



République Algérienne Démocratique et Populaire  
Université Abou Bakr Belkaid– Tlemcen  
Faculté des Sciences  
Département d'Informatique

Mémoire de fin d'études pour l'obtention du diplôme  
Master en Informatique

*Option: Modèle Intelligent et Décision (M.I.D)*

*Thème*

## **Administration Réseaux informatiques**

Réalisé par :

- LAHFA Nadir
- HENAOUI Ali

- *Mme DIDI Fedoua* (Encadreur)
- *Mme Labraoui Nabila* (Présidente de jury)
- *M. Smahi Med Ismail* (Examineur)
- *Mme. Belhabi Amel* (Examinatrice)

Année universitaire: 2012-2013

## Remerciements

Nous tenons à remercier en premier lieu DIEU le tout puissant, pour ses faveurs et ses grâces, et de nous avoir donné le courage, la santé et la patience pour terminer ce travail.

Nos vifs remerciements à Madame **DIDI Fedoua**, notre encadreur, pour nous avoir fait l'honneur de nous encadrer, et nous guider par ses conseils avisés et son aide très précieuse.

Un grand remerciement à nos enseignants et enseignantes qui ont contribué à notre formation, depuis le cycle primaire au cursus universitaire.

Enfin, nos grâces s'adressent également à tous ceux qui nous ont aidés de près ou de loin à achever ce modeste travail.

## **DEDIDACES**

*Je dédie ce mémoire :*

*A mes très chères parents ; qui ont tout fait pour m'encourager  
durant toutes mes études, et grâce a eux que je suis arrivé à  
réaliser ce résultat.*

*A mes très chers frères **HADJ** et **SIDAHMED** pour leurs soutient  
moral indéfectible.*

*Au reste de toute la famille ainsi que mes proches amis qui n'ont  
cessé de m'encourager.*

*A mon collègue et ami LAHFA Nadir qui a été un binomial très  
compétant lors de la réalisation du présent mémoire.*

**Ali**

---

## DEDICACES

*Je dédie ce travail*

*A mes chers **parents** symbole de sacrifice, de tendresse, et qui m'ont donné un magnifique modèle de labeur et de persévérance. J'espère qu'ils trouveront dans ce travail toute ma reconnaissance et tout mon amour.*

*A ma chère **sœur** et mon cher **frère**.*

*Aux êtres chers auxquels je ne saurais exprimer ma gratitude et ma reconnaissance.*

*A mon collègue et ami **HENAOUI Ali** qui a été un binomial très compétant lors de la réalisation du présent mémoire.*

*A mes amis et tous mes camarades de la promotion 2012-2013.*

*Nadir*

## Tables des matieres

Introduction Générale.....	4
I. Chapitre 1 .....	5
A. Introduction .....	5
B. Administration réseau .....	5
B.1. Introduction .....	5
B.2 .Définition .....	5
B.3. L'administrateur réseau .....	6
B.4. Les objectifs de l'administration des réseaux pour un administrateur .....	7
B.5. Les fonctions d'administration .....	7
B.6. Sécurité de réseaux .....	8
B.7. Performance de réseaux .....	8
B.8. Matériel d'interconnexion .....	9
B.9. Visage de l'internet .....	9
B.10. L'architecture TCP/IP .....	9
C. Les Serveurs informatiques .....	11
C.1. Définition .....	11
C.2. Historique .....	12
C.3. Architecture Client/Serveur .....	13
C.4. Types de serveur .....	14
D. Conclusion.....	29
II.Chapitre 2 .....	30
1. Introduction .....	30
2. Serveur Web .....	30
2.1. APACHE .....	30
2.2.PHP .....	34
2.3.MySQL .....	37
2.4 .PHP MyAdmin .....	39
3.Serveur FTP.....	40
3.1 Introduction à FileZilla Server .....	40
3.2.Installation de FileZilla Server .....	41
3.4.Configuration du Serveur FTP .....	42
3.5.Connexion au serveur à partir d'un client FTP .....	45

3.6.Conclusion .....	46
4.Serveur Telnet .....	46
4.1.Activation de Telnet sous Windows Vista/7 .....	46
4.2.Commandes Telnet.....	48
4.3.Connexion à un terminal .....	48
5.Serveur DNS .....	50
5.1.Changer ses serveurs DNS sous Windows .....	50
5.2 .Pourquoi un DNS primaire et un DNS secondaire ? .....	51
5.3.DNS Jumper .....	51
5.4.Présentation de Simple DNS Plus .....	52
5.5.Utilisation de Simple DNS Plus .....	52
5.6.Conclusion.....	55
6.Serveur de messagerie électronique .....	55
6.1.Hmail Server .....	55
6.2.Installation .....	56
6.3.Configuration .....	58
6.4.Configuration du client de messagerie .....	62
6.5.SQUIRRELMAIL Webmail pour serveur Web .....	64
6.5.1.Configuration .....	65
6.5.2.Installation du patch Français .....	67
7. Conclusion.....	68
Conclusion générale.....	69

## **Résumé :**

Le domaine des réseaux informatiques est un domaine trop vaste, qui évolue trop vite. L'administration réseau rassemble les services Internet (*Telnet, ftp, mail,...*) pour tout le réseau qu'elle protège. Ceci nécessite de créer des comptes dédiés à ces services, ces comptes sont accessibles par un certain nombre de personnes du réseau. Nous avons réalisé un projet de fin d'étude visant ce domaine. On s'est proposé comme objectif d'installer et de configurer plusieurs serveurs, qui sont une partie intégrante d'un réseau privé :

- Serveur Web.
- Serveur FTP.
- Serveur Telnet.
- Serveur DNS.
- Serveur de Messagerie électronique.

L'administrateur réseau est l'homme-orchestre du système informatique d'une entreprise. Sa mission principale est de mettre en place un réseau informatique pour échanger des données en interne ou avec l'extérieur.

Cependant la conception du réseau informatique n'est qu'une infime partie du travail d'un administrateur, le plus important est la gestion et l'administration de ce réseau qui se fait en temps réel et en continu afin d'assurer le meilleur des services.

## **Abstract:**

The computer network is a field too large that is moving too fast. Network administration brings Internet services (telnet, ftp, email ...) for the entire network it protects. This requires Creating accounts dedicated to these services, these accounts are accessible to all people in the network. We had to make a final project to study this area, so we had set up a computer network consisting of five parts:

- Web server.
- FTP Server.
- Telnet Server.
- DNS Server.
- E-mail.

The network administrator is the one-man band of a company's IT system. Its main mission is to develop a computer network to share data internally or externally.

However the creation of the computer network is only a small part of the work of a network administrator, the most important thing is the management and administration of the entire network which is done in real time and continuously to ensure the best services.

# **Introduction Générale**

Nul doute que l'informatique a envahit tous les aspects de notre vie quotidienne, avec l'explosion d'internet, les réseaux sociaux, la toile, la ToIP, et tous les services que nous propose ce formidable outil.

Et donc tout naturellement tous les organismes qu'ils soient gouvernementaux, commerciaux ou privés se voient presque obligés de se doter d'un réseau informatique, qui est en passe de devenir un outil incontournable dans tous les domaines, aussi bien pour booster l'économie, que pour rester aussi performant que les autres, dans un monde où la concurrence est féroce et où la planète n'est plus qu'un petit village où l'information circule instantanément. Un réseau a donc nécessairement besoin d'un administrateur réseau et parfois même d'une équipe informatique de gestion de réseau. Son rôle est d'une telle importance et sa responsabilité est si grande que parfois on le considère comme la clef de voûte de l'économie de l'entreprise. Car si le réseau tombe ou à des problèmes, l'entreprise est en danger. Donc l'administration réseau est un ensemble de compétences à acquérir, en perpétuel changement et en évolution constante. Le choix du sujet de notre PFE s'est naturellement porté donc sur cet aspect des réseaux, et on se propose de maîtriser au moins quelques points primordiaux de l'administration réseau, entre autre l'installation, configuration des serveurs les plus importants faisant partie d'un réseau. Qui ne sont autre que le serveur MAIL, WEB, FTP, DNS et TELNET. Pour se faire, nous allons passer en revue dans ce manuscrit quelques notions de base, nécessaires à la compréhension du sujet, au chapitre un et au chapitre deux, nous donnerons le détail des installations et configurations, ainsi que les problèmes auxquels nous avons dû faire face, lors de ce travail. Nous donnerons aussi plusieurs conseils aux étudiants qui voudront approfondir leurs connaissances dans ce domaine passionnant, pour leur faire éviter une trop grande perte de temps à répéter les mêmes erreurs par lesquelles, nous on est passés. On a pris un grand plaisir à mener à bien ce travail, très bénéfique et très riche, mais nous sommes conscients que nul ne peut tout maîtriser et qu'il nous reste encore beaucoup à faire.



# I. Chapitre 1 :

## **A. Introduction :**

Dans ce chapitre, nous allons passer en revue quelques notions, bien que connues car de base, mais que nous jugeons opportun de les rappeler très brièvement pour une meilleure compréhension de l'avancement de l'objectif posé. Ce chapitre est composé de deux parties, la partie A résume ce qu'est l'administration réseau et la partie B s'intéresse aux serveurs, leur rôle et leur importance dans un réseau.

## **B. Administration réseau :**

### **B.1. Introduction :**

Le réseau est devenu une ressource indispensable au bon fonctionnement d'une organisation, une entreprise, une université,...

L'administration du réseau met en œuvre un ensemble de moyens pour :

- Offrir aux utilisateurs une certaine qualité de service;
- Permettre l'évolution du système en incluant de nouvelles fonctionnalités
- Optimiser les performances des services pour les utilisateurs
- Permettre une utilisation maximale des ressources pour un coût minimal
- Assurer la sécurité et la disponibilité des ressources du réseau à tout moment
- Contrôler l'accès au réseau et aux ressources par les utilisateurs

### **B.2 .Définition :**

L'administration Réseau est le processus permettant le CONTROLE d'un réseau de données pour en assurer l'efficacité et la productivité. Le but Final de l'administration Réseau est d'aider à maîtriser la complexité des réseaux de données et d'assurer que les données transitent sur le réseau avec le maximum d'efficacité et de transparence aux utilisateurs.

L'administration des réseaux informatiques évolue sans cesse et elle s'affirme aujourd'hui comme une activité clé de toute entreprise. En plus d'être constamment en évolution, ses outils d'échange de données et de partage d'information en temps réel doivent être en mesure d'offrir une confidentialité maximale et une sécurité à toute épreuve.

L'administration des réseaux informatiques implique le développement et la mise en place d'outils de gestion ayant pour but de tirer le maximum d'Internet, tout en mettant l'accent sur l'aspect sécurité de l'utilisation de cet outil maintenant indispensable.

En jouant un rôle de tout premier plan dans la gestion, l'échange et la transmission de l'information, l'administration *des réseaux informatiques* prend une importance capitale pour un nombre croissant d'entreprises. Bien au fait des produits et des services offerts dans le domaine et constamment à l'affût des innovations.

### **B.3. L'administrateur réseau :**

L'administrateur réseau est responsable de ce qui peut se passer à partir du réseau administré. L'administrateur réseau est chargé de la gestion, des comptes et des machines du réseau informatique d'une organisation. Il est souvent assisté d'un ingénieur qui conçoit l'architecture du réseau. L'administration réseau est une discipline de l'informatique qui peut s'étendre à la téléphonie. L'administrateur réseau est parfois également administrateur système, il gère alors aussi les postes de travail et les serveurs de l'entreprise.

L'administrateur d'aujourd'hui doit arriver à déjouer des envahisseurs virtuels qui disposent de nouvelles armes de plus en plus sophistiquées.

L'utilisation du matériel informatique et des logiciels doit se faire en fonction de l'environnement informatique actuel de chaque entreprise et avec deux mots d'ordre en tête : compatibilité et homogénéité. Outre les diverses installations à coordonner, il faut aussi s'assurer d'effectuer les mises à jour des manufacturiers sur une base constante.

C'est ce qui nous donne l'assurance de profiter d'une performance et d'une sécurité optimale.

La continuité des activités d'une entreprise dépend essentiellement de la qualité de son système d'information, qui repose en grande partie sur l'efficacité avec laquelle est géré son réseau.

Pare-feu, antivirus, gestion des courriers électroniques; voilà autant de produits et de services qui doivent être analysés avant d'être mis en place.

Le rôle d'un administrateur réseau consiste (entre autre) à :

- Mettre en place et maintenir l'infrastructure du réseau (organisation, . . .).
- Installer et maintenir les services nécessaires au bon fonctionnement du réseau.
- Assurer la sécurité des données internes au réseau
- S'assurer que les utilisateurs n'outrepassent pas leurs droits.
- Gérer les "logins" (i.e. noms d'utilisateurs, mot de passe, droits d'accès, permissions particulières, . . .).
- le trafic des données qui circulent sur le réseau.
- la sauvegarde des données.
- la politique de sécurité régissant tous les types d'accès au réseau (accès interne, accès à distance et interconnexion avec des tierces parties).
- la surveillance et l'assurance de la fiabilité générale du réseau.

#### **B.4. Les objectifs de l'administration des réseaux pour un administrateur :**

- Supervision du fonctionnement des réseaux.
- Optimisation pour l'utilisation des ressources.
- Détection et prévision des erreurs.
- Signalisation des pannes.
- Calculs de facturations à l'utilisation des ressources.
- Le support technique pour utilisateurs.

#### **B.5. Les fonctions d'administration :**

La station d'administration doit permettre :

- l'extraction des informations des éléments du réseau au moyen d'outils d'un grand nombre d'informations.
- la réduction du volume d'informations au moyen de filtres afin de sélectionner les informations significatives.
- le stockage des informations retenues dans une base de données d'administration

- des traitements sur ces informations
- offrir des interfaces (utilisateur d'administration administration, opérateur réseau).

### **B.6. Sécurité de réseaux :**

Avec l'apparition des nouvelles technologies et la diversification des types de réseaux comme la multiplication des mobiles connectés et le développement des solutions de cloud computing, la gestion des solutions de sécurité réseau est devenue une tâche complexe.

#### **Menaces**

Voici une liste des menaces pouvant affecter la sécurité d'un réseau informatique.

- Virus
- Cheval de Troie
- Mouchards
- Attaque par déni de service
- Analyseur de paquets
- Ingénierie sociale

#### **Outils**

Voici une liste des outils généralement utilisés pour gérer ces différentes menaces.

- Logiciel antivirus
- Réseau privé virtuel
- Service de vérification d'identité
- Chiffrement
- Gestion de la sécurité

### **B.7. Performance de réseaux :**

L'efficacité des réseaux dépend de la manière dont se font les échanges d'informations. Ces échanges sont effectués grâce à des mécanismes qui président comme les protocoles, ceux-ci représentent l'ensemble des règles décrivant la manière de faire transiter les informations sur un réseau. L'évaluation de la performance d'un réseau peut être effectuée de plusieurs façons et revient à mesurer la rapidité et la fiabilité d'une transmission de données.

## **B.8. Matériel d'interconnexion :**

### Routage et passerelle :

Il est primordial qu'un administrateur réseau soit en mesure de réussir la configuration d'une passerelle, d'un réseau privé VPN, d'un NAT (Network Address Translation), IP masquerading...

### Sécurité dans les réseaux :

La configuration d'un pare-feu (firewall) réussie, en plus de la maîtrise de la manipulation des tables IPTables et des Règles de filtrage, garantit une sécurité efficace.

### Outils de diagnostic :

De tels outils foisonnent sur internet comme nmap, encore faut-il en connaître les avantages et les failles.

### Configuration et manipulation de services spécifiques :

La Gestion d'utilisateurs distants (NIS), par l'utilisation d'un annuaire fédérateur (LDAP), le Transfert de fichiers et autres (FTP, TFTP, NFS, SMB), Connexions à distance (telnet, rlogin, ssh, X11, ...), les serveurs de noms (DNS) sont toutes des notions essentielles à l'administration réseau.

## **B.9. Visage de l'internet :**

Un réseau de réseaux, utilisant un ensemble de logiciels et de protocoles, basé sur l'architecture TCP/IP et fonctionnant en mode Client/serveur, offre un ensemble de services (e-mail, transfert de fichiers, connexion à distance, WWW, Gestion des noms et des adresses. . .). Cependant une somme « d'inventions » qui s'accroissent et qu'il est impossible de maîtriser totalement.

## **B.10. L'architecture TCP/IP :**

Une version simplifiée du modèle OSI, avec toutes les applications qu'on lui connaît comme FTP, WWW, telnet, SMTP, en couche application. La couche transport utilise TCP, UDP (entre 2 processus aux extrémités)

TCP : transfert fiable de données en mode connecté

UDP : transfert non garanti de données en mode non connecté

La couche Réseau responsable du routage essentiellement

L'administration réseau consiste justement à maîtriser tous les services offerts par un réseau, se trouvant essentiellement dans la couche application, mais d'autres services sont aussi nécessaires et se trouvant dans les autres couches.[1]

### Exemples de protocoles applicatifs :

#### HTTP - HyperText Transport Protocol

- Protocole du web
- Échange de requête/réponse entre un client et un serveur web

#### 2. FTP - File Transfer Protocol

- Protocole de manipulation de fichiers distants
- Transfert, suppression, création, ...

#### 3. TELNET - TELetypewriter Network Protocol :

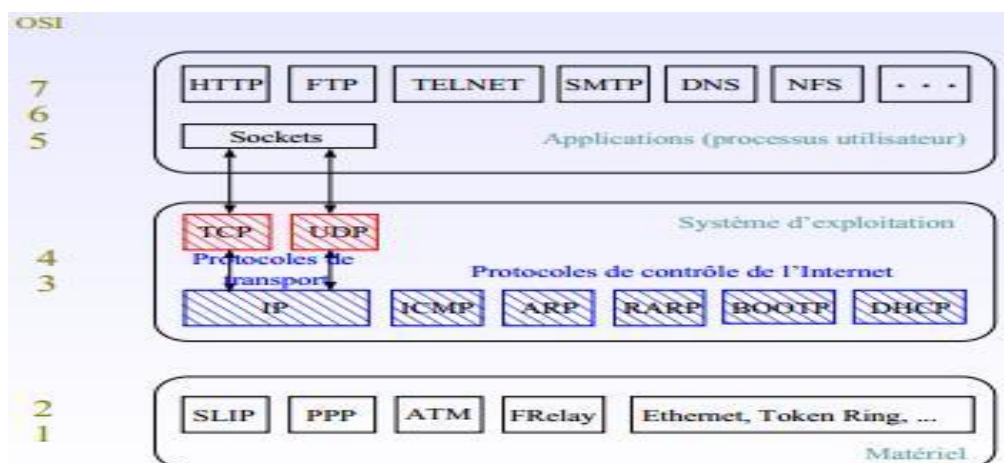
- Système de terminal virtuel
- Permet l'ouverture d'une session distante

#### 4. DNS - Domain Name System :

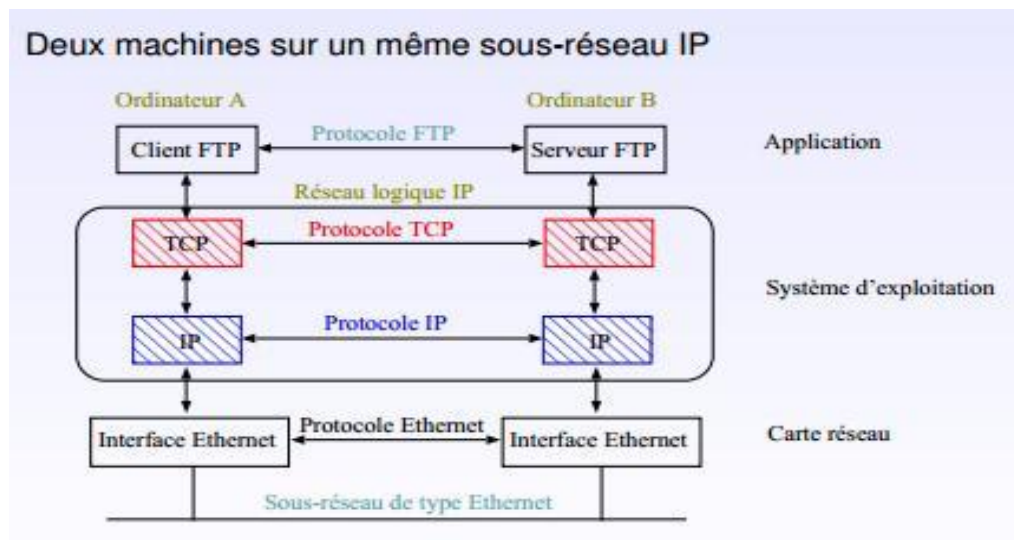
- Assure la correspondance entre un nom symbolique et une adresse Internet (adresse IP), dont les bases de données sont réparties sur le globe

#### 5. L'adressage IP :

Une adresse de transport = une adresse IP + un numéro de port (16 bits)



## Adressage IP fig I.B.1



## Adressage IP fig I.B.2

### C. Les Serveurs informatiques :

#### C.1. Définition :

Un **serveur informatique** est un dispositif informatique matériel ou logiciel qui offre des services, à différents clients. Les services les plus courants sont :

- le partage de fichiers ;
- l'accès aux informations du World Wide Web ;
- le courrier électronique ;
- le partage d'imprimantes ;
- le commerce électronique ;
- le stockage en base de données ;
- le jeu et la mise à disposition de logiciels applicatifs (optique software as a service).

Un serveur fonctionne en permanence, répondant automatiquement à des requêtes provenant d'autres dispositifs informatiques (les clients), selon le principe dit client-serveur. Le format des requêtes et des résultats est normalisé, se conforme à des protocoles réseaux et chaque service peut être exploité par tout client qui met en œuvre le protocole propre à ce service. Les serveurs sont utilisés par les entreprises, les institutions et les opérateurs de télécommunication. Ils sont courants dans les centres de traitement de données et le réseau Internet.

D'après le cabinet Netcraft, il y a en mars 2009 plus de 220 millions de serveurs web dans le monde, et leur nombre est en augmentation constante depuis l'invention du World Wide Web en 1995

### **C.2. Historique :**

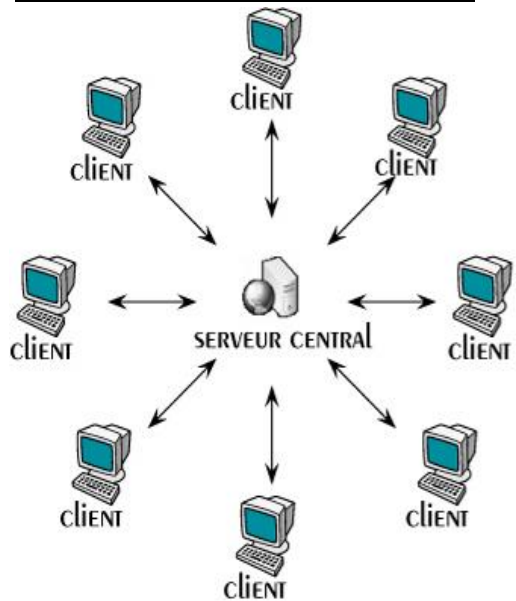
Au départ les utilisateurs accédaient aux serveurs par des terminaux passifs généralement alphanumériques (le minitel en est un exemple) et par la suite avec quelques capacités graphiques. Cette époque a laissé aux gens la notion de serveur central puissant supportant des clients faibles, en termes de capacité de calcul, mais permettant de contrôler le serveur. Aujourd'hui, alors qu'il existe des ordinateurs portables beaucoup plus puissants que les grands ordinateurs centraux des années 1970, et qui peuvent être utilisés en tant que clients pour naviguer sur le World Wide Web, ceci peut prêter à confusion.

L'évolution de la mise en réseau peut se résumer en trois phases:

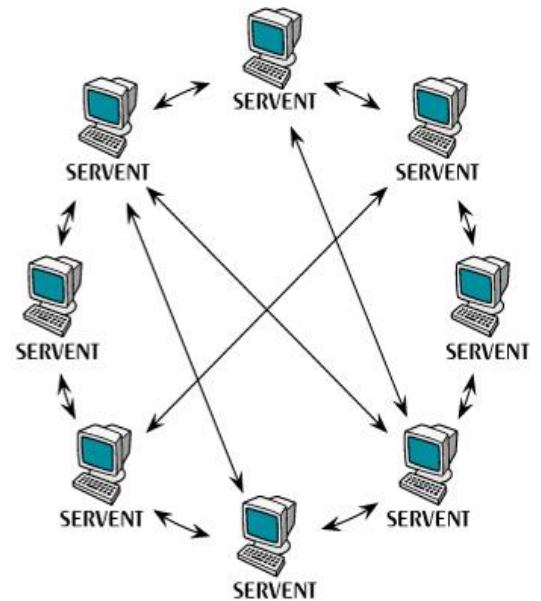
- Dans un premier temps, un réseau est centralisé: le serveur (seul élément de calcul) est le cœur du réseau, autour gravitent les terminaux, parfois très nombreux.
- Dans un second temps, au début des années 1980 les terminaux sont remplacés par des machines actives (les PCs). Le serveur n'est plus alors une machine, mais un programme. Sur la même machine peuvent se trouver plusieurs serveurs et plusieurs clients en même temps. Dans le système d'exploitation Linux par exemple, l'écran graphique est géré sur le mode client serveur. Le serveur fournit un service au client, il n'est plus symbole de pouvoir. L'image du serveur dans une brasserie convient mieux: le client lui demande une bière, et il l'apporte. Sur internet, toute machine peut être à la fois et en même temps cliente et serveur.
- Dans un troisième temps, au milieu des années 1990, les extensions de l'architecture client serveur vont explorer deux directions: la **délégation** (un serveur s'adresse à un autre serveur, comme dans l'architecture **3 ou n-tiers**), et le **client riche**



### C.3. Architecture Client/serveur :



ARCHITECTURE CLIENT-SERVEUR



ARCHITECTURE PAIR-À-PAIR

Architecture Client/serveur fig I.C.3

L'architecture client serveur s'appuie sur un poste central, le serveur, qui envoie des données aux machines clientes, des programmes qui accèdent au serveur sont appelés programmes clients (client FTP, client mail).

L'architecture **client/serveur** désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs postes clients du serveur : chaque logiciel client peut envoyer des requêtes à un serveur. Un serveur peut être spécialisé en serveur d'applications, de fichiers, de terminaux, ou encore de messagerie électronique.

Caractéristiques d'un serveur :

- il est passif ;
- il est à l'écoute, prêt à répondre aux requêtes envoyées par des clients ;
- dès qu'une requête lui parvient, il la traite et envoie une réponse.

Caractéristiques d'un client :

- il est actif ;
- il envoie des requêtes au serveur ;
- il attend et reçoit les réponses du serveur.

#### Avantages de cette architecture :

- Unicité de l'information : pour un site web dynamique par exemple (comme vulgarisation-informatique.com), certains articles du site sont stockés dans une base de données sur le serveur. De cette manière, les informations restent identiques. Chaque utilisateur accède aux mêmes informations.
- Meilleure sécurité : Lors de la connexion un PC client ne voit que le serveur, et non les autres PC clients. De même, les serveurs sont en général très sécurisés contre les attaques de pirates.
- Meilleure fiabilité : En cas de panne, seul le serveur fait l'objet d'une réparation, et non le PC client.
- Facilité d'évolution : Une architecture client/serveur est évolutive car il est très facile de rajouter ou d'enlever des clients, et même des serveurs.

#### Inconvénients de cette architecture :

- Un coût d'exploitation élevé (bande passante, câbles, ordinateurs surpuissants)

### **C.4. Types de serveur :**

#### **C.4.1. Serveur Central :**

En informatique, un serveur "central", centralise un service. Ils sont utilisés dans une architecture "*centralisée*", plutôt que la deuxième solution : une architecture "*décentralisée*".

Il existe différents types de centralisation :

- La centralisation de données ; il s'agit ici d'un mécanisme qui stocke l'ensemble des données sur un serveur concernant l'ensemble des éléments du réseau informatique.
- La centralisation de direction ; un seul serveur décide de qui, ou quoi, faut quoi et quand.
- La centralisation de communication ; toutes les communications passent par le serveur central.

Ces trois centralisations étaient à l'origine, la principale organisation d'un réseau informatique. Dans les années 1980, tous les utilisateurs de serveur se connectaient à lui par un terminal informatique ; cette structure, pratique pour l'organisation, pose toutefois des

problèmes de sécurité du réseau... Exemple : si le serveur central tombe en panne, tout le réseau ne fonctionne plus. Mais encore, pour les pirates, il n'y a qu'une seule cible, et enfin, saturation du serveur central s'il y'a trop de connexion simultanées.

#### **C.4.2. Serveur d'Application:**

Un serveur d'application est un serveur (ordinateur), sur lequel sont installées des applications utilisées par les particuliers. Elles sont accédées par le réseau. Dans l'entreprise actuelle, l'outil informatique a pris une place prépondérante. Des applications qui interviennent dans toutes les étapes du processus métier, de la production à la comptabilité. Ces applications, qui étaient souvent disposées sur différents postes, sont aujourd'hui de plus en plus centralisées sur des serveurs d'application. Ces serveurs sont de larges systèmes contenant les différentes applications de l'entreprise.

Dans une infrastructure régulière, on trouve plusieurs serveurs d'applications, mais il n'est pas impossible qu'il n'y en a qu'un seul, sur lequel toutes les applications seraient installées. Les applications sont chargées sur le serveur et leur résultat est affiché sur les écrans des terminaux utilisés par les clients. Les serveurs d'application sont des logiciels occupant la couche centrale dans une architecture multicouche.

#### **C.4.3. Serveur de fichiers:**

un serveur de fichier permet simplement de partager des données à travers un réseau. Le terme désigne généralement l'ordinateur sur lequel est installé le logiciel applicatif. Cet ordinateur possède généralement un gros espace disque (plusieurs centaines de Go, voire To), où sont déposés les fichiers, que les utilisateurs peuvent récupérer au moyen d'un protocole de partage de fichiers. On utilise généralement l'un des trois :

- FTP (File Transfer Protocol)
- CIFS (Common Internet File System)
- NFS (Network File System)

Le choix du protocole dépend principalement de la méthode d'accès des utilisateurs. CIFS est utilisé par les systèmes d'exploitation Microsoft Windows, NFS est répandu dans le milieu UNIX. Toutefois des implémentations de ces protocoles sont disponibles pour tout type de

système. Ces deux protocoles permettent d'établir des liaisons permanentes entre le client et le serveur. FTP est utilisé pour des connexions ponctuelles lorsque le client n'a pas besoin d'être connecté en permanence au serveur de fichier.

#### **C.4.4. Serveur d'impression:**

Un **serveur d'impression** est un serveur qui permet de partager une imprimante entre plusieurs utilisateurs situé sur un même réseau informatique. Le serveur dispose donc d'une entrée réseau (par exemple un port RJ45 pour le réseau Ethernet) qui gère les protocoles réseaux. Et également, d'une ou plusieurs sorties permettant de connecter chacune des imprimantes (USB généralement).

Le système d'impression qui est le plus utilisé aujourd'hui sous Linux et Unix est CUPS (Common Unix Printing System).

#### **C.4.5. Serveur de messagerie électronique :**

Un **serveur de messagerie électronique** est un logiciel serveur de courrier électronique (courriel). Il a pour vocation de transférer les messages électroniques d'un serveur à un autre. Un utilisateur n'est jamais en contact direct avec ce serveur mais utilise soit un client de messagerie, soit un courriel web, qui se charge de contacter le serveur pour envoyer ou recevoir les messages.

La plupart des serveurs de messagerie possèdent ces deux fonctions (envoi/réception), mais elles sont indépendantes et peuvent être dissociées physiquement en utilisant plusieurs serveurs.

- Pour envoyer et recevoir des messages, il est nécessaire de posséder un compte de messagerie ou boîte mail.

Le compte de messagerie est caractérisé généralement par une adresse de messagerie.

- Les termes e-mail, courrier électronique, messagerie Internet sont aussi utilisés pour définir une adresse de messagerie électronique.

- Un système de messagerie électronique est l'ensemble des éléments contribuant à transmettre un **courriel** (courrier électronique : message transmis via un réseau informatique) de l'émetteur au récepteur. Il y a quatre éléments fondamentaux. Ce sont:

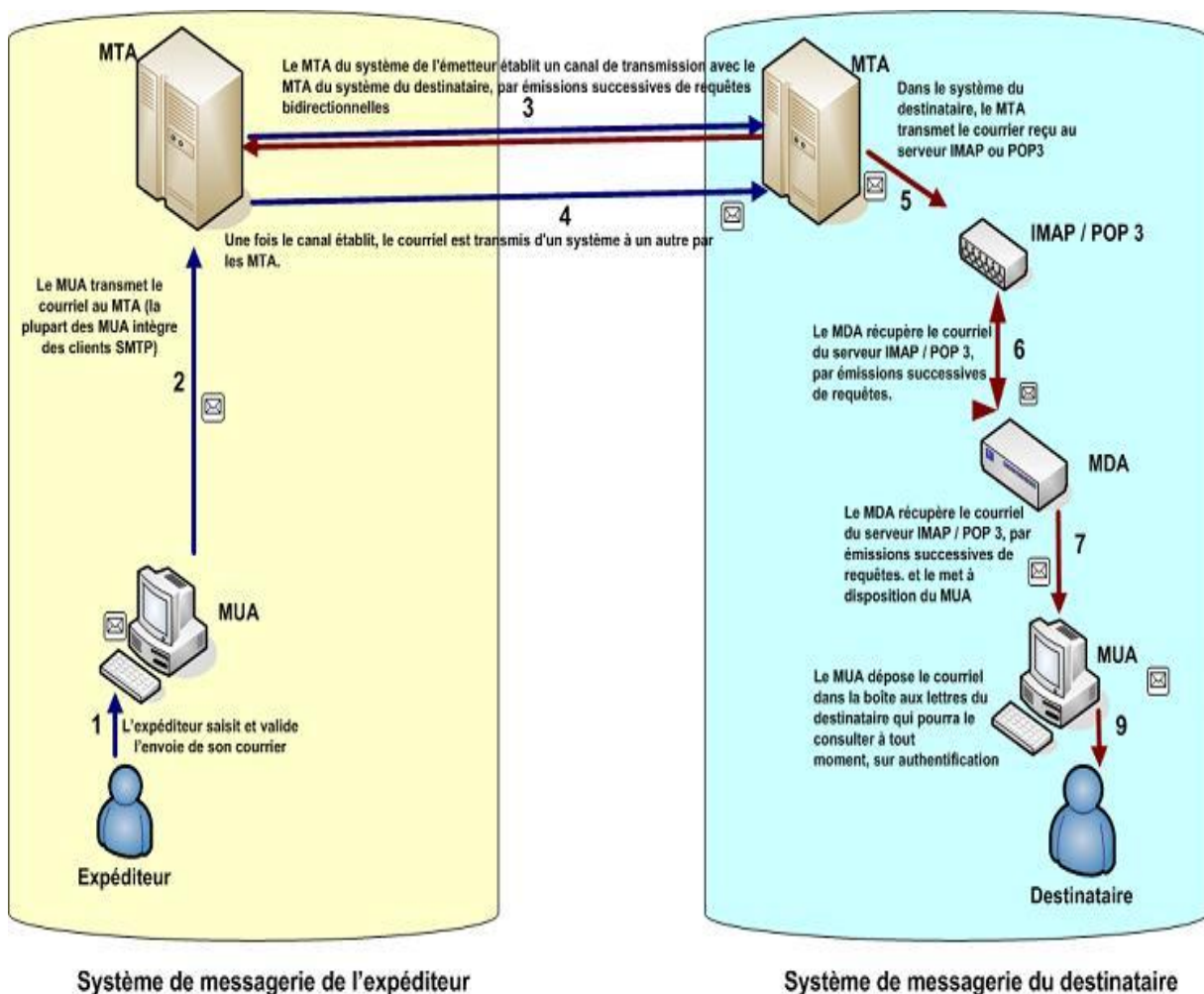
- **Le Mail Transfert Agent ou MTA**
- **Le serveur du protocole entrant**
- **Le Mail Delivery Agent ou MDA**
- **Le Mail User Agent ou MUA**

### Description de l'architecture de fonctionnement :

Les différents éléments du système de messagerie sont agencés selon une architecture logique :

Pour en assurer le fonctionnement.

Nous représentons cette architecture par le schéma suivant



## Architecture et fonctionnement messagerie fig I.C.4

Ce schéma présente le transfert d'un courriel d'un expéditeur à un destinataire.

**1**-L'expéditeur communique son courriel via le MUA.

**2** - Le MUA transmet ce courriel au MTA (la plupart des MUA intègre des clients SMTP).

**3 et 4**- Le MTA du système de l'émetteur établit un canal de transmission avec le MTA du système du destinataire, par émissions successives de requêtes bidirectionnelles.

**5** - Une fois le canal établi, le courriel est transmis d'un système à un autre par les MTA.

**6** - Dans le système du destinataire, Le MTA transmet le courrier reçu au serveur IMAP ou POP3.

**7, 8 et 9** - Le MDA récupère le courriel du serveur IMAP / POP 3, et le met à disposition du MDA.

**10** - Le MDA dépose le courriel dans les boîtes aux lettres du destinataire qui pourra le consulter à tout moment, sur authentification.

### Transfert Agent ou MTA

C'est un agent qui permet d'acheminer le courriel d'un serveur à un autre. Le **MTA** de l'émetteur route le mail sur le MTA du récepteur. Il implémente un protocole sortant. Notons que les protocoles sortants permettent de gérer la transmission du courrier entre les systèmes de messagerie. Le protocole sortant généralement utilisé est le **Simple Mail Transfert Protocol** ou **SMTP**.

SMTP peut être traduit comme *protocole simple de transfert de courriel*. Il est de la famille des protocoles basés sur TCP/IP. Il utilise généralement le port 25. Le mécanisme de fonctionnement de SMTP est qu'il commence d'abord par vérifier l'existence de l'expéditeur et du ou des destinataire (s), indiqués dans l'entête du message, puis il transmet le contenu.

La transmission s'effectue sur un canal de communication établi entre l'émetteur et le destinataire par émission bidirectionnelle de requêtes basées sur des commandes. Il existe plusieurs serveurs MTA qui implémentent SMTP. Parmi les plus connus, il y a *Postfix, Exim, Qmail* et *Sendmail*.

### **Serveur du protocole entrant :**

Les protocoles entrants permettent la réception et la distribution du courriel. Les plus généralement utilisés sont : **Post Office Protocol version 3 (POP3)** et **Internet Message Access Protocol (IMAP)**, qui sont tous deux basées sur TCP/IP. Dans son fonctionnement, POP3 va récupérer le courriel sur un serveur de messagerie.

IMAP est une version améliorée de POP3 qui intègre les opérations suivantes:

- La gestion de plusieurs accès simultanés
- La gestion de plusieurs boîtes aux lettres
- La synchronisation des courriels entre le serveur et le client (les courriels ne sont pas effacés sur le serveur de messagerie comme le fait POP3)
- Le tri du courrier selon des critères

Dans le système de messagerie, il est nécessaire d'avoir un serveur qui implémente un protocole entrant.

Il existe une version sécurisée d'IMAP : **IMAPS** qui intègre le protocole **SSL** (*Secure Socket Layer*).

### **Mail Delivery Agent ou MDA:**

Il s'agit d'un agent qui est en charge de la gestion des boîtes aux lettres. Il est chargé de livrer le courriel dans la boîte à messages du destinataire. Pour cela, il est souvent considéré comme le point final d'un système de messagerie. Dans le MDA, on peut filtrer les courriels; et même supprimer les spams par des anti-spams (comme spam assassin) et contrôler les virus par des antivirus. Il existe plusieurs serveurs MDA, les plus courants sont *procmail*, *maildrop* et *cyrus*.

### **Mail User Agent ou MUA :**

Le **MUA** est un logiciel client de messagerie qui fournit un environnement pour la gestion du courriel (envoi, saisie, réception, suppression, etc.). Il est très proche du MDA.

Tout comme les autres agents, il existe également plusieurs MUA. Un MUA avec une interface Web, est appelé Webmail.

### **Description des protocoles utilisés par les messageries:**

#### **POP3 (post Office Protocol 3) :**

POP3 est un protocole qui permet de récupérer les courriers électroniques situés sur un serveur de messagerie électronique. Le port de communication utilisé par ce protocole est le port 110. Le principe est simple, les messages sont conservés sur un serveur et lorsqu'un client se connecte, les messages lui sont remis. Il est normalement prévu qu'après "livraison" des messages, ces derniers soient supprimés du serveur. Mais depuis la version 3 du protocole, il est possible de laisser une copie du message sur le serveur.

#### **SMTP (Simple Mail Transfert Protocol) :**

SMTP, est un protocole standard de communication permettant de transférer le courrier d'un serveur à un autre en connexion point à point.

Il s'agit d'un protocole fonctionnant en mode connecté, encapsulé dans une trame TCP/IP. Le courrier est remis directement au serveur de courrier du destinataire. Le protocole SMTP fonctionne grâce à des commandes textuelles envoyées au serveur SMTP. Chacune des commandes envoyées par le client (validée par la chaîne de caractères ASCII CR/LF, équivalent à un appui sur la touche entrée) est suivi d'une réponse du serveur SMTP composée d'un numéro et d'un message descriptif.

Ce protocole est le complément idéal des protocoles POP et IMAP.

Le port de communication utilisé par ce protocole est le port 25.

#### **IMAP (Internet Mail Access Protocol) :**

Internet Message Access Protocol (IMAP) est un protocole utilisé par les serveurs de messagerie électronique, fonctionnant pour la réception.



Ce protocole permet de laisser les e-mails sur le serveur dans le but de pouvoir les consulter de différents clients e-mails ou webmail. Il comporte des fonctionnalités avancées, comme les boîtes aux lettres multiples, la possibilité de créer des dossiers pour trier ses e-mails... Le fait que les messages soient archivés sur le serveur fait que l'utilisateur peut accéder à tous ses messages depuis n'importe où sur le réseau et que l'administrateur peut facilement faire des copies de sauvegarde.

**Remarque :**

Si vous hésitez entre **POP** et **IMAP**, préférez **IMAP**, qui est un protocole permettant l'utilisation de la boîte depuis plusieurs machines (client mail sur ordinateur, PDA/téléphone, webmail, etc.) et qui laisse les messages sur le serveur, supportant un très grand nombre de mails à la fois.

**C.4.6. Serveur de base de données :**

Un serveur de base de données répond à des demandes de manipulation de données stockées dans une ou plusieurs bases de données. Il s'agit typiquement de demandes de recherche, de tri, d'ajout, de modification ou de suppression de données.

Le serveur de base de données fait partie d'un système de gestion de base de données (abréviation *SGBD*) — logiciel qui manipule une base de données — qui comporte un logiciel client et un logiciel serveur. Les demandes de manipulation de données sont souvent créées par un logiciel de gestion sous forme de requêtes en langage SQL, puis le client les transmet au serveur en utilisant un protocole propre au SGBD.

Oracle, Microsoft SQL Server ou MySQL sont des SGBD qui comportent un serveur de base de données

Le choix du matériel informatique sur lequel on installe un SGBD est fonction, comme ce dernier, du volume des données stockées dans la base et du nombre maximum d'utilisateurs simultanés.

**C.4.7. Serveur web :**

Un **serveur Web** peut être :

- un ordinateur tenant le rôle de serveur informatique sur lequel fonctionne un logiciel serveur HTTP ;
- un serveur HTTP lui-même ;
- un ensemble de serveurs permettant le fonctionnement d'applications Web.

Le plus souvent, un serveur Web fait fonctionner plusieurs logiciels qui fonctionnent en parallèle. On retrouve la combinaison Apache (serveur HTTP), MySQL (serveur de base de données) et PHP, tous libres. Cette combinaison s'appelle :

- Sous Linux, LAMP (sigle de « Linux, Apache, MySQL, PHP »).
- Sous Windows, WAMP (« Windows, Apache, MySQL, PHP »).
- Sous Mac, MAMP (« Macintosh, Apache, MySQL, PHP »).

La plupart des ordinateurs utilisés comme serveur Web sont reliés à Internet et hébergent des sites Web du World Wide Web. Les autres serveurs se trouvent sur des intranets et hébergent des documents internes d'une entreprise, d'une administration, etc...

### **Serveur HTTP et serveur Web:**

Le serveur HTTP (HyperText Transfer Protocol) a été développé par le World Wide Web. Sa création est due à un défaut du FTP qui ne supportait pas un format de donnée très précis, le HTTP fût en clair une révolution, puisqu'il permet de coder les données en nombre binaire.

Le serveur HTTP a connu une série d'évolution, pour pallier aux défauts des anciennes versions, dont notamment la durée de connexion entre un internaute et le serveur.

Les ordinateurs utilisant le serveur HTTP sont appelés " serveur web". Un certain nombre de ces ordinateurs ont choisi de servir d'hébergeur pour site web.

Les clients du serveur HTTP sont principalement les navigateurs internet comme "Internet Explorer". Ainsi lorsqu'un internaute va taper un nom de domaine quelconque, un serveur HTTP va lui répondre et le navigateur va donc lui afficher une page dont les données sont codées avec le format HTML.

Les serveurs HTTP les plus connus sont Apache HTTP et Internet Information Services.

Un *serveur HTTP* est un logiciel qui, lorsqu'il est lancé sur une machine reliée à Internet, permet à tout utilisateur d'Internet de se connecter sur cette machine avec un client HTTP tel

que Netscape Navigator ou NCSA Mosaic et d'obtenir des informations, en particulier des documents HTML.

On parle de *serveur HTTP* parce que leur rôle est de répondre à des requêtes et d'envoyer des pages d'information en utilisant le protocole HTTP (HyperText Transfer Protocol). Néanmoins, ces serveurs font souvent beaucoup plus que cela, à tel point qu'on pourrait se demander si l'appellation "*serveur HTTP*" n'est pas un peu réductrice. On parlera également de "*Serveur WWW*" ou de "*HTTPd*", le '*d*' signifiant ici *daemon*, terme utilisé dans la terminologie UNIX pour désigner un programme qui attend en permanence des événements (ici des requêtes HTTP) pour entreprendre des actions (ici envoyer des documents par HTTP).

### **Bénéfices :**

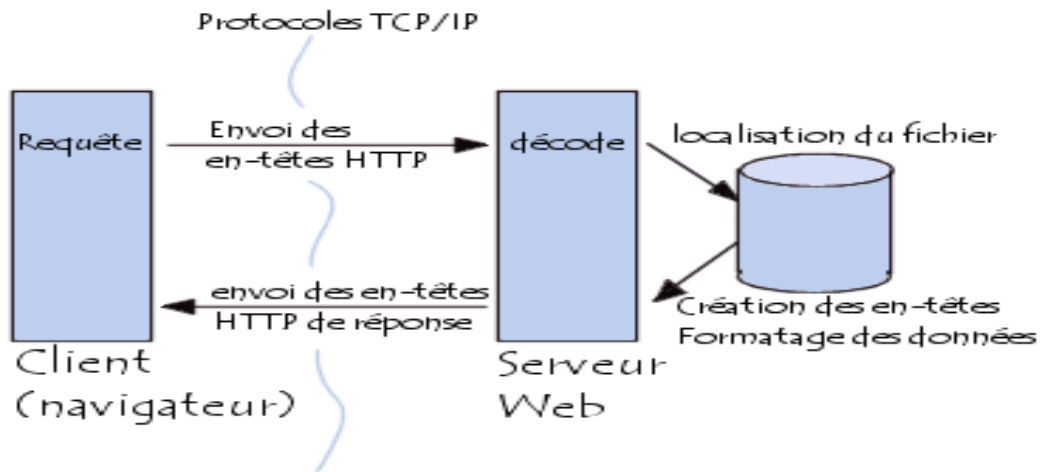
Les bénéfices que nous pouvons tirer de la mise en œuvre de son propre serveur HTTP sont multiples. Tout d'abord, si une organisation ne dispose encore d'aucun serveur web, elle devra choisir entre faire héberger son service web chez un prestataire ou alors installer son propre serveur web.

Par ailleurs, même si une organisation dispose déjà d'un ou de plusieurs sites web hébergés, il peut quand même être intéressant de disposer de son propre serveur, par exemple pour avoir un contrôle direct sur ses programmes CGI ou sur les statistiques de connexions.

Dans tous les cas, les possibilités offertes à l'administrateur d'un serveur WWW sont bien plus larges que celles de la personne qui se contente d'écrire quelques pages HTML et de les placer sur un serveur géré par quelqu'un d'autre.

### **Communication entre navigateur et serveur :**

La communication entre le navigateur et le serveur se fait en deux temps :



Communication Serveur/Navigateur fig I.C.5

Le navigateur effectue une **requête HTTP**

Le serveur traite la requête puis envoie une **réponse HTTP**

En réalité la communication s'effectue en plus de temps si on considère le traitement de la requête par le serveur. Etant donné que l'on s'intéresse uniquement au protocole HTTP, le traitement du côté serveur ne sera pas explicité dans le cadre de cet article... Si ce sujet vous intéresse, référez-vous à l'article sur le traitement des CGI.

### Navigateur Web :

Un **navigateur Web** est un logiciel conçu pour consulter le World Wide Web. Techniquement, c'est au minimum un client HTTP.

Le terme *navigateur Web* est inspiré de Netscape Navigator. D'autres métaphores sont ou ont été utilisées. Le premier terme utilisé était *browser*, comme en anglais. Par la suite, on a vu *fureteur* (surtout utilisé au Québec), *butineur* et *brouteur*, attestés par l'Académie française, *arpenteur*, *fouineur* ou encore *explorateur* (inspiré d'*Internet Explorer*). Le terme *navigateur internet*, bien qu'incorrect, est également souvent rencontré. Aujourd'hui, les termes *navigateur* ou *logiciel de navigation* sont recommandés en France par la DGLFLF, ainsi qu'au Canada par l'OQLF.

Parmi les navigateurs les plus répandus, on cite :

- Mozilla Firefox
- Google Chrome

- Opera
- Apple Safari

Les navigateurs existent pour les tablettes PC, les ordinateurs portables ou ultraportables et la téléphonie.

### **Logiciels de serveur HTTP :**

Les serveurs HTTP les plus utilisés sont :

- Apache HTTP Server de l'Apache Software Foundation, successeur du NCSA HTTPd ;
- Internet Information Services (IIS) de Microsoft ;
- Sun Java System Web Server de Sun Microsystems (anciennement iPlanet de Netscape, puis Sun ONE de Sun Microsystems) ;
- Zeus Web Server de Zeus Technology ;
- lighttpd de Jan Kneschke ;
- nginx d'Igor Sysoev.

Le serveur HTTP le plus utilisé est Apache HTTP Server qui sert environ 60% des sites Web en 2007 selon Netcraft.

Historiquement, d'autres serveurs HTTP importants furent CERN httpd, développé par les inventeurs du Web, abandonné le 15 juillet 1996 et NCSA HTTPd, développé au NCSA en même temps que NCSA Mosaic, abandonné mi-1994, ainsi que WebObjects.

### **C.4.8. Serveur DNS :**

Dans le monde de l'Internet, les machines du réseau sont identifiées par des adresses IP. Néanmoins, ces adresses ne sont pas très agréables à manipuler, c'est pourquoi, on utilise les

noms. L'objectif a alors été de permettre la résolution des noms de domaines qui consiste à assurer la conversion entre les noms d'hôtes et les adresses IP. La solution actuelle est l'utilisation des Dns (Domain Name System). Le système de nom ou DNS est une base de données permettant la traduction de noms d'hôtes en adresse IP.

Le DNS résout le problème essentiel lors de connexion d'ordinateurs à intranet ou Internet : il est en effet aisé de se souvenir de noms que de nombres.

### **Rôle de serveur DNS:**

Le service DNS est essentiel aux réseaux et surtout à Internet. Il est utilisé par tous même si les utilisateurs ne le réalisent pas. En effet, les échanges sur les réseaux se font à partir d'adresses IP.

Chaque utilisateur est connecté par une adresse IP, et lorsque l'on se connecte à un site, on envoie des mails on sollicite en fait une IP. Pourquoi les utilisateurs ne voient pas les IP ? Car pour accéder à Internet, un PC a besoin de l'IP d'un serveur DNS. Ce serveur DNS va faire pour lui les résolutions DNS, c'est-à-dire convertir les noms en IP (et les IP en nom si nécessaire).

Par exemple, lorsque vous ouvrez un navigateur et pointez sur <http://www.google.fr>, votre PC envoie une requête à votre serveur DNS qui lui dit [www.google.com](http://www.google.com) = 216.239.37.99 et votre navigateur sollicite en réalité <http://216.239.37.99>.

### **C.4.9. Serveur FTP :**

FTP obéit à un modèle client-serveur, c'est-à-dire qu'une des deux parties, le *client*, envoie des requêtes auxquelles réagit l'autre, appelé *serveur*. En pratique, le serveur est un ordinateur sur lequel fonctionne un logiciel lui-même appelé serveur FTP, qui rend public une arborescence de fichiers similaire à un système de fichiers UNIX. Pour accéder à un serveur FTP, on utilise un logiciel client FTP (possédant une interface graphique ou en ligne de commande).

Le protocole, qui appartient à la couche application du modèle OSI et à la couche application du modèle ARPA, utilise une connexion TCP. Il peut s'utiliser de deux façons différentes :

- **Mode actif** : c'est le client FTP qui détermine le port de connexion à utiliser pour permettre le transfert des données
- **Mode passif** : le serveur FTP détermine lui-même le port de connexion à utiliser pour permettre le transfert des données (data connexion) et le communique au client

Le port par défaut et le plus souvent utilisé est le port 21

#### **C.4.10. Serveur Telnet**

TELNET est le service permettant l'exécution de programmes à distance, en général sur un hôte de type Unix.

La commande **Telnet** vous permet de vous connecter sur une machine distante et d'y travailler exactement comme si vous étiez devant cet ordinateur.

Lors de la connexion à une machine distante, vous devez fournir un nom d'utilisateur et un mot de passe car l'accès sur le port **23** est contrôlé.

Telnet est un protocole permettant d'émuler un terminal à distance. Cela permet d'exécuter des commandes sur une machine distante par l'entremise d'un bon vieux clavier. Vous pouvez ainsi vous connecter sur un serveur mail, sur votre modem, etc.

#### **Sécurité :**

Le côté sommaire de Telnet fait que toute communication est transmise en clair sur le réseau, mots de passe compris. Des *sniffeurs* comme `tcpdump` ou `Wireshark` permettent d'intercepter les communications de la commande Telnet. Des protocoles chiffrés comme SSH ont été développés pour fournir un accès distant remplaçant Telnet et dont l'interception ne fournit aucune donnée utilisable à un éventuel espion du net

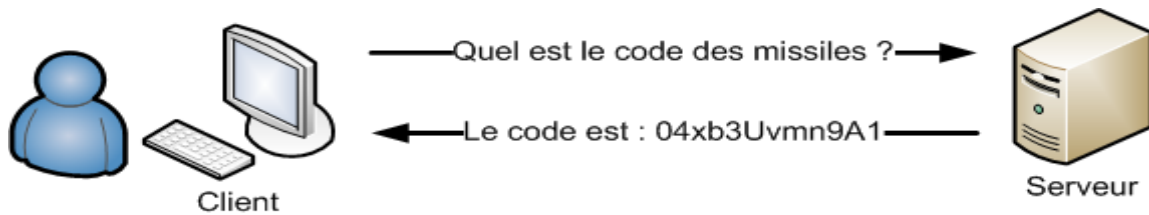
#### **Le protocole Telnet : simple mais dangereux**

Un protocole très simple, très basique, a été créé dans les années 80 : c'est **Telnet**. Il sert juste à échanger des messages simples d'une machine à une autre.

En théorie donc, on peut communiquer avec un serveur à l'aide du protocole Telnet. Le problème de ce protocole... c'est justement qu'il est trop simple :

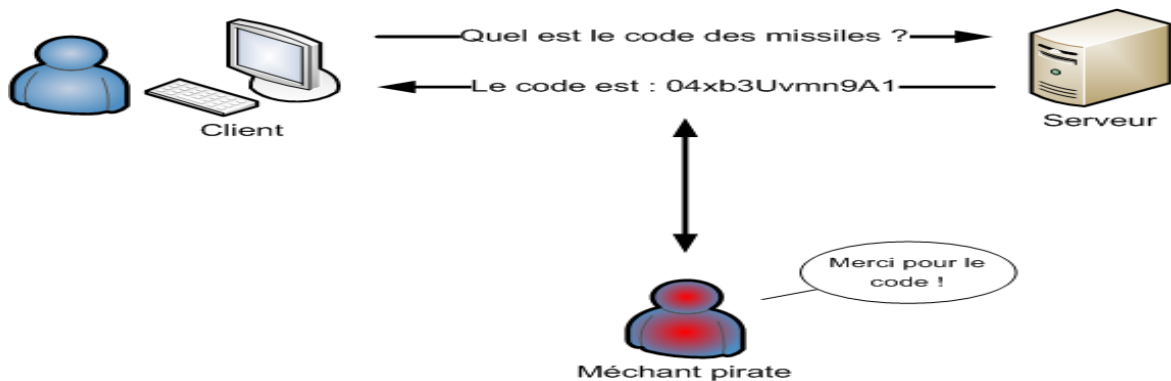
Les données sont transférées en clair sur le réseau. Il n'y a aucun cryptage.

Voici ce qui pourrait se passer. Nous forçons le trait, mais c'est pour vous donner une idée. Imaginez qu'un PC militaire demande à un serveur de l'armée le code de lancement de missiles (nucléaires, soyons fous), comme sur la figure suivante.



Transfert données Telnet fig I.C.6

Après tout, il n'y a rien de choquant. Le message n'est envoyé qu'au client qui l'a demandé. Mais en fait, un pirate aurait la possibilité d'« écouter » ce qui se passe sur le réseau, et donc d'intercepter les données en chemin (fig1.4).

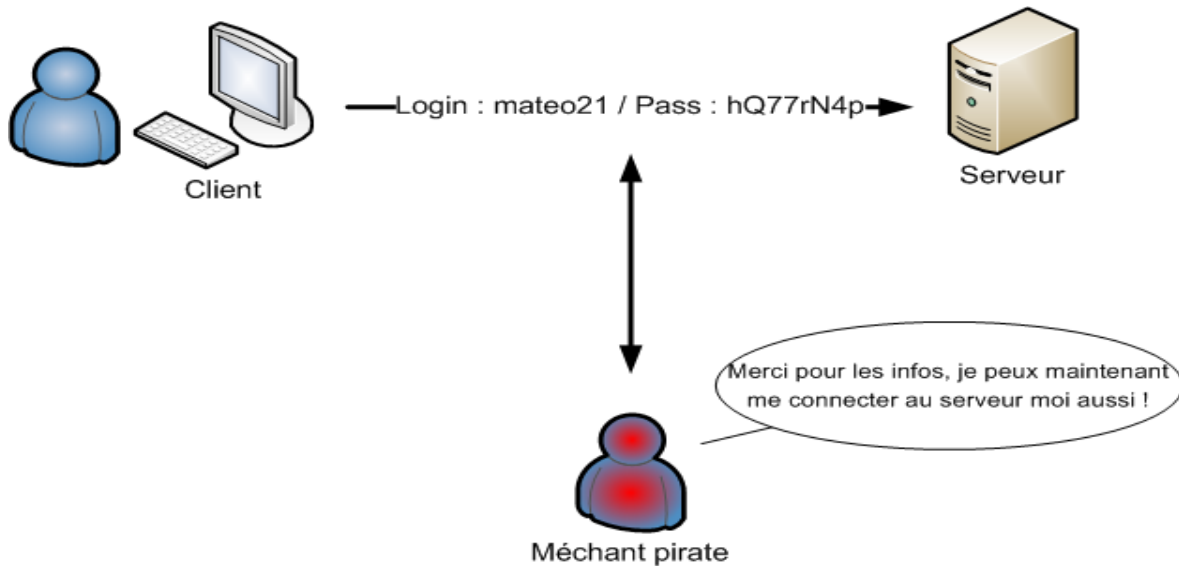


Interception données Telnet fig I.C.7

Ça ne vous dérange pas que l'on vous espionne ? Soit.



Mais quand vous allez vous connecter au serveur, vous allez donner votre login et votre mot de passe. Rien que ça, c'est dangereux (fig1.5). Il ne faut pas que le login et le mot de *pass*e apparaissent en clair sur le réseau !



Interception Login/Pass Telnet fig I.C.8

### **Le protocole SSH : la solution pour sécuriser les données**

Comme on ne peut pas complètement empêcher quelqu'un d'intercepter les données qui transitent sur l'internet, il faut trouver un moyen pour que le client et le serveur communiquent de manière sécurisée. Le cryptage sert précisément à ça : si le pirate récupère le mot de passe crypté, il ne peut rien en faire.

### **D. Conclusion**

Toutes ces notions sont données pour mieux comprendre le rôle d'un administrateur réseau, car comme on peut facilement s'en rendre compte, les outils et les logiciels pour déployer les différents services d'internet sont si nombreux, qu'on s'y perdrait facilement. Le foisonnement est tel, qu'il est parfois très facile de faire des erreurs qui pourraient coûter à l'administrateur réseau très cher. Donc dans ce chapitre, on a fait le panorama non exhaustif de l'existant pour ensuite décider sur quel logiciel va se porter notre choix dans le chapitre suivant

## **II. Chapitre 2 : INSTALLATION ET CONFIGURATION DES DIFFERENTS SERVEURS**

### **1. Introduction :**

Ce chapitre se veut très technique, car il est l'essence même de notre travail, et suite aux recommandations de notre encadreur, on a jugé utile de donner le maximum de détail, car comme nous l'avons déjà dit, ce PFE s'adresse en premier lieu aux étudiants voulant approfondir leurs connaissances sur ce domaine, et donc nous avons fait exprès de détailler toutes les étapes, de telle sorte que quiconque peut reproduire les mêmes installations sans trop de soucis.

### **2. Serveur Web : [2]**

#### **2.1. APACHE :**

APACHE est le serveur web le plus répandu, le plus utilisé, le plus abouti et si simple d'utilisation.

#### **Installation d'APACHE :**

Nous avons téléchargé le programme d'installation d'Apache.

A l'écran 'Server information', nous allons configurer le nom de domaine, le nom du serveur ainsi que l'adresse e-mail de l'administrateur du serveur.

**A - Network Domain :** Nom de domaine du serveur, **Localhost** correspondant à l'hôte local dans le cas présent.

**B - Server Name :** Nom du serveur, ici, nous allons mettre **127.0.0.1** qui correspond à l'ip interne de la machine.

**C - Administrator's Email Address** : l'adresse e-mail de l'administrateur en l'occurrence vous.

**D - For All Users, on Port 80, as Service** : cette option permet d'installer apache pour tout les comptes utilisateurs en utilisant le **port 80** (port utilisé par défaut pour un serveur web) en tant que service (le service est utilisé par Windows XP ou 2000).

**E - only for the Current User, on Port 8080, when started Manually** : cette option permet d'installer apache pour l'utilisateur courant (qui procède à l'installation) en utilisant le **port 8080** et avec un démarrage manuel. Cette dernière est recommandée pour le développement web, et non dans un but de production.

Dans le cas présent, nous procédons à une installation de type « tout utilisateurs, sur le port 80 en tant que service ».

On clique sur **D** ou **E** et on continue l'installation...

Dans le cas présent, on souhaite procéder à l'installation dans un répertoire autre que celui par défaut « **C:\Program Files\Apache Group** ».

On le **Change...** par « **E:\projet** ».

Une fois l'installation terminée à présent, notre serveur est opérationnel, d'ailleurs, si l'installation était standard, les services sont déjà en action.

Note : Si nous souhaitons lancer notre service apache sans pour cela passer par **Apache**

**Service Monitor**. Il nous suffit de faire un **clique droit sur Poste de Travail / Gérer** ensuite, aller dans la rubrique **Services et applications / Services**. Là, on trouve le service

**Apache2** sur lequel on va faire un clique droit, puis **démarrer** ou **arrêter** selon le cas.

Une astuce toute simple est d'ouvrir une invite de commande, pour cela, **Démarrer /**

**Exécuter** puis taper **CMD** (ou cmd, ça n'a pas d'importance). La fenêtre ouverte, on tape

**NET START APACHE2** (pour démarrer le service)

**NET STOP APACHE2** (pour arrêter le service)

Ceci étant nous allons effectuer une vérification pour savoir si notre serveur est bien en cours de fonctionnement.

Pour cela, on ouvre notre navigateur favori (**Firefox** par exemple), puis on tape dans la barre d'adresse : **http://localhost** ou **http://127.0.0.1**

Si lors de l'installation nous avons choisi l'option « **only for the Current User, on Port 8080, when started Manually** », nous aurions du ajouter : **8080** à la fin de l'url comme ceci :  
**http://localhost:8080** ou **http://127.0.0.1:8080**.

Ceci n'étant valide que dans le cas où nous avons entré les mêmes informations indiquées lors de l'installation.

Si nous avons rentré une IP de notre réseau local en lieu et place de **127.0.0.1**. Notre serveur est visible à l'url suivante depuis n'importe quel poste de notre réseau **LAN**.

**http://192.168.0.1** (par exemple).

### Configuration d'APACHE :

La configuration de Apache se fait de manière simple et ce via un seul et unique fichier de configuration. Le fichier de configuration du serveur web se nomme **httpd.conf** (un fichier texte qui sera édité avec le bloc-notes) et est situé dans le sous-répertoire **conf** d'Apache.

Ce fichier contient les principaux éléments pour faire en sorte que votre serveur web tourne sans encombre. Une modification dans ce fichier peut rendre indisponible Apache.

Voyons quelques paramètres paramétrables sans trop de difficulté :

Tout d'abord, une chose bien utile si nous ne souhaitons pas utiliser le répertoire de base de Apache pour nos documents web.

Par défaut, le sous-répertoire qui contient les pages web se nomme **htdocs**, si nous souhaitons modifier cela, repérons le paramètre « **DocumentRoot** » puis modifions comme ceci:

**DocumentRoot** "e:/projet/www"

On aura pris soin de créer le sous-répertoire **www** dans **e:/projet** avant même d'avoir fait la modification dans le fichier de configuration, sinon cela aurait pour effet de générer une erreur lors du lancement d'Apache.

Si pour une raison ou pour un autre, on souhaite modifier l'adresse e-mail de l'administrateur du serveur, on repère le paramètre **ServerAdmin** puis on lui indique en valeur une adresse e-mail (de préférence valide).

**ServerAdmin** phenicien@nomdedomaine.com

Si l'on souhaite indiquer les fichiers qui seront traités comme des fichiers de base du serveur web, c'est-à-dire la page par défaut d'un répertoire web, nous pouvons modifier pour cela le paramètre **DirectoryIndex**.

**DirectoryIndex** index.htm index.html index.php index.php5

Ici, toutes les pages qui se nomment **index.html**, **index.html**, **index.php** ou **index.php5** seront prises en compte par le serveur web comme page par défaut d'un site web.

Pour faire en sorte que le visiteur ait un minimum d'information concernant notre serveur lorsqu'une page d'erreur type 404 s'affiche, nous pouvons modifier la valeur du paramètre

**ServerTokens**.

**ServerTokens** Prod

En donnant la valeur **Prod** cela permet de ne fournir que le nom du serveur, soit dans le cas présent **Apache**, il n'y aura aucune information concernant la version utilisée ni d'autres informations qui pourraient renseigner une personne mal intentionnée.

Par ailleurs, nous veillons à fournir une adresse e-mail qui pourrait permettre au visiteur de nous informer d'un éventuel problème sur le serveur. Pour se faire nous modifions la valeur du paramètre **Server Signature** comme ceci :

**ServerSignature** Email

Ce qui au final lorsqu'un message d'erreur est affiché permet à tout visiteur de pouvoir prévenir l'administrateur du serveur.

Une option qui est très utile est l'utilisation du module status, un module est une fonction qui permet d'ajouter des fonctions à notre serveur web.

Le module status permet d'obtenir des informations en quasi temps réel sur l'état du serveur.

Pour ce faire dans le fichier de configuration, nous recherchons la ligne suivante :

```
# LoadModule status_module modules/mod_status.so
```

Dans le cas présent, la ligne est actuellement en commentaire puisque ayant un # en son début de ligne.

Donc, on dé commente tout d'abord la ligne :

```
LoadModule status_module modules/mod_status.so
```

Puis l'on recherche les quelques lignes ci-dessous (lignes qui dans leur version d'origine sont là aussi commentés #) :

```
<Location /server-status>
```

```
SetHandler server-status
```

```
Order deny,allow
```

```
Deny from all
```

```
Allow from 127.0.0.1
```

```
</Location>
```

Ces quelques lignes permettent de rendre ou non disponible l'état du serveur. Le **Deny from all** permet tout d'abord **interdit l'accès à tout le monde** puis avec l'option **Allow from**

**127.0.0.1** de **l'autoriser uniquement à 127.0.0.1** (la consultation sera donc possible que depuis le serveur et non depuis une machine dans le réseau local par exemple).

Une fois la modification, on enregistre le fichier de configuration puis l'on ouvre son navigateur favori (Opéra par exemple).

Dans la barre d'adresse : **http://localhost/server-status**

Voilà donc en somme un aperçu de la configuration d'un serveur web Apache.

Note : Suite à un problème que nous avons rencontré récemment, nous prendrons soin lors de l'installation de ne pas avoir le logiciel « **skype,msn,...** » En fonctionnement ainsi que tous autres logiciels qui pourraient utilisés le port **80**. Ce qui a pour effet dans le cas présent lorsqu'on veut installer / démarrer le service apache d'afficher un message d'erreur.

## **2.2PHP :**

### **Installation et configuration de PHP :**

Le module PHP est sous forme « .zip », pour l'installer il suffit simplement de l'extraire de l'archive.

L'archive se nomme dans le cas présent : **php-5.1.1-Win32.zip**

Extraire vers : **e:\Projet\php5**

Ceci fait, voyons les deux installations possibles de PHP pour le faire travailler avec Apache :

- Soit installer PHP en tant que **module d'Apache**.
- Soit installer en tant que **programme CGI**

Dans le cas présent, on va procéder à l'installation en tant que module du serveur apache.

Pour cela on va devoir configurer Apache en conséquence.

Ceci étant, on doit éditer le fichier de configuration d'Apache qui se situe dans

**e:\projet\apache2\conf\** et qui se nomme **httpd.conf**.

A celui-ci, on doit lui ajouter les lignes suivantes :

**LoadModule php5\_module "e:/projet/php5/php5apache2.dll"**

**AddType application/x-httpd-php .php**

La première ligne (**LoadModule php5\_module "e:/projet/php5/php5apache2.dll"**) s'ajoute à la suite des autres **LoadModule** qu'on trouvera dans le fichier.

Pour ce qui concerne la deuxième ligne (**AddType application/x-httpd-php .php**), elle s'ajoute à la suite des autres **AddType**.

Ceci fait, on enregistre le fichier de configuration ainsi modifié.

Nous allons à présent passer à la configuration du **php.ini** qui à l'origine se trouve dans le répertoire d'installation sous le nom de **php.ini-dist**, première chose, en faire une copie et renommez la dite copie en tant que **php.ini**.

On édite le fichier **php.ini** pour modifier les lignes suivantes :

```
extension_dir = "./" en extension_dir = "e:\projet\php5\ext"  
;upload_tmp_dir = en upload_tmp_dir = e:\projet\php5\uploadtemp  
;session.save_path = "/tmp" en ;session.save_path = "e:\projet\php5\sessionsave"
```

A noter que les deux dernières lignes ne sont nullement obligatoires, si l'on ne pense pas utiliser les fonctions qui font appel à ces valeurs. Dans un autre cas, on aura pris soin de créer le sous-répertoire **uploadtemp** et **sessionsave**.

Autre point, si on compte utiliser le serveur de données MySQL, il est nécessaire de décommenter la ligne suivante :

```
; extension=php_mysql.dll en extension=php_mysql.dll
```

Par ailleurs, on a pris soin de copier le fichier **libmysql.dll** dans le répertoire système soit **c:\windows\system32** ou **c:\winnt\system32** selon le système d'exploitation utilisé.

Note : Par défaut, **MySQL n'est plus activé dans PHP5** ce qui explique la manipulation vue ci-dessus. Si vous obtenez un message similaire à celui-ci :

```
"Unable to load dynamic library './php_mysql.dll'"
```

C'est tout simplement parce que le fichier **libmysql.dll** n'a pu être trouvé par le système.

Ceci étant nous pouvons à présent, faire un premier test pour savoir si notre serveur Apache à bien pris en compte le support PHP. Pour ce faire, nous allons créer un fichier **info.php** que nous placerons à la racine de notre serveur web.

Dans notre cas, notre racine se situe dans le répertoire suivant **e:\projet\www** (répertoire que l'on a spécifié lors de la configuration d'Apache).

Le fichier **info.php** contiendra la ligne suivante :

```
<? phpinfo(); ?>
```



La ligne ajoutée et le fichier modifié enregistré, on ouvre notre navigateur web favori et rendez-vous à l'URL suivante :

**http://127.0.0.1/info.php** ou **http://localhost/info.php**

Voilà, PHP est à présent installé sur notre serveur web.

### **2.3. MySQL :**

#### **Installation de MySQL :**

On choisit le type d'installation, après on peut changer le répertoire où sera installé notre serveur MySQL.

Dans le cas présent notre serveur sera dans **e:\Projet\MySQL Server 5.0\**

À l'écran MySQL Sign-Up, vous demande si vous souhaitez créer un compte ou si vous en disposez un sur le site officiel mysql.com. Dans le cas présent, je ne souhaite ni créer un compte ni m'authentifier sur le site. Donc, je coche **Skip Sign-Up**.

Dans le cas présent, je souhaite procéder à la configuration de mon serveur.

Dans le dernier écran je coche l'option **Configure the MySQL Server Now**.

#### **Configuration de MySQL :**

Nous avons choisi précédemment l'option **Configure the MySQL Server now**, ceci va nous permettre de configurer le serveur MySQL dès à présent.

Dans le cas présent, on choisit une configuration de type Standard, donc on coche l'option

#### **Standard Configuration.**

À l'écran « **Server instance configuration** », nous pouvons choisir le nom du service (Service Name) qui va être installé, dans le cas présent, MySQL5. Par ailleurs, on peut ajouter une variable **Path** dans la configuration de Windows ce qui aura pour effet de rendre disponibles les commandes MySQL depuis n'importe quel répertoire dans une invite de commandes.

Nous devons déterminer le mot de passe du compte administrateur (root) du serveur.

Par ailleurs, nous pouvons procéder à la création d'un utilisateur anonyme sur le serveur.

Ceci étant, dans le cas présent, nous définissons le mot de passe root mais nous ne créons pas d'utilisateur anonyme qui comme on peut le voir n'est pas des plus prudent (insecure) pour le serveur.

NOTE : un conseil : mettre un mot de passe alphanumérique pour assurer un bon niveau de sécurité sur votre serveur MySQL.

Nos options de configuration définies, MySQL va procéder à son exécution.

On clique sur le bouton **Execute** pour lancer la configuration de ce dernier.

La configuration se passe sans soucis dans le cas présent

L'installation et la configuration du serveur sont à présent terminées.

Notre serveur MySQL ayant été installé en tant que service ce dernier doit être dès cet instant en action.

Si nous souhaitons vérifier que notre serveur MySQL tourne bien, nous pouvons nous connecter au Shell de ce dernier. Pour ce faire on clique sur **MySQL Command Line**

**Client** dans le menu Démarrer / Programmes.

Là, l'invite de commandes s'affiche

On entre le mot de passe du compte **root** qu'on a configuré tout à l'heure.

Nous voilà connecté sur le Shell de notre serveur MySQL.

Ce Shell permet de pouvoir surveiller (monitoré) le serveur MySQL via quelques commandes, la liste des commandes est disponible en tapant **help**.

**mysql> help**

L'écran d'aide qui s'affiche.

Exemple : **status**, qui va me fournir quelques informations concernant mon serveur mysql.

En outre, on obtient l'uptime du serveur, le port utilisé par le serveur de données.

Si nous souhaitons maintenant pouvoir administrer notre serveur MySQL, nous pouvons le faire avec le programme mysqladmin.exe qui se situe dans le sous-répertoire \bin de notre installation.

Ce programme permettra en outre, la création/suppression de base, en bref toutes les commandes dont nous avons besoin pour l'administration du serveur.

#### **2.4 .PHP MyAdmin :**

Une dernière interface graphique que nous affectionnons est **phpMyAdmin**, qui est une interface web donc dans le cas présent, il faut qu'on ait un serveur web (par exemple, Apache) configuré pour PHP. Une fois l'archive récupérée, on la décompresse, pour le mieux on la décompresse dans un des sous-répertoires du serveur web. Et on renomme le répertoire en **phpmyadmin**.

Exemple : **c:\monserveur\phpmyadmin**

Ceci fait, nous devons éditer le fichier de configuration pour ce faire, on fait une copie du fichier **config.default.php** que l'on nommera **config.inc.php**. On édite ce fichier avec le bloc-notes pour modifier quelques informations.

Voici les quelques éléments à modifier :

```
$cfg['PmaAbsoluteUri'] = ''; en $cfg['PmaAbsoluteUri'] = 'http://127.0.0.1/mysql';
```

```
$cfg['blowfish_secret'] = ''; en $cfg['blowfish_secret'] = 'une phrase par exemple';
```

```
$cfg['Servers'][$i]['auth_type'] = 'config'; en $cfg['Servers'][$i]['auth_type'] = 'cookie';
```

```
$cfg['Servers'][$i]['user'] = 'root'; en $cfg['Servers'][$i]['user'] = '';
```

Ceci fait on sauvegarde les modifications. Par ailleurs, dans le fichier de configuration, il faudra ajouter les lignes suivantes (dans la partie Alias de Apache) :

```
Alias /mysql/ "c:/monserveur/phpmyadmin/"
```

```
<Directory "c:/monserveur/phpmyadmin">
```

```
Options Indexes MultiViews
```

```
AllowOverride None
```

Order allow,deny

Allow from all

</Directory>

On sauvegarde le fichier de configuration « **httpd.conf** » après avoir ajouté les quelques lignes (ci-dessus). Une fois ceci fait, il faudra redémarrer le serveur Apache.

Dans notre navigateur favori (firefox par exemple) on tape dans la barre d'adresse :

**http://127.0.0.1/mysql** ou **http://localhost/mysql**

Entrer le nom d'utilisateur (*root* par exemple) et le mot de passe...

On aura alors accès à l'interface d'administration du serveur de données.

Cette interface web, permet de réaliser toutes les opérations d'administration/maintenance sur le serveur de données, entre autres, création/suppression d'utilisateurs, création/suppression de bases, de tables. De quoi ravir toutes personnes souhaitant administrer au mieux son serveur MySQL.

### **3. Serveur FTP: [3]**

#### **3.1 Introduction à FileZilla Server :**

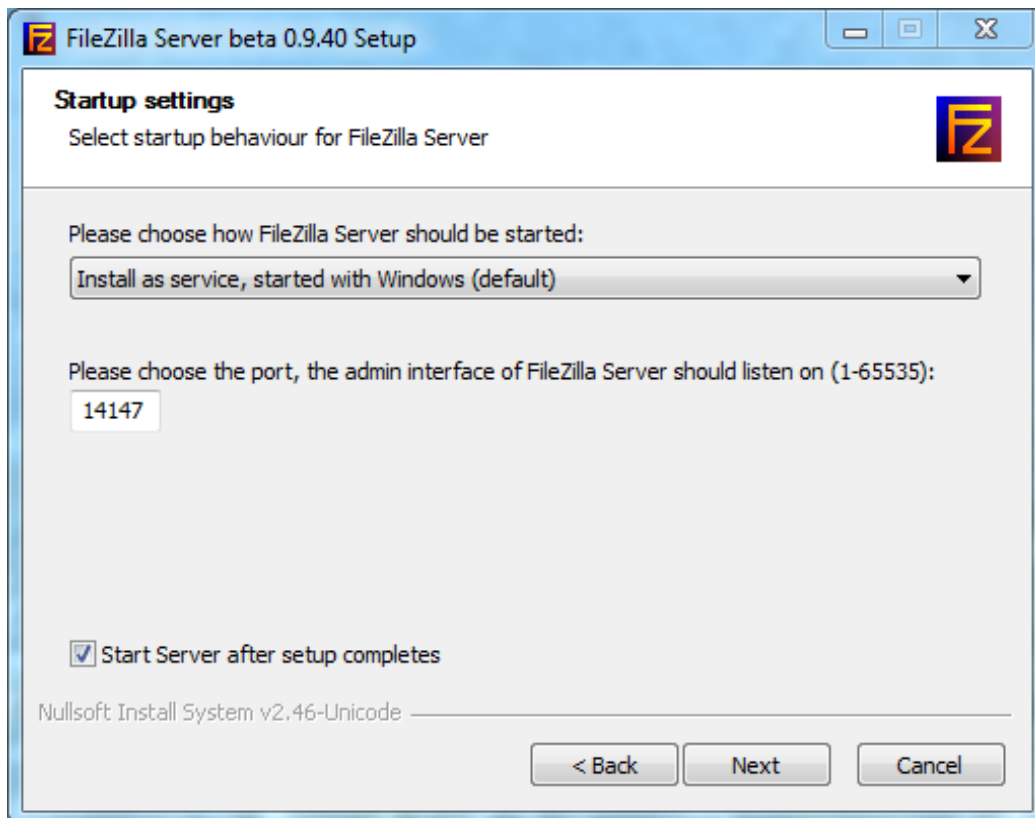
**FileZilla server** est un serveur FTP rapide avec une interface facile à utiliser. Il permettra de partager des fichiers que ce soit sur un réseau local ou sur Internet.

Ainsi, les utilisateurs (côté client) pourront télécharger des fichiers à partir du serveur. Celui-ci peut être démarré avec Windows en tant que service ou bien comme un périphérique.

#### **3.2. Installation de FileZilla Server :**

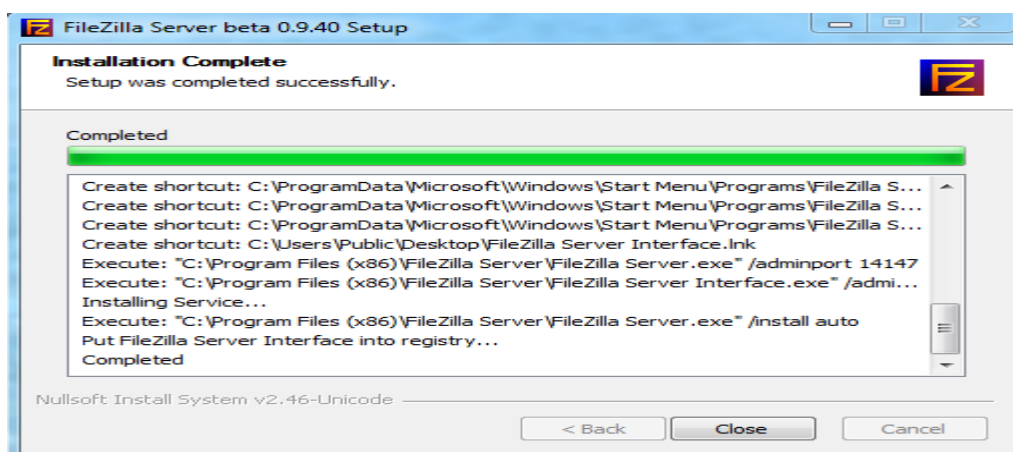
Après avoir téléchargé le fichier d'installation et lancé l'exécutable,

On installera Filezilla server, en prenant compte de quelques petits détails :



Ici, on nous demande de choisir la façon dont notre serveur FTP devra démarrer :

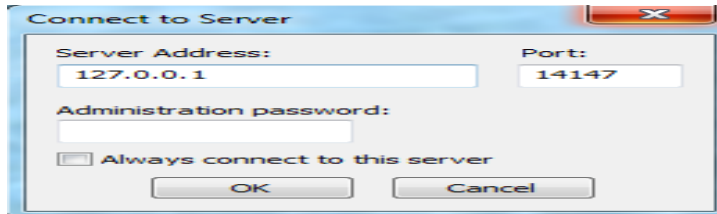
- Par défaut il démarrera en tant que service et en même temps que **Windows**
  - La deuxième option permettra de démarrer manuellement votre serveur
  - La dernière option (non recommandée) permet de ne pas installer le service
- Prenez la première option, et continuer l'installation normalement



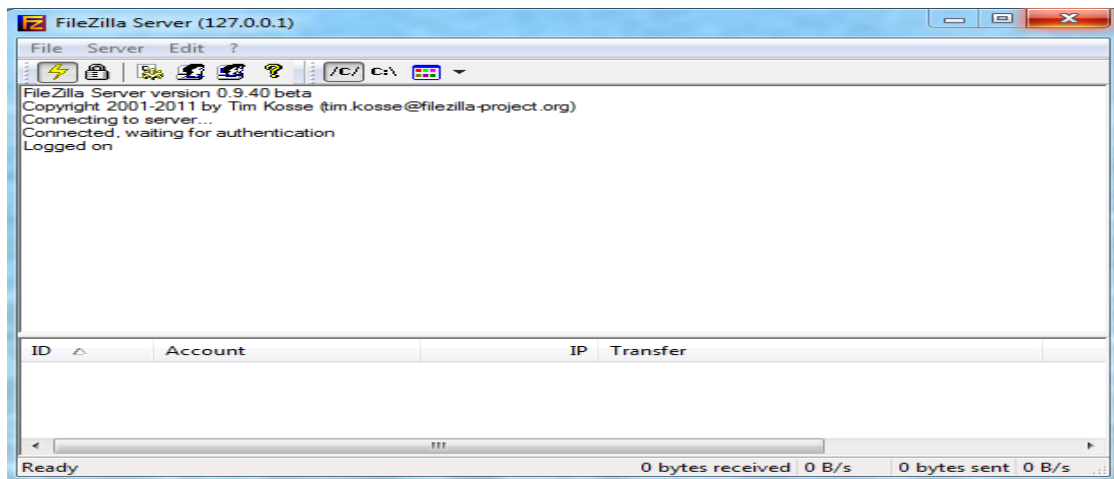
L'installation est terminée, cliquer sur "**Close**".

### 3.4. Configuration du Serveur FTP :

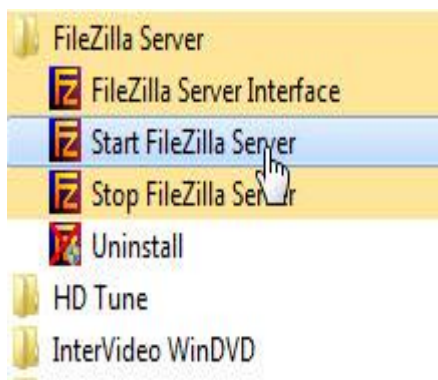
Normalement, si l'option de démarrage du serveur après installation a été définie, le serveur doit être effectif actuellement. On pourra donc accéder directement à l'interface d'administration du serveur FTP juste en cliquant sur **OK**.



Vous aurez ceci :



Si sa ne marche pas, il faut lancer le serveur manuellement en cliquant sur "**Start Filezilla Server**" dans le menu démarrer comme ceci :

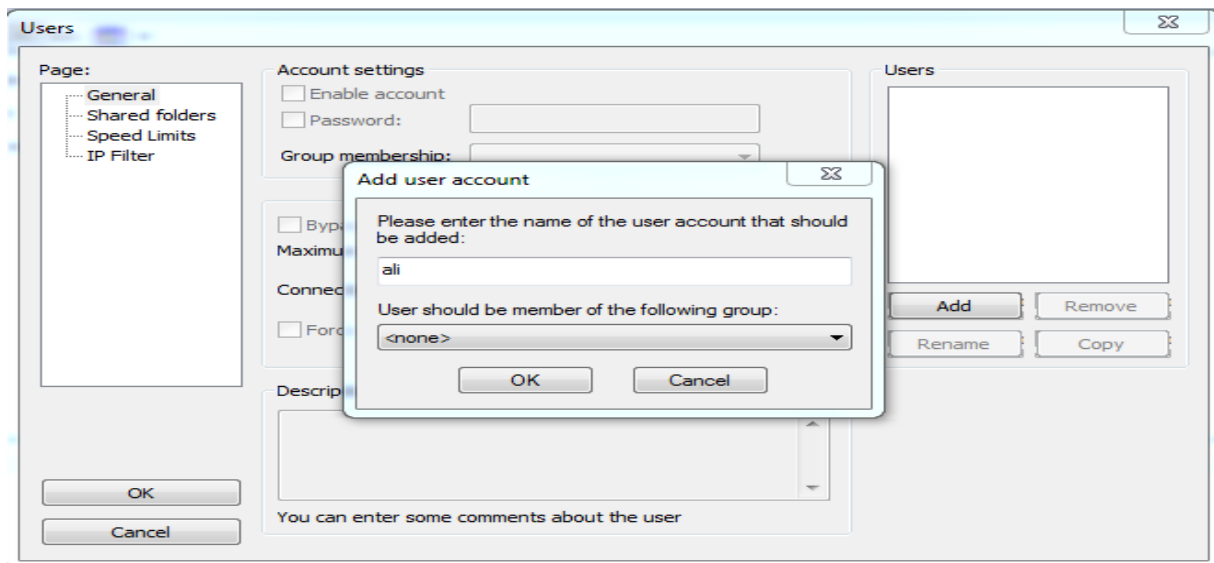


Puis retenter la connexion à nouveau.

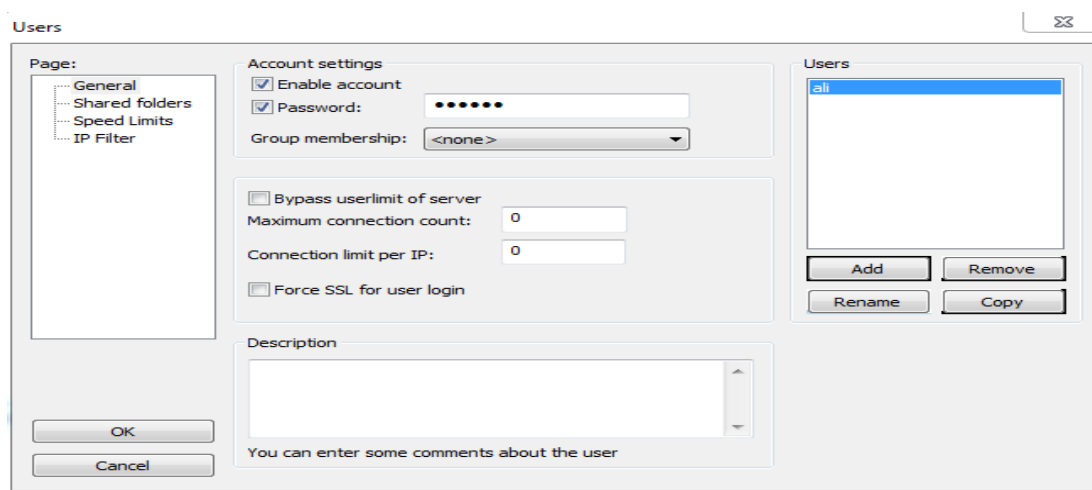
Voyons maintenant comment ajouter un utilisateur afin de se connecter au serveur par la suite.

Pour cela, cliquer sur le menu "**Edit**" puis "**Users**".

Cliquer sur le bouton "**Add**", puis dans la petite fenêtre qu'on apercevra en premier plan, indiquer le nom souhaité de l'utilisateur puis cliquez sur **OK**.

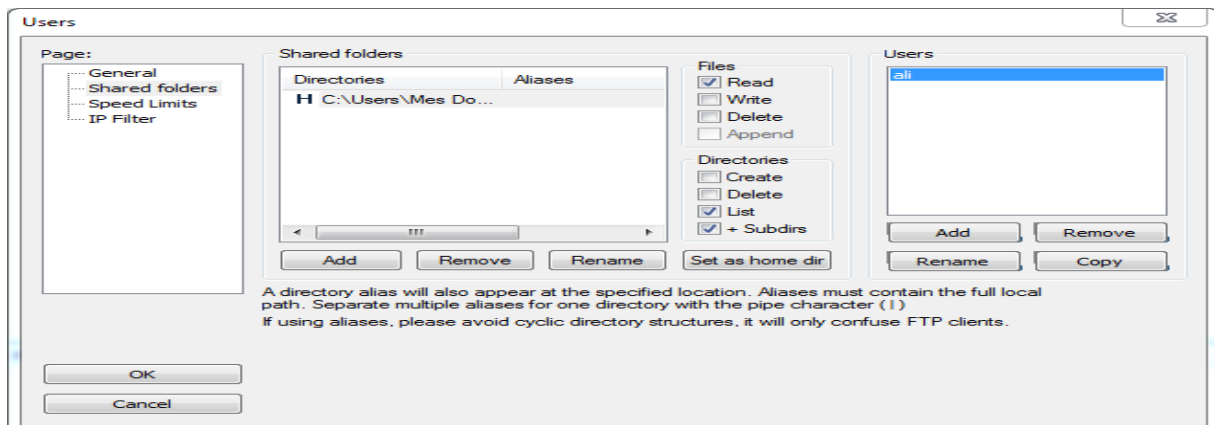


Maintenant, attribuer un mot de passe en sélectionnant la case "**Password**"



A présent, nous devons définir un ou plusieurs dossiers auxquels pourra accéder l'utilisateur lorsqu'il se connectera en tant que client.

Pour cela, cliquer sur "**Shared folders**" puis sur le bouton "**Add**" :

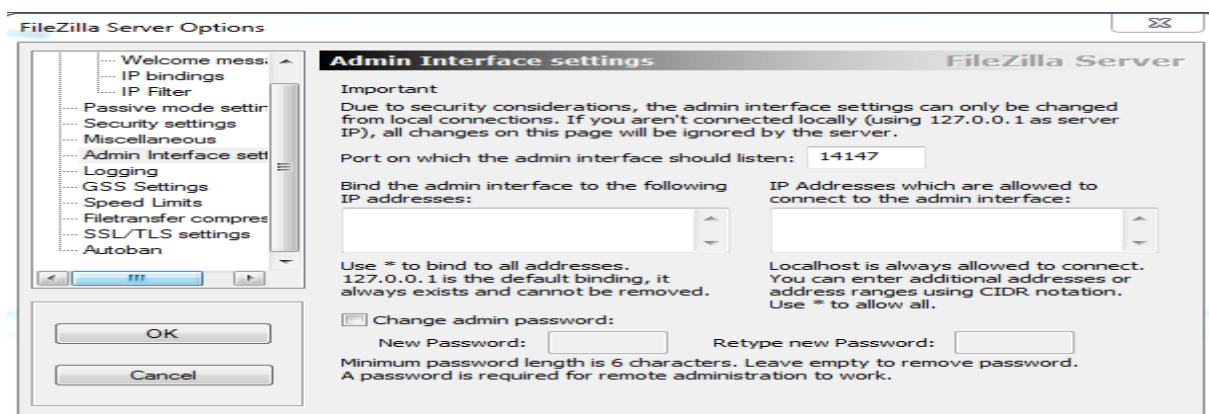


Ensuite, il s'agit de régler les droits d'utilisateurs sur ce dossier, pour que l'utilisateur puisse seulement télécharger des fichiers, alors conserver les droits par défaut comme ci-dessus, mais si on souhaite qu'il puisse également envoyer des fichiers sur le serveur, cocher "**Write**" (écriture) et "**Append**" (reprendre un envoi de fichier).

Directories concerne la création (**create**), suppression (**delete**), lister des répertoires (**list**) et l'accès aux sous répertoires (+ **Subdirs**).

Voilà pour la configuration de base de notre serveur, il faut noter que pour plus de sécurité, on peut ajouter un mot de passe pour sécurisé l'interface d'administration. Pour cela, dans les paramètres,

Cliquer sur **Admin Interface Setting** et indiquer le mot de passe souhaité comme ci-dessous :

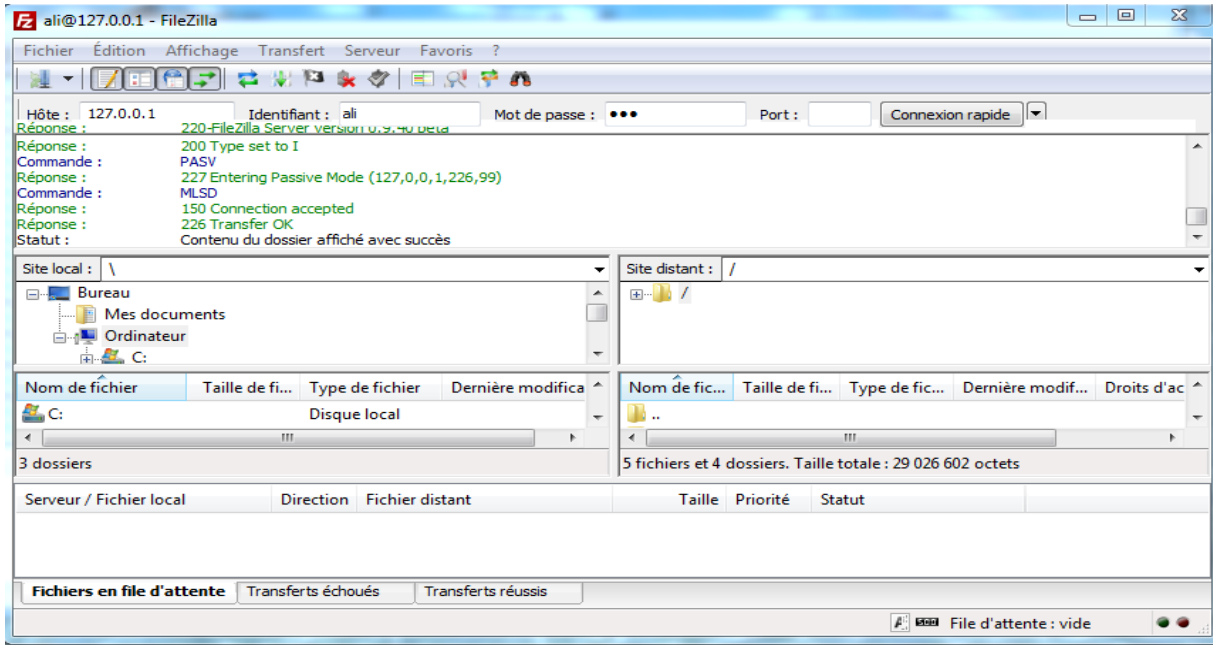




### 3.5. Connexion au serveur à partir d'un client FTP

Pour se connecter au serveur avec un nom d'utilisateur attribué au préalable, on devra être en possession de **Filezilla Client**.

En tant qu'administrateur du serveur, voici la manière dont la connexion se fait :



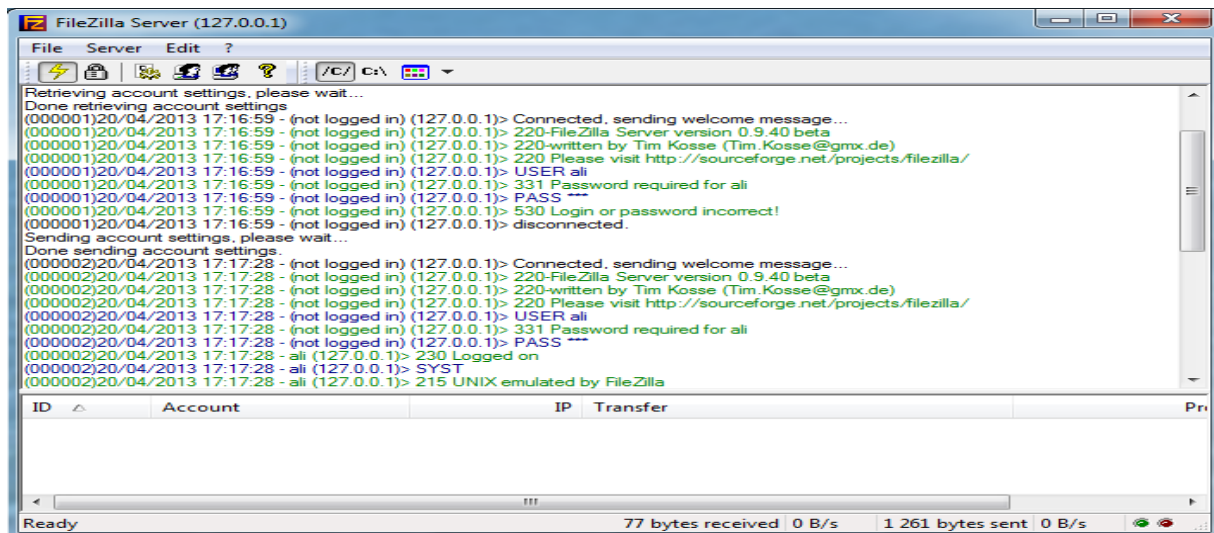
**Hôte** : local host

**Identifiant** : nom d'utilisateur ajouté sur le serveur

**Mot de passe** : \*\*\*\*\*

NB : Vérifiez bien dans ce cas que les ports nécessaires sont ouverts (21 par défaut) sur votre routeur et votre firewall car sans cela personne de l'extérieur ne pourra accéder au serveur.

Enfin, sur la fenêtre principale de l'interface d'administration, on pourra visualiser qui est connecté sur notre serveur et d'autres détails :



### 3.6. Conclusion :

La simplicité de configuration, la grande flexibilité et la robustesse de ses connexions font de Filezilla le serveur FTP idéal pour une utilisation personnalisée de qualité.

## 4. Serveur Telnet :

### 4.1. Activation de Telnet sous Windows Vista/7 : [4]

Microsoft a mis un point d'honneur à améliorer la sécurité de ses systèmes d'exploitation. Cela a parfois pour conséquence de provoquer des situations surprenantes.

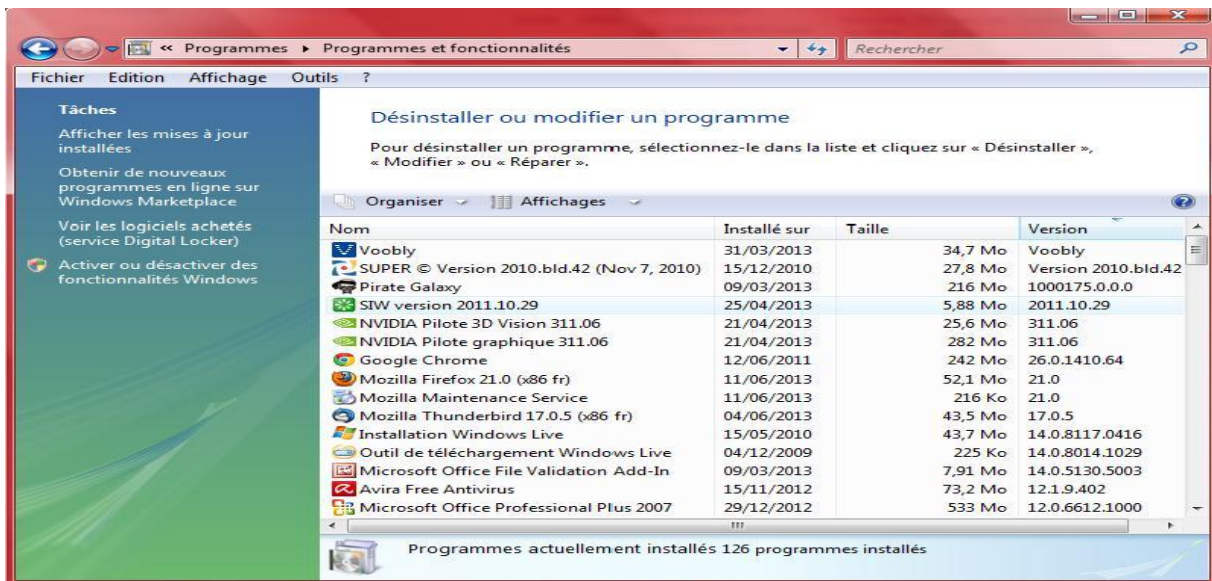
Par exemple Telnet n'est pas installé par défaut et lorsque vous essayez d'utiliser la commande Telnet sur l'invite de commande Dos, vous obtenez ce message (voir image ci-dessous).

```
'telnet' n'est pas reconnu en tant que commande interne
ou externe, un programme exécutable ou un fichier de commandes.
```

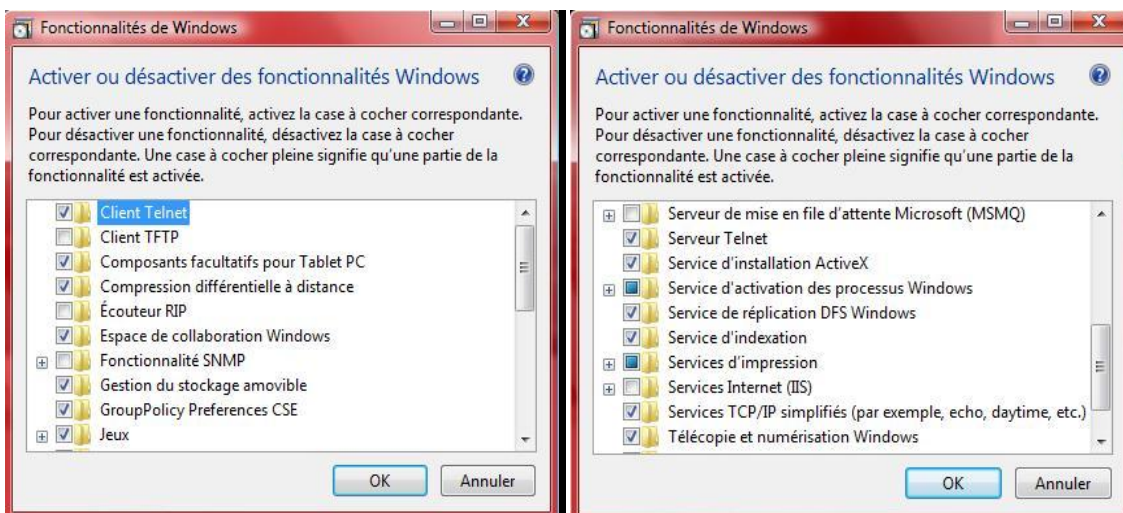
Si nous désirons pouvoir bénéficier de cette fonction, suivons la procédure ci-dessous.

Cliquer sur **Démarrer** et ensuite sur **Panneau de configuration**

Dans la rubrique **Programmes**, cliquer sur **Désinstaller un programme**



Cliquer sur le menu de gauche "Activer ou désactiver des fonctionnalités Windows". Après quelques secondes, la fenêtre ci-dessous s'affiche.



Cocher la case du **Client Telnet** et celle du **Serveur Telnet** et pour finir cliquer sur **OK**

Le client Telnet permet de se connecter à des ordinateurs à distance à l'aide du protocole Telnet, à condition qu'ils soient reliés par un réseau (filaire ou sans fil), et permet aussi aux ordinateurs distants de se connecter à notre ordinateur.

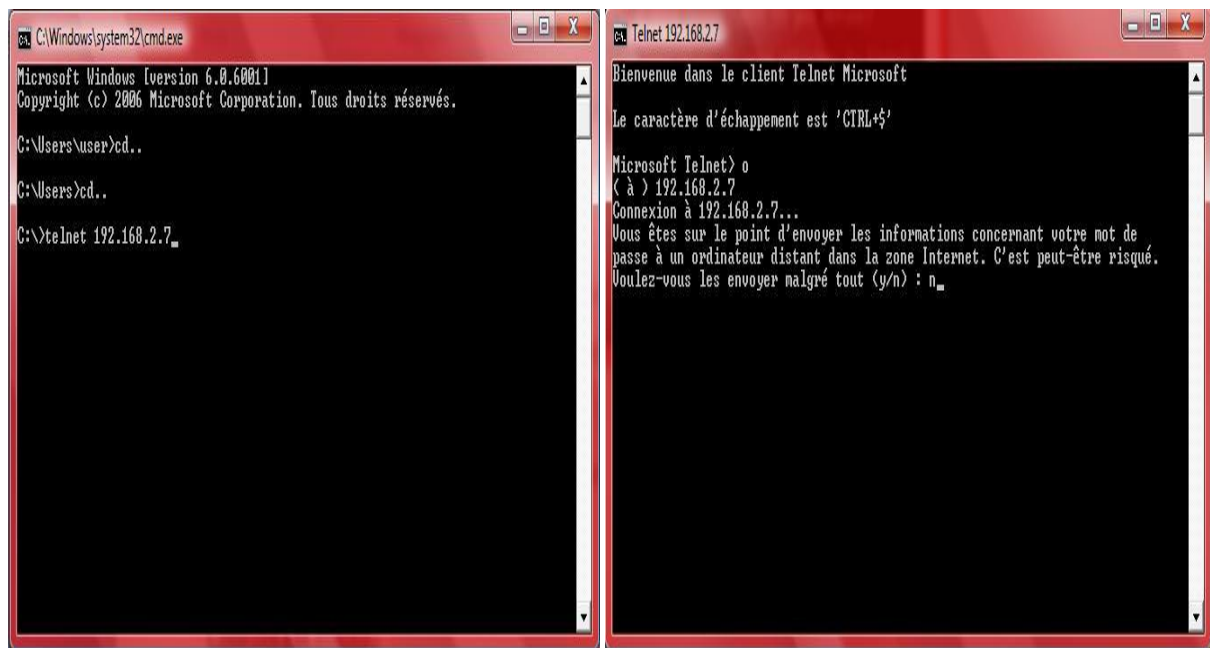
Voilà désormais Telnet est accessible

## 4.2. Commandes Telnet:

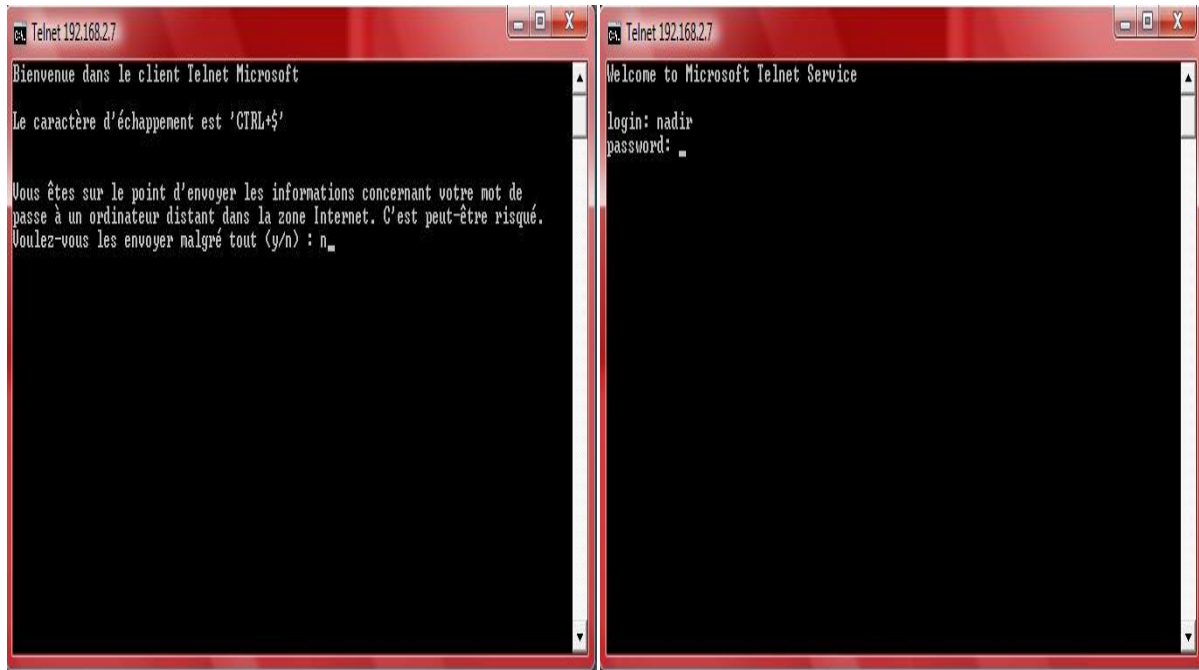
Les commandes peuvent être abrégées. Les commandes prises en charge sont :

```
c - close          ferme la connexion en cours
d - display       affiche les paramètres d'opération
o - open NomHôte [Port] Se connecte à l'hôte nommé (port 23 par défaut)
q - quit         quitte telnet
set - set        définit les options ('set ?' pour afficher la liste)
sen - send       envoie les chaînes au serveur
st - status      affiche les informations d'état
u - unset        annule les options ('unset ?' pour afficher la liste)
?/h - help      affiche des informations d'aide
```

## 4.3. Connexion à un terminal :



Deux manière aboutissent à un même résultat, le 1<sup>er</sup> cas, en utilisant uniquement l'invite de commandes Dos, le 2<sup>ème</sup> cas, en utilisant le client Telnet directement. On entre l'adresse IP ou bien le nom de l'ordinateur distant auquel on veut se connecter.

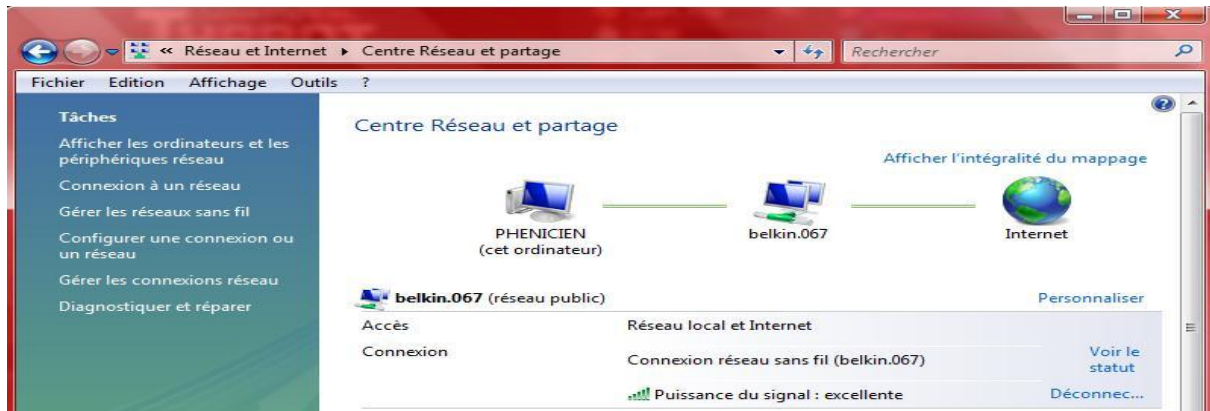


Maintenant nous pouvons faire toutes sortes de commandes sur l'ordinateur distant.

## 5. Serveur DNS :

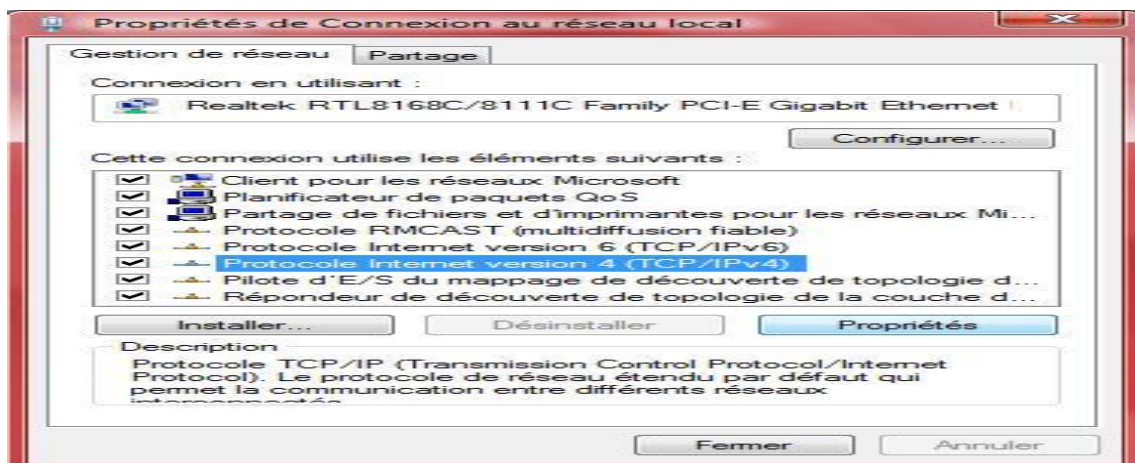
### 5.1. Changer ses serveurs DNS sous Windows :

Tout d'abord, ouvrir le Centre réseau et partage dans le menu du Panneau de configuration. Ensuite, cliquer sur "Gérer les connexions réseau"

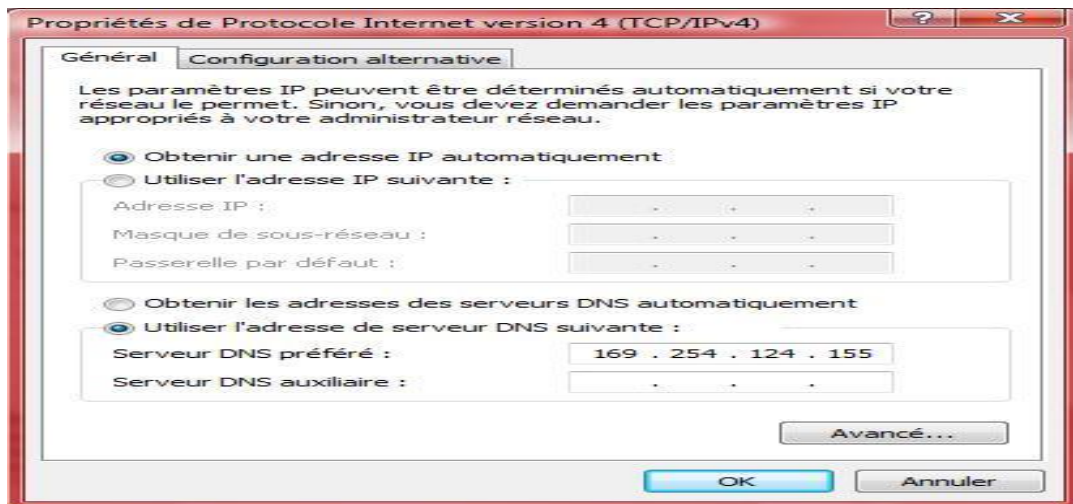


Si notre réseau est configuré pour une connexion sans fil, on modifie les paramètres de la carte sans fil, sinon c'est les paramètres de la carte réseaux filaire qu'on modifie. Faites un clic-droit => Propriétés sur la carte en question.

Choisir la ligne "Protocole Internet version 4 (TCP/IPv4)", et cliquer sur Propriétés



Dans la nouvelle fenêtre qui s'ouvre, toutes les cases sont sûrement sur "Attribuer une adresse IP automatiquement" et "Attribuer les serveurs DNS automatiquement". Cliquer sur "Utiliser l'adresse de serveur DNS suivante" pour pouvoir choisir les DNS qui vous conviennent. Ainsi, les serveurs DNS utilisés ne seront plus ceux distribués par l'ordinateur ou le modem, qu'on ne contrôle pas, mais des serveurs définis par nous. Il faut utiliser un serveur primaire, et un secondaire appelés : "Serveur préféré" et "Serveur auxiliaire".



## 5.2. Pourquoi un DNS primaire et un DNS secondaire ?

Chaque domaine possède un serveur de noms de domaines, appelé DNS primaire, ainsi qu'un serveur de noms secondaire (DNS secondaire ou auxiliaire), permettant de prendre le relais du serveur de noms primaire en cas d'indisponibilité ou de panne de celui-ci.

Sinon on peut utiliser un petit logiciel qui va gérer les changements de serveurs DNS en quelques cliques seulement :

## 5.3. DNS Jumper :

DNS Jumper est un logiciel simple à utiliser, il ne fonctionne malheureusement que sous Windows. Sans nécessiter d'installation, le logiciel est utilisable en français dans une interface sans trop de termes barbares.



#### **5.4. Présentation de Simple DNS Plus :**

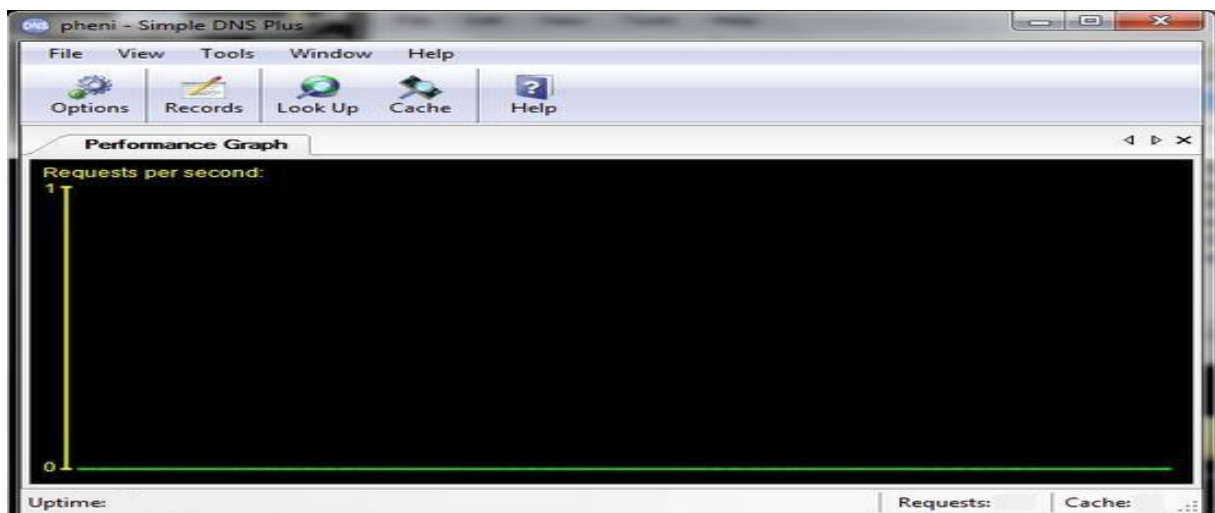
Les serveurs DNS ont été créés pour permettre l'utilisation d'adresses de nom. Ils sont chargés de traduire les noms en adresses lisibles à la machine qui sont nécessaires afin de localiser sur Internet le serveur Web que vous avez demandé.

Simple DNS Plus fournira un puissant serveur DNS personnel. Simple DNS Plus est un outil puissant, pratique et fiable qui peut aider à organiser efficacement des noms de domaine, il apporte un avantage grâce à son interface conviviale et simple d'utilisation.

#### **5.5. Utilisation de Simple DNS Plus : [5]**

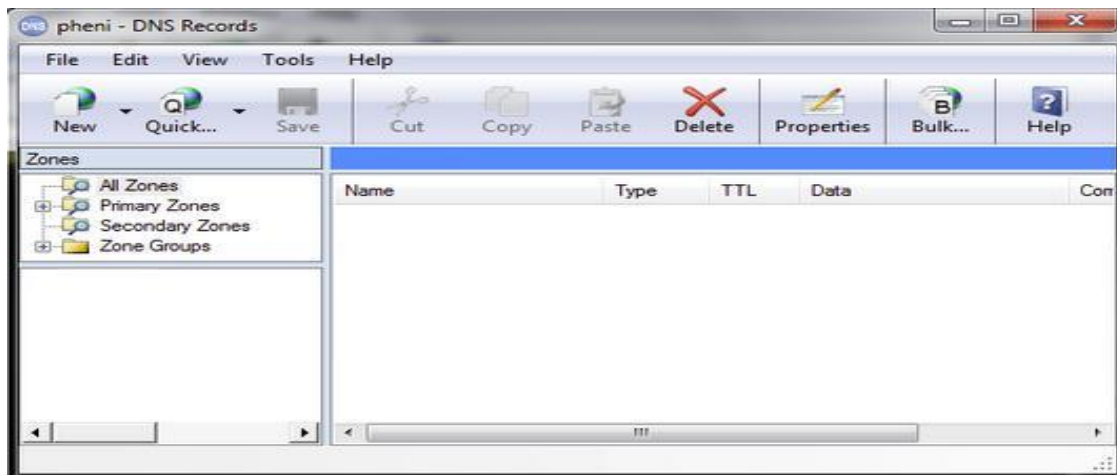
L'installation du logiciel se fait suivant les démarches habituelles comme pour tout autre logiciel.

Une fois l'installation du logiciel et le lancement de ce dernier on obtient :

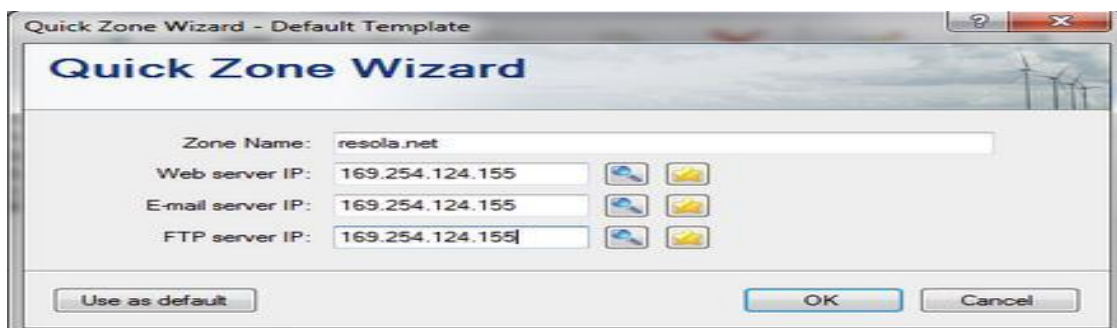


Pour configurer un nom de domaine on a suivi les étapes suivantes :  
Cliquer sur « Records » on aura :



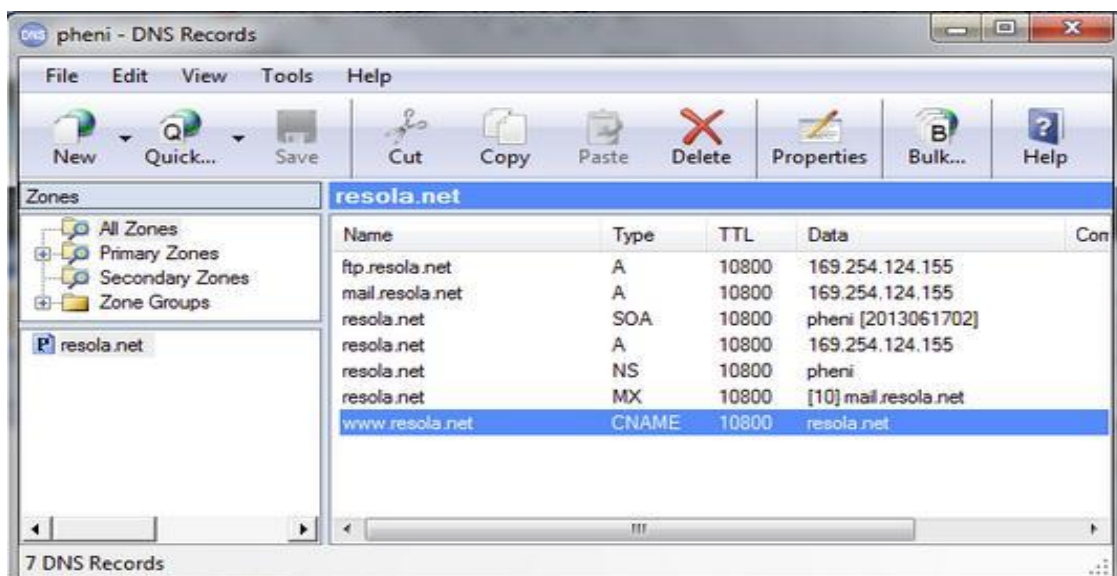


Cliquer sur « Quick » et on aura une petite fenêtre pour faire la configuration :

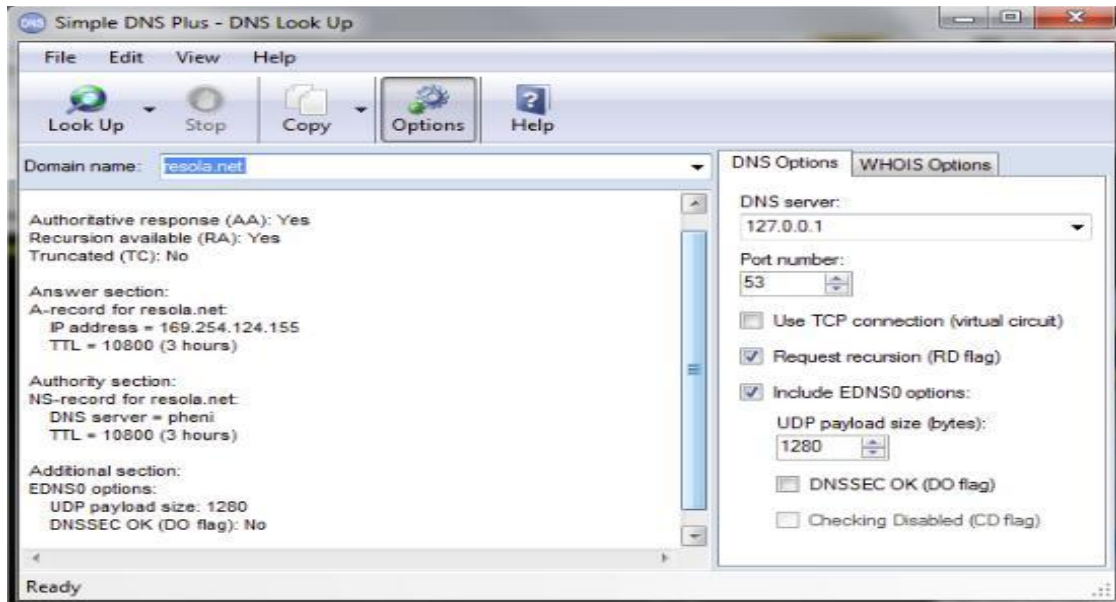


Dans notre exemple on a pris le nom de domaine « resola.net ».

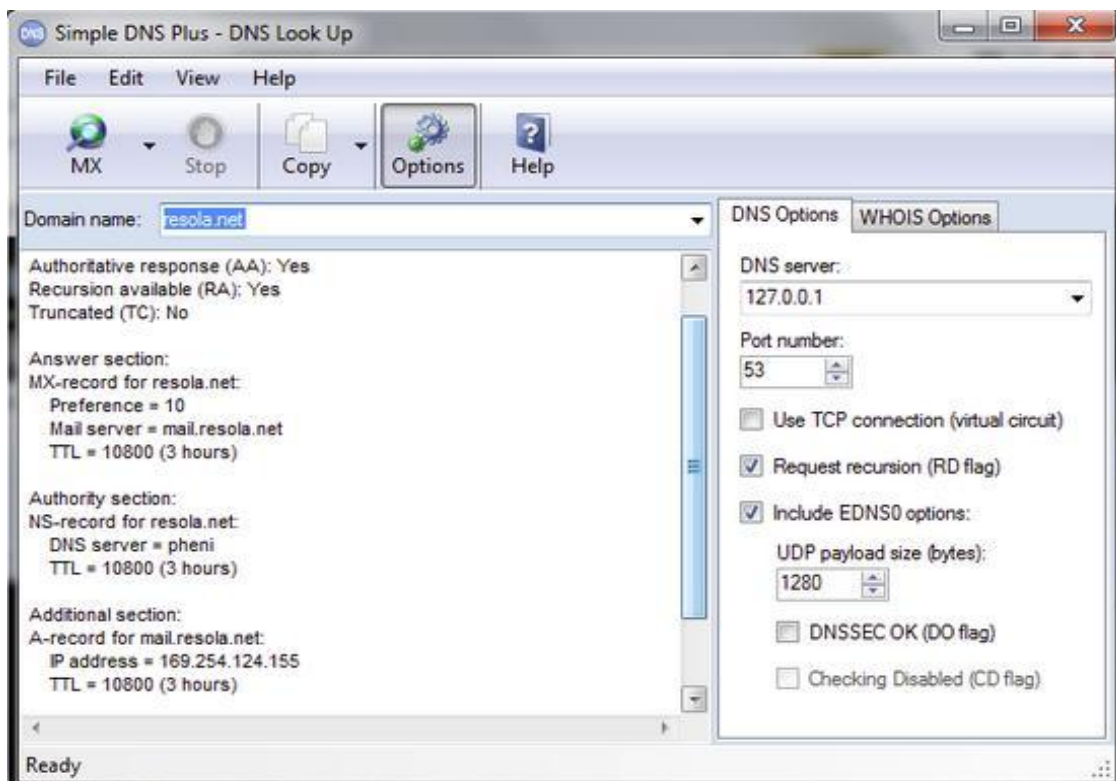
Si le réseau comporte un serveur web, un serveur ftp et un serveur de messagerie sur le même ordinateur (dit serveur), reprendre les paramètres comme montré ci-dessus. Si on ne veut utiliser qu'un serveur Web par exemple, il faut laisser les autres champs vides. Pour finir cliquer sur OK, on aura alors configuré notre nom de domaine :



Pour tester si la configuration a été bien prise en compte, quitter la fenêtre « Records » et cliquer sur « Lookup », en entrant le nom de domaine configuré, on aura une confirmation pareille à ça :



Et pour confirmer le serveur de messagerie aussi, il suffit de remplacer le paramètre « Look Up » en haut à gauche par « MX-record » et vous aurez :



La configuration de Simple DNS Plus est terminée.

Un petit détail avant d'en finir, il faut modifier le fichier « hosts » de Windows qui se trouve à l'emplacement : C:\Windows\System32\drivers\etc

Ouvrir avec un éditeur de texte en mode administrateur, ajouter les lignes suivantes :

- adresse IP      nom de domaine
- adresse IP      nom de serveur de messagerie

Dans notre exemple ça fera :

- 169.254.124.155      resola.net
- 169.254.124.155      mail.resola.net

Enregistrer et fermer le fichier.

Si tout va pour le mieux, en allant sur le navigateur web, en tapant dans la barre d'adresses le nom de domaine qu'on a configuré, on découvrira que c'est juste notre serveur web qui a pris un nom de domaine tout simplement.

## **5.6. Conclusion**

Le système de gestion de noms Dns est efficace et unificateur. De plus, comme tous les systèmes cruciaux d'Internet, il y a au moins une implémentation efficace et gratuite. C'est portable et très complet.

## **6. Serveur de messagerie électronique :**

### **6.1. Hmail Server :** [6]

hMailServer est un serveur de courrier électronique gratuit pour Microsoft Windows. Il est utilisé par les fournisseurs de services Internet, les entreprises, les gouvernements, les écoles et les amateurs de toutes les régions du monde.

Il prend en charge les protocoles de messagerie communs (IMAP, SMTP et POP3) et peut être facilement intégrée avec de nombreux systèmes de messagerie Web existants.

Son utilité est qu'il permet de gérer les mails d'un **Nom de domaine** enregistré, comme il supporte également les domaines virtuels en local.

hMailServer permet de se substituer, à un serveur de mail attaché à un nom de domaine enregistré et déposé sur la toile qu'est Internet pour un moindre coût.

Il intègre aussi un filtre anti-spams et peut être associé à un anti-virus résident dans le pc, les listes noires...

Il travaille avec une base de données (MySQL ou Microsoft SQL), pour les mails ainsi que certaines données liées à sa configuration.

Néanmoins, il n'est pas nécessaire d'avoir une base de données déjà installée sur le pc gérant hMailServer, puisqu'il intègre d'ores et déjà une base de données de type MySQL indépendante.

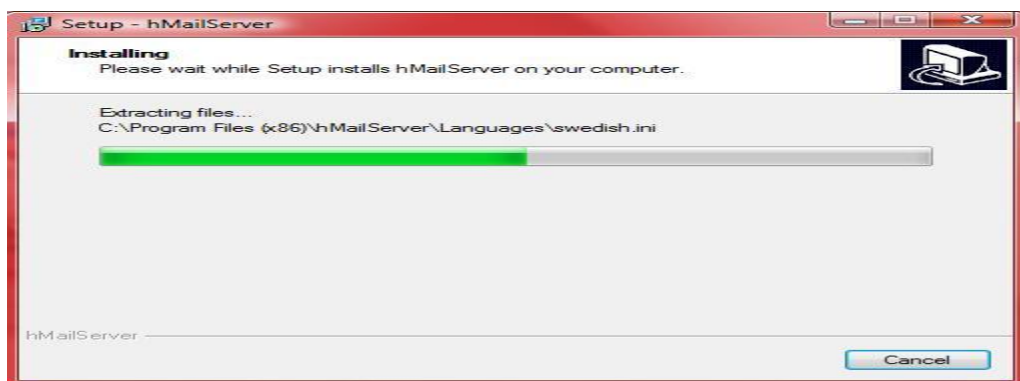
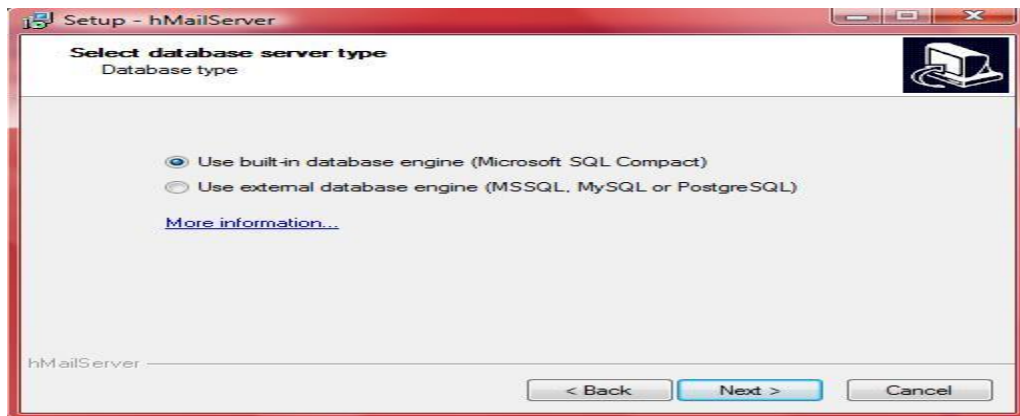
Le fonctionnement du serveur de mail est indépendant de tout autre serveur Web ou Ftp. Par contre pour une utilisation de gestion de l'administration du serveur ou des comptes utilisateurs, à distance par Internet et/ou Intranet ainsi que pour le Webmail, une application serveur Web est nécessaire (Type Apache...).

## **6.2. Installation :**

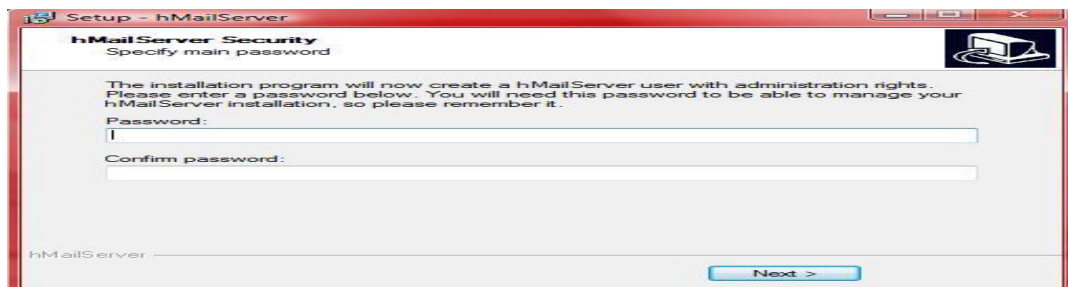
Sélectionner le dossier de destination sur lequel hMailServer doit être installé.

Un détail à prendre en considération est le choix du serveur de base de données :

Il est recommandé de laisser le 1<sup>er</sup> choix comme proposé par le setup, cela permet d'utiliser la base de données incluse dans le serveur et qui la rend indépendante de toute autre base, et si on veut utiliser un autre serveur de base de données, il faut prendre le 2eme choix.

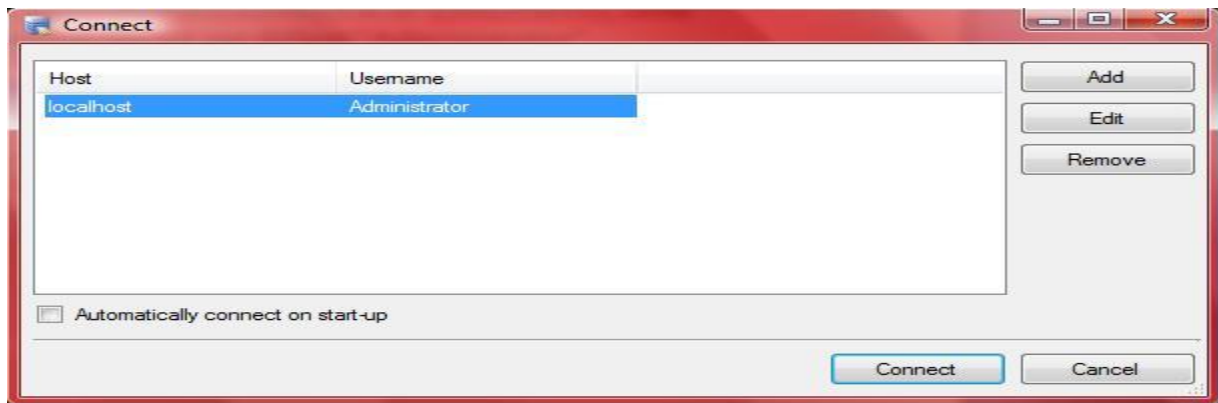


L'installation est presque terminée. Il faut pour sécuriser la configuration, créer un mot de passe pour l'administrateur.



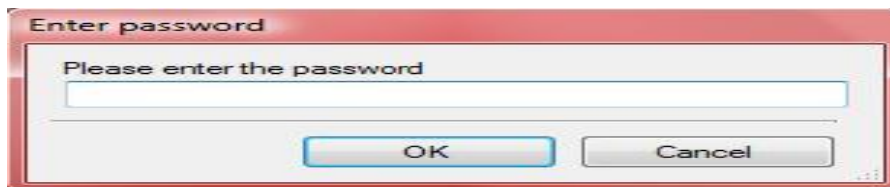
Une fois l'installation terminée, on va lancer le Serveur de mail en tant que service. Celui-ci démarrera automatiquement en même temps que l'ordinateur et la session.

Pour se connecter la première fois au panneau d'administration de hMailServer, cette fenêtre va permettre de sélectionner l'utilisateur qui est par défaut "Administrator".



La possibilité de se connecter automatiquement au serveur est de cocher la case: Connexion automatique au démarrage.

Le mot de passe que vous avez saisi lors de l'installation est maintenant demandé.



Le serveur étant installé, il va falloir le configurer en ajoutant un Nom de Domaine (Réel ou virtuel)

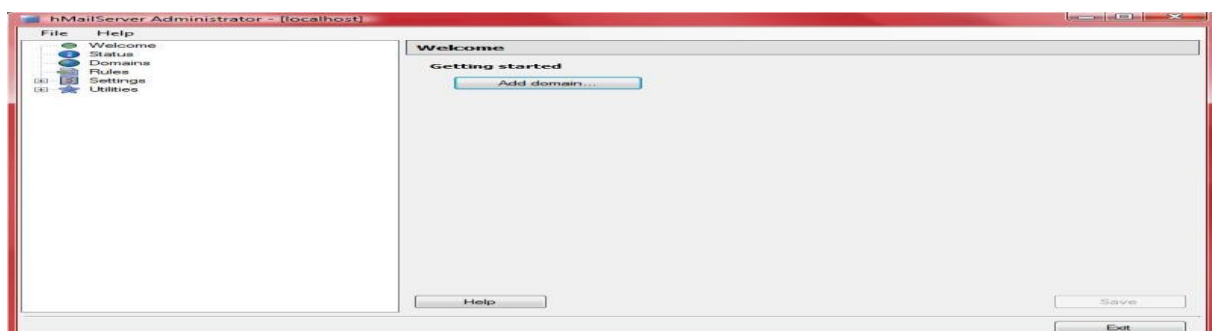
### **6.3. Configuration :**

Le lancement de hMailServer, se fait depuis:

Démarrer > Tous les programmes > hMailServer > hMailServer Administrator

Vous pouvez bien sûr, créer un raccourci sur votre bureau.

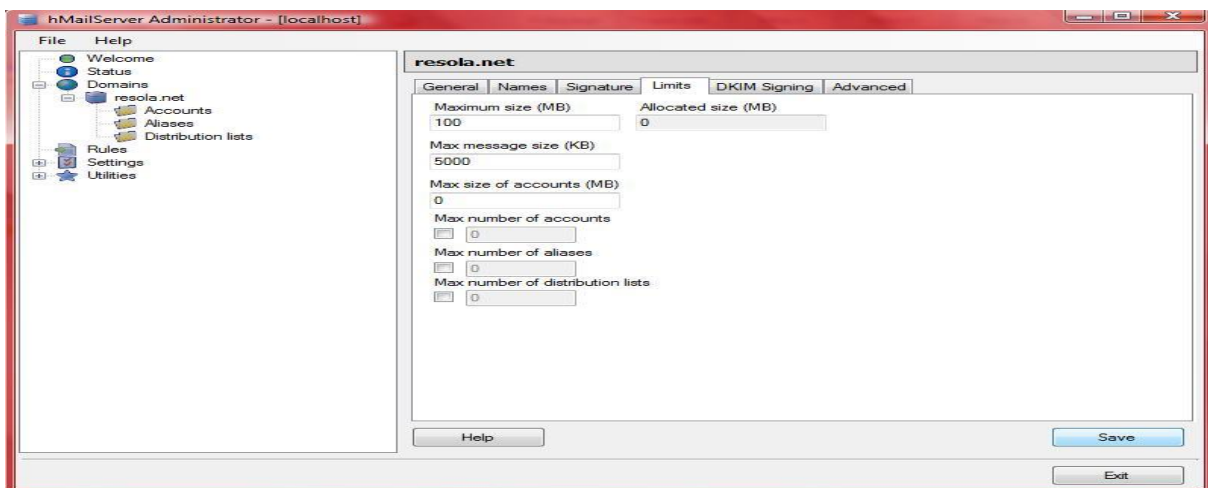
La première fenêtre à s'ouvrir est celle ci-dessous:



Cliquer sur le bouton "Ajouter un domaine..." dans notre exemple "resola.net".

Définir la taille maximale de l'espace alloué à ce domaine pour les mails, (Ici=100Mb) ce qui est suffisant pour une utilisation standard (20 adresses différentes)

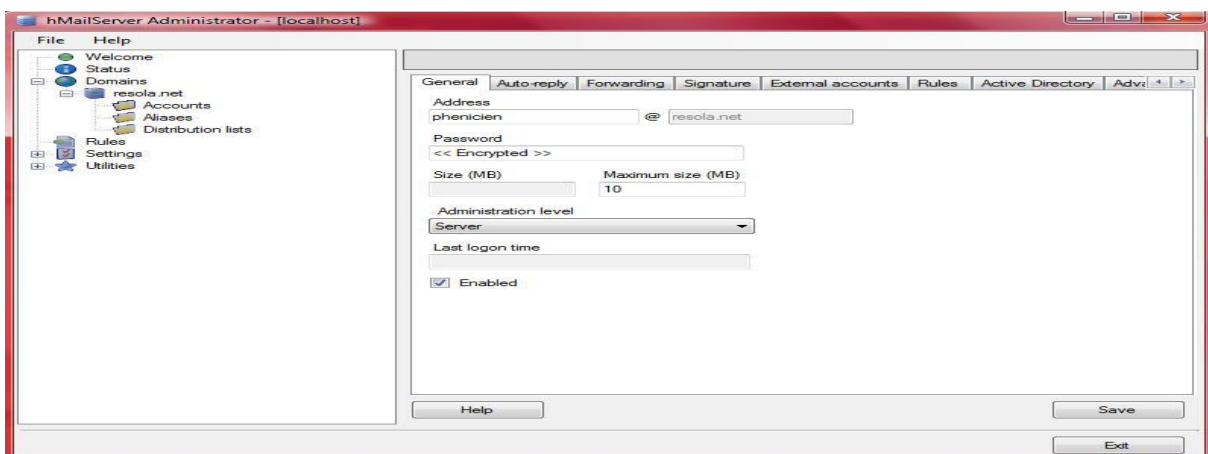
Définir la taille maximale d'un message (Ici= 5000Ko)



Cliquer sur "Enregistrer" afin de sauvegarder les informations.

Ajout d'un utilisateur

Dans le champ "Adresse du compte", il suffit d'indiquer le nom d'un utilisateur, qui va définir l'adresse mail complète, comme par exemple *phenicien@resola.net*. Un mot de passe est bien entendu nécessaire pour la sécurité du compte



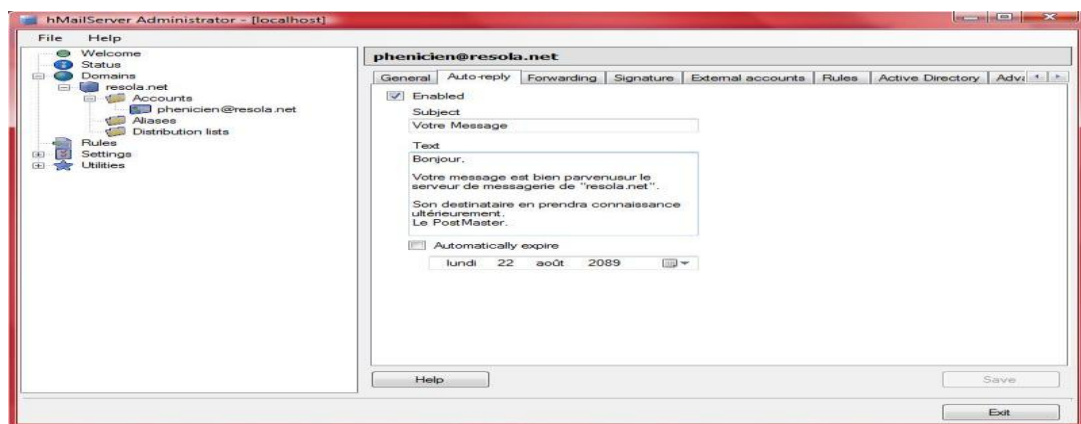
Le niveau d'administration permet de définir les autorisations pour administrer le serveur de messagerie par [phpmailadmin](#) en utilisant un navigateur Internet.

3 niveaux sont offerts:

- **Utilisateur:** Ne peut modifier que certains paramètres de son compte.
- **Domaine:** N'a accès que pour la gestion des Utilisateurs
- **Serveur:** Contrôle total du serveur, sauf que le menu Utilitaires et les fonctions de sauvegarde ne sont pas accessibles.

Cliquer sur "Enregistrer" afin de sauvegarder les informations.

Une réponse automatique peut être envoyée à l'expéditeur. Par exemple :

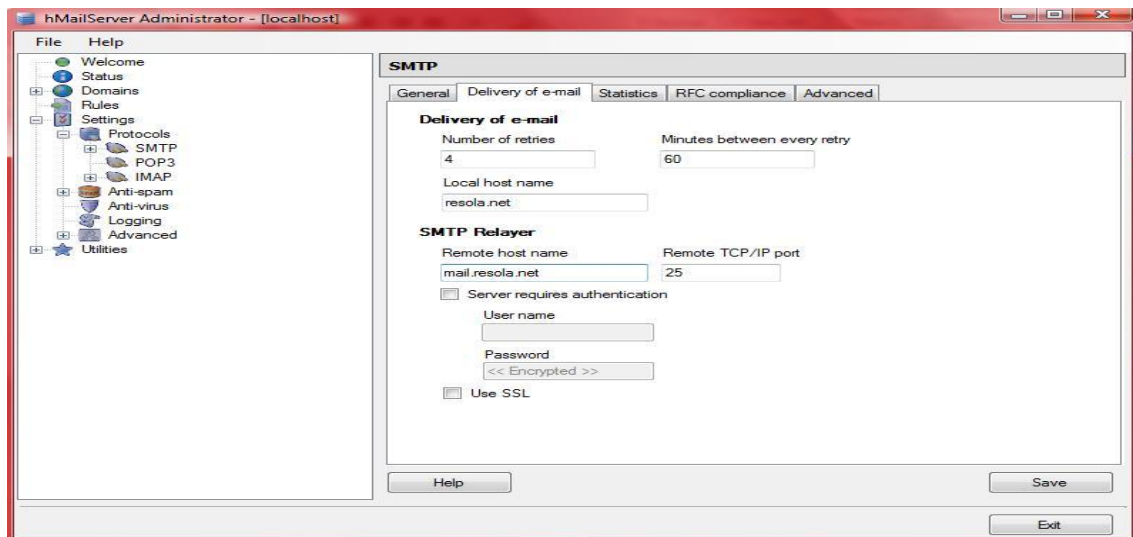


La configuration de la partie "External accounts" n'est nécessaire que si le serveur de mail doit gérer des utilisateurs ayant besoin de se connecter par Internet.

**Ce qui implique que le Nom de domaine soit bien enregistré et actif sur la toile. Il faut que la fonction MX(Mail eXchange) soit activée sur le gestionnaire du Nom de domaine (SimpleDNSPlus) .**

La configuration du compte externe est très simple et ne nécessite pas de commentaire particulier.



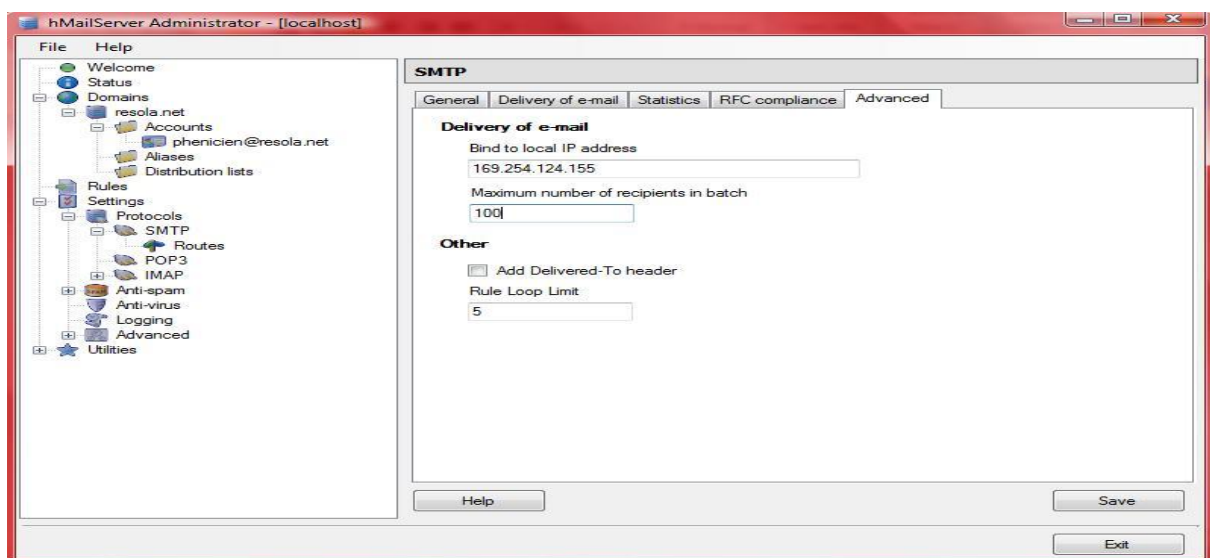


Les champs "Nom du serveur" et "SMTP Relay" devront être renseignés.

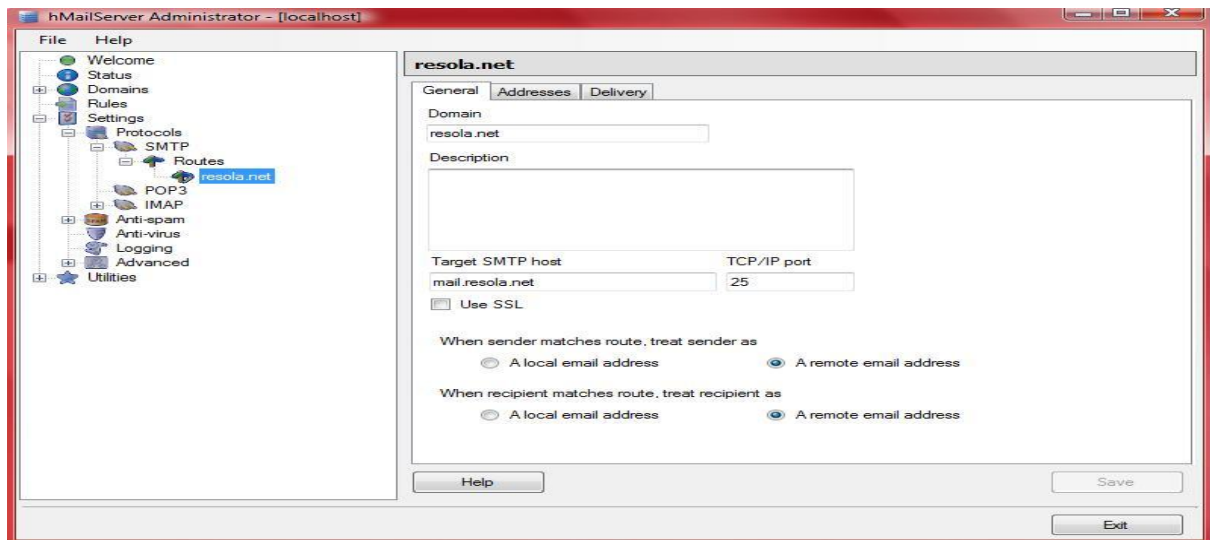
Le nom du serveur peut reprendre votre nom de domaine

Le SMTP Relay peut reprendre le nom du serveur de messagerie.

Afin que les postes connectés sur le même réseau que le serveur de messagerie puissent se connecter facilement à celui-ci, il faut renseigner l'adresse IP du PC serveur.



Un chemin doit aussi être indiqué pour que les mails transitant par le nouveau serveur de messagerie puissent être récupérés par un client (Utilisateur).



Reprendre les mêmes valeurs que précédemment.

Pour les paramètres POP et IMAP, il n'y a besoin de rien modifier.

L'installation et la configuration de hMailServer sont maintenant terminées.

Cependant veuillez vérifier que votre pare-feu autorise ce serveur ainsi que les ports 25 (SMTP) ,110 (POP) ou 143 (IMAP).

#### **6.4. Configuration du client de messagerie :**

Afin de pouvoir récupérer les mails sur son ordinateur, nous allons maintenant voir comment configurer un client de messagerie.

Nous allons prendre pour exemple ThunderBird de Mozilla bien connu par tous les utilisateurs des systèmes d'exploitation Windows.

Une fois ThunderBird ouvert, il faut ajouter un compte de courrier.

#### **Thunderbird Courrier - Dossiers locaux**

##### **Comptes**

 Voir les paramètres pour ce compte

 Créer un nouveau compte

##### **Fonctions avancées**

 Rechercher dans les messages

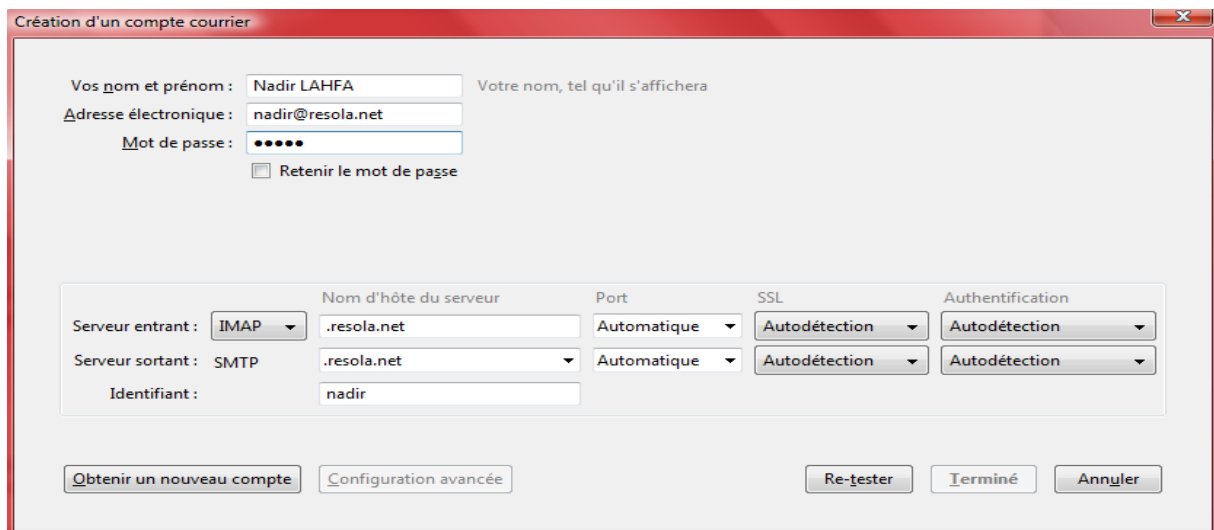
 Gérer les filtres de messages

Pour ce faire, cliquez sur Créer un nouveau compte :



The screenshot shows the 'Bienvenue dans Thunderbird' window. A dialog box titled 'Voulez-vous une nouvelle adresse électronique ?' is displayed. It contains a text input field for 'Votre prénom ou votre pseudo' and a 'Rechercher' button. Below the input field, there is explanatory text: 'En partenariat avec plusieurs fournisseurs, Thunderbird peut vous offrir une nouvelle adresse électronique. Indiquez vos prénom et nom, ou tout autre terme désiré, dans les champs ci-dessous pour démarrer.' Two domain suggestions are shown: 'gandi.net' and 'Hover.com'. At the bottom of the dialog, there are two buttons: 'Passer cette étape et utiliser mon adresse existante' and 'Je configurerai mon compte plus tard.'

Cliquer sur Passer cette étape et utiliser mon adresse existante (les adresses qu'on a déjà créé avec HmailServer) :



The screenshot shows the 'Création d'un compte courrier' dialog box. It contains the following fields and options:

- Vos nom et prénom: Nadir LAHFA (with a note: 'Votre nom, tel qu'il s'affichera')
- Adresse électronique: nadir@resola.net
- Mot de passe: [masked with dots]
- Retenir le mot de passe

Below these fields is a table for server configuration:

	Nom d'hôte du serveur	Port	SSL	Authentification
Serveur entrant: IMAP	.resola.net	Automatique	Autodétection	Autodétection
Serveur sortant: SMTP	.resola.net	Automatique	Autodétection	Autodétection
Identifiant:	nadir			

At the bottom of the dialog, there are buttons for 'Obtenir un nouveau compte', 'Configuration avancée', 'Re-tester', 'Terminé', and 'Annuler'.

En confirmant votre Nom, Prénom, Adresse électronique et mot de passe, ThunderBird va automatiquement rechercher les serveurs entrant/sortant, dans notre cas, on va faire une configuration manuelle :

Création d'un compte courrier

Vos nom et prénom : Nadir LAHFA Votre nom, tel qu'il s'affichera

Adresse électronique : nadir@resola.net

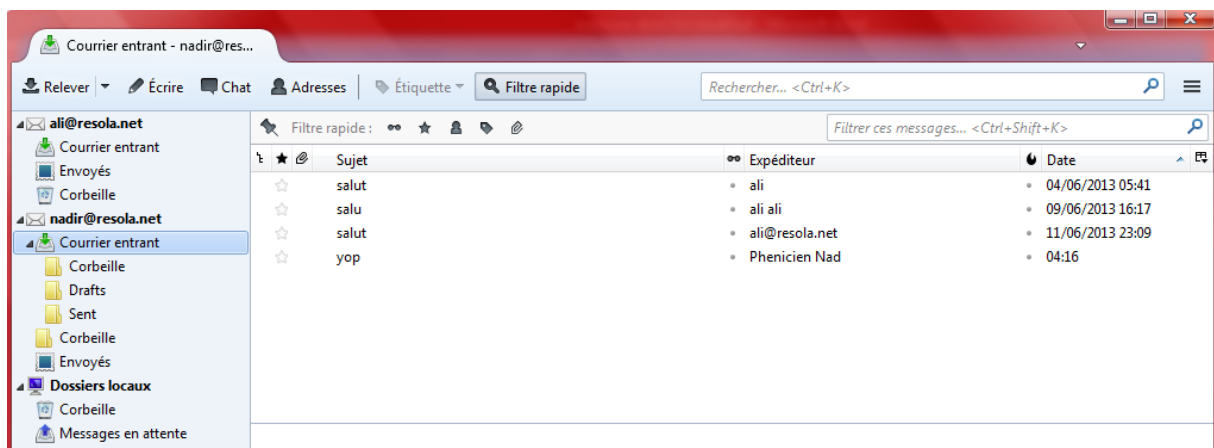
Mot de passe : ●●●●●●

Retenir le mot de passe

Les paramètres suivants ont été trouvés en sondant le serveur indiqué

	Nom d'hôte du serveur	Port	SSL	Authentification
Serveur entrant : IMAP	169.254.124.155	143	Aucune	Mot de passe normal
Serveur sortant : SMTP	169.254.124.155	25	Aucune	Mot de passe normal
Identifiant :	nadir			

En cliquant sur re-tester, le client de messagerie va valider les serveurs IMAP/SMTP, et donc on clique sur terminé. On aura alors notre messagerie :



Les adresses des serveurs peuvent aussi se présenter sous la forme « mail.resola.net »

Le Nom du compte doit être formulé comme l'adresse mail complète.

### **6.5. SQUIRRELMAIL Web mail pour serveur Web : [7]**



SquirrelMail est un webmail gratuit, permettant d'accéder directement aux mails par une interface web à votre serveur de messagerie, sans passer par un client de messagerie (Outlook, Windows Live Mail, ThunderBird, etc.) et dans notre démonstration associé à hMailServer.

SquirrelMail est disponible sous forme d'un fichier compressé en .zip, pour les systèmes tournant sous Windows.

Lorsque le fichier .zip est ouvert, il faut le décompresser vers le répertoire du serveur Web ou vous voulez l'utiliser.

### **6.5.1. Configuration :**

Tel quel on ne pourra rien en faire. Il faut donc le configurer.

Rechercher le fichier config\_default.php qui doit se trouver à une adresse telle que  
...:\www\webmail\config.php

Ouvrir ce fichier avec un éditeur de texte, comme le bloc notes de Windows ou avec un éditeur plus avancé comme NotePad et bien d'autres...

A la fin de sa modification, l'enregistrer sous le nom de config.php

Pour configurer correctement le fichier config.php, il faut modifier les lignes suivantes (ne pas prendre en compte de numéro des lignes, car cela diffère suivant les versions et la configuration mais bien regarder le début de chaque ligne);

```
55 /**
56 * webmail Title
57 *
58 * This is the web page title that appears at the top of the browser window.
59 * @global string $org_title
60 */
61 $org_title = "webmail-resola.net";
62

86 /**
87 * Provider name
88 *
89 * Here you can set name of the link displayed on the right side of main page.
90 *
91 * Link will be displayed only if you have $hide_sm_attributions
92 * option set to true.
93 * @global string $provider_name
94 */
95 $provider_name = 'resola.net';
96

97 /**
98 * Provider URI
99 *
100 * Here you can set URL of the link displayed on the right side of main page.
101 *
102 * Link will be displayed only if you have $hide_sm_attributions
103 * option set to true.
104 * @global string $provider_uri
105 */
106 $provider_uri = 'http://www.resola.net/';
107 $provider_uri = 'http://169.254.124.155/';
108
109
110 /*** Server Settings ***/
111 /**
112 * Default Domain
113 *
114 * The domain part of local email addresses.
115 * This is for all messages sent out from this server.
116 * Reply address is generated by $username@$domain
117 * Example: In bob@example.com, example.com is the domain.
118 * @global string $domain
119 */
120 $domain = 'resola.net';
```

```

144 /**
145 * Your SMTP server (usually the same as the IMAP server).
146 * @global string $smtpServerAddress
147 */
148 $smtpServerAddress = '169.254.124.155';
149 /**
150 * Your SMTP port number (usually 25).
151 * @global integer $smtpPort
152 */
153 $smtpPort = 25;

199 * IMAP server address
200 *
201 * The dns name (or IP address) for your imap server.
202 * @global string $imapServerAddress
203 */
204 $imapServerAddress = '169.254.124.155';
205
206 /**
207 * IMAP server port
208 *
209 * Port used by your imap server. (Usually 143)
210 * @global integer $imapPort
211 */
212 $imapPort = 143;
213
214 /**
215 * IMAP server type
216 *
217 * The type of IMAP server you are running.
218 * Valid type are the following (case is important):
219 *   courier
220 *   cyrus
221 *   exchange
222 *   uw
223 *   macosx
224 *   hmailserver
225 *   other
226 *
227 * Please note that this changes only some of server settings.
228 *
229 * In order to set everything correctly, you need to adjust several
230 * squirrelmail options. These options are listed in doc/presets.txt
231 * @global string $imap_server_type
232 */
233 $imap_server_type = 'hmailserver';

```

A ce stade de la configuration, il faut créer un répertoire à la racine du disque dur (ou lecteur) sur lequel hMailServer et SquirrelMail sont installés. Dans ce répertoire, il faudra aussi créer 2 sous répertoires "Attachments" et "Data".

Le dossier « Attachments » sert à stocker les fichiers temporaires.

Le dossier « Data » sert à stocker les fichiers de préférences et d'autres informations des comptes mail.

On peut les nommer comme bon nous semble, à condition de bien reprendre les adresses correspondantes.

```

499 * @global string $data_dir
500 */
501 $data_dir = 'c:\data_squirrelmail\Data';

517 * @global string $attachment_dir
518 */
519 $attachment_dir = 'c:\data_squirrelmail\Attachments';
520

```

Et le choix de la langue par défaut, ici en Français 'fr-FR' à condition que le patch en Français soit installé.

### 6.5.2. Installation du patch Français :

Le répertoire fr\_FR est à copier dans le répertoire ..: \www\webmail\locale

```

1006 * Default language
1007 *
1008 * This is the default language. It is used as a last resort
1009 * if SquirrelMail can't figure out which language to display.
1010 * Language names usually consist of language code, underscore
1011 * symbol and country code
1012 * @global string $$squirrelmail_default_language
1013 */
1014 $$squirrelmail_default_language = 'fr_FR';

```

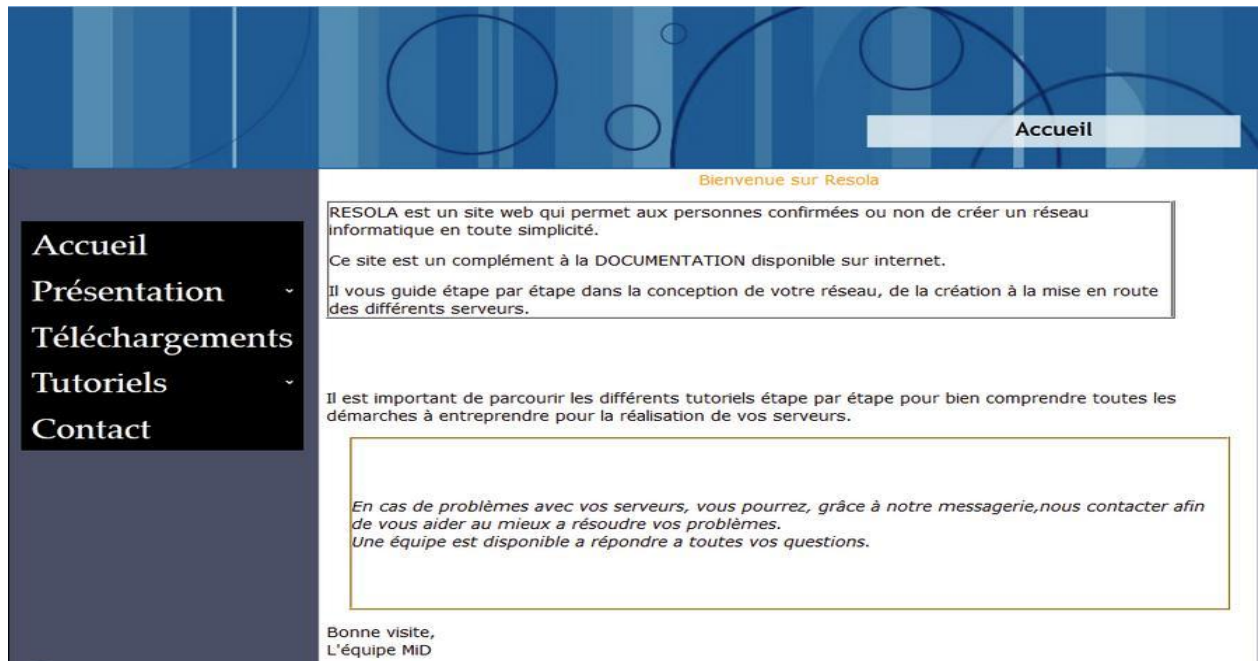
Si tout c'est bien passer, vous devriez en ouvrant votre navigateur internet et à l'adresse du Webmail <http://localhost/webmail> ou <http://www.NomDeDomaine/webmail> avoir une page ressemblante à celle-ci; dans notre cas, sa fera : ”<http://resola.net/webmail>”  
On renseigne notre Adresse de messagerie et notre mot de passe pour y accéder.



Voici une capture d'écran de l'interface une fois un utilisateur connecté.

Il n'est pas très esthétique mais fonctionnel.

Et pour finir, on a rassemblé tout ce travail sur un petit site web assez simple qui va comporter une partie théorique et une partie tutoriels en utilisant nos propres serveurs :



## **7. Conclusion :**

Dans ce chapitre, on a donné l'essence de tous nos efforts consentis pendant toute la période de notre PFE, en espérant qu'il sera d'une grande utilité aux étudiants de tout niveau, spécialement à ceux qui s'intéressent aux réseaux, dont ce n'est qu'un aspect parmi tant d'autres. Personne ne contestera l'utilité d'un réseau, surtout que de plus en plus de gens de tous les horizons utilisent de manière quotidienne un réseau, qui est internet, avec tous les services connus et ceux qui restent plus méconnus, et utilisés que par les spécialistes comme Telnet, DNS...



## **Conclusion Générale**

L'administration réseau est en passe de devenir l'un des profils les plus demandés dans le monde du travail, car pas une société, organisme, banque, université, usine, ou autre ne peut plus se passer d'un réseau informatique reliant tous ses postes de travail, en vue de partager, collaborer, échanger et travailler en réseau, et surtout accéder à internet.

Qui dit réseau, dit tous les services qu'il offre comme le mail, le web, le transfert de fichiers, la prise de commande à distance, la VoIP, la vente en ligne, et tous les nouveaux services qui sont en train de voir le jour ou qui verront le jour dans un futur proche. Et donc pour gérer un réseau, il faut avoir certaines compétences, en l'occurrence dans la sécurité, la gestion et la maintenance des différents serveurs, pour que le réseau soit toujours disponible, performant et agréable à utiliser. Ce qui nous a inspiré pour le choix du sujet de notre PFE, donc l'installation, la configuration et la maintenance d'un serveur mail, FTP, Web, Telnet furent notre objectif principal, sachant que internet foisonne de logiciels et qu'il faut donc faire un choix le plus sage possible, ce que nous espérons avoir fait dans cet humble travail.

## Bibliographies

[1] Sylvain, **le modèle tcp/ip**, mai 2003.

<http://www.frameip.com/>

[2] LAHFA Nadir et Hassaine Walid, **Installation et configuration d'un serveur web sous Windows**.2009.

[3] Matt, **créer son propre serveur FTP avec Filezilla Serveur**, mai 2012.  
<http://www.skyminds.net/creer-son-propre-serveur-ftp-avec-filezilla-server/>

[4] **Activation de telnet sous windows Vista/7**, mars 2010.

[http://technet.microsoft.com/fr-fr/library/cc771275\(v=ws.10\).aspx](http://technet.microsoft.com/fr-fr/library/cc771275(v=ws.10).aspx)

[5] **Utilisation de simple DNS plus** (Anglais).

<http://www.sav25.com/pdf/simpledns-brochure09v1.pdf>

[6] **Installation HmailServer**.2011.

<http://www.jdl68160.com/informatique/hmail/>.

[7] **Installation Squirrelmail**.2013.

<http://www.jdl68160.com/informatique/hmail/webmail.html>.

### Livres:

- **Client / serveur (Guide de survie)**, Auteur(s) : Robert Orfali, Dan Harkey, Jeri Edwards.
- **Apache (version 2) Installation, administration et sécurisation**, Auteur(s) : Mikaël Pirio
- **Php 5**, Auteur(s) : Jean Engels
- **FileZilla kompakt**, Auteur(s) : Peter BERTGES
- **Installer et configurer des serveurs Web, mail et FTP**, Auteur(s) : Jean-Francois Bouchaudy.
- Coffret – Exchange Server 2010 et PowerShell ; **Administrez votre serveur de messagerie** - Coffret 2 volumes, Auteur(s) : Emmanuel Vinazza, Arnaud Petitjean, Robin Lemesle
- **Réseaux informatiques, Notions fondamentales et Administration sous Windows ou Linux** - Théorie et Pratique - 2 livres en 1, Auteur(s) : José Dordoigne
- **Internet information services 8 (IIS 8) - installation, configuration et maintenance du serveur web**, Auteur(s) : Loïc Thobois
- **TCP/IP - Le guide complet Maîtrisez les principes de la communication réseau sous Windows !**, Auteur(s) : Sylvain Caicoya, Jean-Georges Saury
- **L'Architecture client-serveur, aspects techniques enjeux stratégiques**, Alain Lefebvre (Auteur) - (donnée non spécifiée). Paru en 12/1997.

## Listes des figures

Fig I.B.1.Adressage IP .....	10
Fig I.B.2.Adressage IP .....	11
Fig I.C.3.Architecture Serveur/Client .....	13
Fig I.C.4.Architecture et fonctionnement messagerie .....	17
Fig I.C.5.Communication Serveur/navigateur .....	23
Fig I.C.6.Transfert données Telnet .....	27
Fig I.C.7.Interception données Telnet .....	28
Fig I.C.8.Interception Login/Pass Telnet .....	28