie est de permettre à deux personnes, généralement appelés Alice et Bob, de communiquer sur un canal non sécurisé de manière à ce que l'adversaire, Oscar, ne peut pas comprendre ce qui est dit. Ce canal pourrait être une ligne téléphonique ou un réseau informatique, par exemple. Les informations que Alice veut envoyer à Bob, que nous appelons «clair», peut être du texte anglais, des données numériques, ou rien du tout - sa structure est totalement arbitraire. Alice crypte le texte en clair, en utilisant une clé prédéterminée, et envoie le cryptogramme résultant sur le canal. Oscar, en voyant le texte chiffré dans le canal par l'écoute, ne peut pas déterminer ce que le texte en clair était, mais Bob, qui connaît la clé de cryptage, peut déchiffrer le texte crypté et reconstruire le texte en clair.

Ce concept est décrit plus formellement en utilisant la notation mathématique suivante.

2.2 DÉFINITION de la cryptographie

Un système cryptographique est un cinq-uplet P, C, K, E,D lorsque les conditions suivantes sont remplies:

P : est un ensemble fini de textes clairs possibles.

C : est un ensemble fini de cryptogrammes possibles

K : l'espace de clé, est un ensemble fini de clés possibles

pour chaque $K \in \mathcal{K}$, il ya une règle de chiffrement $e_K \in \mathcal{E}$, et une règle de décryptage correspondante $d_K \in \mathcal{D}$. chaque $e_K : \mathcal{P} \to \mathcal{C}$ et $d_K : \mathcal{C} \to \mathcal{P}$ sont des fonctions telles que $d_K(e_K(x)) = x$ pour chaque texte clair $x \in \mathcal{P}$.

Il dit que si un texte clair x est chiffrée en utilisant e_K, et le texte chiffré obtenu est ensuite décrypté à l'aide d_K,, puis les résultats originaux en clair x

Alice et Bob employer le protocole suivant pour utiliser un système de chiffrement spécifique. Tout d'abord, ils choisissent une clé aléatoire $K \in \mathcal{K}$. Cela se fait quand ils sont à la même place et ne sont pas respectées par Oscar, ou, alternativement, quand ils ont accès à un canal sécurisé, auquel cas ils peuvent être dans des endroits différents. À une date ultérieure, supposons que Alice veut communiquer un message à Bob sur un canal non sécurisé. Nous supposons que ce message est une chaîne

pour un entier n ≥ 1, où chaque symbole clair $x_i \in \mathcal{P}$, $\leq i \leq n$. Chaque x_i est chiffrée en utilisant la e_K de règle de chiffrement spécifié par la clé prédéterminée K. Ainsi, Alice calcule $y_i = e_K(x_i)$, $1 \leq i \leq n$, et la chaîne de texte chiffré résultant .

$$\mathbf{y} = y_1 y_2 \cdots y_n$$

est envoyé sur le canal. Quand Bob reçoit Y1Y2. . . yn, il déchiffre en utilisant la fonction de décryptage d_K,, l'obtention de la chaîne en clair d'origine, x1x2. . . xn. Voir la figure 1.1 pour une illustration de la voie de communication. *[2]*

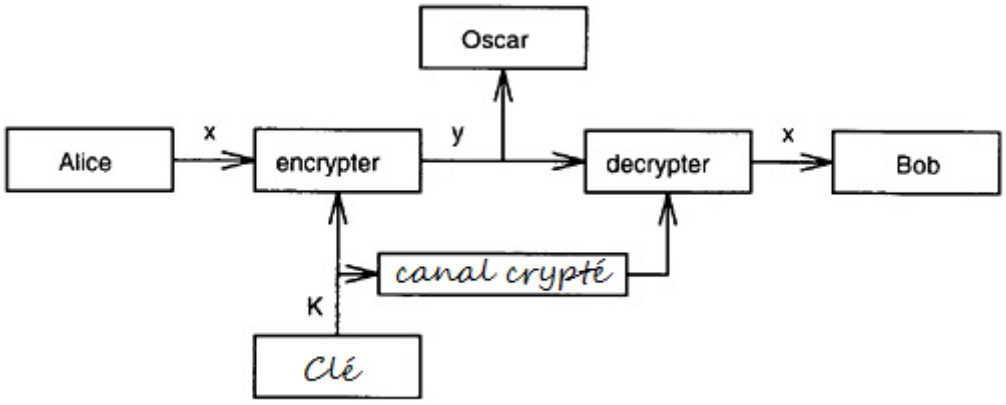


Fig 2.1 : modèle de cryptage entre Alice et Bob

2.3 . L'usage de la cryptographie

La cryptographie est une arte de la cryptologie s'attachant à protéger des messages (assurant confidentialité, authenticité et intégrité) en s'aidant souvent de secrets ou clés.

La confidentialité est le fait de s'assurer que l'information n'est seulement accessible qu'à ceux dont l'accès est autorisé.

L'authentification est la procédure qui consiste, pour un système informatique, à vérifier l'identité d'une entité (personne, ordinateur...), afin d'autoriser l'accès de cette entité à des ressources (systèmes, réseaux, applications...). L'**authentification** permet donc de valider l'authenticité de l'entité en question.

L'identification permet donc de *connaître* l'identité d'une entité alors que l'**authentification** permet de *vérifier* cette identité.
L'intégrité est la motivation première à être conforme à ce que l'on est réellement. L'intégrité est donc le qualificatif donné à ce mécanisme de conformité (à soi-même, non excluant ce qui n'est pas humain).
L'intégrité c'est aussi, l'absence de mauvaise intention. [s5]

2.4 Mécanisme de la cryptographie

Sont des algorithmes de cryptographie ou des fonction mathimatique de chiffrement Utiliser pour le cryptage et décryptage d'une information a partir d'un clé choisier par un utilisateur

La sécurité des données cryptées repose entièrement sur deux éléments : l'invulnérabilité de l'algorithme de cryptographie et la confidentialité de la clé.

2.5 Symétrique vs. Asymétrique Cryptography

Les Algorithmes sont très différents , tels que AES ou DES. les plus algorithmes à clé publique sont basés sur des fonctions de la théorie des nombres. C'est tout à fait différent de chiffrement symétrique, où le but est généralement de ne pas avoir une description mathématique compact entre l'entrée et la sortie. Bien que les structures mathématiques sont souvent utilisés pour petits blocs au sein de chiffrement symétrique, par exemple, dans le AES S-Box, ce ne ne signifie pas que l'ensemble du chiffre constitue une description mathématique compact. [3]

2.6 Introduction à la clé publique

Avant de nous apprendre les bases de cryptographie à clé publique, rappelons-nous que

La cryptographie à clé publique terme est utilisé de manière interchangeable avec la cryptographie asymétrique.

ils désignent tous deux exactement la même chose et sont utilisés comme synonymes.

Cryptographie symétrique est utilisée depuis au moins 4000 ans. Cryptographie à clé publique, d'autre part, est tout à fait nouvelle. Il a été publiquement introduit par Winfield Diffie, Martin Hellman et Ralph Merkle en 1976. Beaucoup plus récemment, en 1997, les documents britanniques qui ont été déclassifiés montré que le chercheurs James Ellis, Clifford Cocks et Graham Williamson du Royaume-Uni de Gouvernement Communications Headquarters (GCHQ) découvert et réalisé le principe de la cryptographie à clé publique quelques années plus tôt, en 1972. Cependant, il est étant encore débattu pour savoir si le bureau du gouvernement reconnaît pleinement l'd'envergure conséquences de la cryptographie à clé publique pour les applications de sécurité commerciale.

Défintion de Cryptographie symétrique

Un tel système est symétrique par rapport à deux propriétés:

1-La même clé secrète est utilisée pour le chiffrement et le déchiffrement.

2-Le cryptage et la fonction de déchiffrement sont très similaires (dans le cas du DES, ils sont essentiellement les mêmes).

Fig2.2 Analogie pour le chiffrement symétrique

Algorithmes symétriques modernes tels que AES ou 3DES sont très sécurisé, rapide et sont largement utilisés. Cependant, il existe plusieurs inconvénients associés à systèmes de clé symétrique . [4]

2.6.1 Principaux mécanismes de sécurité des algorithmes à clé publique

Key Établissement : Il existe des protocoles pour l'établissement de clés secrètes sur un canal non sécurisé. Des exemples de tels protocoles comprennent l'Diffie Helmand (DHKE) ou de transport de clé RSA protocoles.

Non-répudiation : Fournir l'intégrité de la non-répudiation et le message peut être réalisé avec des algorithmes de signature numérique, par exemple, RSA, DSA ou ECDSA.

Identification : Nous pouvons identifier les entités utilisant des protocoles défi-et-réponse avec les signatures numériques, par exemple, dans des applications telles que Smart cartes de banque ou de téléphones mobiles.

Cryptage : Nous pouvons chiffrer les messages en utilisant des algorithmes tels que comme RSA . [3]

2.6.2 Importants algorithmes à clé publique

Il ya seulement trois grandes familles des algorithmes à clé publique qui présentent un intérêt pratique. Elles peuvent être classées en fonction de leur problème de calcul sous-jacent.

Integer-Factorization Schemes : Plusieurs systèmes à clé publique sont basés sur le fait qu'il est difficile de factoriser de grands entiers. Le représentant le plus éminent de cette famille de l'algorithme RSA.

Discrete Logarithm Schemes : Il existe plusieurs algorithmes qui sont sur la base de ce qui est connu comme le problème du logarithme discret dans un corps fini. Les exemples les plus importants incluent l'échange de clés Diffie-Hellman, Chiffrement ElGamal ou l'algorithme de signature numérique (DSA).

Elliptic Curve (EC) Schemes : Une généralisation du logarithme discret algorithme sont les systèmes de clé publique courbes elliptiques. Les exemples les plus populaires inclure l'échange de clés Elliptic Curve Diffie-Hellman (ECDH) et la Algorithme de signature numérique à courbe elliptique (ECDSA).

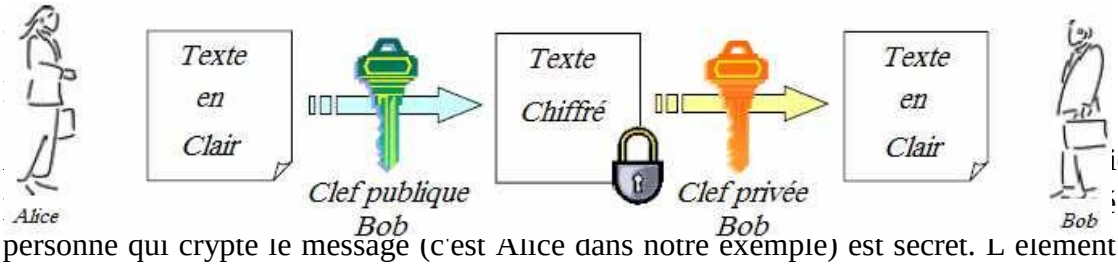
Le tableau 2.1 présente recommandé longueurs de bits pour les algorithmes à clé publique pour les quatre niveaux de sécurité 80, 128, 192 et 256 bit. Nous voyons dans le tableau que les systèmes RSA-comme et systèmes discrets logarithme exiger de très longues opérandes et les touches. La longueur de la clé des systèmes de courbes elliptiques est significativement plus petites, mais encore deux fois plus longtemps que les chiffrements symétriques avec le même force cryptographique.

Famille algorithme	Cryptosystems	Niveau de sécurité (Bit)			
		80	128	192	256
Integer factorization	RSA	1024 bit	3072 bit	7680 bit	15360 bit
Discrete logarithm	DH, DSA, Elgamal	1024bit	3072 bit	7680 bit	15360 bit
Elliptic curves	ECDH, ECDSA	160 bit	256 bit	384 bit	512 bit
Clé-Symmetric	AES, 3DES	80bit	128 bit	192 bit	256 bit

Tab2.1 : Longueurs de bits pour les algorithmes

Afin d'assurer la sécurité à long terme, à savoir, la sécurité pour un laps de temps de plusieurs années, un niveau de 128 bits de sécurité doit être choisi, ce qui nécessite assez longues touches pour tous trois familles d'algorithmes

Une conséquence indésirable de longues opérandes est que les systèmes à clé publique sont très arithmétiquement intensive. Comme mentionné précédemment, il n'est pas rare que un public opération, dire une signature numérique, est de 2-3 ordres de grandeur plus lent que le chiffrement d'un bloc en utilisant AES ou 3DES. En outre, le calcul complexité des trois familles d'algorithmes croît à peu près à la longueur de bits de cube. *[s6]*



ionnaire
par la
personne qui crypte le message (c'est Alice dans notre exemple) est secret. L élement crucial est que Bob, le récepteur, ne peut déchiffrer à l'aide d'une clé secrète.

Pour réaliser un tel système, Bob publie une clé de chiffrement publique qui est connue pour tout le monde. Bob a aussi une clé secrète correspondante, qui est utilisée pour le déchiffrement. Ainsi, K clé de Bob se compose de deux parties, une partie du public, Kpub, et un privé, kpr.
Ce système fonctionne assez de même à la bonne vieille boite aux lettres sur le coin d'une rue: Tout le monde peut mettre une lettre dans la boîte, c'est à dire, chiffrer, mais seulement une personne

Fig 2.3 Protocole de base pour le chiffrement à clé publique

avec une clé privée (secrète) peut récupérer lettres, c'est-à décrypter. Si nous supposons que nous avons cryptographie à cette fonctionnalité, un protocole de base pour le chiffrement à clé publique se présente comme le montre la Fig2 .3.

2.7.1 Définition de cryptographie asymétrique

Une clé privée,vas augmenté la confdentiellité pour chiffré le message envoyé .

2.7.2 Principaux mécanismes de sécurité des algorithmes à clé privé

La cryptographie asymétrique, ou cryptographie à clé publique, est une méthode de chiffrement qui s'oppose à la cryptographie symétrique. Elle repose sur l'utilisation d'une clé publique (qui est diffusée) et d'une clé privée (gardée secrète), l'une permettant de coder le message et l'autre de le décoder [5]

2.7.3 Fonctionnement

La cryptographie asymétrique, ou cryptographie à clé publique est fondée sur l'existence des fonctions à sens unique et à brèche secrète.

Les fonctions à sens unique sont des fonctions mathématiques telles qu'une fois appliquées à un message, il est extrêmement difficile de retrouver le message original.

L'existence d'une brèche secrète permet cependant à la personne qui a conçu la fonction à sens unique de décoder facilement le message grâce à un élément d'information qu'elle possède, appelé clé privée.

Supposons qu'Alice souhaite recevoir un message secret de Bob sur un canal susceptible d'être écouté par un attaquant passif Eve.

- Alice transmet à Bob une fonction à sens unique pour laquelle elle seule connait la brèche secrète.
- Bob utilise la fonction transmise par Alice pour chiffrer son message secret
- Alice réceptionne le message chiffré puis le décode grâce à la brèche secrète
- Si Eve réceptionne également le message alors qu'il circule sur le canal public, elle ne peut le décoder, même si elle a également intercepté l'envoi de la fonction à sens unique, car elle n'a pas connaissance de la brèche secrète.

La terminologie classiquement retenue est :

- pour la fonction à sens unique et brèche secrète : "clé publique"
- pour la brèche secrète : "clé privée"

En pratique, sont utilisées des fonctions de chiffrement classiques, les termes "clé publique" et "clé privée" correspondant alors à des paramètres employés pour ces fonctions. [6]

2.8 Les avantages et inconvénients de la cryptographie à clé publique comparé à la cryptographie à clé privée

Amélioration de la sécurité de l'échange des messages

L'authentification des messages par signature électronique

La longuer de message crypté , la transmission d'information de grandes quantités

La vitesse de transmission qui sont plus rapide. [s7]

2.9 Cryptanalyse

C'est l'étude des faiblesses des systèmes cryptographiques et élaboration des méthodes de protection pour cassé la sécurité de ce système [7]

- ***Cassage complet*** : le cryptanalyse retrouve la clef de déchiffrement.
- ***Obtention globale*** : le cryptanalyse trouve un algorithme équivalent à l'algorithme de déchiffrement, mais qui ne nécessite pas la connaissance de la clef de déchiffrement.
- ***Obtention locale*** : le cryptanalyse retrouve le message en clair correspondant à un message chiffrer.
- ***Obtention d'information*** : le cryptanalyse obtient quelque indication sur le message

2.9.1 Les niveaux d'attaques

il y'a plusieurs méthode pour attaqué un système de cryptage

- ***L'attaque par cryptogramme*** : est un procédé d'attaque utilisé dans l'analyse cryptographique lorsque l'attaquant a accès à un ensemble donné de texte chiffré (s). L'attaquant n'a pas accès à texte clair correspondant à cette méthode; Toutefois, attaquant est réussie quand clair correspdonding peut être déterminée à partir d'un ensemble donné de texte chiffré. De temps en temps, la clé utilisée pour chiffrer le texte chiffré peut être déterminée à partir de cette attaque.
- ***L'attaque à message en clair connu*** : L'attaquant sait ou peut suppose le texte en clair pour certaines parties du texte chiffré. La tâche consiste à décrypter le reste des blocs de texte chiffré en utilisant ces informations. Ceci peut être effectué en déterminant la clé utilisée pour chiffrer les données, ou par l'intermédiaire d'un certain raccourci..
- ***L'attaque à message en clair choisi*** : L'attaquant est en mesure d'avoir un texte qu'il aime crypté avec la clé inconnue. La tâche consiste à déterminer la clé utilisée pour le chiffrement.
- ***L'attaque à message chiffré choisi*** : qui l'inverse de la précédente, le cryptanalyste peut choisir des messages chiffrés pour lesquels il connaitra le message en clair correspondant ; sa tache est alors de retrouver la clef. Ce type d'attaques est principalement utilisé contre les systèmes à clef publique, pour retrouver la clef privée *.[s8]*

2.10 Conclusion

Dans ce chapitre on vus la méchanisme de chiffrement clé public et privé qui joue un rôle important aussi bien au niveau de la sécurité que des performances. Dans le cas de configurations réseaux importantes

Chapitre 3

Méthode de Cryptage En RSA

3.1 Introduction

Après Winfield Diffie et Martin Hellman introduits cryptographie à clé publique dans leur papier repère 1976, une nouvelle branche de la cryptographie soudainement ouvert .
En conséquence, cryptologies commencé à chercher des méthodes avec lesquelles le cryptage A Clé publique peut être réalisé. En 1977, Ronald Rivest, Adi Shamir et Leonard Adleman a proposé un système qui est devenu le plus largement utilisé schéma cryptographique asymétrique, RSA.

Le système de cryptage RSA, parfois appelé l'algorithme Rivest-Shamir-Adleman. Actuellement le système de cryptographie asymétrique le plus largement utilisé, même si les courbes elliptiques et les systèmes de logarithme discret gagnent du terrain. RSA était breveté aux Etats-Unis (mais pas dans le reste du monde) jusqu'en 2000.
Il existe de nombreuses applications pour RSA, mais en pratique, il est plus souvent utilisé pour:

-chiffrement de petits morceaux de données, en particulier pour le transport clé
-les signatures numériques

Elle trouve application dans différents domaines de la vie courante, à savoir :

-guichet automatique (banques, postes…) ;

-achats électroniques ;

-communication mobile (GSM,…) ;

-cartes à puces ;

-numéros de série ; etc.

3.2.1 Présentation

RSA basé sur un chiffrement exponentiel se fait dans l'anneau entier Z*n* et calculs modulaires jouent un rôle central.

RSA chiffrement clair x, où l'on considère la chaîne de bits représentant x soit un élément

Dans $Zn = \{0,1, \ldots, n-1\}$.

En conséquence, la valeur binaire du texte clair x doit être inférieur à n. Il en va de même pour le texte chiffré.
Le chiffrement avec la clé publique et de décryptage de la clé privée sont tels qu'indiqués ci-dessous:

RSA Encryptions Compte tenu de la clé publique key $(n,e) = kpub$ et le texte en clair x, la fonction de chiffrement est la suivante: $y = ekpub\ (x) \equiv xe \bmod n \quad x,y \in Zn$.

Dans la pratique, x, y, n et d sont très grands nombres, généralement de 1024 bits de long ou plus.
La valeur de e est parfois appelé exposant de chiffrement ou exposant public, et la clé privée d est parfois appelé exposant de déchiffrement ou exposant privé. si Alice veut envoyer un message chiffré à Bob, Alice doit avoir son public clé (n, e), et Bob déchiffre avec sa clé privée d D, E et n sont générés.

3.2.2 Rappel mathématique

Th : pour tout (a, b) € N2 avec b ≠ 0, il existe un unique (q, r) € N2 tel que : a = b × q + r et r < b.

Def : on dit alors que a est congru à r modulo b et on note : a = r mod b. *[2]*

3.3 Cryptage et Décryptage

Pour crypter un nombre, il suffit de le mettre à la puissance **e**. Le reste modulo **n** représente le nombre une fois crypté **c = te mod n.**
Pour décrypter, on utilise la même opération, mais en mettant à la puissance **d** :
t = cd mod n
Une fois **e**, **d** et **n** calculés, on peut détruire **p**, **q** et **z**, qui ne sont pas nécessaires pour crypter et décrypter. Pire encore, on peut calculer très rapidement la clé privée **d** à partir de **p** et **q**, il ne faut donc pas conserver ces nombres.
Note : En général, la clé privée est ensuite cryptée à l'aide d'un cryptage symétrique. Cela permet de la conserver de façon sûre, car la clé utilisée par le cryptage symétrique n'a pas à être transmise, et donc ne risque pas d'être interceptée

3.4 .1 Comment ça fonctionne

Puisque e.d = 1 (mod θ) il existe un entier k tel que e.d = 1 + k θ.

Si m est premier avec p, d'après le théorème de FERMAT :

$\boldsymbol{m^{p-1} = 1 \pmod p}$.

On élève les deux membres à la puissance k (q − 1) :

$m^{k(p-1)(q-1)} = 1 \pmod p$.

En multipliant les deux membres par m on obtient :

$m^{ed} = m^{1+k(p-1)(q-1)} = m \pmod p$.

Si m n'est pas premier avec p alors m est un multiple de p et la congruence précédente est encore valide puisque les deux membres sont congrus à 0 modulo p.

On montre de même que $m^{ed} = m \pmod q$. Puisque p et q sont premiers et distincts et $m^{ed} = m \pmod n$. En fin $c^d = (m^e)^d = m^{ed} = m \pmod n$.

Remarques :

– Puisque e est premier avec α(n) l'application m vers c est une bijection sur Zn.

– On peut calculer $c^d \pmod p$ et $c^d \pmod q$et en déduire $c^d \pmod{pq}$.

– Un raisonnement hâtif se rencontre fréquemment dans la littérature. Puisque

$m^{\alpha(n)} = 1 \pmod n$on a $m^{ed} = m^{1+k'(n)} = m \pmod n$. . . sans se préoccuper de vérifier que m est premier avec n.On peut prendre pour contre-exemple $6^{20} = 12 \pmod{33}$. *[8]*

3.4.2 Algorithme RSA pour le cryptage /décryptage

Fonction E Encodage (publique) :

- La clé publique est un couple d'entiers: k = (e, n)

- L'encodage se fait au moyen de l'élévation à la puissance e modulo n:

$Ek(M) = M^e \pmod n$

Fonction D Décodage (secrète)

- La clé secrète est un couple d'entiers: k' = (d, n)

- Le décodage se fait au moyen de l'élévation à la puissance d modulo n :

$Dk'(M) = M^d \pmod n$

Remarque: Les entiers n, e, d doivent être choisis selon des règles précises.

Méthode de choix des clés :

Une particularité de tous les schémas asymétriques, c'est qu'il ya une phase set-up durant laquelle la clé publique et privée sont calculées. En fonction de la clé publique schéma, la génération de clé peut être assez complexe. À titre de remarque, nous notons que la génération de clés n'est généralement pas un problème pour le chiffrement par blocs.

Les étapes à suivre dans le calcul de la clé publique et de la clé privée d'un RSA crypto système

1. Détermination de n :

Trouver deux entiers premiers p et q très grands:

Calculez n = p q

De préférence détruisez p et q.

La sécurité du système repose sur la difficulté de factoriser un grand nombre entier n en deux entiers premiers p et q (taille de n : 320 bits, 512 bits, 1024 bits conditionne également la lenteur des algorithmes).

2. Détermination de la clef publique e

Calculez $\theta = (p-1)(q-1)$; Choisir un entier e premier avec θ.

La clé publique est (e, n)

3. Détermination de la clef privée d

Choisir un entier d tel que : e d □ □1 (mod θ)

(d inverse de e dans l'arithmétique mod θ)

La clé privée est (d, n)

3.Initialisation les clés :

- Chaque utilisateur génère une paire de clés publique / privée par:
  sélection de deux grands nombres premiers au hasard - p, q le calcul de leur module de système

N = P.Q

Note ø(N)=(p-1)(q-1)

-Sélection de manière aléatoire la clé de cryptage e

- où 1<e<ø(N), gcd(e, ø(N))=1

-Résoudre l'équation suivante pour trouver clé de décryptage d

- e.d=1 mod ø (N) and 0≤d≤N

-Publier leur clé de chiffrement publique: KU={e,N}

-Garder le secret clé de chiffrement privée: KR={d, p,q}

Réversibilité de RSA : Fonction d'Euler

Pour n entier $\theta = \alpha(n)$ est le nombre d'entiers premiers avec n.

- si n est premier $\alpha(n) = n-1$

- si $n = pq$ avec p et q premiers ; $\alpha(n) = (p-1)(q-1)$

Théorème d'Euler

Si a et n sont premiers entre eux on a : $\alpha(n) \pmod n = 1$ *[2]*

3.5 .1 Exemple 1 : chiffrer BONJOUR

1) Alice crée ses clés :

- La clé secrète : **p = 53, q = 97** (Note : en réalité, p et q devraient comporter plus de 100 chiffres !)
- La clé publique : **e = 7 (premier avec 52*96), n = 53*97 = 5141**

2) Alice diffuse sa clé publique (par exemple, dans un annuaire).

3) Bob ayant trouvé le couple **(n, e)**, il sait qu'il doit l'utiliser pour chiffrer son message. Il va tout d'abord remplacer chaque lettre du mot BONJOUR par le nombre correspondant à sa position dans l'alphabet :

B = 2, O = 15, N = 14, J = 10, U = 21, R = 18

BONJOUR = 2 15 14 10 15 21 18

4) Ensuite, Bob découpe son message chiffré en blocs de même longueur représentant chacun un nombre plus petit que **n**. Cette opération est essentielle, car si on ne faisait pas des blocs assez longs (par exemple, si on laissait des blocs de 2 chiffres), on retomberait sur un simple chiffre de substitution que l'on pourrait attaquer par **l'analyse des fréquences**.

BONJOUR = 002 151 410 152 118

5) Bob chiffre chacun des blocs que l'on note **B** par la transformation $\mathbf{C = B^e \bmod n}$ (où C est le bloc chiffré) :

$C_1 = 2^7 \bmod 5141 = 128$
$C_2 = 151^7 \bmod 5141 = 800$
$C_3 = 410^7 \bmod 5141 = 3761$
$C_4 = 152^7 \bmod 5141 = 660$
$C_5 = 118^7 \bmod 5141 = 204$

On obtient donc le message chiffré C : 128 800 3761 660 204

3.5.2 Example 2 : chiffrer SUZANNE

p = 3, q = 11, n = 3 x 11, f = (11–1).(3–1) = 20. On choisit d=7 (7 et 20 sont bien premiers entre eux).

e = 3 car

Avant chiffrement		Chiffré			Après chiffrement	
Message	**Bloc numérique B_i**	B_i^3	B_i^3 **mod 33**	$B_i^{'7}$	$B_i^{'7}$ **mod 33**	**Symbolique**
S	**19**	**6859**	**28**	**13492928512**	**19**	**S**
U	**21**	**9261**	**21**	**1801088541**	**21**	**U**
Z	**26**	**17576**	**20**	**1280000000**	**26**	**Z**
A	**01**	**1**	**1**	**1**	**01**	**A**
N	**14**	**2744**	**5**	**78125**	**14**	**N**
N	**14**	**2744**	**5**	**78125**	**14**	**N**
E	**05**	**125**	**26**	**8031810176**	**05**	**E**

3.6 Les Propriétés de RSA

1 . Signature RSA

Le problème de la signature est l'inverse du problème du chiffrement à clef publique. Seul le signataire doit avoir la capacité de signer mais tous peuvent vérifier la signature.

Avec RSA, on a que D(E(m))=m mais aussi E(D(M))=m.

Pour signer un document, on applique l'algorithme de déchiffrement au message et tous ceux qui connaissent l'algorithme public de chiffrement peuvent vérifier la signature.

Pour signer un document, il faut connaître la clef privée!

Vérification :

• Pour vérifier la signature, on calcule $t = s^e \bmod n$ et on vérifie que t = m.

1- Obtenir la clé publique (e, n) du signataire.

2- Calculer $m \approx s^e \bmod n$.

Preuve : $s^e \approx (m^d)^e \bmod n \approx m^{ed} \bmod n \approx m \bmod n$.

– Ceci nous permet de recouvrir le message en même temps.

• Ce schéma de signature nous limite à des messages de taille inférieure à celle de la clé.

– Aujourd'hui, une clé RSA typique utilise 1024 bits. *[5]*

2. Sécurité et performances du RSA :

Utiliser des longueurs de clés de plus en plus importantes

Valeurs est entre 512 bits, 640 bits, 1024 bits (considéré comme assez sûr pour plusieurs années) 2048 bits

Utiliser des circuits intégrés décryptage de plus en plus performants

Il ya dizaine de circuits disponibles.

Vitesse de cryptage de base pour 512bits: de 10 à 30 Kb/s

Évolution en cours de l'ordre de 64 Kb/s

3.Inconvénient de système :

Même sans connaître plus de détails, nous pouvons déjà indiquer quelques exigences pour le système de cryptage RSA:

1-Depuis un attaquant a accès à la clé publique, il doit être impossible de calcul pour déterminer la clé privée d donné les valeurs de clé publique e et n.

2-Puisque x est unique seulement à la taille du module n, nous ne pouvons pas chiffrons plus que l bits avec une cryptage RSA, où l est la longueur de bits de n.

3-Il devrait être relativement facile à calculer *xe* mod n, c'est à dire, pour crypter et yd mod n, c'est-à décrypter. Cela signifie que nous devons une méthode pour exponentiation rapide avec très numéros longs.

4-Pour un n donné, il devrait y avoir beaucoup de paires private-key/public-key, sinon une attaquant peut être capable d'effectuer une attaque par force brute *[4]*

4.Conseils d'utilisation du RSA :

- ✓ Ne jamais utiliser de valeur n trop petite
- ✓ Ne pas utiliser de clé secrète trop courte (< à la racine carré de n)
- ✓ N'utiliser que des clés fortes (p-1) et (q-1) ont un grand facteur premier)
- ✓ Ne pas chiffrer de blocs trop courts

- ✓ Ne pas utiliser de n communs à plusieurs clés
- ✓ Si (d, n) est compromise ne plus utiliser n

3.7 Conclusion

Dans ce Chapitre en a vus commant le système RSA fonctionne et leurs avantage On peut utiliser ce système dans plusieurs domaine de sécurité Pour crypté notre propre information et augmenter la confidentialité entre les utilisateurs dans un réseaux sensible. Le système RSA cryptage à clefs publiques donne une meilleur résultat parapore autre système.

Chapitre 4

Etude le Format des fichiers vidéo

4.1Historique

Avec les progrès rapides dans les ordinateurs dans les années 1980 et 1990 sont venus multimédia applications, où les images et les sons sont combinés dans le même fichier. De tels fichiers ont tendance à être grande, ce qui explique pourquoi les comprimer est devenu une application naturelle.

Une caméra vidéo analogique convertit l'image qu'il "voit" à travers sa lentille à un moteur électrique tension (signal) qui varie avec le temps en fonction de l'intensité et la couleur de la lumière
émise à partir des différentes parties de l'image. Un tel signal analogique est appelé, parce que c'est
analogue (proportionnel) de l'intensité lumineuse. La meilleure façon de comprendre ce signal est de voir comment un récepteur de télévision répond à Analog vidéo

4.2 Introduction

Def 1 : Un fichier vidéo est séquence d'image comprisé qui s'appelle Jpeg avec syncronisation audio wav qui vas jouer en parallèle qui vas enregistrer dans un fichier structuré appelle MPEG

Toute les format d'un fichier vidéo est une compressions propre d'un fichier MPEG comme

FLV AVI 3GP

Def 2 : fichiers vidéo sont des collections d'images, audio et autres données. Les attributs du signal vidéo comprennent les dimensions en pixels, la cadence, canaux audio, et plus encore. En outre, il existe de nombreuses façons de coder et enregistrer des données vidéo. Cette page décrit les principales caractéristiques du signal vidéo, et les formats de fichier utilisé pour capturer, travailler avec, et fournir ces données. *[s9]*

4.3 Les caractéristiques d'un signal vidéo

Chaque fichier vidéo a des attributs qui décrivent ce qui constitue le signal vidéo. Ces caractéristiques comprennent:

• **Châssis de taille**: Il s'agit de la dimension de pixel de l'image

• **Le ratio d'aspect**: C'est le rapport entre la largeur et la hauteur

• **Vitesse de défilement**: C'est la vitesse à laquelle les images sont capturées et destinés à la lecture.
• **Débit**: Le taux de débit ou de données est la quantité de données utilisées pour décrire la partie audio ou vidéo du fichier. Il est généralement mesurée en unités par seconde et peut être en kilo-octets, méga-octets ou giga-octets par seconde. En général, plus la vitesse de transmission, meilleure est la qualité.

• **Le taux d'échantillonnage audio**: C'est à quelle fréquence le signal audio est échantillonné lors de la conversion d'une source analogique à un fichier numérique. *[9]*

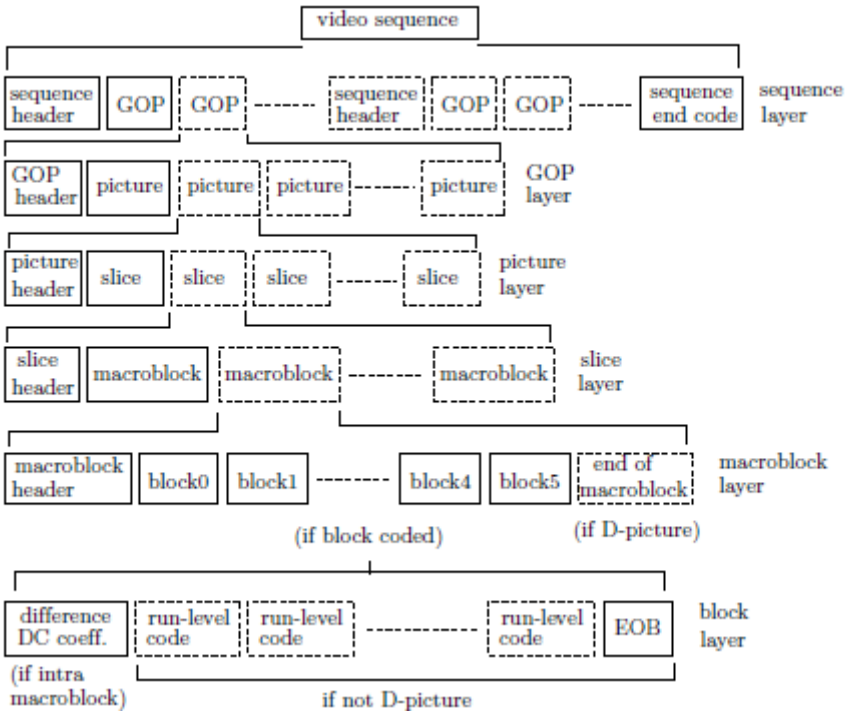


Figure 4.1: Les couches d'un flux vidéo.

4.3.1 L'anatomie d'un fichier vidéo

• Un type de conteneur: AVI et Quicktime MOV sont deux exemples de types de conteneurs.
• Le signal audio et vidéo: données Ce la vidéo réelle et audio, qui a des caractéristiques décrites dans la section suivante.

• Un Codec: Codec désigne le logiciel qui est utilisé pour coder et décoder le signal vidéo. Les applications vidéo utilisent un codec pour écrire le fichier et de le lire. Il peut être intégré dans le programme, ou installé séparément.

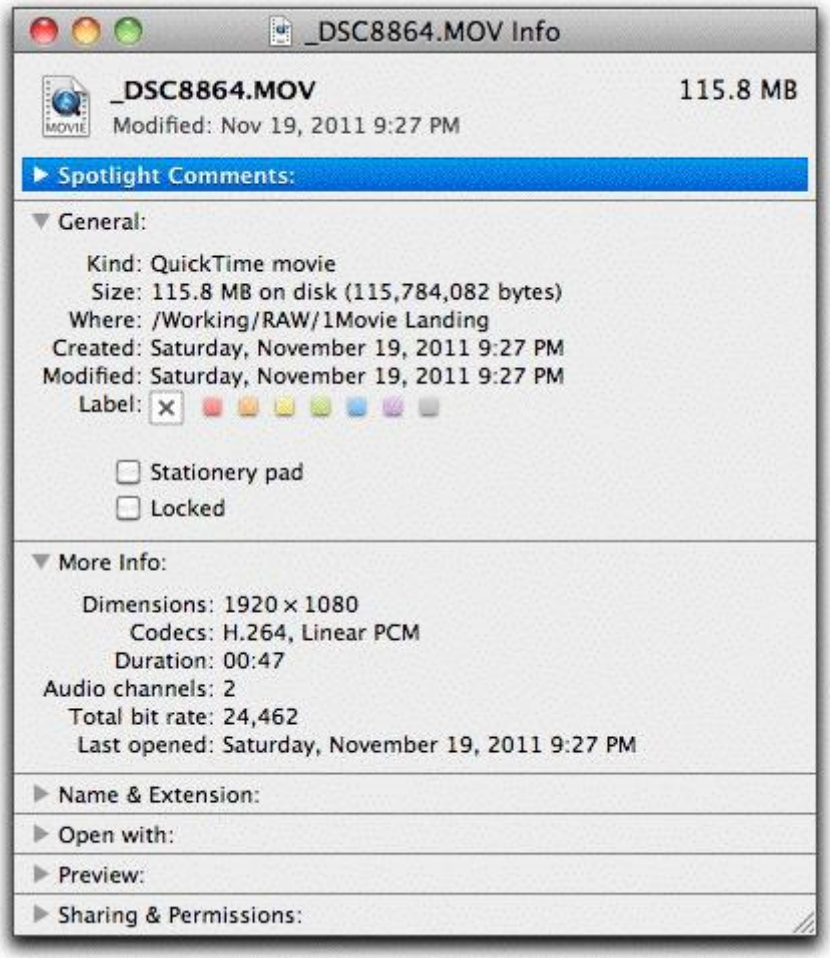


Fig 4 .2 Information général d'un fichier vidéo

4.3.1.1 Codecs

L'intérieur de chaque conteneur de fichiers vidéo sont les données vidéo et audio. Ces données sont créées par un logiciel appelé un codec, abréviation de compresseur / décompresseur (ou compresser / décompresser). Vous pouvez penser que les codecs petites applications d'aide que le programme ou le système d'exploitation utilise pour faire ou lire le fichier vidéo. Sans le bon codec, un fichier vidéo ne peut être lu par un ordinateur. Les codecs vidéo sont souvent propriétaires et peuvent impliquer des frais de licence supplémentaires. Logiciel de montage vidéo (et votre système d'exploitation) seront toujours venir avec au moins quelques codecs intégrés, et généralement vous permettre d'installer d'autres. *[s9]*

4.3.1.2 Compatibilité d'un codec

Le grand nombre de codecs utilisés font compatibilité vidéo une scène très compliquée. Vous ne pouvez pas dire quel codec est utilisé par l'extension de fichier, et le logiciel de votre système peut seulement vous donner des informations partielles. Votre logiciel de montage vidéo peut être en mesure de vous dire quel codec a été utilisé pour créer le fichier, ou vous pourriez avoir besoin pour obtenir des logiciels spécialisés. *[s9]*

4.4.1.3 H.264 codec

Une des familles de codecs les plus polyvalents utilisés aujourd'hui est H.264. (H.264 est également appelé MPEG-4 Part 10 et AVC). Il offre une excellente compression de haute qualité, et il est extrêmement polyvalent. Lorsque H.264 est utilisé avec un débit binaire élevé, il offre vraiment une excellente qualité, comme vous le verrez si vous lisez un disque Blu-ray. Et il est également utile lorsque la compression est la caractéristique la plus importante. Il est le codec utilisé par diffusion sur le Web par des services tels que Vimeo.

Une raison pour laquelle H.264 est si populaire, c'est qu'il travaille avec tant de types de conteneurs. Parmi les types qui prennent en charge l'encodage H.264 sont:

- MOV
- AVC, AVCHD
- MPEG
- Divx

4.4.1.4 Bit rate

Le débit binaire, ou débit de données d'un fichier vidéo est la taille du flux de données lors de la lecture de la vidéo, tel que mesuré en kilobits ou mégabits par seconde (Kbps ou Mbps). Le débit binaire est une caractéristique essentielle d'un fichier car il précise les capacités minimales de la vitesse de transfert d'entraînement ou connexion Internet disque nécessaire pour lire une vidéo sans interruption. Si une partie quelconque du système de lecture ne peut pas suivre le débit, la vidéo bégayer ou de décrochage.

La plupart des logiciels qui peuvent modifier ou transcodage vidéo vous offre la possibilité de définir le débit binaire pour le nouveau fichier. Parfois, ce paramètre vous permet de spécifier un débit binaire constant (c'est ainsi que la vidéo est généralement enregistrée par la caméra). Parfois, ce débit peut être variable, offrant une "cible" pour le logiciel d'essayer de frapper, et un taux maximal admissible pour le courant. Cela peut produire des fichiers qui sont plus optimisés pour la livraison via Internet ou un disque optique.

4.4.1.5 transcodages

Le transcodage est le processus de changement une partie d'un format de fichier vidéo à un autre type. Si vous modifiez la taille du cadre, ou le débit binaire, ou codec, ou signal audio, vous êtes ce que l'être transcodage d'un fichier. Le processus de transcodage est généralement assez de temps, et peut souvent ajouter un obstacle important au processus de production. Le transcodage introduit également la possibilité d'erreurs dans le fichier, en particulier si le signal vidéo est en cours de transformation significatif d'une certaine façon, par exemple un

changement de fréquence d'image. Pour les applications critiques, il est préférable d'examiner toutes les images transcodé avant de jeter le film original ou en relâchant le nouveau métrage à un client. Bien sûr, cela rend le processus de transcodage prendre encore plus de temps. Si vous pouvez accomplir votre travail sans transcodage, vous aurez généralement vous économisez du temps, de l'argent et des coûts de stockage.*[s9]*

4.4.1.6 reconditionnements

Dans certains cas, vous pouvez changer le format d'un fichier sans transcodage. Lorsque vous RÉEMBALLAGE un fichier, vous changez généralement le format de conteneur, mais vous ne changez pas le signal vidéo ou audio lui-même (dans certains cas, vous pourriez faire un changement de l'espace de couleur ou partie audio du fichier). Reconditionnement d'un fichier est beaucoup plus rapide que le transcodage, puisque la plupart des morceaux sont simplement copiés dans le nouveau fichier, plutôt que de traiter d'une certaine façon.

4.4.1.7 Compression réduit le débit binaire

La vidéo est compressée pour réduire le débit de données et donc de la taille d'un fichier. L'objectif est de faire de la petite taille de fichier tout en conservant une qualité visuelle et sonore acceptable. Sans compression, la plupart des caméras vidéo et le matériel seraient tout simplement pas en mesure de suivre les débits de données nécessaires.
En pratique, vous êtes probablement déjà familiers avec la compression. Vous rencontrez compression sans perte lorsque vous prenez des photos en utilisant des formats JPEG. Compression JPEG, en utilisant une variété de techniques, peut faire une seule image beaucoup plus petit en taille, mais peut conduire à une perte considérable de la qualité. Tout comme la compression d'image fixe, la compression vidéo peut prendre l'une des deux formes: sans perte et avec perte.

• **Lossless:** compression sans perte rend un fichier plus petit, mais le fichier décodé est inchangée après la décompression a eu lieu. La façon la plus courante de compression sans perte est fonctionne en utilisant un schéma appelé "codage de longueur d'exécution". Codecs sans perte de vidéo, tels que le codec d'animation (qui est généralement utilisé pour l'archivage ou le transfert de fichiers), ont un faible taux de compression. Habituellement, il n'est pas de plus de 2: 1.

• **Lossy:** Comme le mot l'indique, la compression avec perte signifie que vous perdez une certaine image, une vidéo, ou audio. La compression avec perte est un peu difficile parce que la bonne compression avec perte est une question d'équilibre entre la qualité, le débit / taille du fichier, la profondeur de couleur, et un mouvement fluide. Presque tous les schémas de compression vidéo avec perte. Systèmes de compression avec perte comme le H.264 (utilisé sur les reflex numériques Canon vidéo) peuvent avoir des taux de plus de 100 de compression: 1.

Avec les flux vidéo, la compression peut être appliquée (ou enlevé) à plusieurs étapes.

• **Capture:** Un codec est utilisé pour compresser la vidéo et audio lors de l'acquisition. Par exemple, le Canon 7D DSLR enregistre la vidéo en utilisant le codec H.264.

• **Montage:** La compression peut également être ajouté lors de l'ingestion ou l'édition. Vous pouvez ajouter une compression après le téléchargement, mais ce n'est pas une partie de la plupart des flux de reflex numériques, car les fichiers sont déjà fortement comprimés.

• **Réduction de l'édition de compression:** Il ya aussi des flux de travail qui dépendent de compression réduite au cours du processus d'édition. Dans ses versions 32 bits, Final Cut Pro nécessite généralement des fichiers de transcodage avec le codec de PreRes, ce qui réduit effectivement la compression dans le fichier. (Notez que 64 bits du logiciel NLE fait souvent transcodage, ce qui économise du temps et de l'espace de stockage).

• **Livraison:** Il est extrêmement fréquent pour compresser des fichiers vidéo pour la livraison. Cela permet au processus d'édition de procéder à une version haute qualité des images et d'appliquer la compression uniquement que les fichiers sont préparés pour la sortie dans un but précis.[s10]

4.5 Structure de fichier MPEG

4.5.1 Historique

À partir de 1988, le projet a été développé par MPEG un groupe de centaines d'experts sous les auspices de l'ISO (Organisation internationale de normalisation) et la CEI (Commission électrotechnique internationale). Le nom MPEG est un acronyme pour Moving Pictures Experts Group. MPEG est une méthode de compression vidéo, qui implique la compression des images numériques et le son, ainsi que la synchronisation pour deux. Il existe actuellement plusieurs normes MPEG. MPEG-1 est prévue pour être intermédiaire des débits de données, de l'ordre de 1,5 Mbit / s ou MPEG-2 est destiné à des taux élevés de données à moins 10 Mbit / s. MPEG-3 a été conçu pour la compression de télévision HDTV, mais s'est avérée redondant et a été fusionné avec le MPEG-2. MPEG-4 est destiné à des débits très faibles de moins de 64 Kbit / sec. Un troisième organisme international, l'UIT-T, a été impliqué dans la conception à la fois de MPEG-2 et MPEG-4. Cette section se concentre sur le MPEG-1 et discute seulement ses caractéristiques de compression d'image.[9]

La norme MPEG définit un ensemble d'étapes de codage qui permettent de transformer un **signal vidéo** (numérisé dans un format normalisé) en un train binaire (bit stream) destiné à être stocké sur un support ou transmis dans un réseau. Le train binaire est décrit selon une syntaxe codée d'une manière normalisée pour pouvoir être restituée par n'importe quel décodeur respectant la norme MPEG.[9]

Il est très similaire à la compression

Relation entre MPEG, JPEG :

Le comité de l'ISO MPEG et JPEG (Joint de Photographic Experts Group) à l'origine commencé comme un même groupe, mais avec deux fins différentes. JPEG axée exclusivement sur la compression d'images fixes, tandis que MPEG axé sur l'encodage / synchronisation des signaux audio et vidéo dans un seul flux de données. Bien que MPEG utilise une méthode de compression de données spatiales similaire à celui utilisé pour les fichiers JPEG, ils ne sont pas au même niveau et n'ont pas été conçus pour le même but. [10]

4.5.2 la Compression de MPEG

MPEG utilise une méthode de compression asymétrique. Compression sous MPEG est beaucoup plus compliqué que de décompression, ce qui MPEG un bon choix pour les applications qui ont besoin d'écrire des données une seule fois, mais il faut le lire plusieurs fois. Un exemple d'une telle application est un système d'archivage. Systèmes qui nécessitent des données audio et vidéo à écrire plusieurs fois, comme un système de montage, ne sont pas de bons choix pour MPEG; ils courent plus lentement lorsque vous utilisez le système de compression MPEG.

MPEG utilise deux types de méthodes de compression pour coder des données vidéo: intertrame et intratrame codage. Codage inter est basé sur un codage prédictif à la fois des techniques de codage et d'interpolation, tel que décrit ci-dessous.

Lors de la capture d'images à une cadence rapide (typiquement 30 images / seconde pour la vidéo en temps réel), il y aura beaucoup de données identiques contenues dans deux ou plusieurs images adjacentes. Si une méthode de compression de mouvement est au courant de cette "redondance temporelle», comme de nombreuses méthodes compression audio et vidéo sont, il ne doit pas encoder l'ensemble du cadre de données, comme cela se fait par codage intra. Au lieu de cela, seules les différences (deltas) de l'information entre les cadres est codée. Il en résulte des rapports de compression plus grandes, avec beaucoup moins de données devant être codé. Ce type de codage inter-trame est appelé codage prédictif. Une réduction supplémentaire de la taille des données peut être réalisé par l'utilisation de la prédiction bidirectionnelle. Codage différentiel prédictif code uniquement les différences entre la trame courante et la trame précédente. Prédiction bidirectionnelle code pour la trame courante sur la base des différences entre la trame actuelle, précédente et suivante des données vidéo. Ce type de codage inter est appelé codage interpolation compensée en mouvement. Pour soutenir à la fois inter-et intra-codage, un flux de données MPEG contient trois types de trames codées:

• les trames-I (intra-codées)

• P-frames (codage prédictif)

• les trames-B (codé bi-directionnel)

Une image I contient une seule trame de données vidéo qui ne repose pas sur les informations contenues dans tout autre cadre à coder ou décoder. Chaque flux de données MPEG commence par un I-frame.

Une trame P est réalisé par la prédiction de la différence entre la trame courante et la plus proche précédant I ou trame P. Un B-cadre est construit à partir de deux I plus proche ou P-frames. Le B-cadre doit être placé entre ces I ou P-frames.

Une séquence typique de cadres dans un flux MPEG pourrait ressembler à ceci: IBBPBBPBBPBBIBBPBBPBBPBBI
En théorie, le nombre de trames-B qui peut se produire entre deux I et P-cadres est illimité. Dans la pratique, cependant, il ya généralement douze P et les images B produisant entre chaque trame-I. Un I-frame aura lieu environ toutes les 0,4 secondes de la vidéo de l'exécution.
Rappelez-vous que les données MPEG n'est pas décodé et affiché dans l'ordre dans lequel les images apparaissent dans le flux. Étant donné que les images B reposent sur deux trames de référence pour la prédiction, les deux trames de référence doivent être décodées à partir du premier train de bits, même si l'ordre d'affichage peut avoir une trame B entre les deux images de référence. [10]

Dans l'exemple précédent, la trame I est décodé en premier. Mais, avant que les deux trames-B peut être décodés, le P-cadre doit être décodé et stocké dans la mémoire de l'I-frame. C'est alors seulement que les deux trames-B peuvent être décodés à partir des informations contenues dans le I décodé et les trames-P. Supposons que, dans cet exemple, que vous êtes au début du flux de données MPEG. Les dix premières trames sont mémorisées dans la séquence IBBPBBPBBP (0123456789) ,mais sont décodés dans l'ordre: IPBBPBBPBB (0312645978) et, enfin, sont affichés dans l'ordre:

IBBPBBPBBP (0123456789)

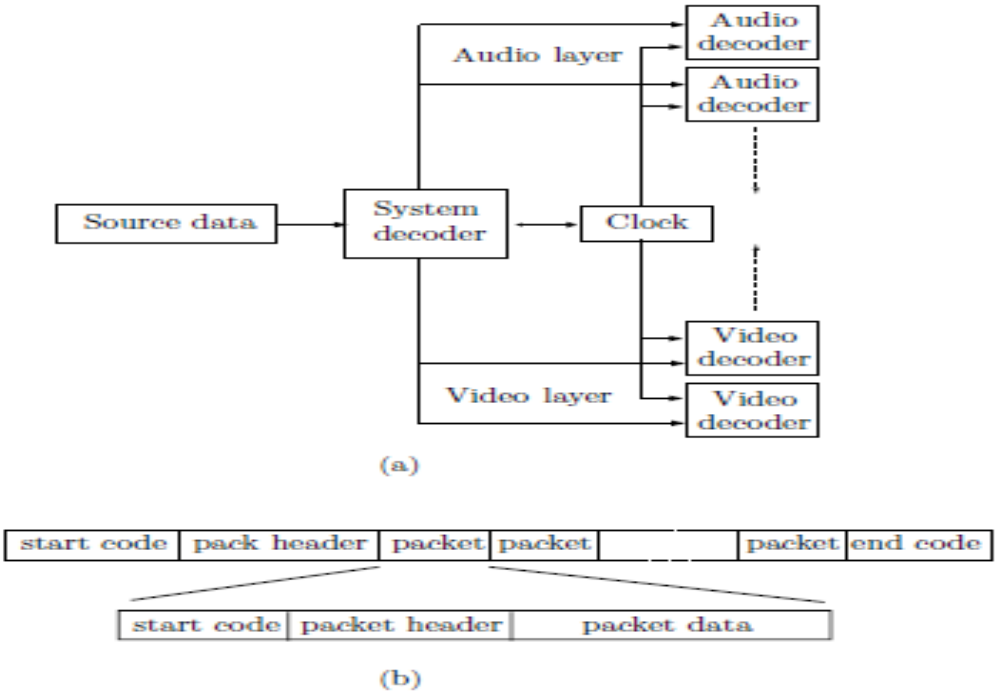
Une fois un I, P ou B-cadre est construit, il est compressé en utilisant une méthode de compression DCT similaire au format JPEG. Lorsque le codage inter-trames permet de réduire la redondance temporelle (données identiques dans le temps), la DCT-codage réduit la redondance spatiale (données corrélées dans un espace donné). Tant le temporel et spatial les informations de codage sont stockées dans le flux de données MPEG.

En combinant le sous-échantillonnage spatial et temporel, la réduction globale de la bande passante obtenue par MPEG peut être considérée comme plus de 200: 1. Toutefois, en ce qui concerne le format final de la source d'entrée, le taux de compression utile a tendance à être entre 16: 1 et 40: 1. Le ratio dépend de ce que l'application d'encodage estime que des résultats plus élevés (vidéo de qualité à des taux de compression plus pauvres) "acceptable" de la qualité de l'image. Au-delà de ces chiffres, la méthode MPEG devient inapproprié pour une application.

Dans la pratique, les tailles des cadres ont tendance à être 150 Kbits pour les trames-I, environ 50 Kbits pour les trames-P et 20 Kbits pour les trames-B. Le débit de données vidéo est généralement limitée à 1,15 Mbits / seconde, la norme pour les fichiers DAT et CD-ROM. La norme MPEG ne prescrit pas l'utilisation de P et B-frames. De nombreux capteurs MPEG éviter la charge supplémentaire de B et P-frames en codant les trames-I. Chaque trame vidéo est capturée, comprimé et stocké dans sa totalité, d'une manière similaire à Motion-JPEG. Trames I sont très semblables à des images codées JPEG. En fait, le Comité JPEG prévoit ajouter MPEG méthodes I-frame à une version améliorée du format JPEG, peut-être à être connu comme JPEG-II.
En l'absence de comparaisons de delta à faire, le codage peut être effectué rapidement; avec un peu d'assistance matérielle, le codage peut se faire en temps réel (30 images / seconde). En outre, l'accès aléatoire du flux de données codées est très rapide car les trames-I ne sont pas aussi complexe et de longue haleine pour décoder comme P et B-frames. Toute trame de référence doit être décodé avant d'être utilisé comme référence par un autre châssis.
Il ya aussi quelques inconvénients à ce régime. Le taux de compression d'un fichier MPEG I-frame-ne sera plus bas que le même fichier MPEG en utilisant une compensation de mouvement. Un fichier d'une minute composée de 1800 images serait d'environ 2,5 Mo en taille. Le même fichier codé en utilisant B et les trames-P serait considérablement plus petite, en fonction du contenu des données vidéo. En outre, ce système d'encodage MPEG peut décompresser plus lentement sur les applications qui allouent une quantité insuffisante d'espace tampon pour gérer un flux constant de données I-frame.[11]

4.5.3 composantes d'un fichier MPEG

L'entrée d'un codeur MPEG est appelé des données source, et la sortie d'un Décodeur MPEG est les données reconstruites. Les données source est organisé en paquets (Figure 3.1 b), où chaque paquet commence par un code de début (32 bits), suivi d'un en-tête, les extrémités avec un code de fin de 32 bits et contient un nombre de paquets entre les deux. Un paquet contient données compressées, soit audio ou vidéo. La taille d'un paquet est déterminé par le MPEG Codeur selon les exigences de support de stockage ou de transmission, qui est pourquoi un paquet n'est pas nécessairement une image complète de la vidéo. Il peut être n'importe quelle partie d'une vidéo image ou d'une partie de l'audio.



Figue 4.1 : (a) MPEG Decoder Organization. (b) Source Format.

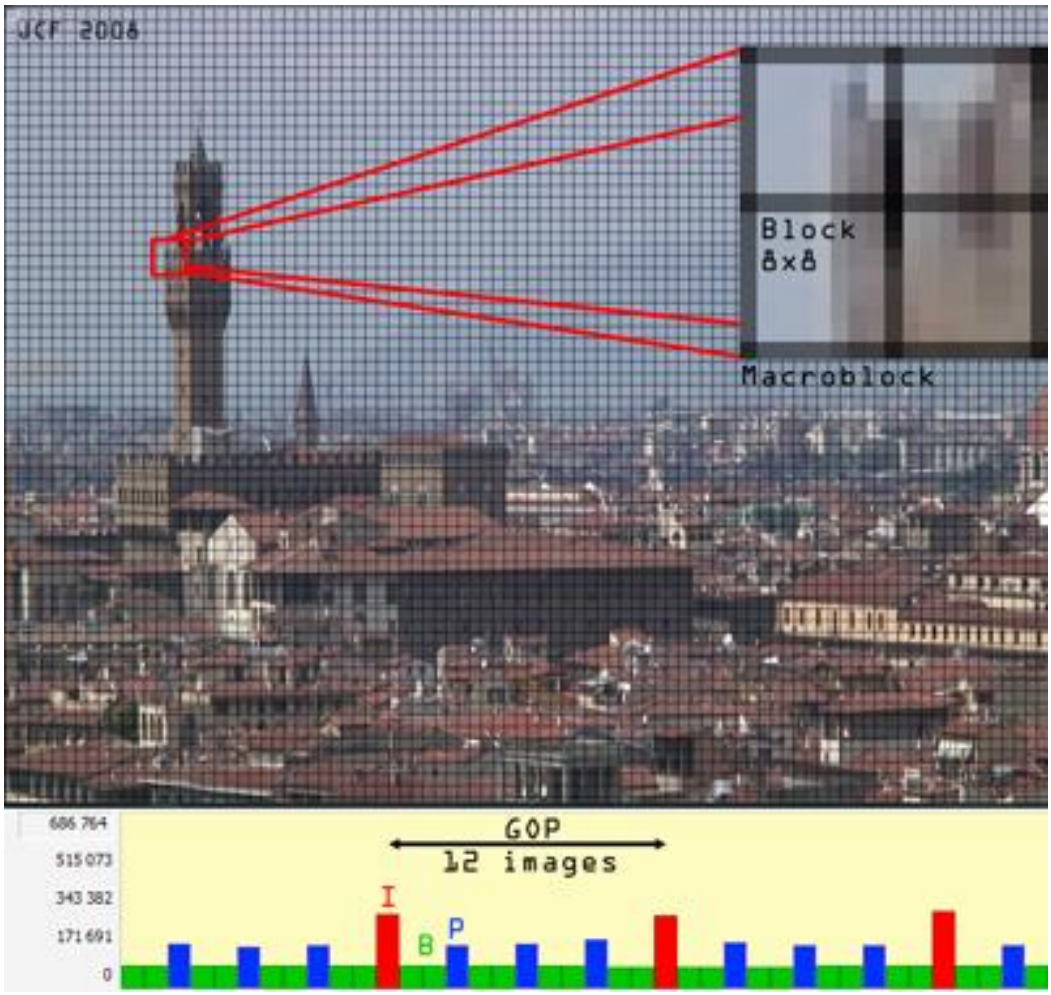
Le décodeur MPEG comporte trois parties principales, appelées couches, pour décoder l'audio,

la vidéo, et les données système. La couche système lit et interprète les différents codes et les en-têtes dans les données sources et les itinéraires des paquets à l'audio ou la vidéo couches à être tamponnées et plus tard décodées. Chacune de ces deux couches se compose de plusieurs décodeurs qui fonctionnent simultanément. [9]

Couche macro-bloc :

Cette couche identifie la position du macro-bloc par rapport à la position du macrobloc courant. Il code les vecteurs de mouvement pour le macrobloc, et identifie le zéro et blocs non nuls dans le macrobloc.

Chaque macrobloc a une adresse, ou indice, dans l'image. Valeurs d'index commencent à 0 dans le coin supérieur gauche de l'image et continuer afin de trame. Lorsque le codeur commence codant pour une nouvelle image

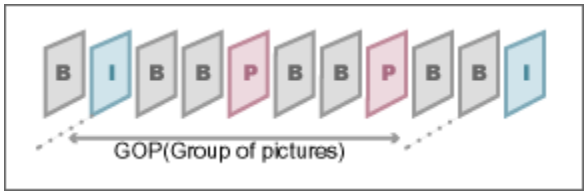


Figue 4.1 Un flux Mpeg standard des blocks 8x8 et un GOP fixe

A la compression spatiale, qui ne travaille que dans le cadre de l'image fixe, s'ajoute une compression qui va travailler dans le cadre de la séquence vidéo, d'une image à l'autre:

La compression temporelle (INTER-images):

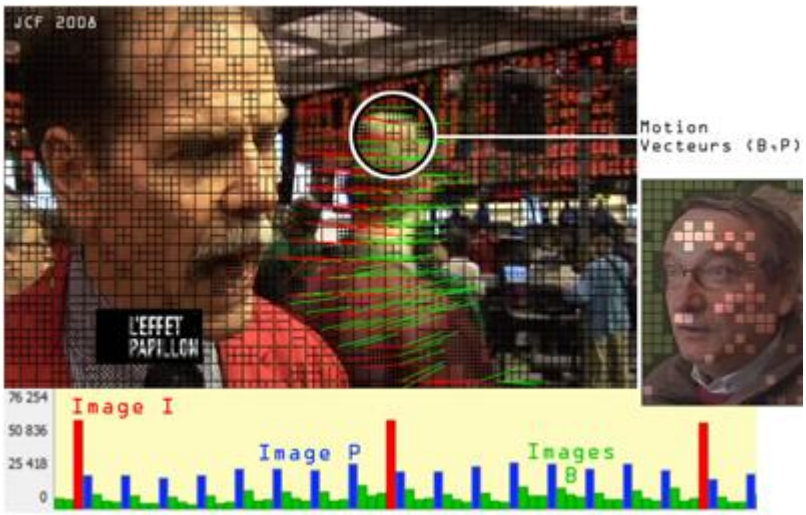
Partant du principe que dans une séquence vidéo, d'une image à l'autre, peu de blocks changent, on ne va coder que les blocks de l'image qui changent (B et P) d'une image à l'autre, les autres blocks proviennent d'images de référence (I et P), placées avant ou après:



G.O.P :

Un groupe d'images (GOP) débute par un en-tête GOP , suivi par un ou plusieurs photos . Chaque image dans un GOP commence par un en-tête d'image, suivie par un ou plusieurs tranches . Chaque tranche , à son tour , est constitué d' un en-tête de tranche , suivi par un ou plusieurs macroblocs de codage , des coefficients DCT quantifiés

Un groupe d'images (G.O.P.), encadré par des images complètes (I), devient alors la structure du flux qui se répète tout au long de la séquence vidéo. Des vecteurs de mouvement permettent ensuite de coder le déplacement des blocks à l'intérieur du GOP. Un GOP "mesure" habituellement 1/2 seconde, soit 12 images en Europe et 15 images aux USA.

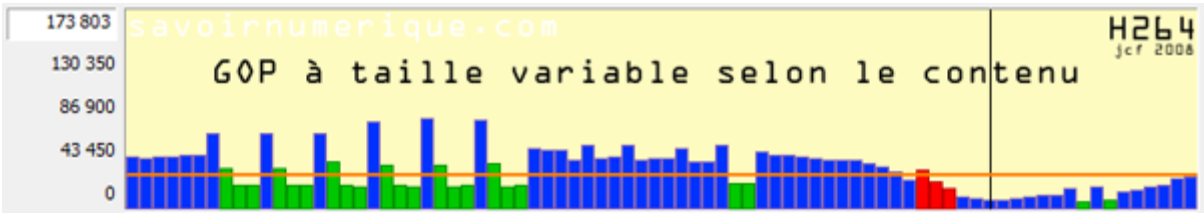


Un flux H264 avec une structure variable selon la complexité de la scène (blocks et macroblocks), et des GOP à structure variable

Les technologies d'encodage Mpeg (Codecs Mpeg1, 2, 4 et H264...) utilisent ces méthodes pour faire baisser le débit général du flux vidéo tout en offrant une bonne qualité visuelle; on passe ainsi d'un flux brut HD SDI 1,5 Gbps à un flux à 100 Mbps en AVCintra (intra seul), et à un flux HDV Mpeg2 à 25 Mbps (intra+inter).

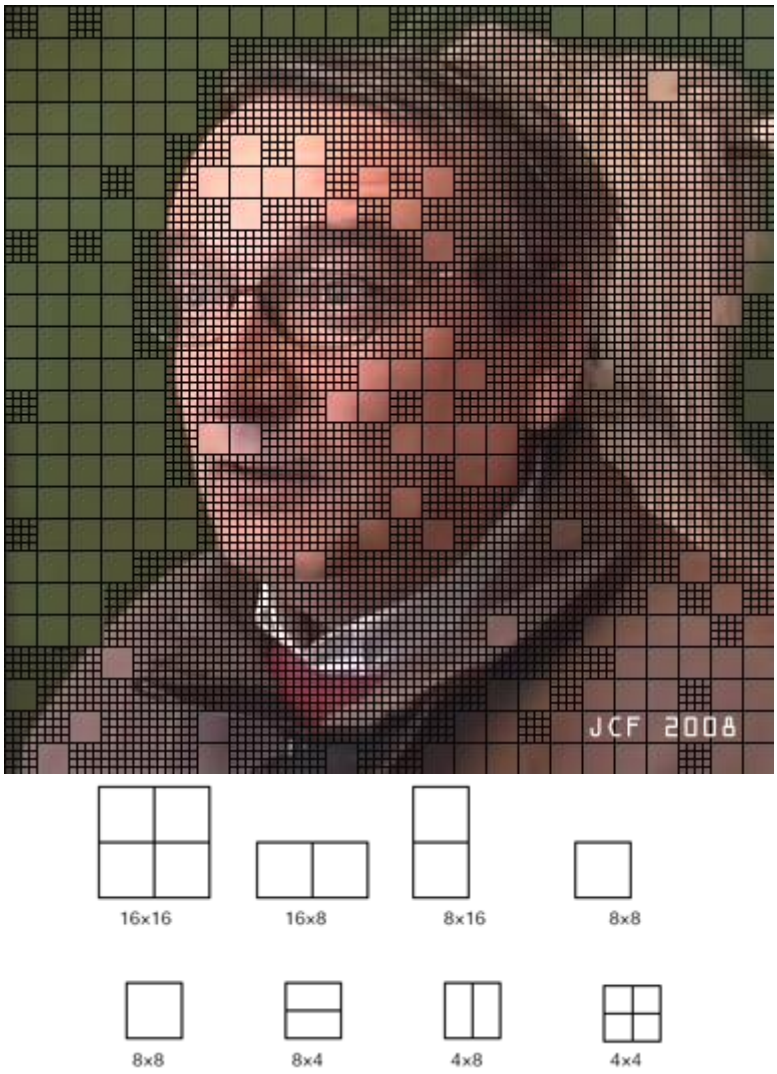
Le HDV, le XDCAM HD et le XDCAM EX utilisent des flux Mpeg2 construits sur des GOP de 6 à 12 images à 18, 25 ou 35 Mbps. Le XDCAM HD offre un débit variable plus confortable dans les scènes dynamiques ou complexes jusqu'à 35 Mbps, là ou le HDV standard à 25 Mbps à bande (contraint par la vitesse fixe de défilement de la bande) peut provoquer parfois des artefacts visibles de compression (dans les scènes subites très dynamique, genre flash de photographes ou animal qui passe devant la caméra).

Les camescopes AVCHD utilisent des flux Mpeg4 H264 avec compression spatiale et temporelle, pour un gain de l'ordre de 50% par rapport aux flux Mpeg2, notamment grace aux nouvelles techniques spatiales (pour 15%, taille de blocks/macroblocks variable...) et temporelle (pour 35%, taille de GOP variable...) du codec H264.



optimisation temporelle du H264: structure variable selon le contenu vidéo

Les camescopes AVCIntra de Panasonic utilisent les seules méthodes H264 de compression spatiales uniquement (AVC = H264, intra = compression spatiale), sans utiliser les méthodes temporelles (pas de GOP, et donc peu de gain vis à vis du codec Mpeg2 de base). [s11]



Optimisation spatiale du H264 selon la complexité de l'image: Blocks et macro blocks variables

4.5.4 Compression spatio-temporelle

- L'image encodée en haut à gauche, avec la découpe en blocks DCT (fixe pour le Mpeg2, variable pour le Mpeg4); L'apparition de flèches rouges et bleues montre à l'image la présence de vecteurs de mouvement

- La structure du GOP en bas, avec la tête de lecture sur l'image I, B ou P
- à droite le type d'image lors de la lecture: I, B ou P

- en haut à droite: un macro block DCT, avec composition fixe (Mpeg2) ou variable (Mpeg4)

- Toutes les images montrent la structure réelle du flux compressé, il ne s'agit pas d'une "simulation" réalisée sur Photoshop ou autre. Les blocks, macroblocks, vecteurs de mouvement et type d'images (GOP) sont ainsi affichés en temps réel par analyse des flux (des logiciels d'analyse comme mosalina sont destinés à la vérification des flux)

La tâche principale du décodeur MPEG est de reconstruire les pixels de la séquence complète de vidéo Cela se fait par la lecture des codes d'un bloc dans le flux compressé, les décoder, les déquantifier, et le calcul de la IDCT. Pour les blocs nonintra P et des images B, le décodeur doit ajouter la prédiction à compensation de mouvement pour les résultats du IDCT. Cette opération est répétée six fois (ou moins, si certains blocs sont complètement zéro)

Discrete Cosine Transform (DCT) :

Une transformation en cosinus discrète (DCT) exprime une séquence finie de points de données en fonction d'une somme de fonctions cosinus oscillant à des fréquences différentes. DCT sont importantes pour de nombreuses applications de la science et de l'ingénierie, de la compression avec perte de données audio (MP3 par exemple) et des images (par exemple JPEG) (où de petits composants à haute fréquence peuvent être jetés), aux méthodes spectrales pour la résolution numérique des équations aux dérivées partielles. L'utilisation de cosinus plutôt que des fonctions sinus est critique pour la compression, car il se trouve (comme décrit ci-dessous) que moins de fonctions cosinus sont nécessaires pour approximer un signal caractéristique, tandis que pour les équations différentielles cosinus exprimer un choix particulier des conditions aux limites.

Le DCT est utilisé en compression d'image JPEG, MJPEG, MPEG, DV, Daala, et la compression vidéo Theora. Là, les deux dimensions de la DCT-II N \ fois N blocs sont calculées et les résultats sont quantifiés et codés entropie. Dans ce cas, N est typiquement de 8 et la formule II-DCT est appliquée à chaque rangée et colonne du bloc. Le résultat est un 8 × 8 transformée tableau coefficient dans lequel l'élément (0,0) (en haut à gauche) est le DC (fréquence nulle) composant et entrées avec des valeurs croissantes de l'indice verticales et horizontales représentent les fréquences spatiales supérieures verticales et horizontales. [11]

Définition mathématique : Les coefficients de la DCT F s'obtiennent à l'aide de l'équation suivante, pour une image de taille N × N dont la valeur du pixel est définie par f :

$$F(u,v) = \frac{2}{N}\left(\frac{1}{\sqrt{2}}\right)^{\delta(u)+\delta(v)} \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) \times \cos(\frac{(2x+1)\pi u}{2N}) \times \cos(\frac{(2y+1)\pi v}{2N})$$

où $\delta(x) = \begin{cases} 0 & \text{si } x = 0 \\ 1 & \text{si } x > 0 \end{cases}$

Cette transformée est inversible et l'expression de la IDCT est :

$$f(x,y) = \frac{2}{N}\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}\left(\frac{1}{\sqrt{2}}\right)^{\delta(u)+\delta(v)} \times F(u,v) \times \cos(\frac{(2x+1)\pi u}{2N}) \times \cos(\frac{(2y+1)\pi v}{2N})$$

4.5.5 Le contrôle de débit dans MPEG

Comme son nom l'indique, le contrôle de débit dans le codeur est l'algorithme qui permet de maîtriser le volume de données généré. Basé essentiellement sur l'étape de quantification, le contrôle de débit a pour but d'optimiser la qualité visuelle de l'image tout en respectant des contraintes de débit exprimées en fonction du système de transport utilisé (e.g. débit réseau) ou du support de stockage (e.g. capacité du CD-ROM). Les premiers réseaux de transmission pouvant être utilisés pour la vidéo étaient à commutation de circuits (e.g. RNIS, liaison spécialisée, liaison satellite...) et offraient un débit constant. Le contrôle de débit dans le codeur est nécessaire pour maintenir constant le débit de sortie. On parle alors de codage à débit constant ou codage CBR (pour Constant Bit Rate). L'avènement des réseaux ATM offre la possibilité de transmettre des débits variables plus adaptés à ce qu'on appelle codage à débit variable ou codage VBR (pour Variable Bit Rate).

4.5.6 Codage à débit constant

L'idée générale d'un codage à débit constant est d'utiliser un buffer à la sortie du codeur (c'est à dire à l'entrée du réseau) qui est continûment vidé au débit fixe du canal de transmission. Le contrôle de débit consiste alors à éviter le débordement du buffer à l'aide d'une boucle de réaction qui agit sur le paramètre de quantification en fonction croissante et très souvent linéaire du remplissage du buffer. L'échelle de temps de la réaction peut aller du macro-bloc à l'image selon la finesse de l'algorithme et la taille du buffer. Pour MPEG, l'algorithme de contrôle tient compte des différents types d'images. En effet, une image I servant de référence pour la restitution du reste des images du GoP, doit subir moins de dégradation que celles de type P ou B. De même pour les images P qui sont plus volumineuses que les images B . [s12]

4.6 Conclusion

Dans ce chapitre on a vus un rôle plus important dans le développement des multimédia La définition de la série de standards MPEG permet de couvrir une très large gamme d'applications vidéo. Très souvent, les applications MPEG déjà existantes n'utilisent que le mode de codage Cela est en partie du à l'absence de réseaux permettant de bien gérer le débit L'avenir de MPEG est donc très lié à l'évolution des réseaux

Chapitre 5

Cryptage de flux video

5.1 Introduction

Dans ce chapitre , nous présentons une application qui vas crypté et sécurisé des flux vidéo par un modèle client /serveur nous avons établi une comparaison sur la qualité et la vitesse de l'opération de chiffrement et de transmission . Nous avons utilisé un ordinateur Intel (R) core 2 duo cadencés à 2.00 MHz avec une RAM de 2.00 Go et un système d'exploitation Windows 7 .

5.2 Présentation de l'application

Langage utilisé

La langage choisi pour réalisation de notre application Microsoft visual c ++/basic studio 2012

Quit permet a créé des applications en windows avec un développement rapidement

grâce a ses différentes bibliothèques et de la réalisation d'une interface facile

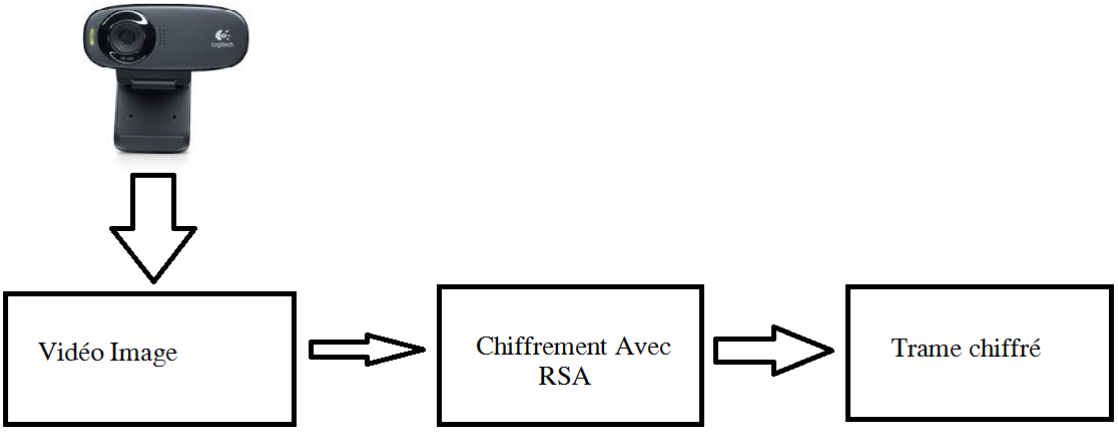


Figure 5.1 : idée général pour réalisé interface de client

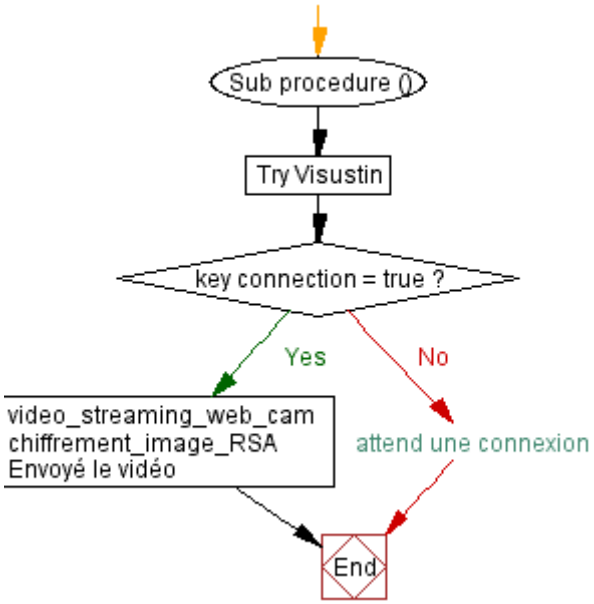


Figure 5.2 :Diagramme de client

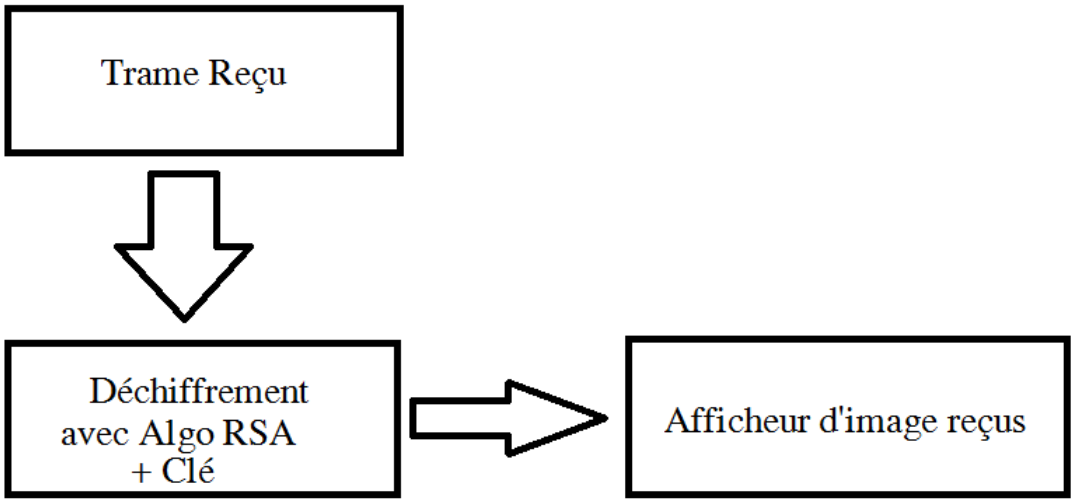


Figure 5.3 : idée général pour réalisé interface de serveur

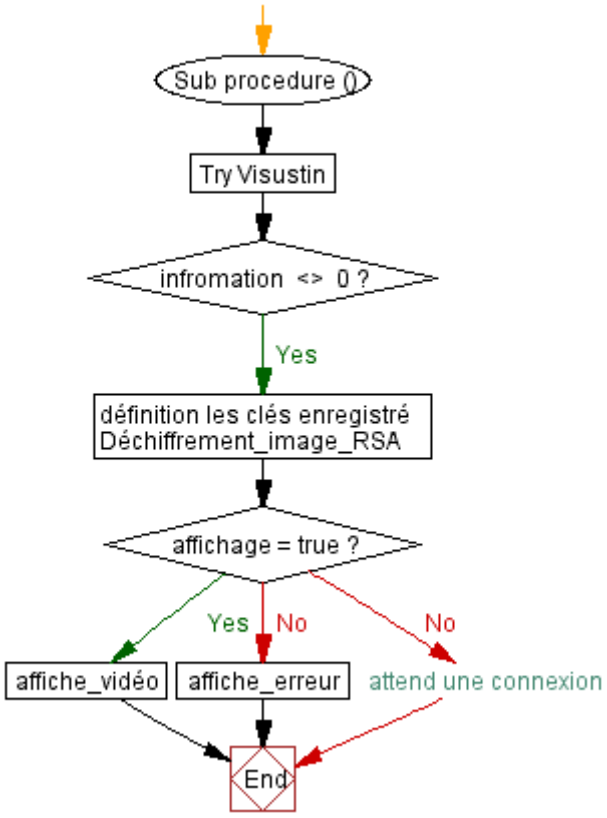


Figure 5.4 :Diagramme de serveur

L'interface de l'application

Client

La fenêtre d'interface de notre Client est présentée ci-dessus par la figure :

Figure 5.5 : interface de Client

Le menu de programme :

Le button Diffuser : permet de etablir une connexion avec l adress ip entré

Le button Utiliser la cam : permet d'ouvrir la cam dans interface de programme

Le button Gérer les clés : permet de gérer les clés de chiffrement

Figure 5.6: la fenêtre de chiffrement

Le button voir les flux : permet de voir les flux vidéo en mode de matrice

Le button quitter : permet de quitter l'application

Serveur

La fenêtre d'interface de notre Client est présentée ci-dessus par la figure :

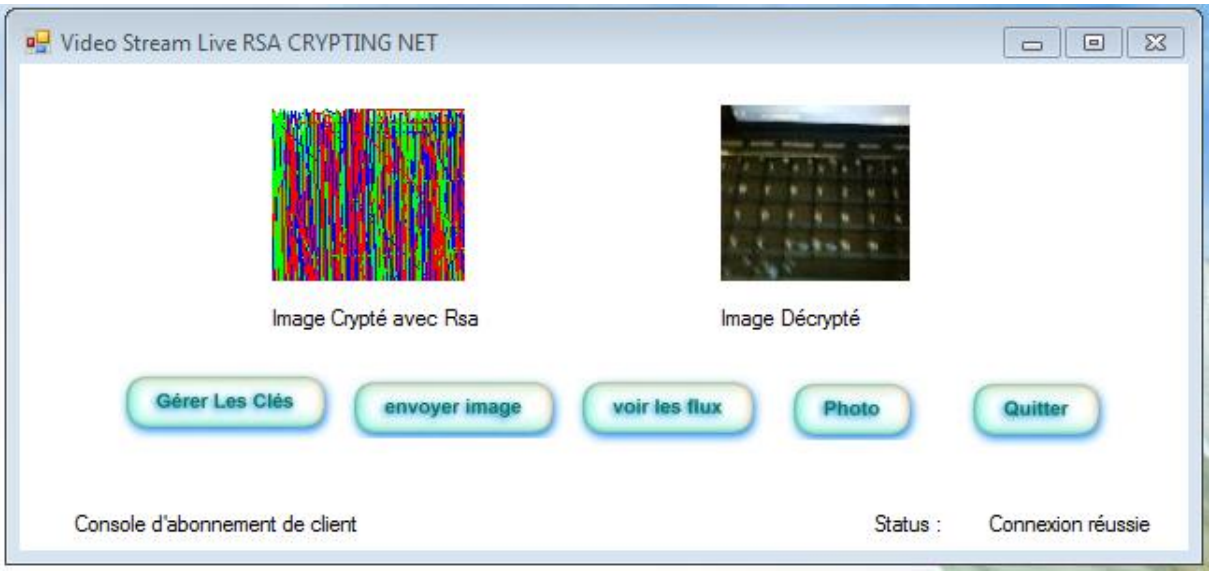


Figure 5.7 : interface de Serveur

Le button Gérer les clés : permet de gérer les clés de chiffrement

Figure 5.8 : la fenêtre de chiffrement

Le button envoyer image : permet de envoyer vidéo chez client

Le button voir les flux : permet de voir les flux vidéo en mode de matrice

Le button photo : permet de garder une photo

Le button quitter : permet de quitter l'application

3 . Les résultats obtenus

Vidéo en couleur

Client :

Figure 5.9 : réalisation une connexion avec serveur

Serveur

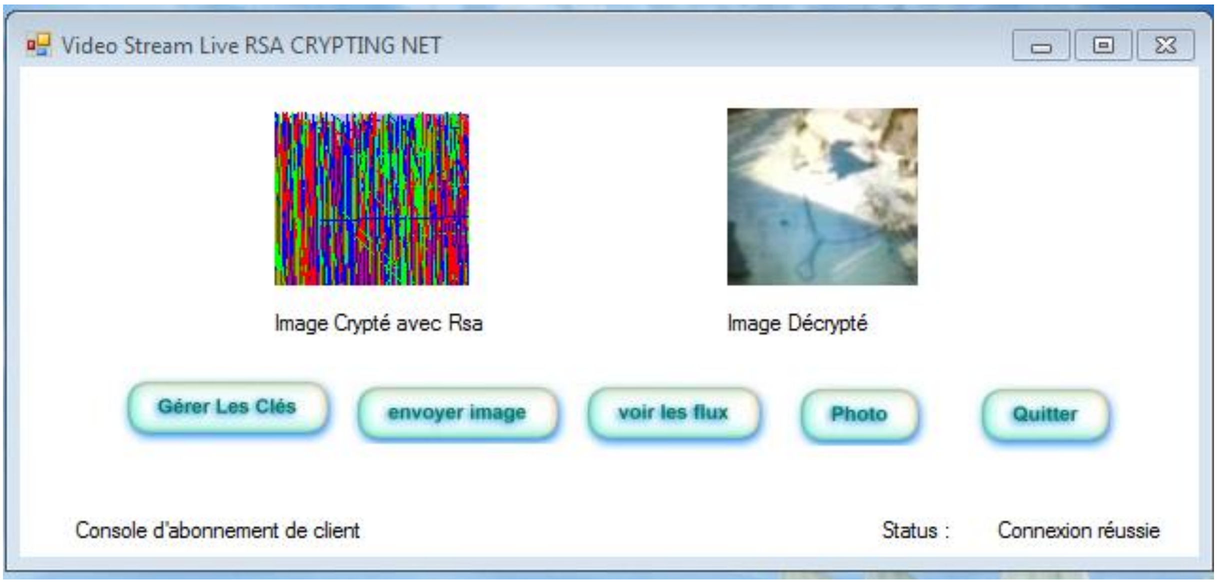


Figure 5.8 : affichage d'un vidéo quand information est bien reçu

Trame vidéo sans cryptage :

*255*216*255*224*0*16*74*70*73*70*0*1*1*1*0*96*0*96*0*0*255*219*0*67*0*8*6*6*7*6*5*8*7*7*7*9*9*8*10*12*20*13*12*11*11*12*25*18*19*15*20*29*26*31*30*29*26*28*28*32*36*46*39*32*34*44*35*28*28*40*55*41*44*48*49*52*52*52*31*39*57*61*56*50*60*46*51*52*50*255*219*0*67*1*9*9*9*12*11*12*24*13*13*24*50*33*28*33*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*50*

Trame vidéo crypté :

2232814+29760179+30351269+30351269+2232814+29760179+20160510+4198187+2232814+29760179+30351269+30351269+2232814+29760179+29760179+27216804+2232814+8689969+2232814+20160510+4198187+2232814+29620497+27216804+2232814+29620497+8689969+2232814+29620497+8171306+2232814+29620497+8689969+2232814+8689969+2232814+20160510+2232814+20160510+2232814+20160510+2232814+8689969+2232814+28506274+4198187+2232814+8689969+2232814+28506274+4198187+2232814+8689969+2232814+8689969+2232814+29760179+30351269+30351269+2232814+29760179+20160510+28506274+2232814+8689969+2232814+4198187+29620497+2232814+8689969+2232814+22456627+2232814+4198187+2232814+4198187+2232814+29620497+2232814+4198187+2232814+30351269+2232814+22456627+2232814+29620497+2232

Remarque :

- Le temp de chiffrement est chargé par le changement de vidéo

  Chaque chiffrement passe entre 290 millisecond dans un dual core
- Le temp de trasmission est chargé aussi par le changement vidéo et aussi la qualité de réseau entre le client et serveur
  dans le cas de local réseau le temps de transsmission et de 240 millisecond
- Le temp de Déchiffrement dans serveur est le meme que le client
- Le temp pour affiché une image dans serveur et de 530 millisecond
- l'algorithme RSA à clé publique c'est l'algorithme le plus lent mais elle donne meilleur sécurité

Vidéo Noir et Blanc

- Le temp de chiffrement est chargé par le changement de vidéo

  Chaque chiffrement passe entre 120 millisecond dans un dual core
- Le temp de trasmission est chargé aussi par le changement vidéo et aussi la qualité de réseau entre le client et serveur
  dans le cas de local réseau le temps de transsmission et de 95 millisecond
- Le temp de Déchiffrement dans serveur est le meme que le client
- Le temp pour affiché une image dans serveur et de 215 millisecond

  Résulat obtenus :
  La taille de vidéo augmante quand les couleurs est bien définit
  Donc la taille de trame augment quand la qualité d'image et plus haut

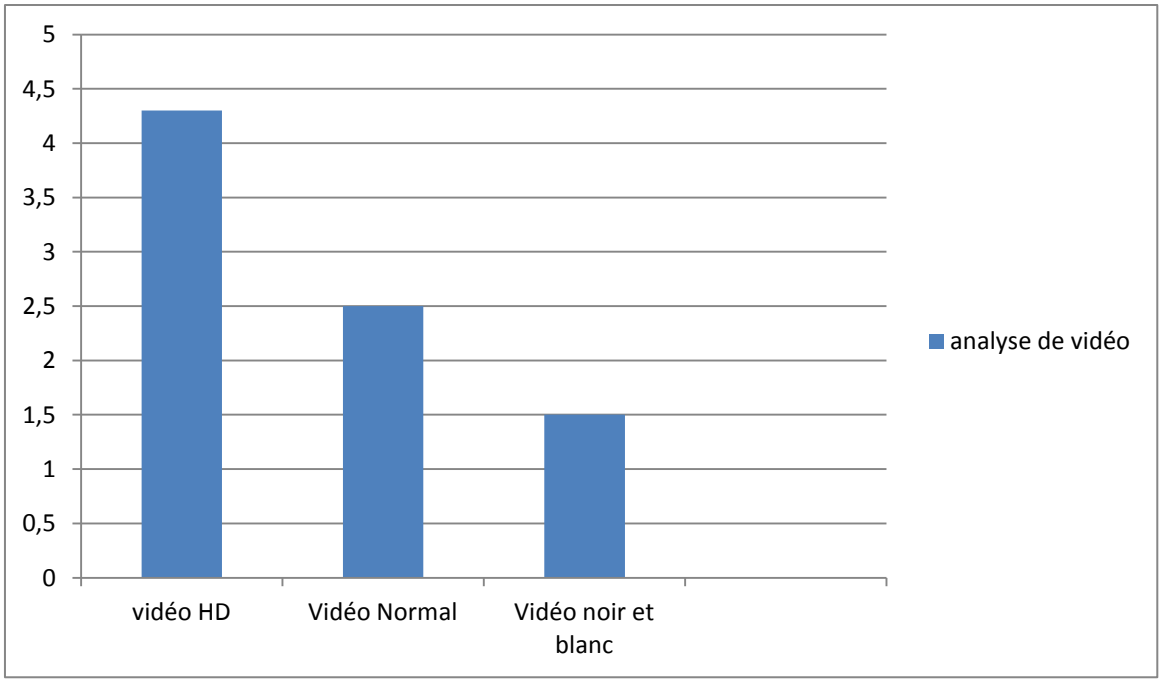


Figure 5.9 : Analyse des information obtenus par Histogramme

5.3 Conclusion

Notre objectif est d'assurer la sécurisé des flux vidéo echangé dans une architecture client et serveur on utilisons une algorithme de cryptage RSA qui garde une performance et une efficacité dans l'opération de chiffrement. Les critères de performance sont basés sur le temps de transmission et la longuer de trame chiffré

Conclusion générale

La sécurisation des informations échangé dans un réseaux , est devenus très intérissant dans

le domaine Infomratique .

La manipulation et la transmission de ces informations à travers des réseaux informatiques risquent d'être attaquées par des personnes non autorisées, dans ce cas là, il faut faire appel aux algorithmes de cryptages des données.

La cryptographie informatique professionnelle est une solution , rendu indispensable du fait que les informations sont accessibles pratiquement à tous par des réseaux publics.

Au cours de cette mémoire, nous avons étudié et implémenté la transmission de vidéo dans une architecture Client/Serveur avec un chiffrement des flux par l'algorithme RSA .

Les critères de comparaisons des cas dans le cryptage des qualité images sont :

- ✓ La vitesse et le temps de l'opération de chiffrement par l' algorithmes RSA étudiés dans notre projet de fin d'étude.
- ✓ La qualité et la performance de l'opération de cryptag .

Notre but principal est assure une connexion a distance sécurisé et authentifié Pour la transmission de flux vidéo , Cette solution est utilisé dans plusieurs domaine commerciel .

Comme percepective de notre travail, nous souhaitons une etude comparative sur la qualité de chiffrement de flux video avec d'autre algorithme de cryptography comme DES et IDEA

Référence

[1] – Network Programming for Microsoft Windows Second Edition ,MSPress2002

[2] - Bart Prenee ,*Understanding Cryptography , Springer* 2010

[3] - Sean Murphy , Cryptography. A Very Short Intro , Oxford University Press 2002

[4] - Douglas Stinson , Cryptography: Theory and Practice ,CRC Press 1995

[5] - William Stallings Cryptography-network-security-5th-edition, Prentice Hall 2011

[6] – Pierre Noizat , Bitcoin book , *Spring* 2012

[7] – Mark Stamp,Informatique security principle and practice , Wiley 2013

[8] - S. Bruce, Cryptographie appliquée- Algorithmes, protocoles,Wiley 1997

[9] - David Salomon , D*ata-compression-the-complete-reference, Springer 2007*

[10] -See R. Steinmetz, K. Nahrstedt. Multimedia: Computing, Communications & Applications. Innovative Technology Series. Prentice Hall P T R. 1995.

[11] -G. Richardson, H.264 and MPEG-4 Video Compression , Wiley 2003

[12]- Jacques Printz , Architecture Logiciel des applications adaptables , Dunod 2006

[13] - Matthew MacDonald ,Microsoft Visual Basic .NET Programmer's Cookbook

Référence site web

*[***s1***]* - *w3.polytech.univ-montp2.fr/~karen.godary/M1/Trans_Client_Serveur.pdf*

[**s2**] - nt.impmc.upmc.fr/impmc/Enseignement/ye/informatique/systemes/chap7/73.html

[s3] - *igm.univ-mlv.fr/~duris/NTREZO/20022003/ClientLeger.pdf*

[s4] - http://www.marthendiaye.com/cours/Coursclserv.pdf

[s5] - www.clusir-rha.fr/sites/default/files/upload/s2/ClusirCryptographie.pdf

[s6] - http://csrc.nist.gov/publications/nistpubs/800-56B/sp800-56B.pdf

[s7] - *math.univ-lille1.fr/~bhowmik/enseignement/Mem12/mem_crypto.pdf*

[s8]- www.student.montefiore.ulg.ac.be/~s091678/files/resume_crypto.pdf

[s9] -http://www.dpbestflow.org/Video_Format_Overview

[s10] - http://www.dpbestflow.org/node/627

[s11] - repaire.net/200806011528/articles/compression_mpeg_comprendre_visualiser.html

[s12] - http://icawww.epfl.ch/hamdi/these/Chap4.html#31763

Résumé

Les communications vidéo sont devenus une chose familière dans notre vie quotidienne de partout à l'heure actuelle. Ils sont devenus indispensables au bon fonctionnement général de nombreuses entreprises et administrations.

La Cryptographie utilisée pour dissimuler une information ,maintenant cette science est devient un domaine trés important dans la science de l' informatique .
L'objectif principal de notre projet est d'assurer le chiffrement des flux vidéo capté par une webcam est transmet sur un réseau de communication . Nous sommers intéressés par une architecture client serveur on utilisant un Algorithme de chiffrement a clé public RSA .

Abstract

Video communications have become a familiar thing in our daily life everywhere today. They have become essential to the overall functioning of many businesses and governments.

Cryptography used to hide something, now that science is becoming very important in the science of information technology. The main objective of our project is to ensure the video stream captured by a webcam transmits a communication network. We are interested in a client-server architecture is using encryption algorithm is RSA public key.

Keyword: server, client, RSA, video, image, MPEG, public key, private key.

خلاصة

أصبحت الاتصالات الفيديو شيء مألوف في حياتنا اليومية في كل مكان اليوم. فقد أصبحت ضرورية لسير العام للعديد من الشركات والحكومات.

التشفير المستخدمة لإخفاء شيء ما، الآن أن العلم أصبح مهم جدا في علم تكنولوجيا المعلومات. الهدف الرئيسي من مشروعنا هو ضمان تدفق الفيديو التي تم التقاطها بواسطة كاميرا ويب تنقل شبكة الاتصالات. نحن سومرز مهتم في الهندسة المعمارية خدمة العملاء وباستخدام خوارزمية التشفير RSA غير المفتاح العام.
الكلمة الرئيسية: الخادم، العميل، ، الفيديو، الصور، ، المفتاح العام، المفتاح الخاص