*RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE*

*MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE*


## UNIVERSITÉ IBN-KHALDOUN DE TIARET

**FACULTÉ DES SCIENCES APPLIQUEES**

**DÉPARTEMENT DE GENIE ELECTRIQUE**


# MEMOIRE DE FIN D'ETUDES

**Pour l'obtention du diplôme de Master**

**Domaine : Sciences et Technologie**

**Filière : Génie Electrique**

**Spécialité : Informatique industrielles**

## THÈME

# *Mise en œuvre d'un Proxy Filtrant sous Linux*


***Préparé par :*** **DJILAILI HANANE**
**CHOUICHI HANANE**


**Devant le Jury :**

| Nom et prénoms | Grade | Qualité |
|---|---|---|
| MAASKRI.M | MAA | Président |
| GOUASMI.M | MAA | Examinateur |
| BENABID.H | MAA | Encadreur |

**PROMOTION 2015 /2016**

# Dédicace

*Je dédie se modeste travail aux êtres les plus chers au monde :*

*A* ***mon grande père « ABI HABIBI HADJ »*** *pour leur sacrifice et amour durant toutes mes années d'étude*

*A mes chers et mes biens aimes,* ***mon père*** *et* ***ma mère*** *qui ont partage mes joies et mes soucis, et qui ont tout sacrifié pour ma réussite.*

*A ma deixieume pére papa* ***Abdalaziz***

*A ma deixieume méme mama* ***Horia***

*A mes frères «****Moulay, Abdalaziz, Manad, Saad, Med, Ahmed et Youcef****»*

*A mes sœurs «Hbiba, Manina, Asmaa, Kheira, Aicha, Fazo et Aya », a qui je souhaite la belle vie avec pleine de joies et de bonheur.*

*A ma très chère et charmante famille.*

*A tout mes amis « Hanane, Sabah, Nacéra et Imane ».*

*ET*

*A tout mes collègues de la promotion M2 Informatique Industrielle*

*Djilaili Hanane*

# *Dédicace*

*JE dédie ce modeste travail à :*

*MA très chère et douce mère, Mon très cher père à qui m'adresse au ciel les vœux les plus ardents pour la conservation de leur santé et de leur vie.*

*Pour mes chers frères : Mohamed, Habib, Amine, Aboubaker et Benchohra*

*Pour mes très chers amis : Hanane, Sabah, Nacera et Djamila*

*a toutes la promotion de M2 Informatique Industrielle : 2015-2016*

*Chouichi Hanane*

# Remerciement

Nous remercions Allah, le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail.

Nous adressons notre reconnaissance, notre gratitude à notre enseignant encadreur Monsieur
BENABID HOUARI .Nous tenons, tout particulièrement et très sincèrement, à lui remercier de nous avoir proposé le sujet et de nous avoir en encadrés. Son suivi, ses encouragements et ses orientations ont été d'un grand réconfort et d'une aide précieuse. Qu'il nous soit permis d'exprimer nos plus vifs remerciements de nous avoir fait bénéficier de ses compétences, ses qualité humaines et de sa disponibilité non seulement pour la réalisation de ce mémoire mais aussi durant tout le parcours de notre formation .

Nos remerciements s'adressent également à Monsieur le président du jury et les membres du jury pour l'honneur qu'ils nous font d'avoir bien voulu étudier ce travaille et de le juger.

Nous n'oublions pas d'adresser un grand merci à tout les enseignants, touts les personnes qui ont contribuées de prés et de loin à l'enrichissement et à notre épanouissement intellectuel durant tout ce parcours universitaire aux Département de Génie Electrique de l'Université de Tiaret.

# Liste d’abréviations

ACL : Access Control List

ACO : Access Control Operator

ARP : Address Resolution Protocol

Arpanet : Advanced Reseach Projet Agency

ATM : Asynchronous Transfert Mode

CERN : Centre Européen de Recherche Nucléaire

CORBA : Common Object Request Broker Architecture

CIDR : Classless Inter-Domain Routing

CR/LF : Carringe return/line feed

CTI : Computer Telephony Integration

DCOM : Distribution Component Object Model

DHCP : Dynamic Host Configuration Protocol

DNS : Domain Name System

DOD : Département américain de la Défense

FDDI : Fiber Distributed Data Interface

FTP : File Transfer Protocol

GNU/GPL : General Public License

GPS : Global Position System

HTTP : Hyper Text Transfer Protocol

ICANN : Internet Corporation for Assigned Names and Numbers

ICMP : Internet Control Message Protocol

IP : Internet Protocol

IPTV : Internet Protocol Télévision

ISA : Internet Security and Acceleration( Microsoft firewall and cache server)

LAN : local area network

# Liste d'abréviations

LDAR : Lightweight  Directory Access Protocol

MAN : Metropolitan Area Network

MCA : Microsoft Certified Architect

MIME : Multipurpose Internet Mail Extensions

MRTG : Multi Router Traffic Grapher

NAT : Network Address Translation

NSF : La National Science Fondation

NTLM : NT LAN Manage

OSI : Open System Interconnections

PCI : Protocol-Control Information

POP 3 : Post Office Protocol

RFC : Requests for Comments

RFID : Radio Frequency Identification

RMI : Revenu Minimum D'insertion

RPC : Remote Protocol Call

RR : Resource Records

SMTP : Simple Mail Transfer Protocol

SNMP : Simple Network Management Protocol

SOAP : Simpele Object Access Protocol

SSL : Secure Sockests Layer

TCP : Transmission Control Protocol

TTL : Time Top Live, traduisez espérance de vie

TLD : Top Level Domain, soit domaines de plus haut niveau

UDDI : Universal Discovery Description and Integration

UDP : User Datagram Protocol

UNIX : Uniplexed Informatique and Computer Service

URL : Uniform Resource Locator

# Liste d’abréviations

VBI : Virtual Desktop Infrastructure

VLN : Virtual Local Area Network

VLB : Vesa Local Bus

VOD : Video en demand

WAIS : Wide Area Information Serves

WAN : Wide Area Network

WEB : World Wide Web

WCCP : Web Cache Communication Protocol

WSDL : Web Servise Description Language

XML : Extensible Markup Language

# Sommaire

# Sommaire

# Sommaire

# Sommaire

# Sommaire

# Sommaire

# Liste de figures :

# Liste de tableau :

# Introduction Générale

## 1.1 Introduction

Pour un administrateur réseau, il est important de contrôler l'utilisation d'Internet dans une entreprise afin de limiter les abus. L'interdiction aux utilisateurs de naviguer sur des sites « sensibles » et la réduction des temps d'accès permet d'assurer la sécurité et le bon fonctionnement d'un réseau d'entreprise sur Internet.

Le serveur proxy permet d'assurer ces fonctions puisqu'il va effectuer une requête sur Internet pour le compte d'un ordinateur du réseau local.

Ce mémoire introduit une réflexion sur les problèmes posés par les interconnexions de réseaux IP de politiques de sécurité différente. Il distingue un réseau « interne » a priori maitrisé d'un réseau « externe » sur lequel il n'est pas fait d'hypothèse. On suppose par contre qu'une politique de sécurité de cette interconnexion a été réalisée au préalable. Cette politique doit avoir identifié les flux autorisés à traverser l'interconnexion, ainsi que les objectifs éventuels d'authentification forte de ces derniers. Ce mémoire propose des architectures d'interconnexion pouvant être utilisées dans la conception de passerelles ou d'équipements intégrés. Il est présenté dans le cadre d'une solution complètement basée sur des outils open source sur les interconnexions de réseaux à titre d'introduction à cette problématique de protection aux limites d'enclaves et ne constitue en l'état qu'un document pour lancer le travail d'identification d'architectures génériques et modulables.

Ce mémoire est organisé en quatre chapitres :

Le premier chapitre représente une généralité réseaux. Il définit les classifications de réseaux, leur matériel ainsi que leur stratégie de connexion, transmission de l'information et architecture client/serveur. Nous terminerons ce chapitre par le modèle TCP/IP.

Le deuxième chapitre : les services réseaux. On va voir les principaux services réseau talques le service de résolution de noms (machines : DNS), service d'attribution d'adresse (DHCP), la messagerie, l'annuaire, et le web.

Le troisième chapitre : considère le serveur Proxy et leurs fonctionnalités, et en a mentionner quelque avantages et inconvénients du proxy, et présenter Squid et Squidguard.

Quatrième chapitre: premièrement nous allons aperçu les technologies que nous avons utilisé. Ensuit, nous allons montrer la politique sécurité proposée.

## 1.2 Problématiques générale :

Les problèmes d'interconnexions interviennent toujours entre deux systèmes d'information entre lesquels doit exister un certain cloisonnement. Il s'agit, par exemple, de la connexion d'un réseau local (LAN) a un réseau étendu (WAN) par une liaison. Les exigences naturelles dans ce type de situation sont que les flux sortants et entrants soient contrôlés. En particulier, certaines informations du réseau local sont considérées comme privées et ne doivent donc pas être accessibles de l'extérieur. Inversement, il peut être interdit par la politique du réseau local d'accéder a certaines données sur le réseau étendu. Par souci pédagogique, nous allons nous concentrer sur le protocole HTTP qui est majoritairement utilisé sur l'Internet pour la consultation et la mise à disposition de données. Notre hypothèse de base est que seul le réseau local est sous contrôle. Le réseau étendu est considéré par hypothèse comme non sur c'est-a-dire en particulier, qu'il n'y a aucune garantie de respect des protocoles de communication. Usuels sur l'interface réseau local – réseau étendu. La politique de sécurité voulue dans notre cas est que le réseau étendu ne doit pouvoir accéder qu'aux données explicitement autorisées au niveau local. Inversement, le réseau local peut accéder à toutes les données disponibles sur le réseau étendu (l'Internet) à l'exception des données interdites. Il y a une dissymétrie imposée par l'hypothèse de travail. En effet, nous supposons qu'il n'est pas possible d'imposer au réseau local de n'accéder qu'à des données explicitement autorisées, puisque l'origine des données issues du réseau étendu ne peut être garantie. Dans ces conditions, on ne peut qu'interdire a posteriori l'accès à certains sites une fois qu'ils ont et identifies.

# Chapitre 01

# Généralités sur

# Les réseaux

## 1. Introduction à la réseautique :

Malgré que l'industrie informatique est plus jeune par rapport à d'autre industrie (automobile, transport aérien,…), elle a fait des progrès spectaculaires dans un petit temps. Pendent ces vingt première années, les systèmes informatiques étaient très centralisés, situés physiquement en général dans une salle. Le concept de salle d'ordinateur comme lieu où les utilisateurs apportaient leurs travaux à traiter est aujourd'hui complètement obsolète. Le modèle ancien d'un unique ordinateur est remplacé par celui d'un ensemble d'ordinateurs séparés mais interconnectés qui exécutent des taches différentes. De tels systèmes sont appelés Réseaux**.**

Dans ce chapitre nous allons parler sur les types, topologies, architectures de réseau, les éléments d'un réseau et enfin les modes de fonctionnement d'un réseau et ces applications. [1]

## 2. Historique général des réseaux :

Internet est issu du réseau Arpanet (de l'Advanced Research Projects Agency), créé en 1968 par le département Américain de la Défense, dans un but stratégique, pour relier ses centres de recherche.

Le réseau initial ne permettait que l'envoi de courrier électronique. C'est en 1972 que commencèrent les spécifications des protocoles TCP/IP avec l'expérience de l'usage de X25 sur ARPANET.

En 1983, c'est au tour de l'Europe et du reste du monde de se connecter à ce réseau de réseaux.

L'outil qui rendit populaire l'internet à partir de 1993 est le WWW

Le premier navigateur WEB graphique a été mis aux points au CERN (centre européen de recherche nucléaire) en 1993.

Un navigateur Web permet de se connecter à une multitude de sites diffusant des informations sans connaissances des règles de communication propre au réseau.

L'internet reliait en 1995 plus de 2 millions d'ordinateurs et plus de 30 millions d'utilisateurs dans 146 pays. [2]

3. **Définition Réseaux :** C'est un ensemble d'ordinateurs (ou de périphériques) autonomes connectés entre eux et qui sont situés dans un certain domaine géographique. Deux stations sont considérées comme interconnectées si elles sont capables d'échanger de l'information.

## 4. Classification des réseaux :

On peut classifier les réseaux selon deux aspects :

- Leurs tailles
- Leurs topologies.

**4.1. Selon leurs tailles :** (en termes de nombre de machine): Leur vitesse de transfert des données ainsi que leur étendue .Les réseaux appartenant à une même organisation. On fait généralement trois catégories de réseaux.

**4.1.1. Réseaux LAN** : Local Area Network(en français Réseau Local)

Il s'agit d'un ensemble d'ordinateurs appartenant à une même organisation et reliés entre eux par un réseau dans une petite aire géographique. La vitesse de transfert de données d'un réseau local peut atteindre 10 Mbps (pour un réseau Ethernet par exemple) et 100Mbps (en FDDI par exemple). Les LANs ne comportent généralement pas de cent ordinateurs.

**4.1.2. Réseau MAN** : (Metropolitan Area Network)

Ils permettent l'interconnexion des entreprises ou des départements sur un réseau spécialisé à haut débit. Ce type correspondent à une interconnexion de quelques bâtiments se trouvent dans une ville (Campus).

**4.1.3. Réseau WAN: (**Wide Area Network)

Fonctionnent grâce à des routeurs qui permettent de choisir le trajet le plus approprié pour atteindre un nœud du réseau. Un WAN (Wide Area Network) ou réseau étendu, sert à relier des LANs et des MANs, les réseaux qui composent un WAN peuvent être situés dans un même pays ou être dispersés dans le mode. [4]

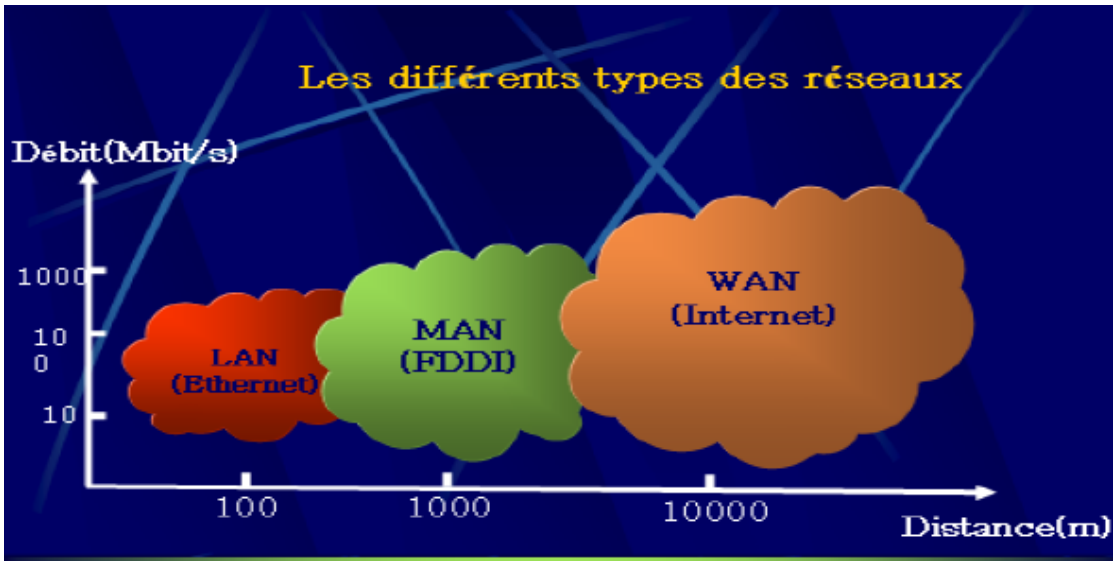


**Figure 1.1 : Classification selon la taille.**

### 4.2. Selon leurs topologies

On peut également différencier les réseaux selon leur structure et plus précisément leur topologie : La topologie est l'organisation physique et logique d'un réseau.

**4.2.1. Topologies physique :** un réseau informatique est constitué d'ordinateurs reliés entre eux par des câblages et des équipements permettant d'assurer la bonne circulation des données. L'arrangement physique de ces éléments est appelé topologie physique. Il existe trois :

**4.2.1.1. Topologie bus :** Une topologie en bus est l'organisation la plus simple d'un réseau. En effet dans une topologie en bus tous les ordinateurs sont reliés à une même ligne de transmission par l'intermédiaire de câble, généralement coaxiale. Le mot « bus » désigne la ligne physique qui relie les machines du réseau. Dans cette topologie, toutes les stations (imprimante, ordinateurs,..)Sont connectés en série de long d'un câble désigné par bus.

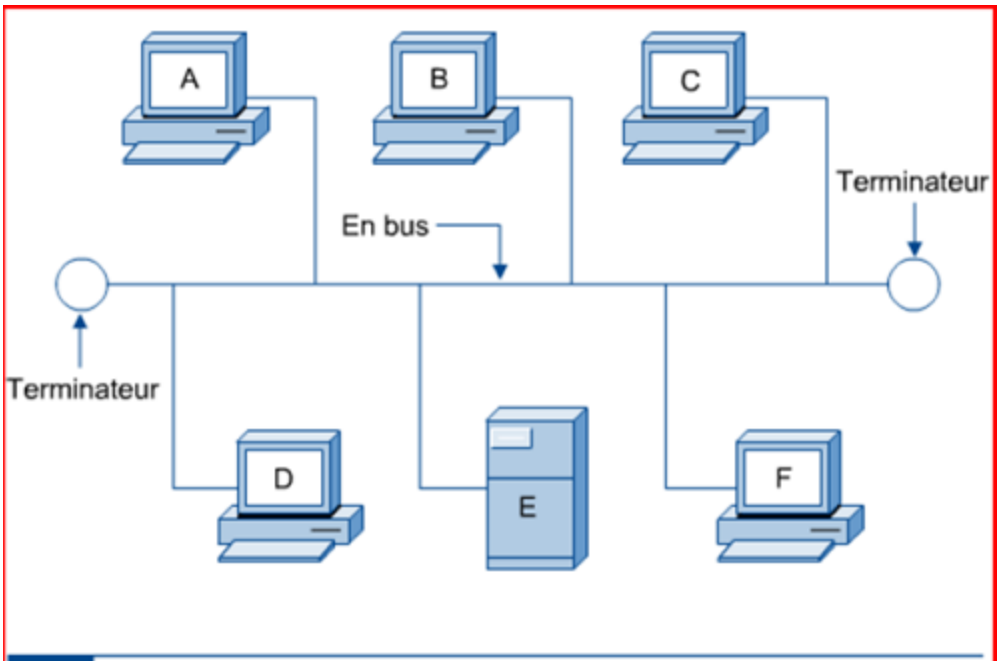


**Figure 1.2 : Topologie bus.**

Cette topologie a pour avantages d'être facile à mettre en œuvre et de fonctionner facilement, par contre elle est extrêmement vulnérable.

**4.2.1.2. Topologie en étoile :** Dans cette topologie, les ordinateurs du réseau sont connectés à un équipement appelé hub ou concentrateur. Il s'agit d'une boite comprenant un certain nombre de jonctions auxquelles on peut connecter les câbles en provenance des ordinateurs. Celui-ci a pour rôle d'assurer la communication entre les différentes jonctions. [3]

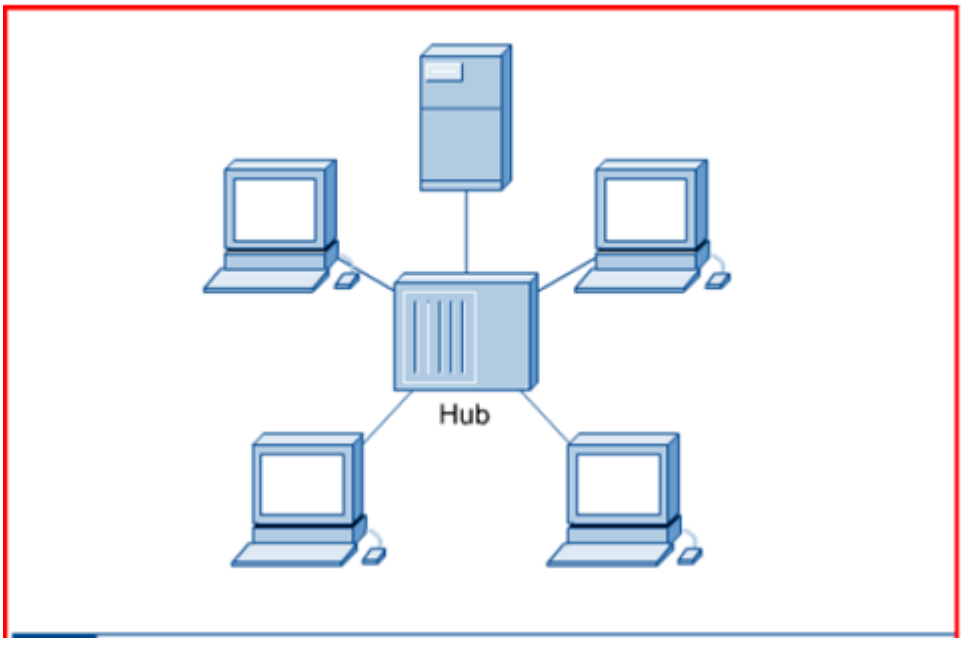


**Figure 1.3 : Topologie étoile.**

#### 4.2.1.3. Topologie en anneau :

Dans cette topologie, les hôtes sont connectés via un cercle physique ou anneau. Cette disposition n'ayant aucun début fin, il n'est pas nécessaire d'équiper le câble d'un terminateur. Contrairement à la topologie en bus, aucune de ses extrémités ne nécessite de terminaison. Les deux principales topologies logiques utilisant cette topologie physique sont TokenRing et FDDI

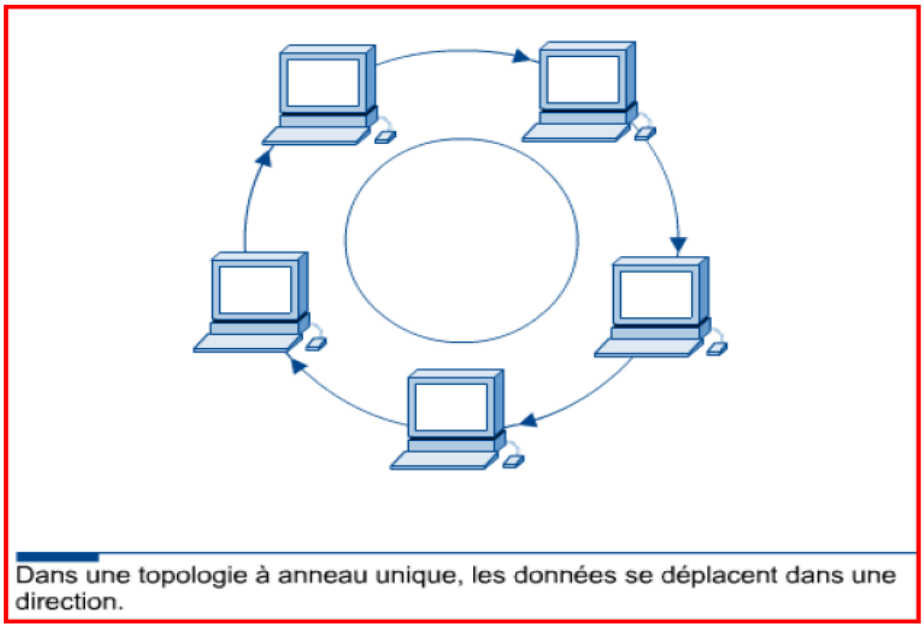


**Figure 1.4.Topologie anneau.**

### 4.2.2. Topologie logique

Représente la façon d'accès au support (câbles). Les topologies logiques les plus courantes sont Ethernet, Token Ring et FDDI.

## 5. Matériel de réseau

Le matériel de réseau est l'ensemble des éléments physique qui composent un réseau, chaque type de réseau nécessite un matériel spécial.

### 5.1. Ordinateur

Un Ordinateur est une machine automatique commandée par des programmes enregistrés dans sa mémoire. Il est capable d'effectuer des opérations variées sur les données proposées, a une grande vitesse, sans risque d'erreur (à condition que les programmes soient corrects). L'utilisateur fournit des données, l'ordinateur effectue sur ces données les traitements.

### 5.2. Câbles

Un câble électrique désigne un regroupement de fils conducteurs avec parfois un blindage électromagnétique extérieur. Un câble électrique peut être utilisé pour le transport d'énergie (en général électrique) mais aussi pour la transmission de données (entre autres téléphoniques et informatiques).

### 5.3. Connecteur :

Les connecteurs informatiques, généralement appelés « connecteurs d'entrée-sortie » (notés E/S), sont des interfaces permettant de relier des équipements à l'aide de câbles. Ils se composent généralement d'une prise mâle, avec des broches (en anglais pin) saillantes

**5.4. Carte réseau :** est matérialisée par un ensemble de composants électroniques soudés sur un circuit imprimé. L'ensemble constitué par le circuit imprimé et les composants soudés s'appelle une carte électronique, d'où le nom de carte réseau. La carte réseau assure l'interface entre l'équipement ou la machine dans lequel elle est montée et un ensemble d'autres équipements connectés sur le même réseau.

### 5.5. Les serveurs :

Un serveur réseau est un ordinateur spécifique partageant ses ressources avec d'autres ordinateurs appelés clients**.** Il fournit un service en réponse à une demande d'un client.

Il existe plusieurs types de serveurs :

**5.5.1 Les serveurs dédiés :** ordinateur situé à distance mis à la disposition d'un seul client par un prestataire. Le client pourra bénéficier pleinement des capacités et des ressources de la machine.

**5.5.2 Les serveurs mutualisés** : Un hébergement mutualisé est un concept d'hébergement internet destiné principalement à des sites web. Ce type de serveur va donc héberger plusieurs sites internet sur un seul et même serveur. Il repose sur le partage équitable des ressources, à savoir la mémoire RAM, le CPU, les espaces disques et la bande passante.

**5.5.3 Les serveurs virtuels :** Un serveur virtuel se comporte comme un serveur dédié, mais le dispositif qui l'héberge est mutualisé. La machine physique héberge plusieurs serveurs virtuels simultanément, d'où son caractère mutualisé. [4]

### 5.6 Supports de transmission :

Chaque carte réseau est interconnectée à l'aide de câble, dont le choix dépend du réseau mis en œuvre.

#### 5.6.1 Câble coaxial :

C'est un câble utilisé également en téléphone et en télévision, il est constitue d'un fil de cuivre dans une gaine isolante, elle même en tournée par une tresse de cuivre. Le tout est couvert d'une gaine isolante. Sa bande passante est de 50 à 400 Mégahertz et son

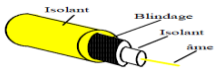


**Figure 1.5: une paire coaxiale.**

#### 5.6.2 Paire torsadée :

Le câble à paire torsadée (TWISTED-PAIR CABLE) est composé de plusieurs éléments :

- Des brins de cuivre entrelacés (torsadés)
- Une enveloppe isolante

Le câble à paire torsadée a été largement diffusé parce qu'il est à l'origine utilisé pour les lignes téléphoniques et qu'il était jusqu'en 1983 systématiquement pré installé dans tous les nouveaux bâtiments américains. Le câble à paire torsadée est le support (le média) le plus utilisé à l'intérieur d'un bâtiment.

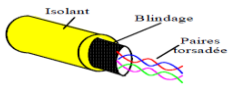


**Figure 1.6: une paire torsadée**

## 6. Logiciels de réseau :

Les logiciels de réseau sont les programmes qui gèrent le réseau, fournissent les services et permettent aux ordinateurs de communiquer et de partager des informations. [4]

## 7. Stratégie de connexion :

#### 7.1.1. Commutation de circuits :

Un chemin physique est établi à l'initialisation de la communication entre l'émetteur et le récepteur et reste le même pendant toute la durée de la communication. Si les deux correspondants n'ont pas de données à transmettre pendant un certain temps, la liaison restera inutilisée. L'idée est de concentrer plusieurs correspondants sur une même liaison. Dans le cas où les communications

seraient nombreuses, il faut prévoir des mémoires pour stocker des informations en attendant que la liaison soit disponible.

#### 7.1.2. Commutation de messages :

Consiste à envoyer un ensemble d'informations (Un message) d'un émetteur vers un récepteur en passant par un ou plusieurs nœuds de commutation. Chacun de ces nœuds attend la réception complète du message avant de le réémettre, cela demande des buffers sur chaque équipement, ainsi qu'un contrôle de flux pour éviter les engorgements. De plus le taux d'erreurs pour des messages de taille importante doit être très bas.

**Remarque** : La commutation de message nécessite la mise en place d'algorithmes de routage.

#### 7.1.3. Commutation de paquets :

Celle-ci reprend la méthode précédente, mais en découpant le message en un nombre de fragment défini. Chaque nœud redirige ces fragments selon ses propres lois (tables de routage), la reprise sur erreur est donc plus simple, cependant le récepteur final doit être capable de réassembler tous ces paquets dans un ordre souvent différent de celui dans lequel il les a reçu.

Cette technique nécessite la mise en place de la numérotation des paquets.

#### 7.1.4. Commutation de cellule :

Commutation de paquets particulière. Tous les paquets ont une longueur fixe de 53 octets (une cellule= un paquet de 53 octets), c'est un mélange de la commutation de circuit et de la commutation de paquets, elle a pour avantage de simplifier le travaille des commutateurs et d'autoriser des débits plus élevés. [1]

## 8. Catégories de réseaux :

- Les réseaux poste à poste (égal à égal),
- Réseaux organisés autour de serveurs (Client/serveur),
- Trois tiers.

### 8.1. Les réseaux poste à poste

Dans une architecture d'égal à égal (où dans sa dénomination anglaise peer to peer), tous les ordinateurs sont égaux, il n y a pas de machine spécifique. Cela signifie que chacun des ordinateurs du réseau est libre de partager ses ressources. Un ordinateur relié à une imprimante pourra donc éventuellement la partager afin que tous les autres ordinateurs puissent y accéder via le réseau.

### 8.2. Architecture client/serveur :

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des machines clientes contactent un serveur, une machine généralement très puissante en termes de capacités d'entrée-sortie, qui leur fournit des services.

Dans un environnement purement Client/serveur, les ordinateurs du réseau (les clients) ne peuvent voir que le serveur, c'est un des principaux atouts de ce modèle.

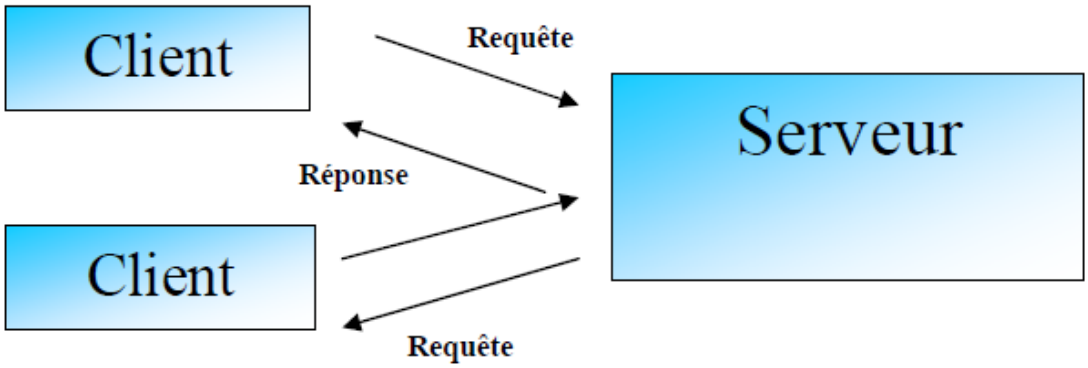


**Figure 1.7 : Architecture client/serveur**

- Le client émet une requête vers le serveur grâce à son adresse, demandant un service.
- Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine.

### 8.3. Architecture Trois tiers

Dans l'architecture à 3 niveaux (appelées architecture 3-tiers), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre:

Le client, le demandeur de ressources.

Le serveur d'application, le serveur chargé de fournir la ressource mais faisant appel à un autre serveur.

Le serveur secondaire, fournissant un service au premier serveur. [1]

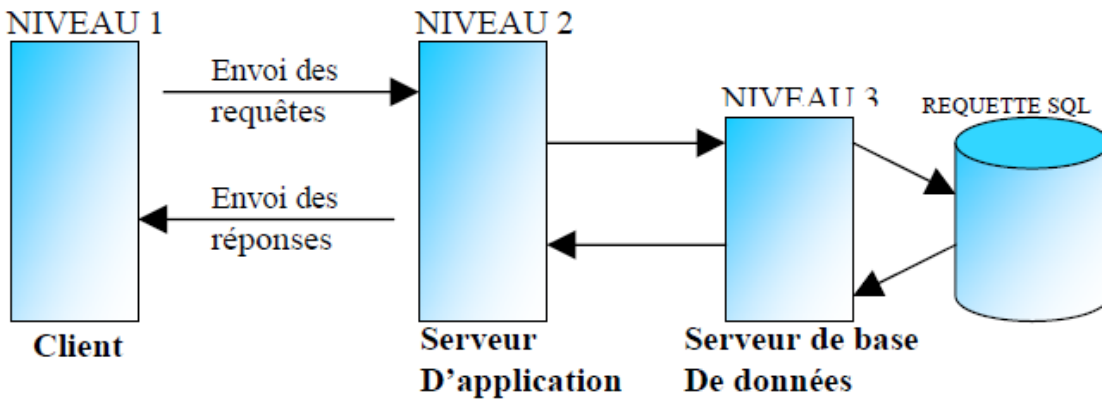


**Figure 1.8 : Architecture Trois tiers.**

**9. La transmission de l'information sur un réseau :**

**9.1. Mode de diffusion :** Consiste à partager un seul support de transmission. Chaque message envoyé par un équipement sur le réseau est reçu par tous les autres. A tout moment chaque équipement à le droit d'envoyer un message sur le support, il faut juste écouter au préalable si la voie est libre, sinon il doit attendre. Les réseaux locaux adoptent pour la plupart des cas, le mode diffusion sur une architecture en bus ou en anneau. La rupture du support provoque l'arrêt du réseau, par contre la panne d'un des éléments ne provoque pas la pane globale du réseau.

- **Adresse physique/logique**: C'est l'adresse spécifique placé dans le message qui permettra à chaque équipement de déterminer si le message lui est adressé ou non.

**9.2. Mode de point à point :** le support physique (câble) relie une paire d'équipement seulement. Quand deux équipements non directement connecter entre eux veulent communiquer, ils le font par l'intermédiaire des autres nœuds du réseau. [1]

**10. Modes de connexions :**

On distingue deux modes de connexion quelle que soit l'architecture physique d'un réseau :

- Le mode connecté.
- Le mode non connecté.

**10.1. Mode connecte :**

Le Processus de communication de ce mode est illustré dans cet algorithme :

1. L'émetteur demande l'établissement d'une connexion avec un hôte.
2. si le récepteur refuse la connexion, celle-ci n'aura pas lieu.
3. Sinon un lien s'établit entre l'émetteur et le récepteur.

4. Les données transitent d'un point à l'autre.

5. la communication est libérée.

La communication téléphonique est le meilleur exemple pour ce mode.

### 10.2. Mode non connecté

Le Processus de communication de ce mode est comme suite :

1. L'émetteur envoie un message sur un support et il espère qu'il arrive.

2. Le message contient les coordonnées du destinataire.

3. Chaque récepteur potentiel possède des coordonnées uniques.

4. le contenu de l'information est inconnu de l'émetteur.

5. le support est inconnu des utilisateurs (applicatifs). [1]

## 11. Qu'apportent les réseaux :

### 11.1. Usage des réseaux :

- Partager des ressources: imprimantes, disque dur, processeur, etc.
- Réduire les coûts: par exemple au lieu d'avoir une imprimante pour chaque utilisateur qui sera utilisée 1 heure par semaine, on partage cette même imprimante entre plusieurs utilisateurs.
- Augmenter la fiabilité: dupliquer les données et les traitements sur plusieurs machines. Si une machine tombe en panne une autre prendra la relève.
- Fournir un puissant média de communication: e-mail, VC …..
- Faciliter la vente directe via l'Internet.
- Accès facile et rapide à des informations distantes
- Communication entre les individus : Vidéoconférence, courrier électronique, groupes thématiques (newsgroups), caviardage (chat), communication poste-à-poste (peer-to-peer), téléphonie et radio via Internet, etc.
- Divertissements et jeux interactifs : vidéo à la carte et toutes sortes de jeux (jeux d'échec, de combats, etc.)
- Commerce électronique (e-commerce) : transactions financières, achats en ligne à partir de son domicile.

### 11.2. Les réseaux permettent :

- Le partage des fichiers.

- Le partage d'application : compilation, SGBD.
- Partage de ressources matérielles : l'imprimante, disque…
- Télécharger des applications et des fichiers.
- L'interaction avec les utilisateurs connectés : messagerie électronique, conférences électroniques, ….
- Le transfert de données en général: réseaux informatiques.
- Le transfert de la parole : réseaux téléphoniques.
- Le transfert de la parole, de la vidéo et des données : réseaux numérique à intégration de services RNIS ou sur IP. [3]

## 12. Architecture de réseaux :

### 12.1. Modèle de référence OSI d'ISO (Open System Interconnection) :

Principal modèle utilisé pour les communications réseau. C'est le meilleur outil pour décrire l'envoi et la réception de données sur un réseau. Composé de 7 couches

- Couches 1 à 4 dites couches basses :

Prennent en charge le transport des données

- Couche 5 à 7 : couches hautes

S'occupent de tout ce qui concerne les applications.

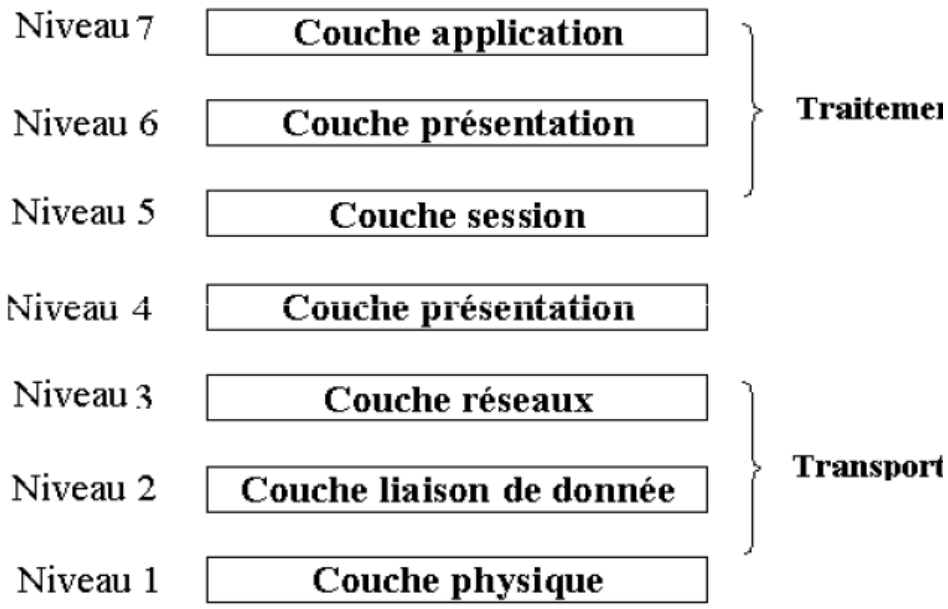


**Figure 1.9 : Le modèle de référence OSI.**

#### 12.1.1. La couche application :

C'est la couche qui est visible par l'utilisateur, elle contient les logiciels capables de fonctionner en réseau.

Protocoles les plus connus : – HTTP, FTP, SMTP, DHCP, DNS,

POP, IMAP, SSH, Telnet, …

#### 12.1.2. La couche présentation :

- Elle permet de mettre en forme les données, elle convertit les données pour qu'elles soient compréhensibles
- Elle (dé)crypte les données
- Protocoles les plus connus : – SSL, TLS, XML, HTTP/HTML …

#### 12.1.3. La couche session:

- Elle gère les sessions entre les hôtes :
- Le démarrage des sessions
- Le (re)synchronisation des hôtes
- La fermeture des sessions

• Protocoles les plus connus : – NFS, NetBIOS, AppleTalk, …

#### 12.1.4. La couche transport :

- Elle gère le transport des données d'un hôte à l'autre
- Elle segmente les données
- Elle vérifie et corrige les erreurs
- Protocoles les plus connus : – TCP, UDP, SPX, …

#### 12.1.5. La couche réseau :

- Elle permet de diriger les paquets vers la bonne destination.
- Elle utilise un système d'adressage.
- Elle gère le routage.
- Protocoles les plus connus : – IP, ICMP, ARP, IPX …

#### 12.1.6. La couche liaison de données :

- Elle gère la bonne transmission des trames sur une liaison physique

- Elle détecte les erreurs et retransmet en cas de problème
- Protocoles les plus connus : – Ethernet, TokenRing, FDDI, PPP,

Frame Relay, …

### 12.1.7. La couche physique :

- Elle définit la partie physique de l'accès au réseau :

– Codage des données

– Protocoles d'échange de bits

- Protocoles les plus connus : – Codage NRZI, Manchester, Bipolaire,

MLT3, …

**13. le modèle TCP/IP :** Il est utilise pour le réseau internet, il est inventé pour la défense Etats-Unis (l'ARPA) pour le réseau ARPANET, ancêtre d'Internet.

Il contient 4 couches.

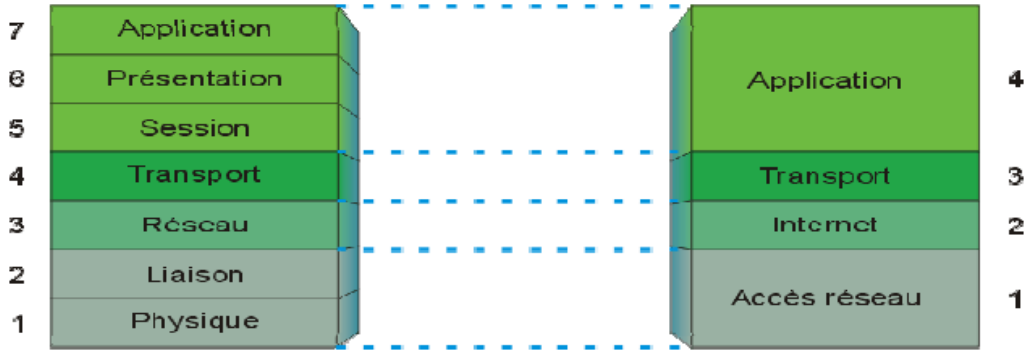


**Figure 1.10 : Le model TCP/IP et le model OSI.**

Le modèle OSI a été mis à côté pour faciliter la comparaison entre les deux modèles.

### 13.1. La couche Application :

- C'est la couche qui est visible par l'utilisateur
- Elle contient les logiciels capables de fonctionner en réseau

• Protocoles les plus connus : – HTTP, FTP, SMTP, DHCP, DNS,

POP, IMAP, SSL, LDAP, …

### 13.2. La couche transport :

- Elle gère le transport des données d'un hôte à l'autre
- Elle segmente les données

- Elle peut vérifier et corriger les erreurs
- Protocoles les plus connus : – TCP, UDP.

### 13.1. La couche internet :

- Elle permet de diriger les paquets vers la bonne destination
- Elle utilise un système d'adressage
- Elle gère le routage
- Protocoles les plus connus : – IP, ICMP, ARP …

### 13.2. La couche hôte-réseau :

- Elle gère la bonne transmission des trames sur une liaison physique
- Elle définit la partie physique de l'accès au réseau :
  -Codage des données -Protocoles d'échange de bits
- Protocoles les plus connus : Ethernet, WiFi, Token Ring, Frame Relay

PPP, PPPoE, …

- Codages les plus connus : Manchester, NRZ, NRZI. [5]

**Conclusion :**

Dans un réseau il faut tout d'abord choisir : une topologie, un bon système de câblage, les meilleur techniques de transmission et de commutation et choisir une architecture conforme aux organismes de normalisation pour les réseaux.

Dans ce qui suit, un état de l'art des quelques services réseaux service Web, DNS, messagerie, DHCP, les annuaires

# Chapitre 2 :

# Les services réseau

## 1. Introduction

Un service réseau est une fonctionnalité assurée par un ordinateur consistant en l'aptitude à la fourniture d'informations à d'autres ordinateurs via une connexion réseau normalisée. Les services réseaux se basent sur des protocoles pour fournir des fonctionnalités qui sont accessibles par l'utilisateur au niveau de la couche 7 du modèle OSI (couche application).

Dans ce chapitre on va voir les principaux services réseau talques le service de résolution de noms (machines : DNS), service d'attribution d'adresse (DHCP), la messagerie, l'annuaire, et le web.

## 2. Architecture client/serveur

L'architecture client/serveur désigne un mode de communication entre plusieurs composants d'un réseau. Chaque entité est considérée comme un client ou un 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.

Donc, Le client pose une question (ou donne un ordre)… et le serveur répond à la question (ou obéit).

### 2.1. Fonctionnement d'un système client/serveur :

Un système client/serveur fonctionne selon le schéma suivant :

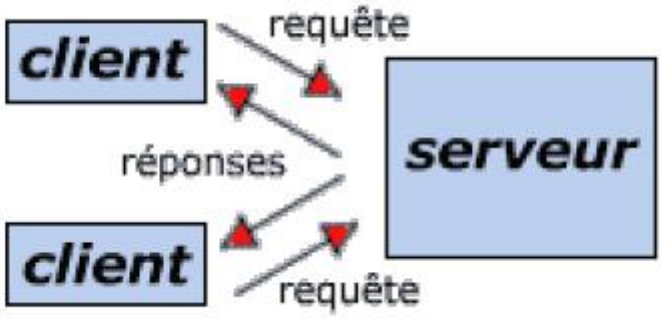


**Figure 2.1 : Fonctionnement d'un système client/serveur.**

- Le client émet une requête vers le serveur grâce à son adresse IP et le port, qui désigne un service particulier du serveur ;
- Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine cliente et son port.

### 2.1. Présentation de l'architecture à 2 niveaux

Ce type d'architecture (2-tier en anglais) caractérise les environnements client-serveur où le poste client demande une ressource au serveur qui la fournit à partir de ses propres ressources.

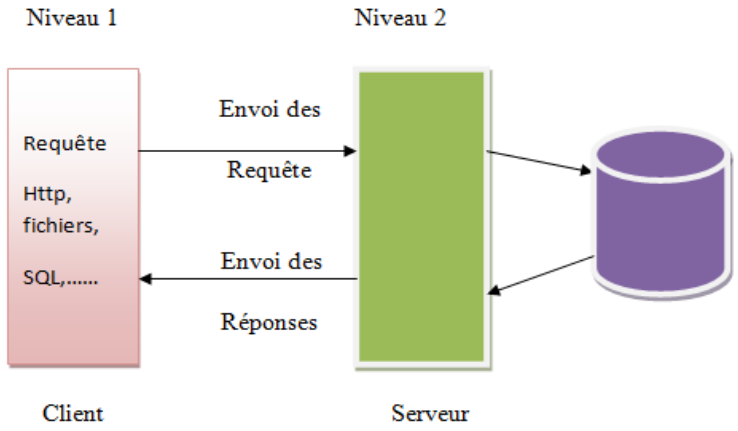


**Figure 2.2: L'architecture à deux niveaux d'un système client/serveur.**

### 2.2. Présentation de l'architecture à 3 niveaux

Dans l'architecture à 3 niveaux (appelée architecture 3-tiers), il existe un niveau intermédiaire, c'est-à-dire que l'on a généralement une architecture partagée entre :

1. Un client, c'est-à-dire l'ordinateur demandeur de ressources, équipée d'une interface utilisateur (généralement un navigateur web) chargée de la présentation.
2. Le serveur d'application (appelé également middleware), chargé de fournir la ressource mais faisant appel à un autre serveur
3. Le serveur de données, fournissant au serveur d'application les données dont il a besoin. [6]

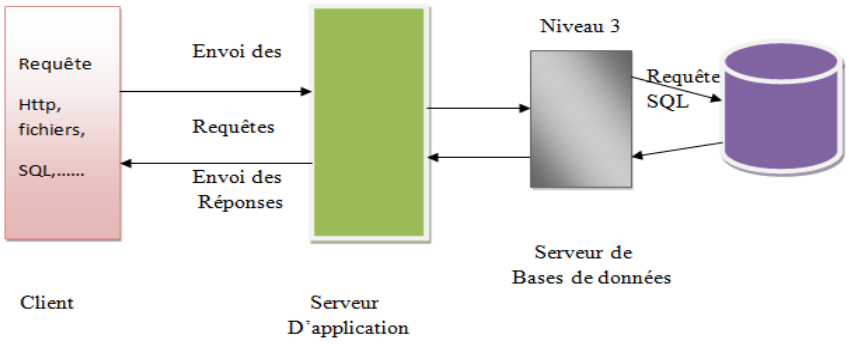


**Figure 2.3: L'architecture à trois niveaux d'un système client/serveur.**

Etant donné l'emploi massif du terme d'architecture à 3 niveaux, celui-ci peut parfois désigner aussi les architectures suivantes :

- Partage d'application entre client, serveur intermédiaire, et serveur d'entreprise ;
- Partage d'application entre client, serveur d'application, et serveur de base de données d'entreprise.

### 2.3. . Comparaison des deux types d'architectures :

L'architecture à deux niveaux est donc une architecture client/serveur dans laquelle le serveur est polyvalent, c'est-à-dire qu'il est capable de fournir directement l'ensemble des ressources demandées par le client.

Dans l'architecture à trois niveaux par contre, les applications au niveau serveur sont délocalisées, c'est-à dire que chaque serveur est spécialisé dans une tâche (serveur web/serveur de base de données par exemple). L'architecture à trois niveaux permet :

- Une plus grande flexibilité/souplesse ;
- Une sécurité accrue car la sécurité peut être définie indépendamment pour chaque service, et à chaque niveau ;
- De meilleures performances, étant donné le partage des tâches entre les différents serveurs.

## 3. Les services réseaux :

### 3.1. Les services web

#### 3.1.1. Définition

Les Web Services sont des services offerts via le web Par exemple, un client demande le prix d'un article en envoyant un message sur le web. Ce message contient la référence de l'article. Le Web Service va recevoir la référence, effectuer le traitement du service et renvoyer le prix au client via un autre message.[7]

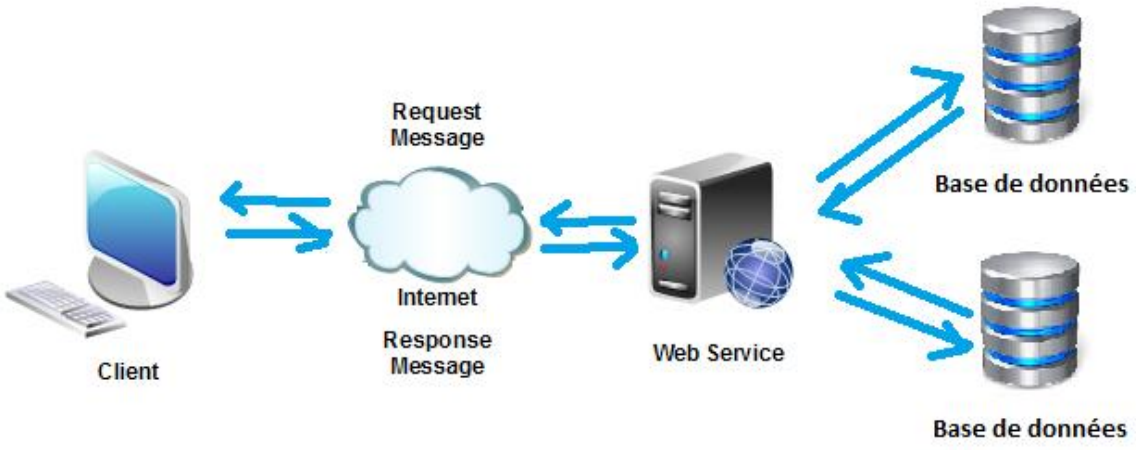


**Figure 2.4: Le service web.**

### 3.1.2. Les technologies concernées :

Les Web services ne sont pas un standard mais plutôt un ensemble de standards associés à trois spécifications XML :

**3.1.2.1. SOAP** (simple Object Access Protocol) est un protocole standard de communication. C'est l'épine dorsale du système d'interopérabilité. SOAP est un protocole décrit en XML et standardisé par le W3C. Il se présente comme une enveloppe pouvant être signée et pouvant contenir des données ou des pièces jointes.

Il circule sur le protocole HTTP et permet d'effectuer des appels de méthodes à distance.

**3.1.2.2. WSDL** (Web Service Description Language) est un langage de description standard. C'est l'interface présentée aux utilisateurs. Il indique comment utiliser le service Web et comment interagir avec lui. WSDL est basé sur XML et permet de décrire de façon précise les détails concernant le service Web tels que les protocoles, les ports utilisés, les opérations pouvant être effectuées, les formats des messages d'entrée et de sortie et les exceptions pouvant être envoyées.

**3.1.2.3. UDDI** (Universal description, Descovery and Integration) est un annuaire de services. Il fournit l'infrastructure de base pour la publication et la découverte des services Web. UDDI permet aux fournisseurs de présenter leurs services Web aux clients.

Les informations qu'il contient peuvent être séparées en trois types :

- les pages blanches qui incluent l'adresse, le contact et les identifiants relatifs au service Web;
- les pages jaunes qui identifient les secteurs d'affaires relatifs au service Web ;
- les pages vertes qui donnent les informations techniques.

Nous allons étudier plus en détail, ces trois dernières technologies.

### 3.2. Les serveurs de noms :

Les serveurs de noms (également appelés serveurs DNS) permettent de retrouver votre nom de domaine sur Internet. Ils assurent la conversion du nom de domaine en l'adresse IP de l'ordinateur cible (où ce trouve par exemple le site Web). Pour qu'un nom de domaine puisse fonctionner sur Internet, il faut qu'au moins un serveur de noms soit configuré pour ce nom de domaine et indiqué à l'enregistrement du nom de domaine.

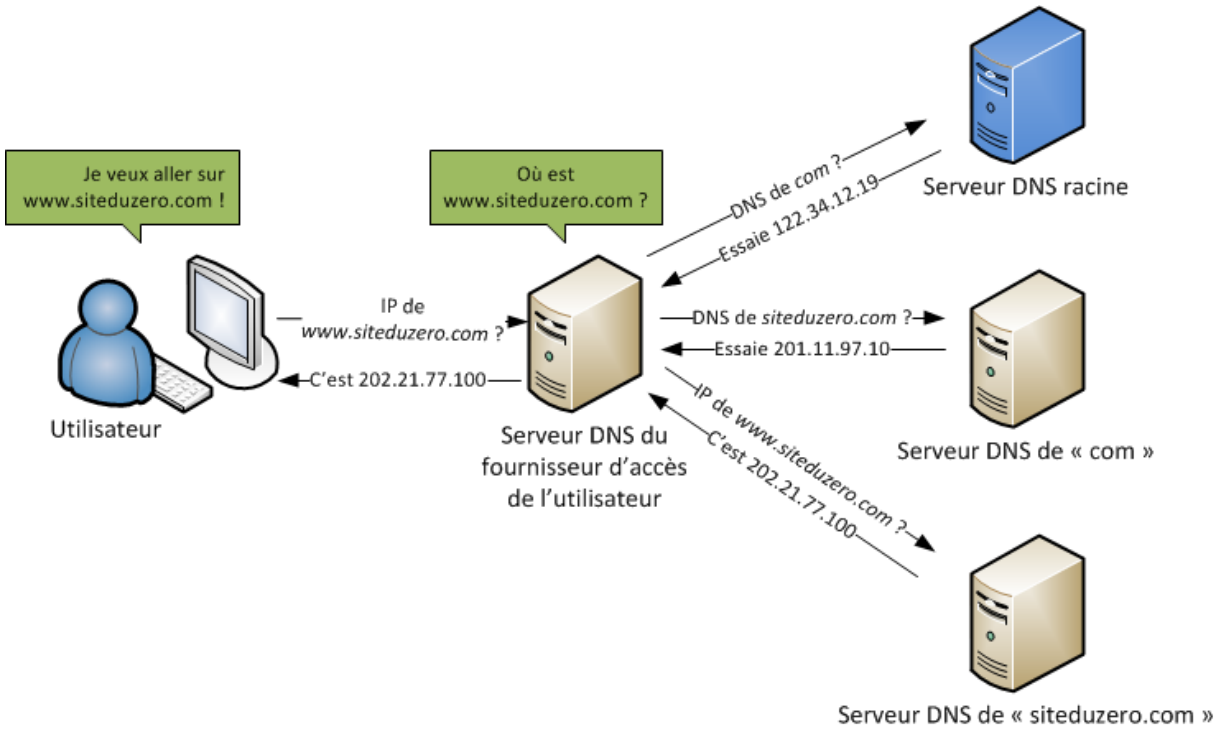


**Figure 2.5: Système de Nom de Domaine DNS.**

#### 3.2.1. Résolution de noms de domaine :

Le mécanisme consistant à trouver l'adresse IP correspondant au nom d'un hôte est appelé « résolution de nom de domaine ». L'application permettant de réaliser cette opération (généralement intégrée au système d'exploitation) est appelée « résolveur ».

Lorsqu'une application souhaite se connecter à un hôte connu par son nom de domaine (par exemple « www.commentcamarche.net »), celle-ci va interroger un serveur de noms défini dans sa configuration réseau. Chaque machine connectée au réseau possède en effet dans sa configuration les adresses IP de deux serveurs de noms de son fournisseur d'accès.

Une requête est ainsi envoyée au premier serveur de noms (appelé « serveur de nom primaire »). Si celui-ci possède l'enregistrement dans son cache, il l'envoie à l'application, dans le cas contraire il interroge un serveur racine (dans notre cas un serveur racine correspondant au TLD « .net »). Le serveur de nom racine renvoie une liste de serveurs de noms faisant autorité sur le domaine.

Le serveur de noms primaire faisant autorité sur le domaine va alors être interrogé et retourner l'enregistrement correspondant à l'hôte sur le domaine (dans notre cas *www*).

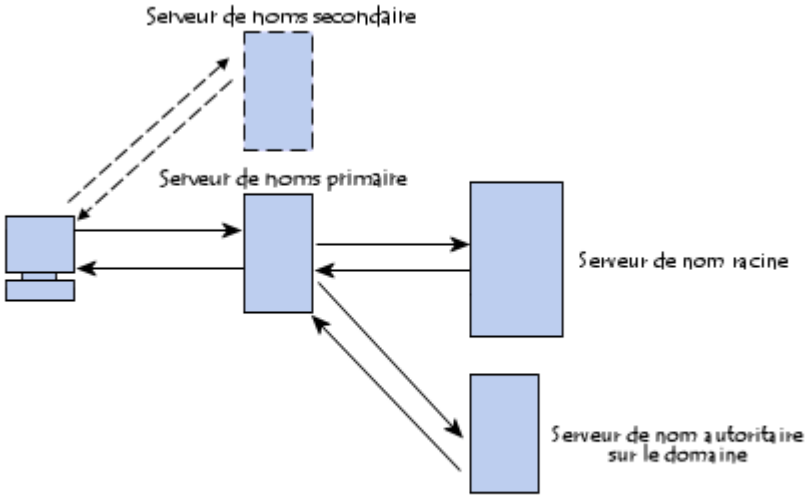


**Figure 2.6: Résolution de noms de domaine.**

### 3.3. La messagerie électronique :

Un serveur de messagerie électronique est 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 courrieller 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.

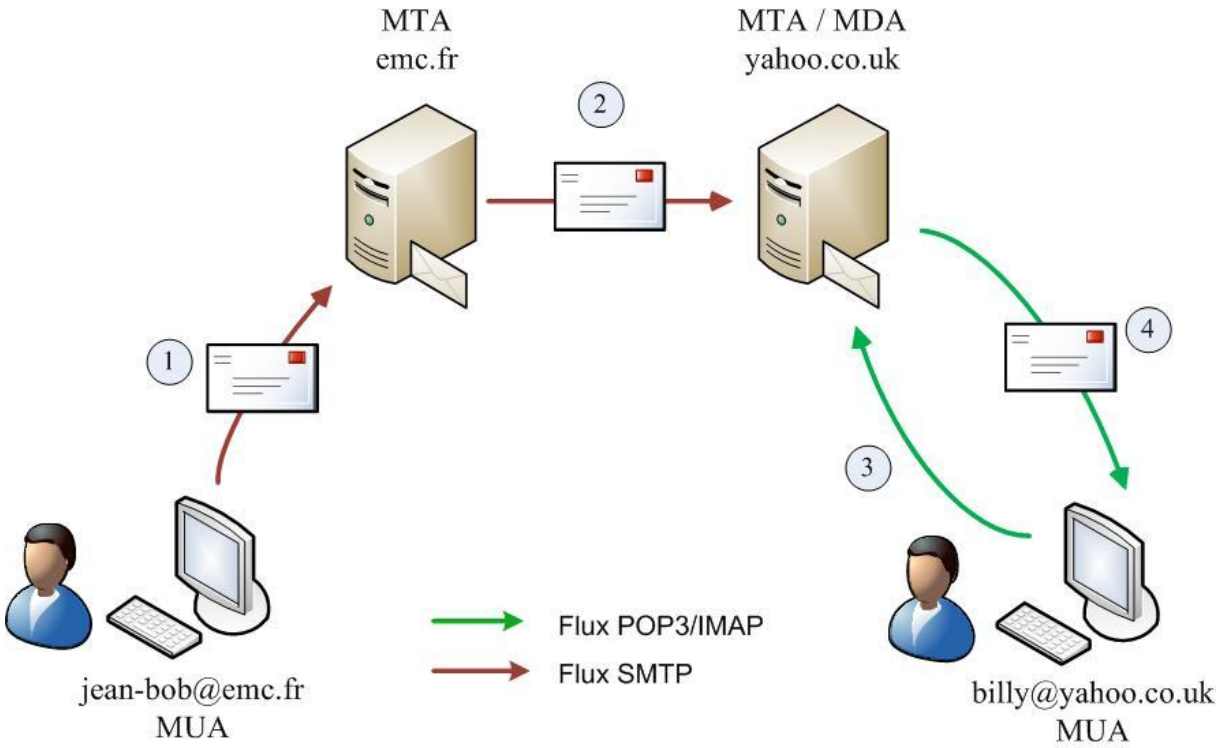


**Figure 2.7: La messagerie électronique.**

### 3.3.1. Les protocoles de communications :

Deux grands types de protocoles et de serveurs sont utilisés pour le courrier électronique:

**3.3.1.1. Les protocoles sortants :** permettant de gérer la transmission du courrier entre les serveurs. Le principal protocole sortant est SMTP.

**3.3.1.1.1. SMTP :(**Simple Mail Transfer Protocol, ou Protocole Simple de Transfert de Courrier) : L'un des standards d'Internet, c'est un protocole de communication pour le courrier permettant d'établir l'interface entre un réseau local et Internet.

Le serveur SMTP (Configuration de la messagerie) est le serveur "sortant", servant à la gestion du courrier entre différents serveurs de messagerie.

Que fait le serveur SMTP :

a. il s'occupe du transport des messages entre serveurs de messagerie (par exemple entre le serveur gmail et le serveur Yahoo)

b. il réceptionne et centralise les messages envoyés, sans vérifier l'identité des émetteurs.

c. il ne différencie pas les destinataires des différents champs : To, Cc, etc.

d. il ne gère pas les Dates ni les Sujets des messages.

**3.3.1.2. Les protocoles entrants :** qui gèrent l'envoi des messages dans les messageries personnelles. Deux protocoles entrants sont utilisés, au choix, dans les systèmes de messagerie : POP ou IMAP. Le principe des protocoles entrants est le même : gérer la communication entre l'utilisateur et le serveur de messagerie et permettre aux utilisateurs d'aller récupérer leurs messages. Ce sont des protocoles de réception et de distribution du courrier. Pour reprendre l'analogie avec le système postal, le protocole et le serveur "entrant" correspondent au centre de tri postal dont vous dépendez et au facteur, qui vient vous apporter le courrier. S'ils ont globalement la même fonction, des différences importantes existent entre les protocoles POP et IMAP, entrainant des possibilités d'utilisation différentes de la messagerie, pour les utilisateurs.

**3.3.1.2.1. POP (Post Office Protocol, ou Protocole de Bureau de Poste)** :

Le serveur POP est le serveur « entrant », recevant les messages et redistribuant vers les comptes de messagerie gérés par le serveur.

- Que fait le serveur POP :

- il gère l'authentification du titulaire d'un compte de messagerie, avec l'identifiant et le mot de passe
- il authentifie les destinataires, met les messages en attente
- Avec un serveur POP, les messages reçus peuvent : être envoyés en bloc sur votre ordinateur : on peut les lire "hors ligne" conservés ou effacés sur le serveur

**3.3.1.2.2. IMAP (Internet Message Access Protocol)** :

Autre protocole entrant, alternative de POP, et offrant plus de fonctionnalités.

- Que fait le serveur IMAP ?

- Sur un serveur IMAP, les messages restent toujours sur le serveur de courrier : seuls les en-têtes sont téléchargés en local, puis les messages eux-mêmes.
- Il gère plusieurs accès simultanés : possibilité de récupérer son courrier à partir de plusieurs postes.
- comme les messages restent sur le serveur, leur gestion peut se faire également sur le serveur : possibilité de tri, de classement, à partir du lieu de travail ou de chez soi, etc.
- assure une mise à jour depuis la dernière connexion

Mais IMAP demande plus de ressources côté serveur (pour l'accès simultané, le tri...) et peut augmenter le temps de téléchargement côté utilisateur (pour les connexions par Modem) Tous les FAI ne prennent pas en charge les serveurs IMAP.

### 3.4. Les annuaires

#### 3.4.1. Définition :

Un annuaire électronique est un système de stockage de données, dérivé des bases de données hiérarchisées, permettant en particulier de conserver les données pérennes, c'est-à-dire les données n'étant que peu mises à jour (historiquement, sur une base annuelle, d'où le nom) :

- il est optimisé pour la lecture; l'ajout et la modification de données peuvent être coûteuses;
- il fournit des fonctions de recherches plus avancées;
- les données sont stockées sur un modèle distribué et des techniques de réplications sont possibles, ce qui facilite un passage à l'échelle efficace.
- la structure des données stockées, appelée schéma, peut être étendue en fonction de besoins locaux;
- il est basé sur des standards établis qui assurent l'interopérabilité entre plusieurs implémentations sur plusieurs supports (notamment OS)

Ainsi le but d'un annuaire électronique est approximativement le même que celui d'un annuaire papier, si ce n'est qu'il offre une grande panoplie de possibilités que les annuaires papier ne sauraient donner.

L'utilisation d'annuaire ne se limite pas à la recherche de personnes ou de ressources. En effet, un annuaire peut servir à :

- constituer un carnet d'adresse
- authentifier des utilisateurs (grâce à un mot de passe)
- définir les droits de chaque utilisateur
- recenser des informations sur un parc matériel (ordinateurs, serveurs, leurs adresses IP et adresses MAC, …)
- décrire les applications disponibles

### 3.5. DHCP

#### 3.5.1. Définition

DHCP signifie Dynamics Host Configuration Protocol. Il s'agit d'un protocole qui permet à un ordinateur qui se connecte sur un réseau local d'obtenir dynamiquement et automatiquement sa configuration IP. Le but principal étant la simplification de l'administration d'un réseau. On voit généralement le protocole DHCP comme distribuant des adresses IP, mais il a été conçu au départ comme complément au protocole BOOTP (Bootstrap Protocol) qui est utilisé par exemple lorsque l'on installe une machine à travers un réseau (on peut effectivement installer complètement un ordinateur, et c'est beaucoup plus rapide que de le faire en à la main). Cette dernière possibilité est très intéressante pour la maintenance de gros parcs machines. Les versions actuelles des serveurs DHCP fonctionne pour IPv4 (adresses IP sur 4 octets). Une spécification pour IPv6 (adresses IP sur 16 octets) est en cours de développement par l'IETF.

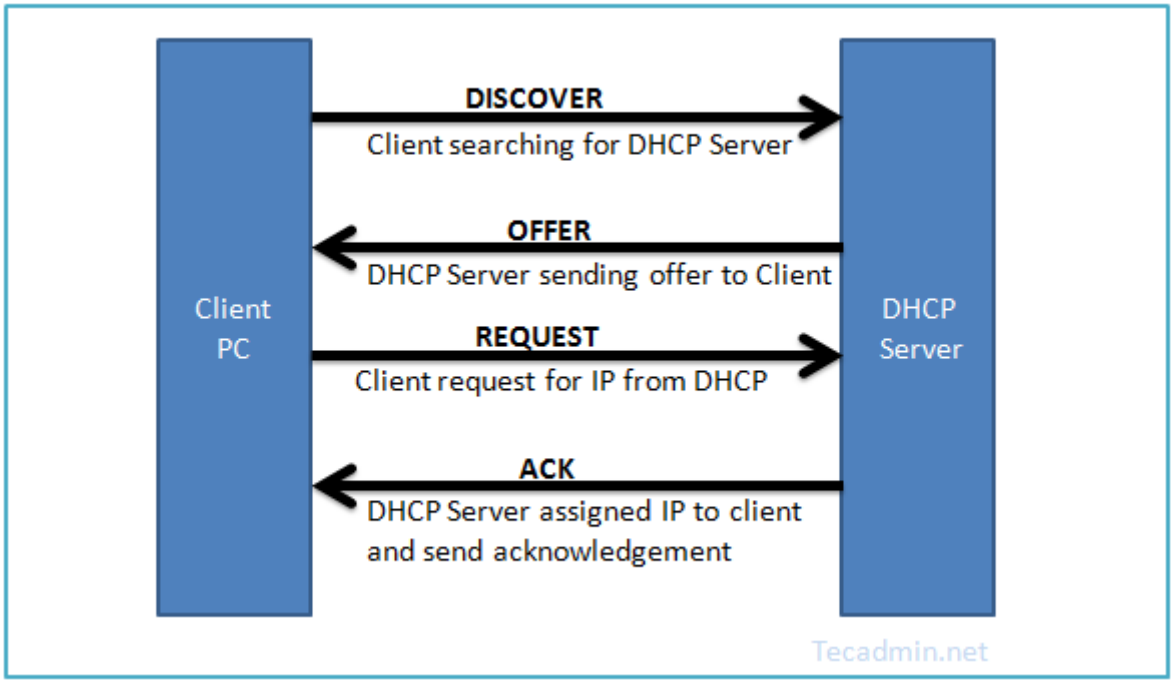


**Figure 2.8: Signifie Dynamics Host Configuration Protocol.**

#### 3.5.2. Fonctionnement :

DHCP fonctionne sur le modèle client-serveur : un serveur, qui détient la politique d'attribution des configurations IP, envoie une configuration donnée pour une durée donnée à un client donné (typiquement, une machine qui vient de démarrer). Le serveur va servir de base pour toutes les requêtes DHCP (il les reçoit et y répond), aussi doit-il avoir une configuration IP fixe. Dans un réseau, on peut donc n'avoir qu'une seule machine avec adresse IP fixe : le serveur DHCP.

Le protocole DHCP s'appuie entièrement sur BOOTP : il en reprend le mécanisme de base (ordre des requêtes, mais aussi le format des messages). DHCP est une extension de BOOTP. Quand une machine vient de démarrer, elle n'a pas de configuration réseau (même pas de configuration par défaut), et pourtant, elle doit arriver à émettre un message sur le réseau pour qu'on lui donne une vraie configuration. La technique utilisée est le broadcast : pour trouver et dialoguer avec un serveur DHCP, la machine va simplement émettre un paquet spécial, dit de broadcast, sur l'adresse IP 255.255.255.255 et sur le réseau local. Ce paquet particulier va être reçu par toutes les machines connectées au réseau (particularité du broadcast). Lorsque le serveur DHCP reçoit ce paquet, il répond par un autre paquet de broadcast contenant toutes les informations requises pour la configuration. Si le client accepte la configuration, il renvoi un paquet pour informer le serveur qu'il garde les paramètres, sinon, il fait une nouvelle demande. Les choses se passent de la même façon si le client a déjà une adresse IP (négociation et validation de la configuration), sauf que le dialogue ne s'établit plus avec du broadcast.

### 3.5.3. Les requêtes et les messages DHCP :

On pourrait croire qu'un seul aller-retour peut suffire à la bonne marche du protocole. En fait, il existe plusieurs messages DHCP qui permettent de compléter une configuration, la renouveler... Ces messages sont susceptibles d'être émis soit par le client pour le ou les serveurs, soit par le serveur vers un client :

| Nom | Description |
|---|---|
| DHCPDISCOVER | Pour localiser les serveurs DHCP disponibles et demander configuration |
| DHCPPOFFER | Réponse de serveur a un message DHCPDISCOVER .qui contient les premier paramètres |
| DHCPREQUEST | Requête diverse du client pour par exemple prolonger son bail |
| DHCPDECLINE | Le client annonce au serveur que l'adresse est déjà utilisée |
| DHCPCK | Réponse du serveur qui contient des paramètres et l'adresse IP du client |
| DHCPNAK | Réponse du serveur pour signaler au le client que son bail est échu ou si le client annonce une mauvaise configuration réseau |
| DHCPRELEASE | Le client libère son adresse IP |
| DHCPLNFORM | Le client demande des paramètres locaux, il déjà son adresse IP |

**Tableau 2.1** : **Les requêtes et les messages DHCP**

La valeur entre parenthèses est utilisée pour identifier ces requêtes dans les messages DHCP. Voir les options DHCP.

La première requête émise par le client est un message DHCPDISCOVER. Le serveur répond par un DHCPOFFER, en particulier pour soumettre une adresse IP au client. Le client établit sa configuration, demande éventuellement d'autres paramètres, puis fait un DHCPREQUEST pour valider son adresse IP. Le serveur répond simplement par un DHCPACK avec l'adresse IP pour confirmation de l'attribution. Normalement, c'est suffisant pour qu'un client obtienne une configuration réseau efficace, mais cela peut être plus ou moins long selon que le client accepte ou non l'adresse IP ou demande des infos complémentaires.

**Conclusion :**

Les services réseaux simplifient grandement l'administration et la gestion du réseau dans ce chapitre, nous avons exposé le fonctionnement de système client/serveur, ses déférents types d'architecture et la déférence entre ces types puis on a mentionné quelques serveurs réseaux et leur fonctionnalité

# Chapitre03 :

# Le serveur PROXY

## 1. Introduction :

Un serveur proxy est utilisé sur Internet comme mémoire cache, dans le sens tampon et caché. Son utilisation principale est la navigation en html mais il peut être implanté en HTTP ou FTP. L'internaute n'est pas directement connecté sur le site via un serveur DNS mais sur le serveur proxy. C'est lui qui se connecte sur le site et récupère les données avant de les renvoyer à l'internaute.

La première utilisation d'un proxy est interne à un réseau et sert de barrière protectrice. Elle est implantée depuis Windows server 2003, des solutions logicielles sont également possibles comme Wingate. Deux cartes Ethernet sont utilisées, une connectée sur Internet, l'autre reliée au réseau local. Ils servent aussi de routeurs logiciels. Cette solution permet d'interdire l'accès à certains sites pour des utilisateurs, le plus souvent paramétrables pour chaque utilisateur par l'administrateur du serveur. De plus, elle permet d'enregistrer les sites visités par chaque utilisateur. C'est une solution de sécurité utilisée

Dans ce chapitre on va définir dans une première section le concept du proxy ces fonctionnalités, les avantages et les inconvénients et puis on va montrer comment modifier les paramètres Proxy, et enfin on va représenter le serveur Squid.

## I. Serveur Proxy

### 1. Définition :

Un proxy (serveur mandataire) est un serveur situé entre un réseau privé et Internet.

Constituant une protection pour le réseau d'une entreprise, il peut également faire office de cache. Dans ce dernier cas, il enregistre les pages Web transférées par les utilisateurs pour les délivrer sans qu'il soit nécessaire de se connecter sur le serveur initial. Ainsi, lorsqu'un utilisateur se connecte à Internet à l'aide d'une application cliente configurée pour utiliser un serveur proxy, celle-ci va se connecter en premier lieu au serveur proxy et lui donner sa requête. Le serveur proxy va alors se connecter au serveur que l'application cliente cherche à joindre et lui transmettre la requête. Le serveur va ensuite donner sa réponse au proxy, qui va à son tour la transmettre à l'application cliente. [8]

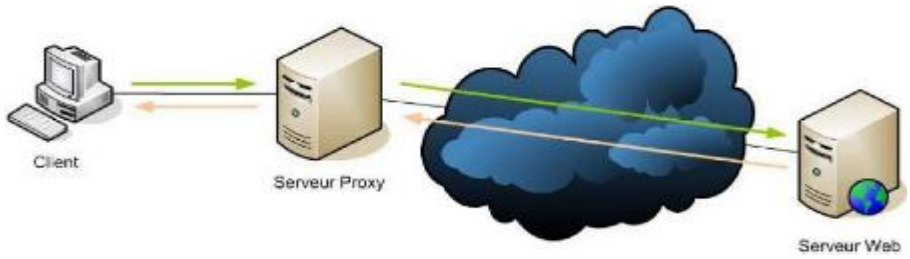


**Figure 3.1: serveur proxy.**

## 2. Les fonctionnalités d'un serveur Proxy :

**2.1. La fonction de cache :** dans ce mode, les pages visitées passent par le proxy qui enregistre le contenu. A chaque nouvel appel d'une page connue, le serveur ne demandera plus le contenu au site mais renverra le contenu enregistré dans sa propre mémoire. Cette solution semble plus rapide mais uniquement si la page est déjà enregistrée. Dans le cas contraire, le transfert est ralentit. Autre défaut, les pages sécurisées (protocole httpS) et les contenus dynamiques (réponses à une requête spécifique) ne peuvent pas utiliser le proxy.

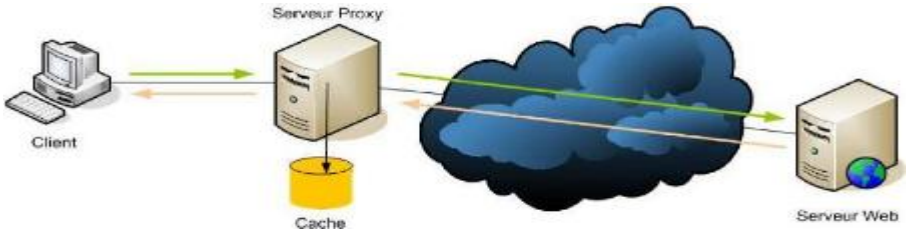


**Figure 3.2: serveur proxy avec service cache**

**2.2. La fonction d'enregistrement** : le proxy enregistre les informations qui transitent par lui: adresse IP du client, date et heure de la communication, adresse des pages consultées, ... Quelques Firewalls hardware le permettent également.

**2.3. La fonction de filtrage :** D'autre part, grâce à l'utilisation d'un proxy, il est possible d'assurer un suivi des connexions (en anglais logging ou *tracking*) via la constitution de journaux d'activité (logs) en enregistrant systématiquement les requêtes des utilisateurs lors de leurs demandes de connexion à Internet.

Il est ainsi possible de filtrer les connexions à Internet en analysant d'une part les requêtes des clients, d'autre part les réponses des serveurs. Lorsque le filtrage est réalisé en comparant la requête du client à une liste de requêtes autorisées, on parle de liste blanche, lorsqu'il s'agit d'une liste de sites interdits on parle de liste noire. Enfin l'analyse des

réponses des serveurs conformément à une liste de critères (mots-clés, ...) est appelé filtrage de contenu.

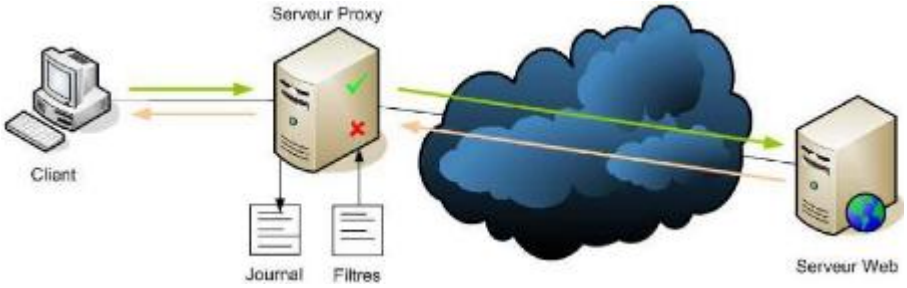


**Figure 3.3 : serveur proxy avec service de filtre et journal**

**2.4. L'authentification :** Le fait d'authentifier les utilisateurs va permettre une surveillance accrue de notre réseau et surtout, la possibilité de gérer des règles quels que soient le poste de travail et le navigateur utilisé.

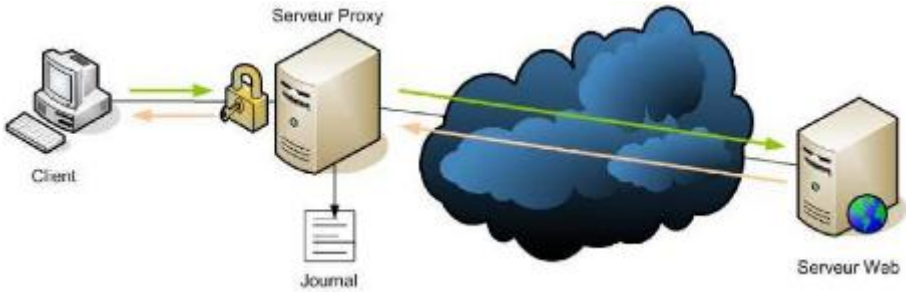


**Figure 3.4 : serveur proxy avec service d'authentification**

**2.5. La fonction de reverse-proxy :** On appelle reverse-proxy(en français le terme de relais inverse est parfois employé) un serveur proxy-cache, c'est-à-dire un serveur proxy permettant non pas aux utilisateurs d'accéder au réseau Internet, mais aux utilisateurs d'internet d'accéder indirectement à un certains serveurs internes.

## 3. Les avantages du proxy

- Les avantages sont nombreux:
- le surf anonyme : Ce n'est pas votre adresse qui est vue sur les sites, mais l'adresse du proxy. Vous êtes ainsi « quasiment anonyme » ou « complètement anonyme ».
- la protection de votre ordinateur : Ce n'est pas vous qui êtes en première ligne sur Internet, vous êtes donc mieux protégé.

- le masquage de votre lieu de connexion : Le proxy peut être dans un pays différent du votre. Lorsqu'il se connecte à un site, c'est la géo localisation du proxy qui est vu, pas la votre. Cela peut être utile sur certains sites qui filtrent les connexions suivants les lieux d'où elles proviennent.
- le filtrage : comme toutes les requêtes et les réponses passent par le proxy, il est possible de filtrer ce que l'on autorise à sortir ou à entrer, c'est le cas dans de nombreuses entreprises.
- peut accélérer la navigation si le proxy est plus "près" de toi que la plupart des sites (proxy de ton FAI)
- fait économiser beaucoup de bande passante à ton FAI (c'est généralement ce qui le motive d'où parfois des proxys transparent obligatoire)

**4. Les inconvénients du proxy :**

- Qui dit avantages, dit également inconvénients :
- Le proxy, c'est lui qui fait l'intermédiaire entre vous et le web, donc il voit et peut enregistrer tout ce qui circule entre votre ordinateur et le web, cela peut être risqué
- Si vous utilisez un proxy, il doit être irréprochable car lorsque vous vous connectez à votre banque, votre proxy pourrait très bien enregistrer vos codes (même si ceux-ci sont émis dans des flux https) ! Il faut donc utiliser un proxy donc vous êtes sûr, ou alors ne pas l'utiliser ;
- Un autre inconvénient des proxys est la technologie utilisée sur les sites web. En effet, certains sites peuvent utiliser des technologies de connexion directes entre votre ordinateur et le serveur Web, dans ce cas, il peut être impossible de se connecter à ce genre de sites si vous êtes caché derrière un proxy. Vous devrez là encore mettre le site concerné en exception proxy ;
- Si le serveur proxy est très sollicité, il peut éventuellement mettre plus longtemps à répondre, donc il est possible que le surf à travers un proxy soit un peu plus lent que le surf direct sur Internet ;

**5. Les exceptions proxy**

Pour certains sites, il soit essentiel de ne pas utiliser le proxy : il faut alors utiliser des exceptions proxy.

Une exception proxy est une adresse IP ou une URL pour laquelle votre navigateur ne va pas utiliser le proxy, mais ce connecter directement à Internet.

-Voici un exemple d'exception proxy pour 2 banques et un réseau IP:

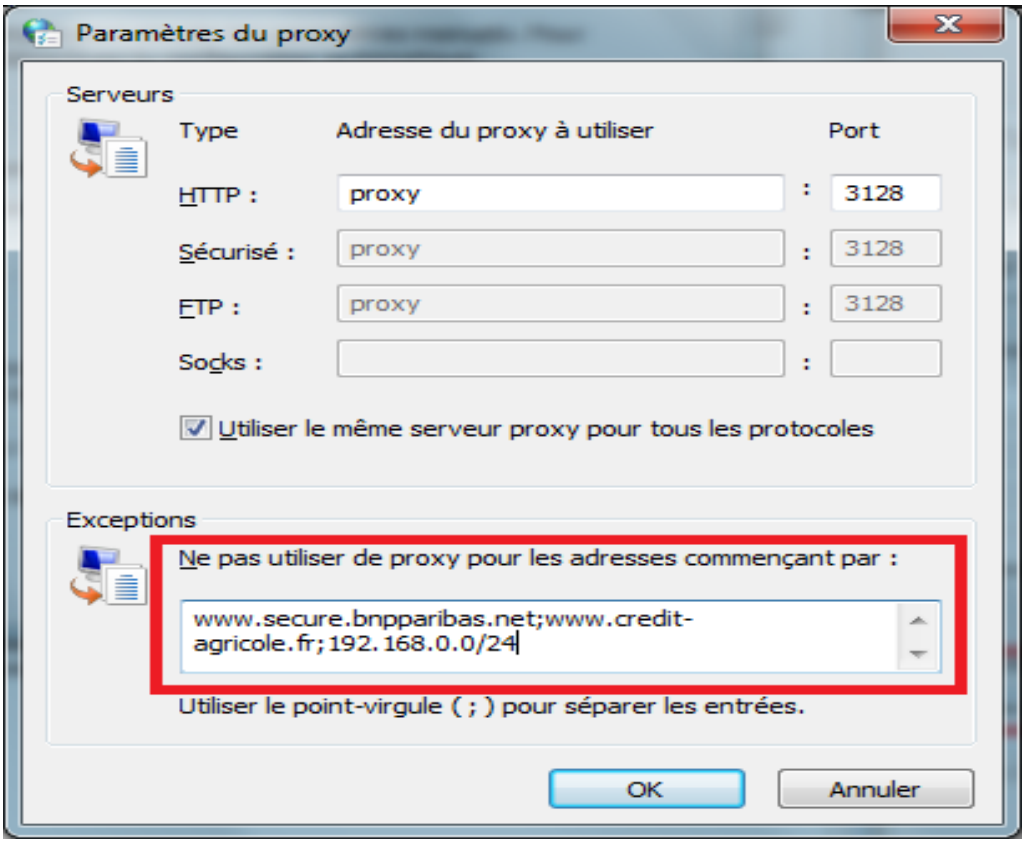


**Figure 3.5: comment utiliser des exceptions proxy**

### 6. Modifier mes paramètres Proxy :

Tout d'abord, il faut accéder aux Options Internet suivant les navigateurs :

- Internet Explorer
- Google Chrome
- Mozilla Firefox

#### 6.1. Paramétrez un proxy dans Google chrome :

Cliquez sur le menu Google Chrome. Il se trouve en haut à droite du navigateur et ressemble à trois traits horizontaux.

Choisissez « Paramètres ». Un nouvel onglet s'ouvre. Cliquez sur le lien en bas de page marqué « Afficher les paramètres avancés … »

Cliquez sur le bouton « Modifier les paramètres du proxy ». Il se trouve dans « Réseau ». La boîte de dialogue « Propriétés Internet » s'affiche.

Nota bene : les paramètres de proxy de Chrome étant liés au système d'exploitation de votre ordinateur, toute modification de ceux-ci affectera tous vos programmes qui travaillent avec internet. Si vous ne voulez pas que cela se produise, téléchargez et installez une extension Chrome du genre Proxy Switch Sharp ou Proxy Helper. Cliquez sur le bouton « Paramètres réseau ». Une nouvelle fenêtre s'ouvre. Cochez la case dans la partie « Serveur proxy » et décochez la case « Détecter automatiquement les paramètres de connexion ». Renseignez les informations nécessaires. Entrez l'adresse du serveur (IP ou de domaine) ainsi que le numéro de port auquel vous voulez vous connecter. Cliquez sur « Ok ». Lorsque vous en avez fini avec votre proxy et que vous désirez revenir à une connexion classique, cliquez à nouveau sur le bouton « Paramètres réseau », cochez la case « Détecter automatiquement les paramètres de connexion » et décochez la case dans la partie « Serveur proxy »

### 6.2. Paramétrez un proxy dans Firefox :

Cliquez sur le menu de Firefox. Ce dernier est situé dans le coin supérieur gauche de Firefox. Cliquez sur « Options ».

Sélectionnez « Avancé ». Cette option se trouve en haut de la fenêtre « Options » à l'extrême droite.

Sélectionnez l'onglet « Réseau ». Cliquez sur le bouton « Paramètres … » en haut de la page « Réseau », partie « Connexions ».

- Choisissez « Configuration manuelle du proxy ». Vous allez activer des champs-textes dans lesquels vous allez pouvoir entrer les informations du proxy.

  Renseignez les paramètres du proxy. Dans le champ « HTTP », entrez soit l'adresse IP soit le nom de domaine du serveur proxy et mettez éventuellement le numéro du port. Si vous avez besoin d'un autre proxy pour utiliser les protocoles FTP ou SSL, entrez les informations dans le champ qui se trouve en dessous. Sinon, cochez la case « Utiliser ce serveur proxy pour tous les protocoles ». Vous utiliserez le même proxy pour tous les types de connexion.

  Vous n'avez changé les paramètres proxy que sur votre Firefox. Vos autres navigateurs sont, eux, connectés directement à l'Internet.

- Appuyez sur « Ok » pour sauvegarder vos modifications. Il faut ensuite redémarrer Firefox pour que votre navigateur prenne en charge les changements.

Vous n'avez changé les paramètres proxy que sur votre Firefox. Vos autres navigateurs sont, eux, connectés directement à l'Internet.

- Appuyez sur « Ok » pour sauvegarder vos modifications. Il faut ensuite redémarrer Firefox pour que votre navigateur prenne en charge les changements.

**6.3 Paramétrez un proxy dans Internet explorer :**

Cliquez sur « Outils ». Selon les versions de votre navigateur, ce menu se trouve soit dans la barre des menus classique soit il apparaît en cliquant sur l'icône en forme de clé dans le coin supérieur droit.

Sélectionnez « Options Internet ». Cette option se trouve, dans tous les cas, en fin de menu. **:**
Cliquez sur l'onglet « Connexions ». Dans la fenêtre qui s'est ouverte, repérez la section « Paramètres du réseau local ». Cliquez sur « Paramètres réseau ». Une nouvelle fenêtre s'ouvre. Activez les paramètres de proxy. Cochez la case « Utiliser un serveur proxy … » dans la partie « Serveur proxy » et décochez la case « Configuration automatique ». Renseignez les paramètres du proxy. Dans le champ « Adresse », entrez soit l'adresse IP soit le nom de domaine du serveur proxy et mettez éventuellement le numéro du port. Cliquez sur « Ok » quand c'est fait. Il faut ensuite redémarrer Internet Explorer pour que votre navigateur prenne en charge les changements.

- N'hésitez pas à essayer plusieurs proxies, car ils sont souvent lents et de plus, un proxy affecte le trafic de vos autres connexions !
- Quand vous ne voulez plus de la connexion via proxy, reparamétrez votre connexion. Décochez la case « Utiliser un serveur proxy … » dans la partie « Serveur proxy » et recouchez la case « Configuration automatique »

## II. Présentation de Squid et Squidguard

1. **Squid** : est un proxy (serveur mandataire en français) cache sous linux. De ce fait il permet de partager un accès Internet entre plusieurs utilisateurs avec une seule connexion. Un serveur proxy propose également un mécanisme de cache des requêtes, qui permet d'accéder aux données en utilisant les ressources locales au lieu du web, réduisant les temps d'accès et la bande passante consommée, il est possible aussi d'effectuer des contrôles de sites. Enfin il

Permet de partager une connexion à Internet à l'aide SQUID. [8]

## 2. Les autres services de Squid :

### 2.1. Le cache :

Le principe de cache est assez simple, prenons par exemple le réseau de l'université de Tiaret .Ce dernier est composé d'une dizaine de postes équipés de cartes réseaux Ethernet à 100Mb/s. Ce réseau est relié via un routeur à internet en utilisant une ligne spécialisée. Nous lui avons affecté une plage d'adresses IP non routable (172.16.17.x). Avant la mise en place de notre proxy, toutes les personnes qui allaient sur internet, récupèrent par l'intermédiaire de leurs navigateurs les objets (html, images, ...) qu'ils avaient consultés, mais tous ces objets qui ont été téléchargé n'étaient pas accessible par les autres navigateurs (c'est à dire les autres utilisateurs du réseau), ce qui implique que les autres utilisateurs devraient eux aussi récupérer leurs propres objets. Ce qui réduit la consommation de bande sur notre ligne spécialisée et produit un gain de temps pour les utilisateurs Internet. Bien sur le serveur proxy vérifie, avant de donner les objets qui possèdent sur son disque, s'il n'y a pas de version plus récente de l'objet demandé sur Internet. S'il y a une version plus récente il va la télécharger sinon il donne à l'utilisateur l'objet qui avait enregistré.

#### 2.1.1. Le support des protocoles liés aux caches :

SQUID supporte beaucoup de protocoles liés aux caches: ICP, HTCP, CARP, Cache Digests, WCCP. Ces protocoles permettent les échanges entre les différents caches et ainsi favoriser les flux de données les plus proche du site plutôt que de solliciter un serveur qui sera peut être plus éloigné et plus lent à répondre qu'un serveur cache.

### 2.2. L'authentification :

SQUID permet d'authentifier les clients avant qu'ils accèdent à la ressource qu'ils demandent. L'authentification s'effectue pour les modes proxy et "http d'Accelerator". Il devient alors de plus en plus avantageux d'utiliser SQUID en frontale d'un serveur web car ce dernier n'aura à assumer que les rôles primordiaux de service web dynamique. SQUID supporte beaucoup de protocoles liés à l'authentification (Basic, Digest, LDAP, NTLM, Radius, MySQL). L'authentification est réalisée via des codes exécutables externes à SQUID, chaque protocole d'authentification ayant son propre code exécutable. Ces programmes d'authentification ont un format d'utilisation très simple : ils lisent sur STDIN les informations d'authentification sous la forme "login, Mot De Passe" et retourne sur STDOUT "OK" ou "ERR" en fonction des informations introduites (correctes ou non).

### 2.3. Le filtrage :

SQUID offre la possibilité de filtrer les requêtes des clients. Ainsi, il est possible de restreindre l'accès aux ressources en fonction de différents paramètres. Voici une liste de paramètre pouvant intervenir dans le rejet d'une requête répondant à l'un des critères :

- L'URL contient un mot interdit.
- L'adresse IP source/destination est interdite.
- Le domaine de source/destination est interdit ou contient un mot interdit.
- La date de la demande. Par exemple, SQUID peut interdire l'accès à Internet durant
- certaines heures (comme le soir entre 20h et 6h du matin).
- Le port de destination.
- Le protocole utilisé peut permettre de bloquer les transferts FTP par exemple.
- La méthode utilisée peut permettre d'empêcher les méthodes HTTP comme POST par exemple.
- Le type du navigateur utilisé peut permettre d'empêcher l'utilisation d'IE par exemple.
- Ce filtrage est basé sur des ACL. SQUID n'est capable de filtrer que les requêtes de ses clients, pas le contenu de ce qu'il relaye à ceux-ci (bien qu'un proxy filtrant le contenu de page revienne à multiplier la charge d'administration par le nombre d'interdiction malencontreuse). [8]

## 3. Les protocoles de Squid :

- **HTTP/1.0** (le serveur est compatible avec un peu plus de 80% des nouvelles fonctionnalités HTTP/1.1).
- **FTP** via HTTP pour le chargement et le téléchargement de fichier. La modification et la suppression à distance ne sont pas supportées.
- **ICP** (Inter Cache Protocol), utilisé pour communiquer avec d'autres serveurs cache.
- **SSL** pour les connexions sécurisées.
- Selon les besoins, plusieurs serveurs cache peuvent être mis en place dans un réseau. Ces différents serveurs peuvent être amenés à échanger des données que certains auraient en cache mais pas d'autres. Pour cela, plusieurs protocoles peuvent être utilisés. Voici ceux qui sont supportés par Squid pour les échanges inter-cache :

- **HTTP** : le protocole Web est ici utilisé par un serveur pour récupérer un objet depuis un autre serveur cache.
- **ICP** : ce protocole permet de questionner un serveur cache voisin pour savoir s'il possède la ressource recherchée.
- **Cache Digests** : ce protocole permet de questionner un serveur en utilisant une empreinte de l'objet recherché. Le serveur distant regarde alors dans son index s'il possède l'empreinte de la ressource demandée et répond positivement le cas échéant.
- **Simple Network Management Protocol (SNMP)** : ce protocole est plutôt destiné aux administrateurs et permet d'accéder à différentes informations du serveur.

**4. Limitations :**

Squid n'est pas un Firewall. Il peut limiter les possibilités des clients mais il ne protège pas l'accès aux personnes extérieures au réseau. La gestion du pare feu sous Linux est réalisée par ipchains pour les noyaux 2.2 ou iptables pour les noyaux 2.4. Squid ne supporte pas tous les protocoles. Il est ainsi incapable de gérer les protocoles de news, real audio et de vidéo conférences. Il est nécessaire d'utiliser d'autres caches applicatifs. [9]

**1. Sécurité :**

Le système revient clairement à charger dans le navigateur un script qui va analyser les URL demandées par le client, et les transmettre, suivant le cas, à un serveur mandataire, et ce, de façon invisible pour l'utilisateur. En d'autres termes, le trafic HTTP (et https) peut être dérouté sur un serveur intermédiaire, sans que l'utilisateur en ait connaissance. Les questions que l'on devrait se poser seront les suivantes :

- si nous sommes sur un réseau « de confiance » et que l'administrateur a clairement annoncé ses intentions, tout va encore à peu près pour le mieux, si l'on admet que le réseau ne peut être compromis
- si nous sommes sur un réseau dont nous ne savons rien, nous ne savons pas par où nous passons (ce peut être le cas aussi avec un proxy transparent, mais ce dernier est facilement repérable, par exemple avec un tcptraceroute). Dans ce cas un indélicat pourrait facilement nous espionner ;
- sommes-nous certains que nos navigateurs sont assez sécurisés pour ne pas accepter n'importe quoi comme fonction FindProxyForURL ?

Le cas le plus intéressant serait sans doute sur un réseau Wi-Fi non sécurisé, ouvert à tous. [11]

2. **SquidGuard**: propose un filtrage puissant d'accès au web, en fonction des groupes d'utilisateurs, des listes de domaines et d'URL, des plages horaires,…

**2.1. Fonctionnement**

SquidGuard propose un filtrage puissant d'accès au web, en fonction :

- de groupes d'utilisateurs, définis de diverses manières. Ici, nous nous baserons sur des IPs ou des groupes d'IPs, mais il est possible d'utiliser l'authentification des utilisateurs mise en place sur Squid,
- de listes de domaines et d'URI qui serviront à définir soit des cibles autorisées, soit des cibles interdites,
- de listes de domaines et d'URI qui ne serviront qu'à interdire l'accès aux cibles spécifiées,
- de plages horaires pendant lesquelles l'accès sera autorisé ou interdit

**2.2. SQUID avec SquidGuard :**

SquidGuard est un module pour le serveur proxy SQUID. Ce module ajoute des fonctionnalités plus avancées en matière de filtrage basé sur une liste noire de sites web à bloquer SquidGuard est ainsi un programme redirecteur distribué sous licence GPL, c'est-à-dire que toutes les trames HTTP seront redirigées vers SquidGuard pour être analysées puis filtrées. Il est nécessaire pour cela d'indiquer à Squid que les trames devront transitées par SquidGuard avant de passer dans le cache du Proxy. La fonction de filtre de SQUID est alors optimisée dans le programme SquidGuard ce qui lui permet d'analyser des listes d'URLS en un temps record. Une fois lancé, SquidGuard apparaîtra comme un processus fils de SQUID. [10]

## III. NAT :

1. **Définition :**(Network Adress Translation soit « traduction d'adresse réseau») lorsqu'il fait correspondre les adresses IP internes privées (non-uniques et souvent non routables) d'un intranet à un ensemble d'adresses externes publiques (uniques et routables). Ce mécanisme permet notamment de faire correspondre une seule adresse externe publique visible sur Internet à toutes les adresses d'un réseau privé, et pallie ainsi l'épuisement des adresses IPv4.
2. **Fonctionnement de NAT**

Lorsqu'un client du réseau interne entre en communication avec une machine sur Internet, il envoie des paquets IP à destination de cette machine. Ces paquets contiennent toutes les

informations sur leur émetteur et leur destinataire nécessaires à leur bon acheminement. La NAT prend en compte les informations suivantes :

- L'adresse IP source (192.168.1.x par exemple)
- Le port TCP ou UDP source (2132 par exemple)

Lorsque les paquets passent à travers la machine-passerelle NAT, ils sont modifiés de telle façon à ce qu'ils semblent provenir de la passerelle NAT. Cette passerelle enregistrera les modifications effectuées dans sa table d'état afin de :

1. inverser les modifications pour les paquets de retour

2. s'assurer que les paquets de retour sont autorisés à traverser le pare-feu et ne sont pas bloqués Par exemple, les modifications suivantes peuvent être effectuées :

- adresse IP source : remplacée par l'adresse externe de la passerelle (24.5.0.5 par exemple)
- Port source : remplacé par un port non utilisé sur la passerelle et choisi de manière aléatoire (53136 par exemple).

Aucune des deux extrémités de la communication ne se rend compte de ces modifications. Pour la machine interne, le système qui effectue la NAT est simplement une passerelle Internet. Pour l'hôte sur Internet, les paquets semblent provenir directement du système de NAT; il ne sait même pas que la machine interne existe. Lorsque l'hôte sur Internet répond aux paquets de la machine interne, ils seront envoyés à l'adresse IP externe de la passerelle NAT (24.5.0.5) sur le port de traduction (53136). Dès réception de ces paquets, la passerelle NAT cherchera dans sa table d'état si ces paquets de retour correspondent à une connexion déjà établie. Une correspondance unique sera trouvée, basée sur la combinaison IP/port qui permet à la passerelle de voir que les paquets appartiennent à une connexion initiée par la machine interne192.168.1.10. La passerelle effectuera alors les modifications inverses à celles effectuées sur les paquets sortants puis enverra ces paquets à la machine interne. [12]

### 3. NAT pour proxy

Le routeur NAT (Network Address Translation) , fait réellement du routage de paquets IP, c'est à dire qu'il transmet bien les paquets reçus par un ordinateur du LAN vers Internet (et inversement), contrairement à un serveur proxy qui ouvre une seconde connexion. Les programmes des machines du réseau local ne sont pas configurés de manière particulière. Le configuration se limite a déclarer l'adresse de la machine NAT comme routeur par default et DNS.

**Conclusion :**

Le serveur proxy importe d'installer. Le serveur proxy peut jouer un rôle de cache efficace pour les ISP. Le firewall, par contre, est un dispositif de sécurité recommandable à toute entreprise qui veut protéger son réseau local.

Comme nous avons pu le deviner, le serveur proxy ne fait pas partie des dispositifs informatiques que j'adore le plus.

# Chapitre 04 :

# L'Architecture Proposée

## 1. Introduction

Le serveur Proxy le plus utilisé par les fournisseurs d'accès et les administrateurs de réseaux locaux s'appelle Squid. Il fonctionne sous Unix. Squid est un logiciel permettant la réalisation d'un cache pour les clients web. Squid peut aussi jouer le rôle de filtre http.

Dans ce chapitre, nous dressons la liste des outils choisis pour la réalisation de notre système, ainsi la configuration de notre proxy.

## 2. Technologies utilisées pour le développement :

Nous présentons dans cette section les outils utilisés pour le mise en ouvre de notre Proxy (ubuntu, squid, squidGuard, webmin….ect), Avant d'entamer la phase réalisation, nous allons présenter les technologies et les outils utilisés durant cette phase.

### 2.1. Ubuntu :

(prononciation [u.bun.tu], « ou-boun-tou » en français) est un système d'exploitation open source basé sur la distribution Linux Debian. Son nom provient d'un ancien mot bantou qui signifie « je suis ce que je suis grâce à ce que nous sommes tous. Dans le même ordre d'esprit, les utilisateurs sont encouragés à étudier son fonctionnement, le modifier, l'améliorer et enfin de le redistribuer.

### 2.2. Squid :

Squid est un serveur proxy/cache libre très connu du monde Open Source.

Ce serveur est très complet et propose une multitude d'options et de services qui lui ont permis d'être très largement adopté par les professionnels mais aussi dans un grand nombre d'école ou administrations travaillant avec systèmes de type Unix.

Squid est capable de relayer les protocoles HTTP, FTP, SSL et Gopher. Le protocole Socks, lui, n'est pas supporté par Squid actuellement.

### 2.3. Webmin :

Webmin est une interface web, sous licence BSD, qui permet d'administrer simplement un serveur UNIX ou Linux à distance via n'importe quel navigateur web.

### 2.4. squidGuard :

SquidGuard est un plugin qui se greffe sur un serveur Squid. Il permet la gestion plus poussée des ACL par le biais notamment de la gestion de Blacklists compilées. SquidGuard permet

également de gérer des accès horaires, des conditions plus poussées et la modification de la page par une autre page.

### *3.* Configuration matérielle et logicielle

Avant d'installer et d'utiliser SQUID, il est recommandé de bien choisir son matériel et le système d'exploitation du serveur. Durant son lancement, SQUID a besoin de faire beaucoup d'actions, notamment en rapport avec les caches, qui seront consommateurs de CPU. Ainsi, le matériel minimum recommandé est un Pentium III 550MHz, la RAM est un composant vital pour SQUID car il réside en mémoire RAM et y stocke ses composants et les objets les plus demandés par les clients.

Il faut ainsi noter que le choix de la configuration matérielle du serveur est important. Le choix d'une bonne carte réseau est nécessaire parce qu'une carte réseau sous dimensionnée par rapport au réseau sur lequel est branché le Proxy provoquerait très rapidement un engorgement important et ralentira les demandes des clients.

#### 3.1. Installation manuelle de Squid3 :

Avant de commencer, il faut au préalable télécharger la dernière version stable de SQUID. On peut l'acquérir sur le site officiel de SQUID qui est http://www.squid-cache.org. La dernière version actuelle stable est la version 3.5.4 Une fois le logiciel au format tar.gz (c'est à dire compressé sous ce format) téléchargé, on pourra alors l'installer et on suppose que le fichier décompressé sera placé dans le répertoire */usr/local/src*. On le décompresse via la commande :

**#tar -zxvf squid-3.5.4.tar.gz --directory=/usr/local/src**

On doit obtenir ce répertoire : **/usr/local/src/squid3.**

Il convient maintenant de compiler SQUID. On se place dans le répertoire de SQUID :

**#cd /usr/local/src/squid3**

On passe ensuite à la configuration des options de compilation. Squid sera par défaut installé dans le répertoire **/usr/local/squid3** mais on peut utiliser un autre répertoire via l'option --prefix. Si on souhaite avoir les messages d'erreurs en français, on ajoute l'option --enable-err-language=French mais on peut aussi faire cela au niveau de la configuration de SQUID. La commande pour la compilation est alors :

**#./configure --prefix=/usr/local/squid3--enable-errlanguage=French**

Si tout c'est bien passé, il ne reste plus qu'à compiler avec la commande :

**#make all**

Puis procéder à l'installation avec la commande :

**#make install**

Le fichier INSTALL situé dans la racine du répertoire **/usr/local/src/squid3** reprend en partie ce qui vient d'être expliqué.

A la fin de l'installation, les répertoires suivants seront créés :

**/usr/local/squid3** : répertoire de base de SQUID

**/usr/local/squid3/etc** : répertoire contenant la configuration de SQUID **/usr/local/squid3/bin** : répertoire contenant les binaires et les scripts /usr/local/squid3/logs : répertoire contenant les logs

### 3.2. Installation automatique :

L'installation automatique de SQUID est relativement facile. Il suffit juste de disposer d'une connexion internet, de se connecter en tant que root sur le terminal de la machine et de taper la commande : **#apt-get install squid3**

## 4. Configuration de Squid:

Par défaut Squid est configuré et fonctionnel. Cependant, on peut apporter quelques modifications afin de l'optimiser ou mieux l'adapter à certains environnements.

Le fichier de configuration de Squid est **/etc/squid/squid.conf ou /etc/squid3/squid.conf.** Pour toute configuration, éditer donc ce fichier.

**Remarque :**

- Penser à effectuer une sauvegarde de ce fichier avant toute modification :

**sudo cp /etc/squid/squid.conf/etc/squid/squid.conf.bak**

Effectuer l'opération inverse pour restaurer le fichier.

- Après toute modification du squid.conf, redémarrer Squid :

**#sudo service squid3 restart**

Les lignes débutant par un '#' sont des commentaires. Les options non définies auront toujours une valeur par défaut. Toutes les options de ce fichier sont dispersées dans plusieurs parties distinctes. Nous détaillerons les options les plus intéressantes de ces parties. Notons que le fichier de base fourni avec les sources est très largement commenté et contient par défaut 4963 lignes. Pour plus de

renseignement et des paramètres plus poussés, il est donc préférable de consulter la documentation officielle.

- **acl mynetworks src**

Cette liste de contrôle d'accès introduite avec l'option acl désigne les adresses réseau utilisées dans l'infrastructure. Si on se réfère au plan d'adressage de cette infrastructure, on considère que le trafic Web est susceptible de provenir de toutes les classes d'adresses IP privées désignées dans le document standard RFC1918.

- L'option src indique que ce sont les adresses IP sources qui sont utilisées comme critère d'accès au service proxy.

- **http_access allow mynetworks**

- La directive http_access contrôle l'accès au service via le protocole HTTP. Dans le cas présent, il s'agit d'ouvrir l'accès aux réseaux de l'infrastructure. La configuration par défaut, telle que fournie lors de l'installation du paquet, interdit tout accès au service. Cette règle restreint donc l'accès aux seules adresses IP utilisées .

- **icp_access allow mynetworks**

- La directive icp_access joue le même rôle que la précédente pour le protocole ICP. L'Internet Cache Protocol est utilisé pour le dialogue entre services mandataires. Tout comme dans le cas précédent, on autorise le fonctionnement du protocole à partir des adresses IP de l'infrastructures.

- **http_port :** définit le port d'écoute de Squid pour les requêtes HTTP. Par défaut, le port sera 3128. Pour plus de sécurité, nous pouvons changer ce port par défaut et nous prenons par exemple le port 8080.

http_port 8080

- **icp_port :** définit le port d'émission et d'écoute des requêtes ICP. Par défaut, le port est 3130. Ceci nous permet de communiquer avec des proxy parents ou voisins. La valeur 0 permet de ne pas utiliser ce service
- **visible_hostname :** Nous indiquons ici le nom de notre serveur proxy. Nous l'appelons firewall_esp
- **error_directory :** Nous indiquins via cette option le répertoire où se trouvent les messages d'erreurs destinés à l'utilisateur. Par défaut, on a : **/usr/share/squid/errors/**en et pour avoir ces

messages en français, on met le répertoire **/usr/share/squid/errors/fr** à la place. Les messages d'erreurs qui apparaitront sur le navigateur du client seront ainsi en français.

- **forwarded_for off:** Pour ne pas inclure l'adresse IP ou le nom du système dans les requêtes HTTP, dans la partie.
- **cache_mem** : correspond au cache mémoire, la valeur dépend de votre système. Par défaut SQUID utilise 256 MB. Cette taille doit être la plus grande possible afin d'améliorer les performances. Nous décidons d'utiliser cette taille
- **maximum_object_size_in_memory:** Définition de la taille maximum d'un objet conservé en mémoire cache. Les objets de taille supérieure ne sont pas conservés en mémoire vive.
- **maximum_object_size**: permet de spécifier la taille maximale des objets qui seront stockés dans le cache. La taille par défaut est de 4 MB.
- **minimum_object_size**: permet de spécifier la taille minimale des objets qui seront WintA iEQADMIKI I7 MIENISEU iéIIXWINiil lM %11FHIXLMJQIIIFD4Mj\ IaESIN ieE minimum pour les objets.
- **cache_replacement_policy heap GDSF**: Définition de la politique utilisée pour remplacer les objets stockés dans le cache disque lorsqu'il est nécessaire de trouver de la place libre. L'option retenue ici **Least Frequently Used with Dynamic Aging**. Elle correspond à un usage de type pile basé sur le taux d'utilisation d'un objet du cache combiné à son ancienneté.
- **cache_swap_low 87, cache_swap_high 90 :** Lancer la purge des trucs de cache lorsque l'utilisation du disque atteint 87%
- **Cache_dir Type RépertoireSource MOctets Level1 Level2** : permet de définir un cache. il est possible de définir plusieurs fois cette option afin de multiplier le nombre de cache. Détaillons les options de ce paramètre :
- **Type :** le type de stockage qui sera utilisé par Squid pour écrire ses données. Ce paramètre modifie le comportement de SQUID lors de l'écriture sur le disque, ses accès au disque, sa répartition de données sur l'espace qui lui est consacrée, etc.. Une liste des types supportés est détaillée dans le fichier de configuration de SQUID. Le type utilisé par défaut est ufs
  - **RépertoireSource :** le répertoire source de l'arborescence du cache. Le cache de SQUID se présente sous forme d'une arborescence dans laquelle les objets sont répartis
  - **MOctets :** la taille en Mo à réserver sur le disque pour ce cache ;
  - **Level1 :** le nombre de répertoire de niveau 1 dans l'arborescence du cache.
  - **Level2 :** le nombre de répertoire de niveau 2 dans l'arborescence du cache.

Par défaut, on a un cache de 100 MB se trouvant dans le répertoire **/var/spool/squid**. On peut ajouter si on veut un autre cache avec le répertoire de son choix avec la taille que l'on veut.

**cache_dir ufs /var/spool/squid3 2048 16 256**

- **access_log DirectoryPath/filename** : l'Hst le chemin vers le fichier de access.log qui contient tous les accès au cache. Par défaut c'est le fichier **/var/log/squid3/access.log**
- **cache_log DirectoryPath/filename :** idem que pour access_log avec le fichier **/var/log/squid3/cache.log** qui contient toutes les informations des activités de SQUID.

Pour ces deux options précédentes, si on ne souhaite pas avoir de log, on met le paramètre none à la place du nom de fichier.

- **url_rewrite_program /usr/bin/squidGuard -c /etc/squid/squidGuard.conf :** La directive url_rewrite_program désigne l'outil utilisé pour le filtrage des URL : squidGuard.
- **url_rewrite_children 20** : Définition du nombre de processus enfants générés pour répondre aux requêtes transmises par le démon squid. La valeur donnée doit permettre d'optimiser les temps de réponse en lançant suffisamment d'instances squidGuard tout en utilisant raisonnablement les ressources du système.
- **request_header_max_size 64 KB**

- Définition de la taille maximum d'un en-tête HTTP lors d'une requête.
- Le choix d'une taille de 32 Ko doit permettre de laisser passer le trafic «normal» sans entraver les accès. À voir en fonction d'utilisations particulières du protocole HTTP.

- **reply_header_max_size 64 KB**

- Définition de la taille maximum d'un en-tête de réponse HTTP. La valeur retenue est identique à celle du paramètre précédent.

- **pipeline_prefetch on**

- Directive utilisée pour doper les performances et se rapprocher d'une navigation Web sans service mandataire.

Cette section permet de configurer les différents timeouts de SQUID

- **connect_timeout** : le temps d'attente d'une réponse du serveur distant avant de retourner une page d'erreur de type "connection timeout" au client ;

- **request_timeout** : le temps d'attente de Squid entre deux requêtes HTTP avant de fermer la connexion ;
- **client_lifetime** : le temps maximum qu'un client a le droit de rester connecter à SQUID
- **pconn_timeout** : le temps d'attente de SQUID avant de fermer une connexion de type persistante ;
- **ident_timeout** : le temps maximum d'attente d'une authentification.

**5. Forcer le passage par SQUID :**

### 5.1. Proxy Transparent

Utiliser un proxy nécessite normalement qu'on configure manuellement les navigateurs de tous nos utilisateurs de manière à ce qu'ils interrogent toujours le proxy, quelle que soit la cible. Cette triche s'avère alors difficile si nous avons un très grand nombre d'utilisateurs et aussi nos utilisateurs ont la main sur ce paramétrage, et pourront probablement passer outre le proxy, s'ils le décident, contournant par le fait toutes vos stratégies. Il existe cependant un moyen d'éviter ceci en rendant le proxy transparent, ce qui veut dire que configurés ou non, les requêtes http passeront quand même par le proxy. Pour que Squid fonctionne comme un serveur mandataire transparent :

- il faut ajouter à la fin du port de Squid :

**http_port 3128 transparent**

- Puis exécuter la commande suivante pour indiquer à **iptables** qu'il doit rediriger les requêtes provenant du port 80 sur celui de Squid, le 3128 :

**# echo 1 > /proc/sys/net/ipv4/ip_forward**

**# iptables –A FORWARD –s 192.168.1.0/24 –j ACCEPT**

**#iptables –t nat –A PREROUTING –si eth0 –s 192.168.1.0/255.255.255.0 –p tcp –m tcp –dport 80 –j REDIRECT –to-port 3128**

**#iptables -t nat -A POSTROUTING -s 192.168.1.0/24 –d 0.0.0.0/0 –j MASQUERADE**

**6. Squid avec SquidGuard:**

SquidGuard est un logiciel URL de redirection, qui peut être utilisé pour le contrôle du contenu des sites Web les utilisateurs peuvent accéder. Il est écrit en tant que plug-in pour Squid et utilise des listes noires pour définir les sites pour lesquels l'accès est redirigé. SquidGuard doit être

installé sur un ordinateur UNIX ou GNU / Linux comme un ordinateur serveur. Une fois lancé, SquidGuard apparaîtra comme un processus fils de SQUID.

SquidGuard propose un filtrage puissant d'accès au web, en fonction :

- de groupes d'utilisateurs, définis de diverses manières. Ici, nous nous baserons sur des IPs ou des groupes d'IPs, mais il est possible d'utiliser l'authentification des utilisateurs mise en place sur Squid,
- de listes de domaines et d'URI qui serviront à définir soit des cibles autorisées, soit des cibles interdites,
- de listes de domaines et d'URI qui ne serviront qu'à interdire l'accès aux cibles spécifiées,
- de plages horaires pendant lesquelles l'accès sera autorisé ou interdit

### 7. Installation et configuration

L'installation se fait de manière automatique et simple avec la commande :

**#apt-get install squidguard**

L'installation a créé un répertoire **/var/lib/squidguard/db**, mais il est vide. Il est destiné à contenir nos listes noires et blanches et deux scripts cgi dont nous verrons l'utilité plus tard. Elle a également créé un fichier de configuration **/etc/squid/squidGuard.conf.**

Nous n'avons pas encore les moyens de travailler efficacement, nous n'avons pas encore de base de données de destinations, mais nous pouvons déjà écrire un fichier de configuration pour SquidGuard, pour nous mettre un peu dans le bain.

### 8. Exemple de configuration minimum de SquidGuard :

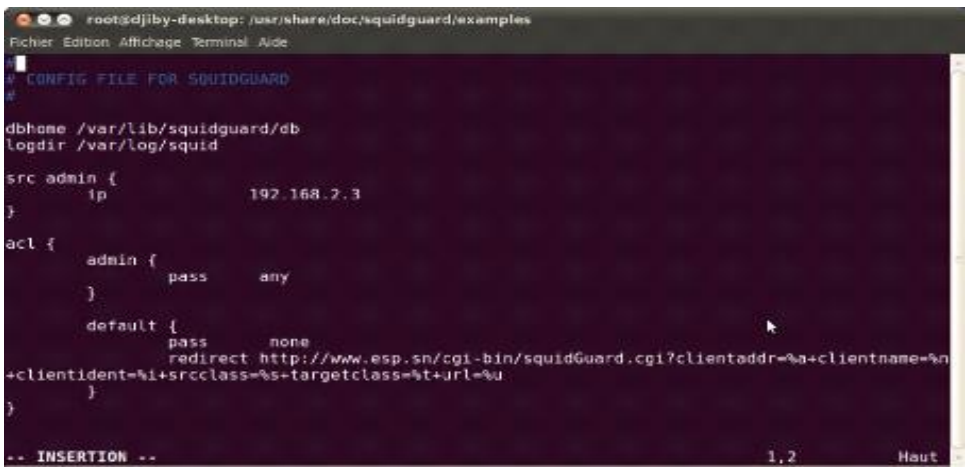


**Figure 4.1: Configuration minimum de SquidGuard**

Il faut indiquer à squidGuard où trouver la base de données des listes (que nous n'avons pas encore), ainsi que l'endroit où l'on désire récupérer les logs.

Les sources sont là pour définir des groupes de clients. Les sources définies par des adresses IP sont les plus simples à mettre en place. Enfin, les ACL permettent de définir quelle source peut aller (ou ne pas aller) vers quelle(s) destination(s). Dans cet exemple : les sources admin peuvent accéder aux toutes les autres destinations. la source default s'applique à tous les clients qui ne font pas l'objet d'une ACL particulière.

Enfin, il faut configurer squid3 pour qu'il invoque SquidGuard en ajoutant ces lignes à la fin de **/etc/squid3/squid.conf :**

**url_rewrite_program /usr/bin/squidGuard -c /etc/squid/squidGuard.conf**

**url_rewrite_children 5**

### 9. Téléchargement de blacklists

Nous allons maintenant configurer notre SquidGuard afin qu'il prenne en compte les listes que nous venons de récupérer à partir de Blacklists (Un ensemble de destinations est activement maintenu à jour par le Centre de Ressources Informatiques de l'Université de Toulouse), nous allons choisir l'archive qui les contient toutes les listes noires et les installer là où c'est prévu, dans **/var/lib/squidguard/db**.

```
#cd /var/lib/squidguard/db/
#wget ftp://ftp.univ-tlse1.fr/blacklist/blacklists.tar.gz
#tar xzf blacklists.tar.gz
#cd blacklists
```


**Figure 4.2: Les listes de destination de SquidGuard**

Nous avons toutes les destinations souhaitables qui sont organisées par catégorie par exemple : audio-video, chat, child, drogue, forums, games, publicite, radio, violence etc… Avant d'oublier ce détail majeur, tout le contenu de **/var/lib/squidguard/db/blacklists** doit être accessible en lecture et en écriture par l'utilisateur sous l'identité duquel SQUID tourne. Pour nous, c'est l'utilisateur « proxy » :

**#cd /var/lib/squidguard/db/ #chown -R proxy:proxy blacklists**

SquidGuard, pour pouvoir travailler rapidement, n'utilise pas les fichiers texte, mais des bases de données. Nous devons alors construire ces bases avant le démarrage de Squid(et donc de SquidGuard) par la commande ci-dessous puis on redémarre le serveur.

**#squidGuard -C all**

Lorsqu'on teste en entrant par exemple le site de « chat.org» ), nous nous rendons compte que l'accès à ce site est bloqué

SquidGuard offre ainsi un filtrage très poussé. SquidGuard permet un filtrage par URL, par adresses IP, par authentification, par plage horaire etc…

### 10. Script de mise à jour de la Blacklist de SquidGuard :

```
#!/bin/bash
   cd /var/lib/squidguard/db
   #if [ -f blacklists.tar.gz]
   #then
          rm -f blacklists.tar.gz
   #fi
   wget -q ftp://ftp.univ-tlse1.fr/blacklist/blacklists.tar.gz
   tar zxvf blacklists.tar.gz
   /usr/bin/squidGuard -C all -d
  chown -R  proxy:proxy /var/lib/squidguard/db/
  service squid3 restart
```

**Figure 4.3 : script de mise à jour de la blacklists.**

**Conclusion :**

A travers ce chapitre, nous avons présenté une démarche pour mettre en place un proxy filtrant avec Squid et SquidGuard pour filtrer et bloquer les sites web à contenu indésirable, la mise en place d'un serveur mandataire http (proxy http) présente de nombreux avantages, aussi bien en termes de sécurités que de « contrôle parental », surtout dans le cadre d'établissements qui offrent à des mineurs la possibilité d'accès à Internet (écoles, collèges, lycées, association diverses …).

Nous utiliserons bien entendu des solutions libres, à savoir Squid pour le proxy http et SquidGuard pour l'élément de Filtrage.

# Conclusion Générale

Nous avons essayé dans ce mémoire de mettre en place un proxy Filtrant sous Linux. De nos jours l'Internet, qui est un réseau mondial public avec plusieurs menaces, est très utilisé et tout le monde y a accès.

Notre choix de proxy c'est ainsi porté sur le proxy SQUID. Nous avons vu dans ce document que son rôle primordial est le filtrage c'est-à-dire filtrer les requêtes des clients. Ainsi, il est possible de restreindre l'accès aux ressources en fonction de plusieurs paramètres différents, Il joue aussi le rôle de sécurité. Nous avons vu que SQUID peut bloquer l'accès à l'Internet à certains utilisateurs selon des critères bien définis ou même bloquer l'accès à certains sites que l'on juge dangereux ou inutiles. Nous avons aussi vu que SQUID pouvait être gardé les pages HTTP en local et les restituer aux clients

Néanmoins, nous tenons à rappeler qu'un proxy n'est pas une solution absolue et complète de filtrage. Comme on l'a déjà dit, il joue primordialement une fonction de filtre. Donc il n'assure pas entièrement le filtrage. Il faut en plus trouver d'autres moyens de filtrage en mettant par exemple en place un firewall correctement configuré qui remplie entièrement un rôle de filtrage. Notons aussi qu'en matière de filtrage informatique, il n'y a ni recette miracle, ni solution définitive.

# Bibliographie et Webographie

## Bibliographie

[1] : Généralité sur les réseaux informatiques-Université BOUMARDES UMBB, RIAHLA Med Amine.

[2] : Réseaux HTML, PHP, Bases de données partagées PostgreSQL, D.Gonzalez, Université de Lille3-Charles de Gaulle, Janvier 2004.

[3] : Cours des réseaux Informatiques, Rziza Mohammed, (2010-2011).

[4] : la messagerie électronique,(2003-2004)

[5] : Généralité sur les réseaux informatiques, Necolas Dewaele.

[6] : Services et Protocoles Applicatifs sur Internet, Olivier Gluck, © 2014 M2 SIR/RTS.

[7] : système multi-agent pour la composition et l'exécution de web services (2008-2011).

[8] : mise en place d'un serveur proxy sous Ubuntu/Debian- BTS Informatique de Gestion –Option administrateur réseaux.

[9] : Le proxy squid – stage administrateurs réseaus.

[10] : Mise en œuvre d'un proxy Filtrant sous Linux (2014/2015).

[11] : squid proxy libre pour Unix et Lunix –Scurinets club de la sécurité informatique INSAT.

[12] : NAT (network Adress Translation) 2014

# Bibliographie et Webographie

**Webographie :**


http://www.les-infostrateges.com/article/0405186/la-messagerie-electronique-principes- techniques, consulté le 18/02/2016.

http://www.memoireonline.com/11/11/4952/m_Conception-et-deploiement-dune-architecture-reseau-securisee--cas-de-SUPEMIR11.html, consulté le 20/02/2016

https://www.nic.ch/reg/cm/wcm-page/faqs/technical/nameserver.jsp?lid=fr, consulté le 22/02/2016.

http://www.commentcamarche.net/contents/518-dns-systeme-de-noms-de-domai, consulté le 22/02/2016.

http://www.frameip.com/dhcp/ , consulté le 31/02/2016.

https://fr.wikipedia.org/wiki/Serveur_de_messagerie_%C3%A9lectronique, consulté le 10/03/2016.

http://emmanuel.marcillac.free.fr/?p=129, consulté le 11/03/2016.

http://www.commentcamarche.net/contents/610-serveur-proxy-et-reverse-proxy, consulté le 18/03/2016.

http://www.materiel-informatique.be/proxy.php, consulté le 19/03/2016.

http://www.materiel-informatique.be/proxy.php consulté le 19/03/2016.

http://www.culture-informatique.net/cest-quoi-un-serveur-proxy/, consulté le 19/03/2016.

http://fr.wikihow.com/changer-les-param%C3%A8tres-de-proxy, consulté le 22/03/2016.

http://www.generation-nt.com/reponses/q-proxy-utilite-quels-avantages-quels-inconvenients-entraide-3477901.html, consulté le 22/03/2016.

http://www.memoireonline.com/07/11/4611/m_Mise-en-place-dun-proxy-Squid-securise-avec-authentification-LDAP3.html, consulté le 28/05/2016.

https://doc.ubuntu-fr.org/squid, consulté le 29/05/2016.

http://caleca.developpez.com/tutoriels/squid-squidguard/, consulté le 30/05/2016.

http://www.malekal.com/squid-squidguard/, consulté le 30/05/2016.

http://lesaventuresdeyannigdanslemondeit.blogspot.com/2013/03/configuration-de-squidguard-comme.html , consulté le 31/05/2016.

http://www.commentcamarche.net/contents/302-cables-et-connecteurs, consulté le 01/06/2016.

https://fr.wikipedia.org/wiki/C%C3%A2ble_%C3%A9lectrique, consulté le 01/06/2016.

# Bibliographie et Webographie

https://fr.wikipedia.org/wiki/Carte_r%C3%A9seau, consulté le 01/06/2016.

http://coursz.com/definition-d-un-ordinateur, consulté le 01/06/2016.

# Résumé

Les systèmes d'information de nos établissements comportent de plus en plus d'applications Web. L'ensemble doit être rendu cohérent pour l'internaute. Ces applications peuvent être lourdes et nécessiter plusieurs machines, doc éventuellement une répartition de charge. En insérant un système intermédiaire entre l'internaute et les serveurs Wed , il est possible d'agréger des ressources diverses en un site Web unique et de répartir la charge entre plusieurs serveurs.

Les firewalls installés il y a quelques années ne protègent plus guère ces serveurs Web : les filtres au niveau IP s'assurent bien que le port de destination est 80 ou 443, au mieux ils vérifient les options TCP. Un filtrage au niveau de l'adresse IP s'avère donc aujourd'hui indispensable.

Nous disposons pour cela de nombreux logiciels open source tant pour et répartir la charge entre des serveurs Web que pour router et répartir la charge entre des serveurs Web que pour authentifier les utilisateurs ou filtrer les requêtes malicieuses.

**Mots-clés :** Proxy, Firewall application Web, Répartition de charge, NAT, Reverse proxy.