T.S.S.I. 2005

Techniciens Supérieurs en Support Informatique

Groupe 01

DNS, FTP, WEB et DHCP : Mise en œuvre de serveurs sous Linux

Eric BOUTIN – Didier KREMER – Patrick MOLLET – Olivier RICHARTE – Philippe RICHOZ

TABLE DES MATIERES

Mise en œuvre d'un serveur DNS, FTP, WEB et DHCP pour Linux

CAHIER DES CHARGES

Les serveurs :

L'entreprise Axius possède un parc de machines avec des postes clients Linux et Windows.

Cette entreprise voudrait que tous les utilisateurs puissent récupérer une adresse IP automatiquement, accéder à des pages Web locales et sur Internet et télécharger des fichiers sur le site de l'entreprise.

Le réseau comportera des adresses IP du type 172.16.11.x *(le n° 11 correspond au groupe de travail n° 1).*

Les postes clients auront des adresses délivrées par le serveur **DHCP** dans une plage comprise entre 172.16.11.200 et 172.16.11.254 et récupérerons les informations du serveur **DNS**. Tous les serveurs se verront attribuer toujours les mêmes adresses via le DHCP.

Concernant le site **Web**, voici ce qu'il faut retenir :

- Il sera installé sur un serveur dédié.
- Le nom de domaine sera **axius-1.fr**.
- Une page d'accueil par défaut possédant des liens vers les différents services sera créée à l'adresse www.axius-1.fr.
- Une page d'accueil pour chaque service de l'entreprise possédant des liens vers des téléchargements sera créée aux adresses suivantes :
    - www.production.axius-1.fr
    - www.comptabilite.axius-1.fr
    - www.secretariat.axius-1.fr
    - www.direction.axius-1.fr
- L'accès au serveur **FTP** de la comptabilité sera soumis à authentification.
- Les accès aux pages seront limités aux utilisateurs authentifiés.
- Si une adresse est erronée, une page d'erreur personnalisée sera affichée avec un lien vers la page d'accueil générale.
- Un utilitaire permettra d'analyser les logs et donnera les statistiques des clients repérés par le nom DNS.

La directive *ErrorDocument* permettra de router les erreurs 404.

Afin d'héberger plusieurs sites avec la même adresse IP, des hôtes virtuels seront créés qui pointeront chacun vers un endroit différent du disque.

La directive *CustomLog* permet de tracer les connexions des clients à partir de leur @IP.

La variable *HostnameLookup* permet de tracer les clients à partir de leur nom DNS.

Webalizer est un utilitaire d'analyse des logs et de statistiques.

Les clients :

Le mot de passe des utilisateurs est leur nom de login

Il existe 4 services dans l'entreprise : production, comptabilité, secrétariat et direction

- **ydurand** est le responsable du service production
- **jdupont** fait partie du service comptabilité
- **hriviere** est le responsable du service comptabilité
- **drameau** est la responsable du secteur secrétariat
- **amartin** fait partie du service secrétariat
- **ggros** est le directeur d'Axius

Configuration du groupe de travail TSSI01

Domainname : axius-1.fr

Server **dhcp**
Service **dhcpd**

SERVEUR DHCP

@IP 172.16.11.12
Mask 255.255.0.0
Name asus

Plage 172.16.11.200 à 172.16.11.254

Fichiers de configuration :

/etc/**dhcpd.conf**
/etc/rc.d/init.d/**dhcpd**

/etc/dhclient.conf (postes clients)

Server **bind**
Service **named**

SERVEUR DNS

@IP 172.16.11.9
Mask 255.255.0.0
Name SRV-DNS

Fichiers de configuration :

/var/named/**axius-1.fr**
/var/named/**axius-1.fr.reverse**
/var/named/**named.ca**
/var/named/**named.local**
/etc/sysconfig/**network**
/etc/**named.conf**
/etc/**nsswitch.conf**
/etc/**resolv.conf**

Server **proftp**
Service **proftpd**

SERVEUR FTP

@IP 172.16.11.1
Mask 255.255.0.0
Name SERVER-LINUX-1

Fichiers de configuration :

/etc/**proftpd.conf**
/etc/**ftpusers**

Server **apache2**
Service **httpd**

SERVEUR WEB

@IP 172.16.11.6
Mask 255.255.0.0
Name SERVER-LINUX-106

Fichiers de configuration :

/etc/httpd/conf/**httpd2.conf**
/etc/httpd/conf/vhosts/**Vhosts.conf**
/etc/**nsswitch.conf**

.htaccess
.htpasswd

POSTES CLIENTS

@IP DHCP

Accès : Internet
Site Intranet *(authentifié)*
FTP *(authentifié)*

RAPPELS LINUX

DHCP			
Fichiers de conf.	**Nom du Service**	**Daemon**	**Paquetages**
/etc/dhcpd.conf /etc/rc.d/init.d/dhcpd /etc/dhclient.conf *(clients)*	DHCP	**dhcpd**	dhcp-common dhcp-server

DNS			
Fichiers de conf.	**Nom du Service**	**Daemon**	**Paquetages**
/var/named/axius-1.fr /var/named/axius-1.fr.reverse /var/named/named.ca /var/named/named.local /etc/named.conf /etc/sysconfig/network /etc/resolv.conf /etc/nsswitch.conf	BIND	**named**	bind bind-utils caching-nameserver

WEB			
Fichiers de conf.	**Nom du Service**	**Daemon**	**Paquetages**
/etc/httpd/conf/httpd2.conf /etc/httpd/conf/vhosts/Vhosts.conf /etc/nsswitch.conf .htaccess .htpasswd	APACHE 2	**httpd**	apache-conf apache2 apache2-common apache2-modules libapr-util0 libapr0

FTP			
Fichiers de conf.	**Nom du Service**	**Daemon**	**Paquetages**
/etc/proftpd.conf /etc/ftpusers	PROFTP	**proftpd**	Proftpd proftpd-anonymous

Groupe 01

SERVEURS DNS

Domain Name System

1. Préparation

Rappel :

Distribution	Mandriva 2005 (Mandrake 10.2)
Nom du serveur	SERVEUR-LINUX-1
Adresse IP du serveur	172.16.11.1
Nom du domaine	axius-1.fr

Avant d'installer le serveur DNS, certaines opérations sont à effectuer...

Définition du nom du serveur
Editer le fichier /etc/sysconfig/**network** et le renseigner ainsi :

```
HOSTNAME=SERVEUR-LINUX-1                    (Définition du nom du serveur)
NETWORKING=yes
GATEWAY=172.16.0.254                        (Passerelle)
```

Définition de l'adresse IP du serveur
Taper la commande **ifconfig** de la façon suivante :

```
# ifconfig eth0 172.16.11.1                 (eth0 correspond au nom du point de montage de la carte réseau par défaut)
```

pour le masque de sous-réseau, taper

```
# ifconfig eth0 netmask 255.255.0.0
```

Remarque : si le système ne garde l'IP de la machine après un redémarrage se servir, en mode console, de l'utilitaire **drakconf** et paramétrer les données ci-dessus)

2. Installation

Vérification et installation des paquetages nécessaires, à savoir : **bind**, **bind-utils** et **caching-nameserver**.

La commande **urpmi -a [mot recherché]** se charge de cette opération. Elle indique si un paquetage correspondant existe, si il est installé, et si d'autres paquetages sont nécessaires à son bon fonctionnement. Puis elle nous demande d'insérer le disque d'installation adéquate.
L'attribut **–a** indique que si plusieurs paquetages coïncident avec la sous chaîne donnée. Il faut tous les prendre.

Exemple :

```
# urpmi -a caching
Pour satisfaire les dépendances, les deux paquetages suivants vont être installé (11 Mo) :
bind-9.3.0-3mdk.i586
catching-nameserver-9.2-2mdk.noarch
Est-ce correct ? (O/N)

# urpmi -a bind-utils
Tous les paquetages sont déjà installés
```

3. Création et Paramétrage du Serveur DNS

Le cache DNS Internet

Dans un premier temps, il est nécessaire d'arrêter le service **named** :

```
# service named stop
```

Sur une autre console, il peut être intéressant d'afficher les messages système se rapportant aux dernières actions effectuées afin d'être sûr qu'il n'y a pas d'erreur. La commande est la suivante :

```
# tail -f /var/log/messages
```
*(la fonction **tail** affiche les dix dernières lignes du fichier et l'attribut –f demande de boucler indéfiniment)*

exemple :

```
service named stop
Arrêt de named :                                           [  OK  ]

# tail /var/log/messages
Jul 19 13:41:24 SERVEUR-LINUX-1 named[2236]: zone 11.16.172.in-addr.arpa/IN: loaded serial
1997022700
Jul 19 13:41:24 SERVEUR-LINUX-1 named[2236]: zone axius-1.fr/IN: loaded serial 1997022700
Jul 19 13:41:24 SERVEUR-LINUX-1 named[2236]: running
Jul 19 13:41:24 SERVEUR-LINUX-1 named: Démarrage de named succeeded
Jul 19 13:41:52 SERVEUR-LINUX-1 named[2236]: shutting down: flushing changes
Jul 19 13:41:52 SERVEUR-LINUX-1 named[2236]: stopping command channel on 127.0.0.1#953
Jul 19 13:41:52 SERVEUR-LINUX-1 named[2236]: no longer listening on 127.0.0.1#53
Jul 19 13:41:52 SERVEUR-LINUX-1 named[2236]: no longer listening on 172.16.11.1#53
Jul 19 13:41:52 SERVEUR-LINUX-1 named[2236]: exiting
Jul 19 13:41:52 SERVEUR-LINUX-1 named:  succeeded
```

Vérifier l'existence et le contenu du fichier /**etc**/**named.conf** (commande vi)
Il devrait contenir les zones suivantes :

```
zone "." {
        type hint;
        file "named.ca";
};

zone "0.0.127.in-addr.arpa" {
        type master;
        file "named.local";
};
```

Ces lignes font référence aux deux fichiers suivant :

- **named.ca** contenant la liste des serveurs DNS racines
- **named.local** où sont définis les paramètres locaux du serveur

Ils se trouvent dans le chemin /**var**/**named** qui est indiqué un peu plus haut dans le fichier dans la ligne :

```
options {
              directory « /var/named » ;
```
(c'est le chemin dans lequel seront rangés les fichiers de configuration de zone)

En utilisant les serveurs DNS de notre FAI comme « **forwaders** », il est possible de rendre les réponses plus rapides et alléger la charge du réseau :

Ex :supposons que le FAI aie deux serveurs DNS payants pour IP 193.252.19.3 et 193.252.19.4 alors, dans notre fichier **named.conf** dans la section appelée **options** nous insérerons les lignes :

```
forwarders {
              193.252.19.3 ;
              193.252.19.4 ;
} ;
```

Remarque :
Attention de ne pas oublier les points virgules !…

Maintenant, nous devons modifier les lignes du fichier **named.local** pour obtenir :

```
@       IN      SOA     SERVEUR-LINUX-1.axius-1.fr. patrick.axius-1.fr.
                                        IN      NS      SERVEUR-LINUX-1.
```

Remarque :
Attention de ne pas oublier les points !…

Ne pas toucher au reste du fichier pour le moment.

Nous devons dire au serveur qu'il est le serveur DNS. Pour cela il faut modifier le fichier /**etc**/**resolv.conf**. Il doit contenir :

```
search axius-1.fr                       (axius-1.fr étant le nom de domaine)
nameserver 127.0.0.1                    (correspond au serveur DNS)
```

Relancer le service named :

```
# service named start
```

Test du bon fonctionnement du cache DNS. Il faut utiliser la fonction **nslookup**.
Exemple :

```
# nslookup

> www.google.fr
Server:         127.0.0.1
Address:        127.0.0.1#53

Non-authoritative answer:
www.google.fr   canonical name = www.google.com.
www.google.com  canonical name = www.l.google.com.
Name:   www.l.google.com
Address: 66.249.87.99
Name:   www.l.google.com
Address: 66.249.87.104
:q

> www.pagesjaunes.fr
Server:         127.0.0.1
Address:        127.0.0.1#53

Non-authoritative answer:
Name:   www.pagesjaunes.fr
Address: 193.252.242.142
Name:   www.pagesjaunes.fr
Address: 193.252.242.225
> exit
```

La ligne **Non-authoritative answer** signifie que l'adresse IP est obtenue depuis le cache de notre serveur. Il fonctionne donc !...
Remarque : Le cache est stocké en mémoire vive. Autrement dit, tout est perdu lors de l'arrêt et du redémarrage du serveur. Cela n'a aucune conséquence.

Le DNS du réseau local

Revenons au serveur DNS principal… Il faut que toutes les machines clientes reconnaissent le serveur DNS nouvellement installé et puissent l'utiliser.

Il faut éditer le fichier /etc/**named.conf**

```
# vi /etc/named.conf
```

On va y mettre les zones directes et inversées correspondant à notre domaine **axius-1.fr.**

On va donc y rajouter les lignes qui suivent :

```
zone "axius-1.fr" {                              (zone d'accès direct)
        notify no;
        type master;
        file "axius-1.fr";                       (référence au fichier axius-1.fr)
};

zone "11.16.172.in-addr.arpa" {                  (zone d'accès inversé)
        notify no;
        type master;
        file "axius-1.fr.reverse";               (référence au fichier axius-1.fr.reverse)
};
```

notify no indique que ce serveur va travailler pour son compte. Cette option est utile lorsque plusieurs DNS doivent se synchroniser. Ce n'est pas notre cas.)

type master indique que nous sommes le serveur d'autorité pour cette zone.

Il convient maintenant de créer les deux fichiers référencés ci-dessus : **axius-1.fr** et **axius-1.fr.reverse.**
Ils doivent être dans /**var**/**named**/.

Pour aller plus vite dans cette démarche, on peut copier le fichier named.local tout en le renommant.
Exemple :

```
# cp /var/named/named.local    /var/named/axius-1.fr
```

Il faut maintenant éditer ces fichiers :

```
vi /var/named/axius-1.fr
```

On peut découper ce fichier en deux parties:

- **L'en-tête**: qui commence au début du fichier $TTL et se finit à la ")"
- **Les enregistrements de la zone**, à savoir le reste du fichier.

Concernant l'entête :

```
$TTL 1d
@       IN      SOA     SERVEUR-LINUX-1.axius-1.fr. patrick.axius-1.fr.  (
                                2005072201 ; Serial
                                28800      ; Refresh
                                14400      ; Retry
                                3600000    ; Expire
                                86400      ; Minimum
                                           )
```

.
$TTL : l'indication du TTL (Time To Live) ou la durée de vie de la zone, exprimée en secondes par défaut, ou dans une autre unité si on la spécifie comme dans l'exemple, ici le d spécifiant que l'unité est en jours, donc 1 jour pour notre exemple.

SOA :est l'abréviation de « Start of Authority » (origine de l'autorité).Le @ est une notation spéciale qui désigne l'origine. Et comme le nom de la zone est axius-1.fr, la première ligne signifie donc :
axius-1.fr IN SOA ……..

SERVEUR-LINUX-1.axius-1.fr. : nom du serveur faisant autorité pour la zone.

patrick.axius-1.fr. : email du responsable technique de la zone en remplaçant le “@” de l'email par un “.”

Serial : Le numéro de version de la zone. Ce chiffre est particulièrement important. A chaque fois que l'on modifie quoi que ce soit dans un fichier de zone, on doit impérativement incrémenter ce numéro, autrement les changements ne seront pas pris en compte par le reste du monde et particulièrement par le serveur secondaire. C'est ce numéro, s'il est incrémenté, qui indique au reste du monde que notre zone a subit un changement et que donc les autres serveurs DNS doivent redemander la zone de notre serveur pour prendre en compte ces changements.

On a l'habitude de suivre une règle simple pour être sûr d'incrémenter ce numéro de version, on le compose par la suite des chiffres de l'année en cours, mois, jour et le nombre de changements effectués ce jour là: *Donc par exemple si on modifie la zone le 10 juillet 2005 , on va mettre 2005071001, puis on le modifie à nouveau le même jour, donc ce serial devient 2003071002, et si on modifie à nouveau le fichier le 15 août de la même année le serial devient alors 2003081501.*

Refresh : Temps en secondes d'attente du serveur secondaire avant de contrôler si le serveur primaire a subi une modification au niveau de sa zone. Sur 8 chiffres max.

Retry : Temps d'attente du serveur secondaire avant de faire à nouveau une demande si le serveur primaire n'a pas répondu à une requête. Sur 8 chiffres max.

Expire : Temps pendant lequel le serveur secondaire va conserver les données en cache. Si ce délai est dépasse et que le serveur secondaire n'a pas pu contacter le serveur primaire, il va alors considérer que les données qu'il a en cache ne sont plus fiable et ne pourra plus servir de serveur secondaire pour la zone tant qu'il n'aura pas réussi à contacter le serveur primaire. Sur 8 chiffres max.

Minimum : Valeur par défaut de ttl des enregistrements. On peut spécifier les ttls au niveau de chaque enregistrement, mais d'une manière générale on défini ici un ttl qui vaut pour tous les enregistrements.

Concernant la suite :

```
                                IN    NS              SERVEUR-LINUX-1.
```

Nous indiquons ici que le serveur DNS (type NS) de la zone (wildcard @) axius-1.fr est SERVEUR-LINUX-1

```
Localhost                       IN    A               127.0.0.1
SERVEUR-LINUX-1                 IN    A               172.16.11.1
```

Nous indiquons ici que l'hôte SERVEUR-LINUX-1 correspond à l'adresse 172.16.11.1 idem pour localhost

Tous les enregistrements d'une zone suivent cette syntaxe :

hôte_ou_wildcard *(ttl facultatif)* **classe** **type** *(priorité_si_besoin)* **valeur**

hôte_ou_wildcard : indique si on défini une machine ou un ensemble de machine.

classe : type de classe, a comme valeur IN pour l'Internet.

Type : Indique quel type d'enregistrement nous sommes en train de définir. Les types les plus utilisés sont A pour une adresse, **CNAME** pour un alias de nom, **NS** pour un serveur de nom, **MX** pour un serveur de Mail, **TXT** pour des commentaires.

(priorité_si_besoin) : Si le type à besoin d'une priorité, nous l'indiquons ici.

Valeur : la valeur ou la donnée de l'enregistrement que nous définissons.

Dans notre exemple ci-dessus, nous indiquons ici que l'hôte par ex, « www.direction » correspond à l'hôte « SERVEUR-LINUX-1.axius-1.fr ». On aurait pu aussi le définir en type A et mettre l'adresse ip correspond à « SERVEUR-LINUX-1.axius-1.fr » comme valeur. Mais dans ce cas, si la machine « SERVEUR-LINUX-1.axius-1.fr »change d'adresse IP on doit modifier la valeur de « www.direction » pour la zone « axius-ally.fr » De plus, comme cette machine héberge environ 200 sites web différents, il y aurait 200 fichiers de zones à modifier avec la nouvelle IP. Pour éviter cela, on définit alors« axius-ally.fr » comme CNAME de « SERVEUR-LINUX-1.axius-1.fr »

Donc le champ CNAME (canonical name) sert à donner plusieurs noms à la même machine.

Enfin il est sage de suivre la règle selon laquelle un champ CNAME, MX ou SOA ne doit jamais se référer à un champ déjà défini avec CNAME mais toujours se référer à un champ A

```
www                             IN    CNAME      SERVEUR-LINUX-1
www.comptabilite                IN    CNAME      SERVEUR-LINUX-1
www.secretariat                 IN    CNAME      SERVEUR-LINUX-1
www.direction                   IN    CNAME      SERVEUR-LINUX-1
www.production                  IN    CNAME      SERVEUR-LINUX-1
```

Puis:

```
vi /var/named/axius-1.fr.reverse
```

```
$TTL 1d
@       IN      SOA     SERVEUR-LINUX-1.axius-1.fr. patrick.axius-1.fr.  (
                                    1997022700 ; Serial
                                    28800      ; Refresh
                                    14400      ; Retry
                                    3600000    ; Expire
                                    86400     ; Minimum
                                                )
              IN      NS      SERVEUR-LINUX-1.
1       IN      PTR     SERVEUR-LINUX-1.axius-1.fr.
```

Remarque :
Notez bien le point à la fin des noms complets. Si on l'omet, le nom de domaine « axius-1.fr » viendra se greffer à ce dernier.

PTR dit que l'adresse « 1 » dans le sous réseau 11.16.172.in-addr.arpa, donc 172.16.11.1 est appelé SERVEUR-LINUX-1.axius-1.fr ; c'est la résolution inverse.

Il est ensuite nécessaire de modifier la ligne **host** du fichier /**etc**/**nsswitch.conf** afin que les postes clients se servent d'abord du serveur **DNS** pour la résolution de nom plutôt que dans files (fichier hosts) :

Le fichier de configuration /**etc**/**nsswitch.conf** détermine l'ordre dans lequel sont effectuées les recherches de certaines informations, en fonction des données et des services, de la même manière que le fichier /**etc**/**host.conf** détermine la façon dont les recherches de noms de machines se font. Par exemple, la ligne **hosts: files nis dns** indique que la recherche d'un nom de machine sera d'abord effectuée dans le fichier local /**etc**/**hosts**, puis dans la table **NIS** et enfin en utilisant le **DNS** (/**etc**/**resolv.conf** et **named**). Si aucune machine ne correspond, alors une erreur est renvoyée. Ce fichier doit être accessible en lecture pour tous les utilisateurs !

```
hosts:      dns files nisplus nis
```

(Le DNS doit être en première position)

Cas d'un serveur DNS secondaire

Il peut s'avérer utile d'installer un serveur DNS secondaire. En effet, si le primaire tombait en panne, le secondaire pourrait prendre le relais de la résolution de nom.

L'installation est la même que pour le serveur DNS Primaire. Par contre le paramétrage du fichier /**etc**/**named.con**f est différent :

```
zone "." {
        type hint;
        file "named.ca";
};

zone "0.0.127.in-addr.arpa" {
        type master;
        file "named.local";
};

zone "axius-1.fr" {                          (zone à ajouter dans le fichier)
        type slave;                          (slave montre qu'il s'agit d'un DNS secondaire)
        file " axius-1.fr ";                 (fichier qui sera créé dans /var/named)
        masters { 172.16.11.6; };            (adresse IP du DNS principal)
};

zone "11.16.172.in-addr.arpa" {              (Idem ci-dessus)
        type slave;
        file " axius-1.fr ";
        masters { 172.16.11.6; };
```

Bien entendu, certaines modification sont à apporter dans le fichier /**etc**/**named.conf** du DNS principal. Il faut ajouter les commandes **allow-transfer** et **notify yes** de la manière suivante :

```
zone "axius-1.fr" {
        notify yes;
        type master;
        allow-transfer { 172.16.11.21; };          (adresse IP du DNS secondaire)
        file "axius-1.fr";
};

zone "11.16.172.in-addr.arpa" {
        notify yes;
        type master;
        allow-transfer { 172.16.11.21; };          (adresse IP du DNS secondaire)
        file "axius-1.fr.reverse";
};
```

Paramétrés ainsi, le DNS secondaire se mettra à jour automatiquement. Si le DNS primaire tombe en panne, il pourra prendre le relais sans problème.

Tests de bon fonctionnement

Nous allons utiliser la fonction **nslookup** :

```
# service named restart
Arrêt de named :                           [  OK  ]
Lancement de named :                       [  OK  ]


# nslookup
> www.comptabilite.axius-1.fr                    (test de résolution de nom)
Server:         127.0.0.1
Address:        127.0.0.1#53                     (utilisation du port d'écoute par défaut : 53)

www.comptabilite.axius-1.fr     canonical name = SERVEUR-LINUX-1.axius-1.fr.
Name:   SERVEUR-LINUX-1.axius-1.fr
Address: 172.16.11.1

> www.axius-1.fr                                 (test de résolution de nom)
Server:         127.0.0.1
Address:        127.0.0.1#53

www.axius-1.fr  canonical name = SERVEUR-LINUX-1.axius-1.fr.
Name:   SERVEUR-LINUX-1.axius-1.fr
Address: 172.16.11.1

> 172.16.11.1                                    (test de résolution inversée)
Server:         127.0.0.1
Address:        127.0.0.1#53

1.11.16.172.in-addr.arpa        name = SERVEUR-LINUX-1.axius-1.fr.

> exit
```

Ici, tout fonctionne bien...

4. Configuration des clients

Dans le fichier /**etc**/**resolv.conf,** il faut que l'on ait les lignes suivantes :

```
search axius-1.fr
nameserver 172.16.11.1
```

Dans le fichier /**etc**/**sysconfig**/**network** il faut renseigner la passerelle ainsi :

```
GATEWAY=172.16.0.254
```

Il suffit ensuite d'utiliser **nslookup** comme précédemment pour tester les connexions...

T.S.S.I. 2005

Techniciens Supérieurs en Support Informatique

Groupe 01

SERVEURS FTP

File Transfer Protocol

1- Installation

Installer **Proftp** avec la commande :

```
# urpmi -a proftp
```

2 - Configuration

2.1 – L'Utilisateur « nobody »

Tester si le groupe et l'utilisateur **nobody** existent :

```
# cat /etc/group | grep nobody et
# cat /etc/passwd | grep nobody
```

S'il n'y a pas de réponse, c'est que l'un ou l'autre sont à créer, pour cela :

```
# groupadd nobody
# useradd nobody -d / -s /bin/false
```

Puis affecter l'utilisateur au groupe :

```
# usermod nobody -g nobody
```

2.2 - Les Autres Utilisateurs

Nous allons créer deux utilisateurs :
adminftp, pour ajouter des fichiers au FTP
userftp, pour se loguer simplement au FTP

Voici la marche à suivre :

```
# useradd adminftp -d / -s /bin/false
# useradd userftp -d / -s /bin/false
```

Pour ajouter les mots de passe à ces utilisateurs :

```
# Passwd adminftp et
# Passwd userftp
```

Remarque : Les /bin/false au lieu de /bin/bash limitent les accès au FTP et non au réseau Linux. Par contre, il faut penser ensuite à éditer le fichier /etc/shells et ajouter la ligne suivante :

```
/bin/false
```

2.3 – Les Partages

Il faut à présent créer un dossier ou seront partagés les fichiers du FTP, par exemple /**mnt/ftp**. Nous y inclurons les dossiers **direction**, **comptabilité**, **secrétariat**, **production** et **docs**. Pour cela, voici la marche à suivre :

```
# mkdir /mnt/ftp
# cd /mnt/ftp
# mkdir direction comptabilite secretariat production docs
# chmod -R 777 /mnt/ftp                    (l'accès aux dossiers est sans restrictions pour le moment...)
```

2.4 – Configuration

Nous allons devoir certainement créer deux fichiers nécessaires à la configuration de proftp. Il s'agit de :
/etc/ftpusers et **/etc/proftpd.conf**.

Tout d'abord, voyons **/etc/ftpusers**. C'est le fichier des utilisateurs exclus de notre FTP. En fait, nous allons y mettre tous les utilisateurs définis dans le fichier **passwd**. Evidement, en procédant ainsi, on exclut tout le monde du FTP ! Il suffira alors d'éditer le fichier et d'ôter les lignes des utilisateurs pouvant aller sur le FTP... Voici la commande :

```
# cp /etc/passwd /etc/ftpusers
```
(copie de passwd et collage dans ftpusers)

Reste donc à ouvrir le fichier **ftpusers** et enlever les lignes des utilisateurs auxquels on veut accorder l'accès au FTP, notamment **adminftp** et **userftp**. Nous devrions avoir un fichier qui ressemble à ceci :

```
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
news
uucp
operator
games
nobody
rpm
postfix
vcsa
xfs
messagebus
haldaemon
apache
rpc
rpcuser
sshd
ftp
gdm
named
```

Cette opération permet de faire en sorte que les utilisateurs du FTP ne soient que des utilisateurs de FTP, et rien d'autre.
*Remarque : Ci-dessus, il n'y a que les noms des utilisateurs exclus. On peut laisser les lignes complètes venant du fichier **passwd**, cela ne gène en rien. La lisibilité du fichier est par contre moins bonne.*

Ensuite, il faut éditer (créer s'il n'existe pas) le fichier **proftpd.conf**.

```
# vi /etc/proftpd.conf.
```

C'est le fichier de la configuration générale de **Proftp**.

Voici un descriptif des principales lignes qu'il contient :

```
# le nom du serveur
ServerName "Kernel's ProFTP server"

# le daemon reste en mémoire et écoute les connections
ServerType standalone

# Autoriser l'usage de /etc/ftpusers
UseFtpUsers on

# Répertoire dans lequel arrivent les utilisateurs acceptés
DefaultChdir /mnt/ftp

# Répertoire racine, les connectés au ftp ne verrons que lui et son contenu
DefaultRoot /mnt/ftp

# Utile surtout pour les "virtuals hosts" mais laissez ainsi
DefaultServer on

# le daemon écoute sur le port 21
Port 21

# On autorise les reprises des téléchargements interrompus :
AllowStoreRestart on

# Les logs des connections et transferts
SystemLog /var/log/proftpd.log
TransferLog /var/log/xfer.log

# Seul le propriétaire d'un fichier peut le modifier.
Umask 022

# Nombre de processus fils maximum que va utiliser proftpd, laissez ainsi.
MaxInstances 30

# Proftp sera lancé avec les privilèges (c'est à dire aucun) de nobody
User nobody
Group nobody

# Nombre maximum de clients simultanés (sur ADSL ça fait 5ko/s par utilisateurs)
MaxClients 3

# Nombre maximum de clients ayant le même login
MaxClientsPerHost 3

# Message d'accueil après une connexion réussie
AccessGrantMsg "Connexion reussie pour %u"

# Pour ne pas donner d'info sur le serveur
DeferWelcome off

# Un utilisateur peut écraser ses propres fichiers
AllowOverwrite on

#Seul l'utilisateur adminftp a le droit d'écrire dans /mnt/ftp
<Directory /mnt/ftp>

<Limit WRITE>
AllowUser adminftp
DenyAll
</Limit>

# Avec les lignes suivantes, on pourrait restreindre l'accès au FTP au machines dont l'IP commence (ici) par 172.16.11
#<Limit LOGIN>
#Allow 172.16.11
#Deny All
#</Limit>

</Directory>
```

Au final, le fichier correspond à ceci :

```
ServerName              "FTP Axius 01"
ServerType              standalone
UseFtpUsers             on

DefaultChdir            /mnt/ftp
DefaultRoot             /mnt/ftp
DefaultServer           on

Port                    21
AllowStoreRestart       on
SystemLog               /var/log/proftpd.log
TransferLog             /var/log/xfer.log
Umask                   022
MaxInstances            30
User                    nobody
Group                   nobody

MaxClients              5
MaxClientsPerHost       5

AccessGrantMsg          "Connexion réussie. Bienvenue %u!"
DeferWelcome            off
AllowOverwrite          on

<Directory /mnt/ftp>
<Limit WRITE>
AllowUser adminftp
DenyAll
</Limit>

</Directory>
```

3 – Lancement du Service (daemon)

On va lancer proftp en standalone, il faut donc le supprimer de la liste des daemons lançables par **xinetd**. Ce service est utilisé fréquemment pour l'accès à de nombreux services réseau. Selon la distribution on peut trouver soit un daemon inetd, soit un daemon **xinetd** (c'est notre cas...). Dans l'absolu, inetd et **xinetd** agissent comme une standardiste. Ils reçoivent des requêtes de clients, extérieurs pour la plupart, qui demandent un accès à un service réseau déterminé (ex : ftp, telnet, ssh...). Le super daemon va, en fonction des instructions qu'on lui aura données (fichiers de configuration) transmettre ou rejeter l'appel.

Déplacez le fichier **/etc/xinetd.d/proftpd-xinetd** vers un autre répertoire (s'il n'y a pas de fichier /etc/xinetd.d/proftpd , ne faîtes rien). Voici les étapes :

```
# mv /etc/xinetd.d/proftpd-xinetd /root/        (Déplacement du fichier vers un autre répertoire)

# killall -HUP xinetd                           (Arrêt de xinetd)
# killall -HUP proftpd                          (Arrêt de proftpd)
```

killall *envoie un signal à tous les processus en train d'exécuter les commandes mentionnées. Les signaux peuvent être indiqués soit par leur nom (par exemple -HUP) soit par leur numéro (par exemple -1).*

```
# /usr/sbin/proftpd                             (Lancement de proftpd)
```

Et vérifier qu'il est bien lancé par :

```
# ps ax | grep proftp
```

S'il restait sans réponse, c'est que **xinetd** vous bloque, refaites les étapes ci-dessus.

Plus tard, si vous faites des modifications dans **/etc/proftpd.conf** pour qu'elles soient prises en compte par le daemon déjà lancé) il vous suffira de taper :

```
# killall -HUP proftpd
```

4 – Contrôle du FTP

Afin de tester la configuration, il faut tester le service **proftpd**.

Pour rappel, si on veut savoir si **proftd** tourne, il faut taper l'instruction suivante :

```
# ps ax | grep proftd
```

Si rien n'apparaît, c'est qu'il ne fonctionne pas, sinon il va afficher quelque chose comme ce qui suit :

```
26786 pts/0    D+     0:00 grep proftd
```

Ensuite, on va taper l'instruction suivante :

```
# ftp 172.16.11.1
```

On utilisera alors le login **userftp** (par exemple) et son mot de passe pour entrer.

Voici ce qui apparaît lors la connexion :

```
Connected to 172.16.11.1.
220 ProFTPD 1.2.10 Server (FTP Axius 01) [172.16.11.1]
500 AUTH not understood
500 AUTH not understood
KERBEROS_V4 rejected as an authentication type
Name (172.16.11.1:patrick): userftp            (login userftp)
331 Password required for userftp.
Password:                                      (password de userftp)

230 Connexion réussie. Bienvenue userftp!
Remote system type is UNIX.
Using binary mode to transfer files.
```

On peut ensuite contrôler le contenu de notre dossier FTP. Il devrait contenir les dossiers précédemment créés :

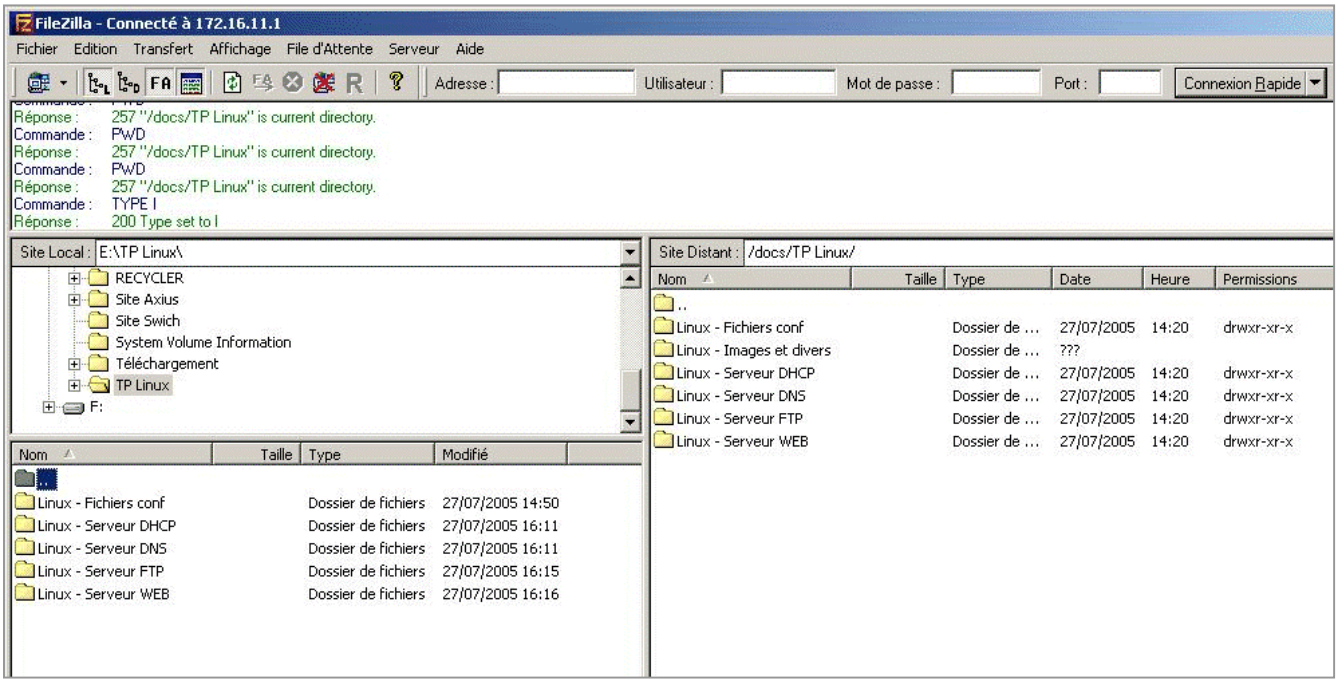
```
ftp> ls                                        (Demande d'affichage des dossiers de FTP)
200 PORT command successful
150 Opening ASCII mode data connection for file list
drwxrwxrwx   2 root     root         4096 Jul 20 12:17 comptabilite
drwxrwxrwx   2 root     root         4096 Jul 20 12:17 direction
drwxrwxrwx   2 root     root         4096 Jul 20 12:17 docs
drwxrwxrwx   2 root     root         4096 Jul 20 12:17 production
drwxrwxrwx   2 root     root         4096 Jul 20 12:17 secretariat
226 Transfer complete.
```

Ici nous voyons bien apparaître nos dossiers, comme prévu...

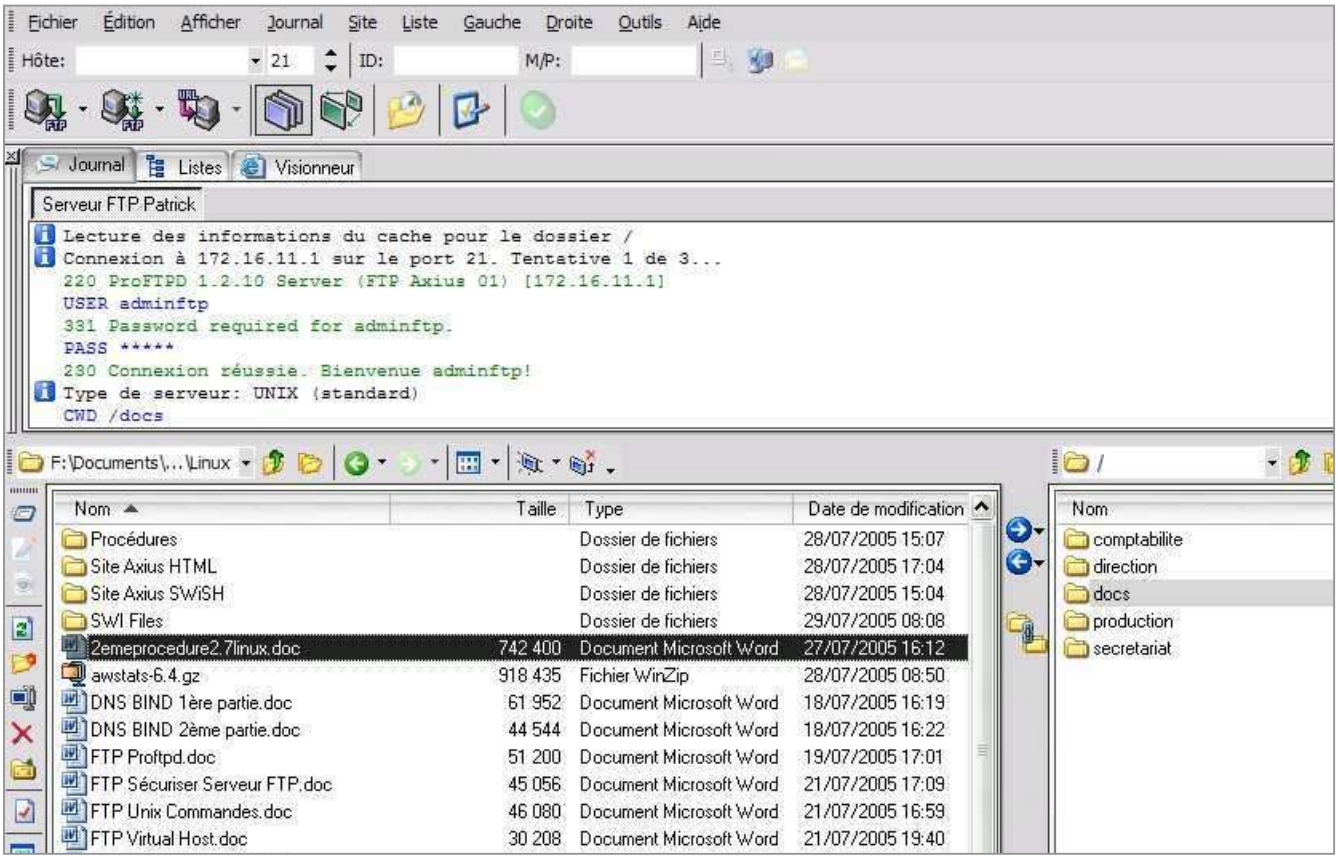
Pour quitter le FTP, il faut utiliser la commande suivante :

```
ftp> bye
221 Goodbye.
```

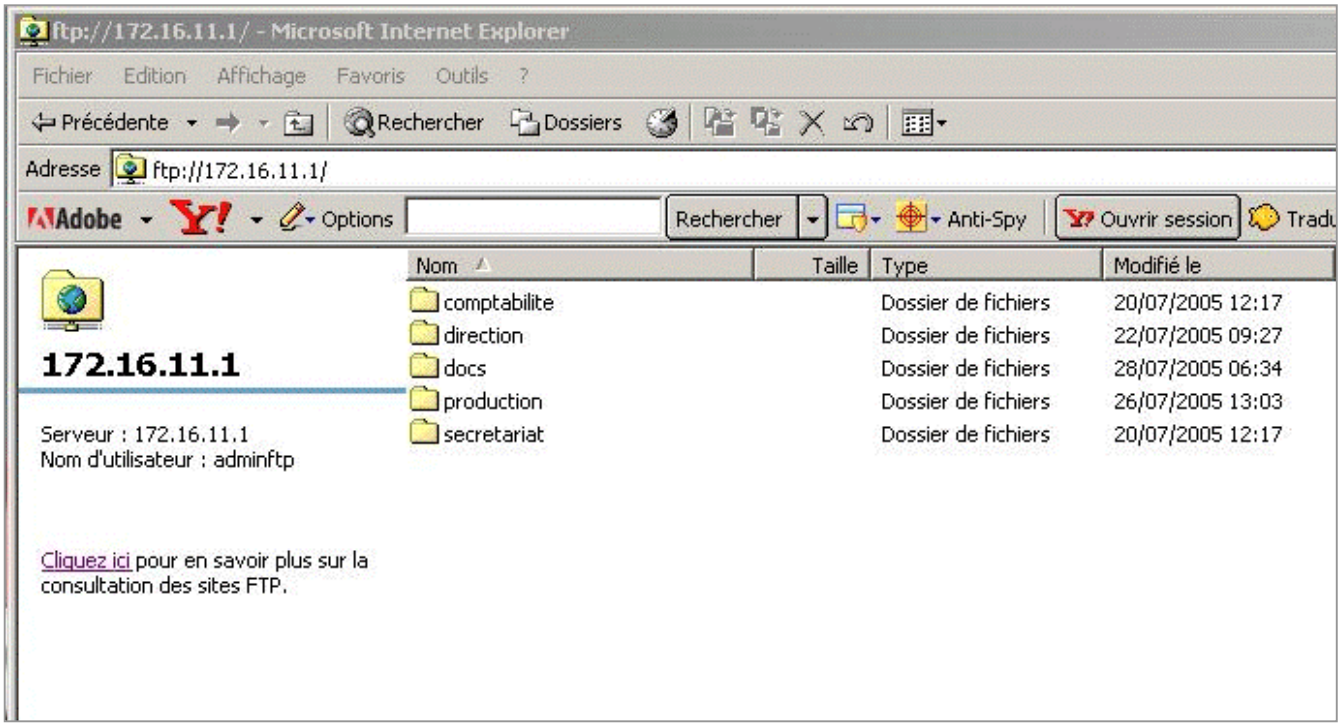
Il est possible de se loguer dans le FTP avec **FileZilla** (client FTP pour Linux) :



Il est aussi possible de se loguer dans le FTP avec **CuteFTP** (client FTP pour Windows) :



Ou encore avec **Internet Explorer** :



5 – Aller plus Loin

5.1 – Attribuer un Dossier Racine en Fonction du Login

Nous pouvons faire en sorte qu'un utilisateur aille sur un dossier bien particulier quand il se connecte au FTP, sans qu'il puisse remonter plus haut dans l'arborescence.

Dans le fichier /etc/proftpd.conf, voici la ligne concernée :

 # Répertoire racine, les connectés au ftp ne verrons que lui et ne pourrons pas remonter plus haut dans l'arborescence (par exemple, ils ne pourrons pas aller dans le dossier mnt).

```
DefaultRoot /mnt/ftp
```

Pour personnaliser, il faut taper :

DefaultRoot [Chemin] [Username]ou[Group]

Résultat :

```
DefaultChdir            /mnt/ftp
DefaultRoot             /mnt/ftp secretariat
DefaultRoot             /mnt/ftp direction patrick          (nom de groupe et d'utilisateur)
DefaultRoot             /mnt/ftp secretariat
DefaultRoot             /mnt/ftp/comptabilite comptabilite
DefaultRoot             /mnt/ftp
```

Là, les membres du groupe Compabilité (comptabilite) iront directement sur leur dossier (mnt/ftp/comptabilite) lorsqu'ils se loguerons sur le FTP. La dernière ligne indique la racine des utilisateurs authentifiés ne faisant parti d'aucun des groupes (et des utilisateurs) définis dans les lignes précédentes.

5.2 - Restrictions

Compliquons un peu les choses...

Nous avons six utilisateurs qui sont répartis dans quatre groupes distincts de la façon suivante :

Direction : ggros

Comptabilité : jdupont, hriviere

Production : ydurand

Secretariat : amartin, drameau

Nous voulons que ces utilisateurs authentifiés puissent accéder à tous les dossiers sauf celui de la comptabilité que seuls les comptables peuvent utiliser...

Création des groupes :

```
# groupadd direction
# groupadd production
# groupadd comptabilite
# groupadd secretariat
```

Création des utilisateurs :

```
# useradd ggros -g direction -s /bin/false
# useradd ydurand -g production -s /bin/false
# useradd jdupont -g comptabilite -s /bin/false
# useradd hriviere -g compabilité -s /bin/false
# useradd drameau -g secretariat -s /bin/false
# useradd amartin -g secretariat -s /bin/false
```

Ensuite, il faut penser à leur attribuer un mot de passe

(commande : passwd [nom de l'utilisateur])

Nous allons maintenant restreindre l'accès au dossier comptabilité

```
# cd /mnt/ftp
# ll
total 20
drwxrwx---  2 root comptabilite 4096 jui 20 14:17 comptabilite/
drwxrwxrwx  2 root direction    4096 jui 22 09:45 direction/
drwxrwxrwx  2 root root         4096 jui 20 14:17 docs/
drwxrwxrwx  2 root production   4096 jui 20 14:17 production/
drwxrwxrwx  2 root secretariat  4096 jui 20 14:17 secretariat/
```

Ici nous avons changé les groupes ayant des droits

(commande : chown :[nom du groupe] [nom du répertoire]).

Nous avons également modifié les attributs du dossier comptabilité (drwxrwx---). En raison des restrictions apportées au dossier « comptabilite », seul ceux du groupe Comptabilité pourront y accéder.

(Commande : `chmod o-rwx [nom du dossier]`)

Enfin, nous allons modifier le fichier **proftd.conf** :

```
DefaultChdir            /mnt/ftp
DefaultRoot             /mnt/ftp secretariat
DefaultRoot             /mnt/ftp direction
DefaultRoot             /mnt/ftp secretariat
DefaultRoot             /mnt/ftp comptabilite
DefaultRoot             /mnt/ftp
```

Ici, tous les utilisateurs arrive sur la même racine lorsqu'ils se loguent au FTP.

Nous avons laissé les lignes correspondant aux différents groupes, au cas où nous voudrions ultérieurement modifier la racine par défaut d'un groupe en particulier (cf. 5.1).

6 – Les Commandes FTP

Pour rappel, le lancement de la connexion FTP se fait en tapant la commande :

```
# ftp [adresse IP] ou [nom de machine] (exemple : # ftp 172.16.11.1)
```

Voici quelques commandes utiles dans le FTP en mode console (à l'invite ftp>) :

Commande	Syntaxe associée	Fonction
disconnect quit bye close		Fermer la session en cours
recv get	remote-file [local-file]	Télécharger un fichier
send put	local-file [remote-file]	Envoyer (uploader) un fichier
cdup		Remonter dans l'arborescence
mkdir	[directory]	Créer un répertoire distant
rmdir	[directory]	Effacer un répertoire distant
rename	[from] [to]	Renommer un fichier distant
delete	[file]	Effacer un fichier distant
size	[file]	Connaître la taille d'un fichier
system		Connaître l'operating system
reget	remote-file [local-file]	Reprendre un téléchargement interrompu
status		Connaître quelques paramètres du système
? help	[command]	Obtenir l'aide concernant une commande
ls		Afficher la liste des fichiers et répertoires

Groupe 01

SERVEURS WEB

1- Installation

Lancer la recherche et l'installation des paquetages nécessaires :

```
# urpmi -a apache
```

```
[root@asus ~]# urpmi apache-1
Pour satisfaire les dépendances, les 7 paquetages suivants vont être installés (
2 Mo):
apache-1.3.33-6mdk.i586
apache-conf-2.0.53-5mdk.i586
apache-modules-1.3.33-6mdk.i586
apache2-common-2.0.53-9mdk.i586
libapr-util0-0.9.6-4mdk.i586
libapr0-0.9.6-3mdk.i586
libmm1-1.3.1-1mdk.i586
Est-ce correct ? (O/n)
```

```
Pour satisfaire les dépendances, les 7 paquetages suivants vont être installés (
2 Mo):
apache-1.3.33-6mdk.i586
apache-conf-2.0.53-5mdk.i586
apache-modules-1.3.33-6mdk.i586
apache2-common-2.0.53-9mdk.i586
libapr-util0-0.9.6-4mdk.i586
libapr0-0.9.6-3mdk.i586
libmm1-1.3.1-1mdk.i586
Est-ce correct ? (O/n) o
Veuillez insérer le média nommé « Installation CD1  » dans le périphérique [/dev
/hdc]
Appuyez sur la touche Entrée quand vous êtes prêt...

installation de apache-conf-2.0.53-5mdk.i586.rpm apache-modules-1.3.33-6mdk.i586.rpm libapr-uti
l0-0.9.6-4mdk.i586.rpm libmm1-1.3.1-1mdk.i586.rpm apache2-common-2.0.53-9mdk.i586.rpm libapr0-0
.9.6-3mdk.i586.rpm apache-1.3.33-6mdk.i586.rpm depuis /var/cache/urpmi/rpms
Préparation ...                  ##########################################
      1/7: libapr0               ##########################################
      2/7: libmm1                ##########################################
      3/7: libapr-util0          ##########################################
      4/7: apache2-common        ##########################################
      5/7: apache-modules        ##########################################
      6/7: apache-conf           ##########################################
Création de httpd-perl.conf à partir du fichier compat
Création de httpd.conf à partir d'un fichier compacté
      7/7: apache                ##########################################
Re-creating /etc/httpd/modules
```

Pour Apache 2, taper :

```
# urpmi apache2
```

```
[root@asus ~]# urpmi apache2
Pour satisfaire les dépendances, les 6 paquetages suivants vont être installés (
2 Mo):
apache-conf-2.0.53-5mdk.i586
apache2-2.0.53-9mdk.i586
apache2-common-2.0.53-9mdk.i586
apache2-modules-2.0.53-9mdk.i586
libapr-util0-0.9.6-4mdk.i586
libapr0-0.9.6-3mdk.i586
Est-ce correct ? (O/n)
```

L'installation de ces paquetages a créé les dossiers suivants :

/var/www/ *(dossier de vos pages Web)*
/etc/httpd/ *(dossier de configuration des services)*

Pour vérifier si ce service est lancé, taper la commande :

```
# ps ax | grep httpd
```

Si il n'y a aucune réponse, lancer le service apache :

```
service httpd start
```

Si un mauvais paquetage a été installé, utiliser les commandes :

 # rpm -q [paquetage] *(Vérifier si le paquetage est installé)*
 # rpm -e [paquetage] *(Désinstaller un paquetage. Ne pas oublier de détruire éventuellement les dossiers créés par ces paquetages)*

2. Vérification

Créer un fichier **/var/www/html/index.html**.

```
# vi index.html
```

(taper du texte dedans)

Vérifier le fonctionnement du service Web en tapant :

```
# lynx localhost
```

3. Configuration

Il existe deux fichiers de configuration importants :

/etc/httpd/conf/httpd.conf (pour apache)
/etc/httpd/conf/httpd2.conf (pour apache2)
et
/etc/httpd/conf/vhosts/Vhosts.conf

3.1 – httpd.conf

Il faut éditer le fichier **/etc/httpd/conf/httpd2.conf** et mettre la ligne :

```
ServerName              SERVEUR-WEB-LINUX
```

Voici quelques lignes de commandes du fichier **/etc/httpd/conf/httpd.conf**

```
Fichier  Édition  Affichage  Terminal  Onglets  Aide
### Main Configuration Section
### You really shouldn't change these settings unless you're a guru
###
ServerType standalone
ServerRoot /etc/httpd
#ServerName localhost
#LockFile /etc/httpd/httpd.lock
PidFile /var/run/httpd.pid
ScoreBoardFile /etc/httpd/httpd.scoreboard
ErrorLog logs/error_log
LogLevel warn
ResourceConfig /dev/null
AccessConfig /dev/null
DocumentRoot /var/www/html
```

Ici, le chemin de travail peut être personnalisé.

Si le chemin de travail doit être modifié, il faut aussi modifier le fichier **commonhttpd.conf** (ligne 107), auquel **httpd.conf** fait référence grâce à la commande `include`.

```
###
### Global Configuration
###
# We now support multiple apache configurations on the same server. In
# common.conf, we put all directives that are common to all implementations
# (httpd, httpd-perl, etc.)
Include conf/commonhttpd.conf
```

Ce fichier est requis pour le bon fonctionnement, il inclut des paramètres.

```
###
### IP Address/Port and Proxied configuration section
###
# The APACHEPROXIED setting can be set in /etc/rc.d/init.d/httpd if you
# are using a proxy or accelerator, like the Apache-SGI or khttpd, so that
# the fast web server serves static content while Apache handles the
# cgi or php files

#BindAddress *
<IfDefine APACHEPROXIED>
    Port 8080
    Listen 8080
</IfDefine>
<IfDefine !APACHEPROXIED>
    Port 80
    Listen 80
</IfDefine>
```

```
### Virtual Hosts
###
# We include different templates for Virtual Hosting. Have a look in the
# vhosts directory and modify to suit your needs.
Include conf/vhosts/Vhosts.conf
#Include conf/vhosts/DynamicVhosts.conf
#Include conf/vhosts/VirtualHomePages.conf
```

Ce fichier est requis pour le bon fonctionnement, il inclut des paramètres.

3.2 – Vhosts.conf

A présent, il convient d'éditer le fichier **/etc/httpd/conf/vhosts/Vhosts.conf** . Ce fichier de configuration permet de créer et de gérer des sites web virtuels. Autrement dit, il permet d'héberger plusieurs sites sur le même serveur, avec une seule adresse IP.

Voici comment déclarer des sites virtuels dans **Vhosts.conf** :

```
NameVirtualHost 172.16.11.6                    (adresse IP du serveur web)

<VirtualHost 172.16.11.6>                      (adresse IP du premier site virtuel)
ServerName axius-1.fr                          (nom du premier site)
DocumentRoot /var/www/html                     (emplacement des pages du site)
DirectoryIndex index.html index.htm index.shtml index.php
ErrorDocument  404   /erreur.html
#ErrorLog logs/axius-error_log
#CustomLog logs/axius -access_log common
#TransfertLog /var/www/html/logs/acces_log
#ErrorLog  /var/www/html/logs/error_log
</VirtualHost>

<VirtualHost 172.16.11.6>                      (adresse IP du second site virtuel)
ServerName www.production.axius-1.fr           (nom du second site)
DocumentRoot /var/www/html/production          (emplacement des pages du second site)
</VirtualHost>

<VirtualHost 172.16.11.6>                      (troisième site virtuel)
ServerName www.secretariat.axius-1.fr
DocumentRoot /var/www/html/secretariat
</VirtualHost>

<VirtualHost 172.16.11.6>                      (quatrième site virtuel)
ServerName www.direction.axius-1.fr
DocumentRoot /var/www/html/direction
</VirtualHost>

<VirtualHost 172.16.11.6>                      (cinquième site virtuel)
ServerName www.comptabilite.axius-1.fr
DocumentRoot /var/www/html/comptabilite
</VirtualHost>
```

Grâce à cette configuration, tous les sites suivants seront héberger sur la même machine, et ils auront tous la même adresse IP, pourtant ils seront bien des sites distincts.

www.axius-1.fr
www.comptabilite.axius-1.fr
www.direction.axius-1.fr
www.production.axius-1.fr
www.secretariat.axius-1.fr

Cependant pour que ces « *virtual hosts* » fonctionnent, il faut modifier d'autres fichiers : un sur le serveur web lui-même (/**etc**/**nsswitch.conf**), et un autre sur le serveur DNS dont il dépend (/**var** /**named**/**axius-1.fr**).

Sur le serveur web, il faut donc éditer le fichier /**etc**/**nswitch.conf**, afin qu'il utilise la résolution de nom fournit par le serveur DNS, au lieu d'utiliser ses fichiers propres (à savoir /etc./hosts). Voici donc comment modifier le fichier **nsswitch.conf** :

A la place de la ligne :

```
hosts: files nisplus nis dns
```

Il faut taper :

```
hosts: dns files nisplus nis
```

Sur le serveur dns, il faut éditer le fichier /**etc**/**named**/**axius-1.fr** pour que celui-ci soit capable de faire correspondre les noms des sites virtuels au serveur qui les héberge. On doit donc obtenir ceci :

```
$TTL 1d
@       IN      SOA     SERVEUR-LINUX-DNS.axius-1.fr. admin.axius-1.fr.  (
                                        1997022700 ; Serial
                                        28800      ; Refresh
                                        14400      ; Retry
                                        3600000    ; Expire
                                        86400 )    ; Minimum

                        IN      NS      SERVEUR-LINUX-DNS.

SERVEUR-LINUX-DNS   IN      A       172.16.11.1
SERVEUR-LINUX-WEB           IN      A       172.16.11.6

www                         IN      CNAME   SERVEUR-LINUX-WEB
www.comptabilite            IN      CNAME   SERVEUR-LINUX-WEB
www.secretariat             IN      CNAME   SERVEUR-LINUX-WEB
www.direction               IN      CNAME   SERVEUR-LINUX-WEB
www.production              IN      CNAME   SERVEUR-LINUX-WEB
```

Ne pas oublier non plus de créer les répertoires /var/www/html/**comptabilite**, /var/www/html/**direction**, /var/www/html/**production**, /var/www/html/**secretariat**, et d'y déposer dans chacun un fichier **index.html** .

Maintenant, il faut redémarrer le service apache afin de prendre en compte tous les changements :

```
# service httpd restart
```

On peut à présent tester le bon fonctionnement de la configuration :

```
# lynx www.axius-1.fr
# lynx www.comptabilite.axius-1.fr
# lynx www.direction.axius-1.fr
# lynx www.productin.axius-1.fr
# lynx www.secretariat.axius-1.fr
```

Si tout a été correctement réalisé, la page **index.html** de chaque site doit s'afficher (penser à personnaliser chaque page pour être sûr qu'elle correspond bien au site choisi).

4. Authentification

Dans le but de sécuriser l'accès à un site web, il est possible de mettre en place un système d'authentification. Deux fichiers prennent en charge cela : **.htaccess** et **.htpasswd** .

4.1 – htaccess

Un fichier **.htaccess** est à créer et à placer dans le dossier web que l'on souhaite sécuriser. Par exemple, si l'on souhaite sécuriser l'accès à la page concernant la direction (qui se trouve **/var/www/html/direction/index.html**), il faudra placer un **.htaccess** à cet endroit : **/var/www/html/direction/** .

Voici comment configurer le fichier **.htaccess** :

```
AuthType            Basic
AuthUserFile /var/www/html/conf/.htpasswd
#AuthGroupFile    /etc/apache/group
AuthName            "Accès Sécurisé"
<Limit GET POST>
      require user utilisateur1 utilisateur2
(ou    require valid-user)
      #require group groupe1 groupe2
</Limit>
```

Explication des lignes de **.htaccess** :

```
AuthUserFile        /var/www/html/conf/.htpasswd
```

Correspond au chemin absolu vers le fichier .passwd contenant la liste des utilisateurs et leur mot de passe crypté.

```
AuthGroupFile              /etc/apache/group
```

Correspond au chemin du fichier group contenant la liste des groupes d'utilisateurs (inutilisé dans cette configuration).

```
AuthName                   "Accès Sécurisé"
```

Permet de choisir le texte qui apparaîtra dans la boîte de dialogue de l'authentification.

```
require user        [utilisateur1] [utilisateur2]
```

Permet de lister tous les utilisateurs autorisés à entrer.

```
require valid-user
```
Permet de laisser entrer tous les utilisateurs qui se sont déjà authentifiés. Autrement, il demandera l'authentification. Il rend inutile l'utilisation de `require group`.

```
require group              [groupe1] [groupe2]
```
Permet de lister tous les groupes autorisés à entrer (inutilisé dans cette configuration).

4.2 – htpasswd

Comme décrit plus haut, le fichier **.htpasswd** est le partenaire indispensable de **.htaccess** pour la mise en œuvre d'un système d'authentification. Le fichier .htaccess créé précédemment y fait référence dans la ligne :

```
AuthUserFile        /var/www/html/conf/.htpasswd
```

Pour que cette ligne soit valide, il faut donc créer maintenant un dossier confidentiel qui contiendra .htpasswd et que l'on nommera **conf**. Taper les commandes suivantes.

```
# cd /var/www/html
# mkdir conf
```

Maintenant que le dossier **conf** existe, il faut s'assurer qu'il est correctement paramétré. Pour permettre une utilisation normale, il doit offrir les droits suivants :

```
drwxr-xr-x   root   root   conf/
```

Si jamais cela ne correspond pas, modifier ses droits avec les commandes :

```
# cd /var/www/html
# chmod o+rx conf          (car ici ce sont les droits de lecture et d'exécution pour others qui comptent)
```

Le dossier **conf** est à présent bien configuré. Passons à son contenu.

Créer le fichier **.htpasswd** en créant le premier utilisateur grâce à la commande :

```
# htpasswd -c .htpasswd [utilisateur1]
```
Puis, rentrer le mot de passe correspondant à cet utilisateur.

Rajouter les autres utilisateurs en tapant :

```
# htpasswd .htpasswd [utilisateur2]
# htpasswd .htpasswd [utilisateur3]
```
etc…

Le fichier **.htpasswd** est maintenant renseigner des noms d'utilisateur et des mots de passe cryptés avec la syntaxe suivante :

```
[utilisateur1]:[motdepassecrypté]
[utilisateur2]:[motdepassecrypté]
```
etc…

Pour sécuriser le serveur web, le dossier **conf** doit être protégé des regards indiscrets par la création d'un fichier .htaccess qui comprendra uniquement la ligne suivante :

```
Deny From All
```

Ceci interdira l'accès à ce dossier au cas où un visiteur web tape l'url : **http://www.axius-1.fr/conf/**
Toute personne tentant d'accéder au contenu de ce dossier sera donc rejeté.

Redémarrer le service :

```
# service httpd restart
```

Tester le bon fonctionnement de l'authentification en surfant sur le site.

Au final l'arborescence de votre site web doit ressembler à cela :

@ racine du site

```
        |----------<> conf
        |                 |---- .htaccess              (Deny From All)
        |                 |---- .htpasswd
        |
        |----------<> comptabilite
        |                 |---- .htaccess              (AuthUserFile    /var/www/html/conf/.htpasswd)
        |                 |---- index.html
        |
        |----------<> direction
        |                 |---- .htaccess              (AuthUserFile    /var/www/html/conf/.htpasswd)
        |                 |---- index.html
        |
        |----------<> production
        |                 |---- .htaccess              (AuthUserFile    /var/www/html/conf/.htpasswd)
        |                 |---- index.html
        |
        |----------<> secretariat
        |                 |---- .htaccess              (AuthUserFile    /var/www/html/conf/.htpasswd)
        |                 |---- index.html
        |
        |---- index.html
        |---- .htaccess                    (ErrorDocument 404 http://www.axius-1.fr/notfound.html)
        |---- notfound.html
```

5. Page d'Erreur 404 Personnalisée

Lorsqu'un visiteur web entre un url erroné ne correspondant à aucune page sur le site, généralement une page d'erreur 404 par défaut s'affiche. Il est possible de personnalisé cette page, voici comment procéder.

A la racine du dossier web (/**var**/**www**/**html**/), créer un fichier **notfound.html** qui sera votre page d'erreur personnalisée. Eventuellement, il conviendra d'y inclure un lien permettant au visiteur égaré de retourner sur la page d'accueil du site.

Au même endroit, créer un fichier **.htaccess** qui comprendra la ligne suivante :

```
ErrorDocument 404          http://www.axius-1.fr/notfound.html
```

Redémarrer le service :

```
# service httpd restart
```

Tester la page d'erreur en rentrant un url erroné du type :

```
# lynx www.axius-1.fr/indx.html
```

Normalement la page d'erreur personnalisée devrait s'afficher.

6. Aller plus Loin

En ce qui concerne l'authentification, il est possible de mettre en place plusieurs niveaux de sécurité.

Par exemple, en reprenant l'arborescence décrite plus haut, comment faire pour :

- restreindre l'accès des pages de la comptabilité aux seuls membres de ce service et au patron.
- restreindre l'accès des pages de la direction au patron et à sa secrétaire.
- restreindre l'accès d'un sous-dossier de la direction uniquement au patron.

@ Racine du site

```
|----------<> conf
|                |---- .htaccess                (Deny From All)
|                |---- .htpasswd
|                |---- .htpasswdcompta
|                |---- .htpasswddir
|                |---- .htpasswdboss
|
|----------<> comptabilite
|                |---- .htaccess (1)            (AuthUserFile  /var/www/html/conf/.htpasswdcompta)
|                |---- index.html
|
|----------<> direction
|                |----------<> boss
|                |                |---- .htaccess(2)        (AuthUserFile  /var/www/html/conf/.htpasswdboss)
|                |                |---- index.html
|                |
|                |---- .htaccess (3)            (AuthUserFile     /var/www/html/conf/.htpasswddir)
|                |---- index.html
|
|----------<> production
|                |---- .htaccess (4)            (AuthUserFile     /var/www/html/conf/.htpasswd)
|                |---- index.html
|
|----------<> secretariat
|                |---- .htaccess (4)            (AuthUserFile     /var/www/html/conf/.htpasswd)
|                |---- index.html
|
|---- index.html
|---- .htaccess                                 (ErrorDocument 404  http://www.axius-1.fr/notfound.html)
|---- notfound.html
```

Voici le détail des fichiers **.htaccess** :

(1)

```
AuthType            Basic
AuthUserFile /var/www/html/conf/.passwdcompta
AuthName            "Accès Sécurisé"
<Limit GET POST>
      require user comptable1 comptable2
</Limit>
```

(2)

```
AuthType            Basic
AuthUserFile /var/www/html/conf/.passwdboss
AuthName            "Accès Sécurisé"
<Limit GET POST>
      require user boss
</Limit>
```

(3)

```
AuthType            Basic
AuthUserFile /var/www/html/conf/.passwddir
AuthName            "Accès Sécurisé"
<Limit GET POST>
      require user boss secretaire
</Limit>
```

(4)

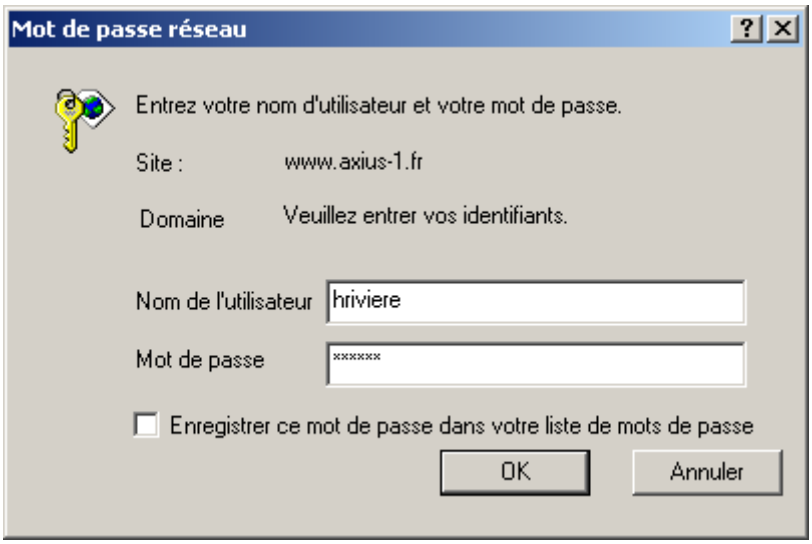
```
AuthType            Basic
AuthUserFile /var/www/html/conf/.passwd
AuthName            "Accès Sécurisé"
<Limit GET POST>
       require valid-user
</Limit>
```

Ceci fait, il suffit de déclarer les utilisateurs dans les bons fichiers :

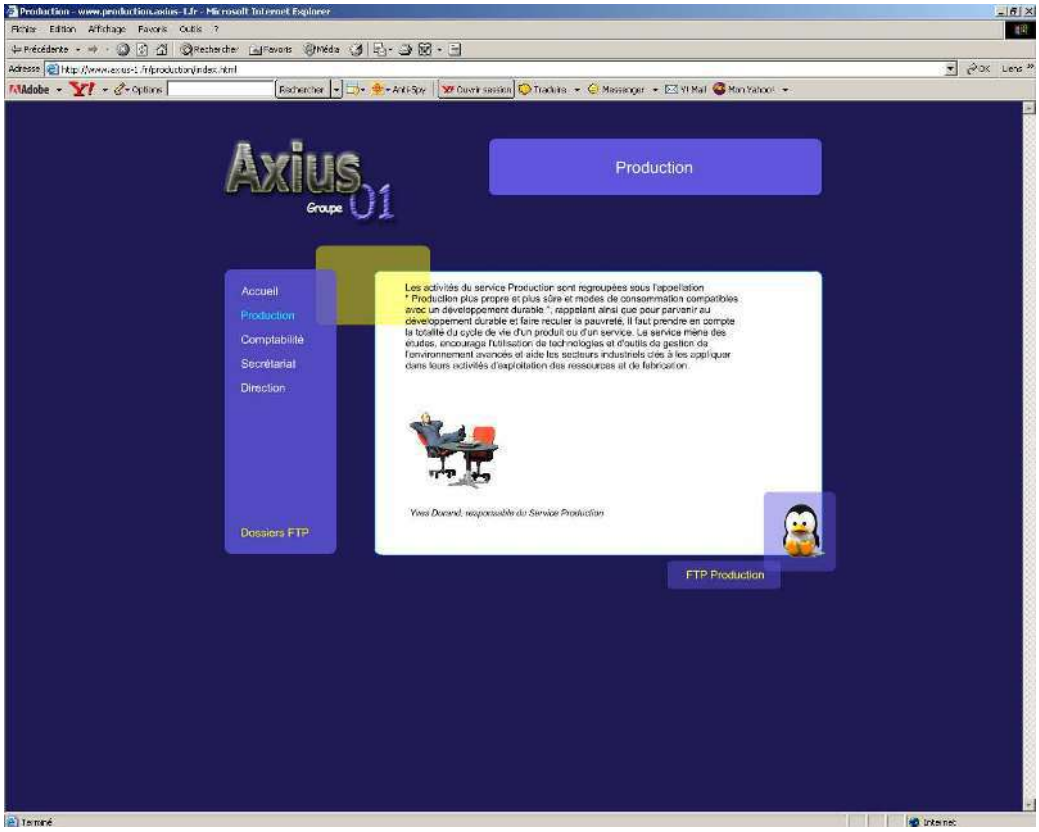
```
# htpasswd -c .htpasswd producteur1
# htpasswd .htpasswd producteur2
# htpasswd .htpasswd producteur3
# htpasswd .htpasswd secretaire1
# htpasswd .htpasswd secretaire2

# htpasswd -c .htpasswdcompta comptable1
# htpasswd . htpasswdcompta comptable2

# htpasswd -c .htpasswddir boss
# htpasswd .htpasswddir secretaire

# htpasswd -c .htpasswdboss boss
```

Page d'accueil

Fenêtre d'authentification pour entrer sur les pages des services. La page de la comptabilité n'est accessible qu'au groupe comptabilité.



Page Production

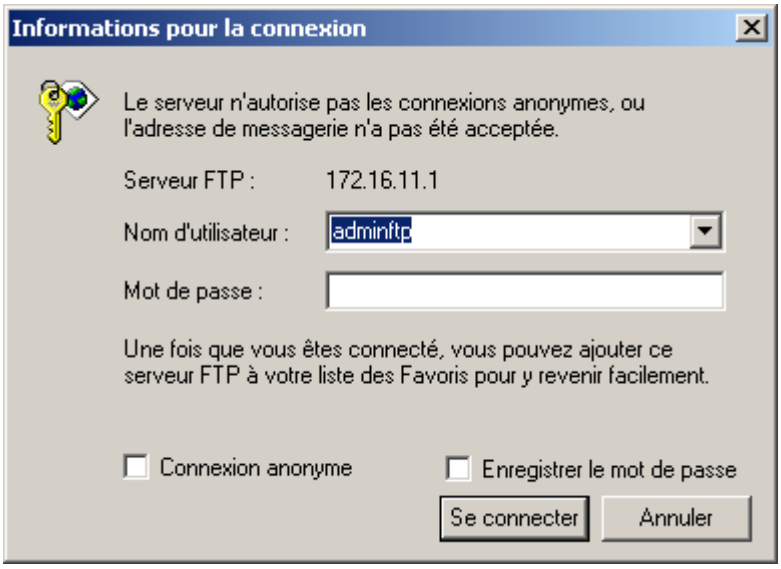


Page Comptabilité

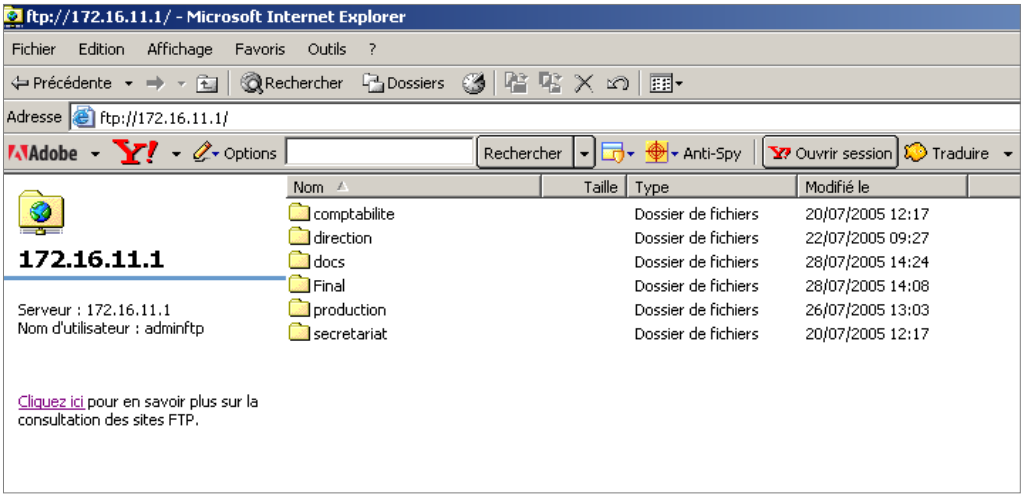
Page Secrétariat

Page Direction

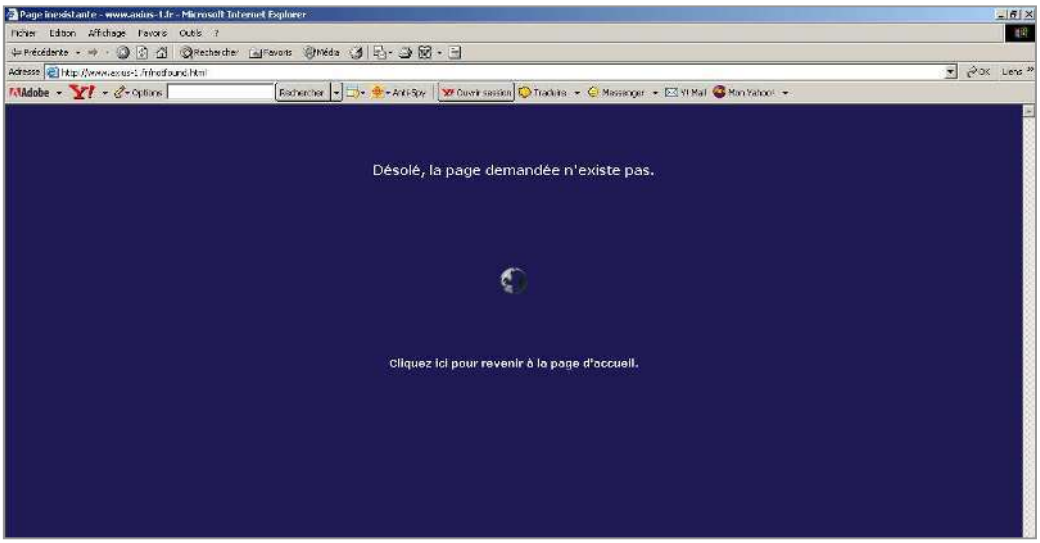
Fenêtre d'authentification pour entrer sur les pages FTP



Lien vers le FTP



Page de re-direction en cas d'erreur 404



Groupe 01

SERVEURS DHCP

Dynamic Host Configuration Protocol

1. Préparation

Rappel :
Distribution : Mandriva 2005 (Mandrake 10.2)
Nom du serveur : SERVEUR-LINUX-1
Adresse IP du serveur : 172.16.11.9
Nom du domaine : axius-1.fr
Plage d'adresses IP : 172.16.11.200 à 172.16.11.254

Attribution d'adresses fixes aux différents serveurs FTP, DNS et WEB

2. Installation

Vérification et installation des paquetages nécessaires, à savoir : **dhcp-common , dhcp-server**.

La commande **urpmi -a [mot recherché]** se charge de cette opération. Elle indique si un paquetage correspondant existe, si il est installé, et si d'autres paquetages sont nécessaires à son bon fonctionnement. Puis elle nous demande d'insérer le disque d'installation adéquate.

L'attribut **–a** indique que si plusieurs paquetages coïncident avec la sous chaîne donnée. Les prendre tous.

3. Création et Paramétrage du Serveur DHCP

Dans un premier temps, il est nécessaire de créer le fichier de configuration du DHCP à l'endroit **/etc /dhcpd.conf**
un exemple se trouve dans **/etc/dhcpd.conf.sample**.

```
# mise a jour dynamiques du dns

ddns-domainname "axius-1.fr";
ddns-update-style none;
ddns-updates off;

# tous les clients sont acceptes ,meme si l'on ne connait pas leur adresse mac.

allow unknown-clients;

# duree de vie du bail

max-lease-time 604800;
default-lease-time 86400;

# les options que l'on va donner aux clients

option domain-name-servers 172.16.11.9;
option domain-name "axius-1.fr";
option subnet-mask 255.255.0.0;
# gateway
option routers 172.16.11.9;

# definition du sous reseau
# plage d'ip a distribuer

subnet 172.16.11.0 netmask 255.255.255.0 {
range  172.16.11.200 172.16.11.254;


        # adressage fixe

        host oliveportable {
                hardware ethernet 00:0A:E4:48:D8:3E;
                fixed-address 172.16.11.180;
        }
}
```

Note importante : le daemon DHCPd écoute par défaut sur toutes les interfaces réseau actives sur le serveur. Ce n'est pas forcement souhaitable, c'est même assez souvent ennuyeux.
Fort heureusement, ce comportement par défaut peut être modifié, mais pas dans le fichier de configuration. Il faut utiliser un paramètre dans la ligne de commande qui va démarrer DHCPd.

Dans le cas de Mandrake, il faut éditer le script **/etc/rc.d/init.d/dhcpd** Et modifier INTERFACES= « eth0 »

Dans le fichier **/etc /dhcpd.conf**, il y a des directives, qui sont obligatoires :

- les directives ddns-xxx serviront plus tard, ce sera la cerise sur le gâteau, pour ceux qui utilisent BIND 9 (le serveur DNS). Elles doivent cependant figurer dans la configuration pour que le démon dhcpd puisse démarrer,
- **allow unknown-clients**
  C'est en principe la configuration par défaut, mais autant le spécifier. Ca veut dire que le DHCP acceptera tous les clients qui feront une requête DHCP. Dans le cas contraire, le serveur ne répondrait qu'aux machines dont il connaît l'adresse MAC.
- Il existe une subtile différence entre les directives max-lease-time et default-lease-time, la page "man dhcpd.conf" vous indiquera quelle est cette différence.

Et des options qui seront dans la pratique, des paramètres de configuration optionnels. Ici :

- **domain-name-servers**
  qui attribuera aux hôtes une adresse de DNS. Dans l'exemple, notre DNS à nous. Si nous n'en avons pas, il faudra ici mettre l'IP du DNS de notre fournisseur d'accès. Eventuellement, nous pouvons spécifier plusieurs DNS.
- **domain-name**
  est vraiment optionnel, ça permettra aux clients de savoir dans quel domaine ils sont enregistrés
- **routers**
  c'est la passerelle par défaut. Il pourrait y avoir plusieurs routeurs, mais tous les systèmes ne savent pas gérer de façon efficace une information contenant plusieurs passerelles.

Toutes les options qui figurent avant le paragraphe "subnet 172.16.11.0 netmask 255.255.255.0" sont des options globales, il n'y a ici aucune option d'étendue (de sous-réseau) de définie.
Cette configuration doit nous permettre de fonctionner dans notre contexte. Il nous suffit de lancer ou de relancer le serveur : `# service dhcpd restart`

4. Configuration Clients Windows

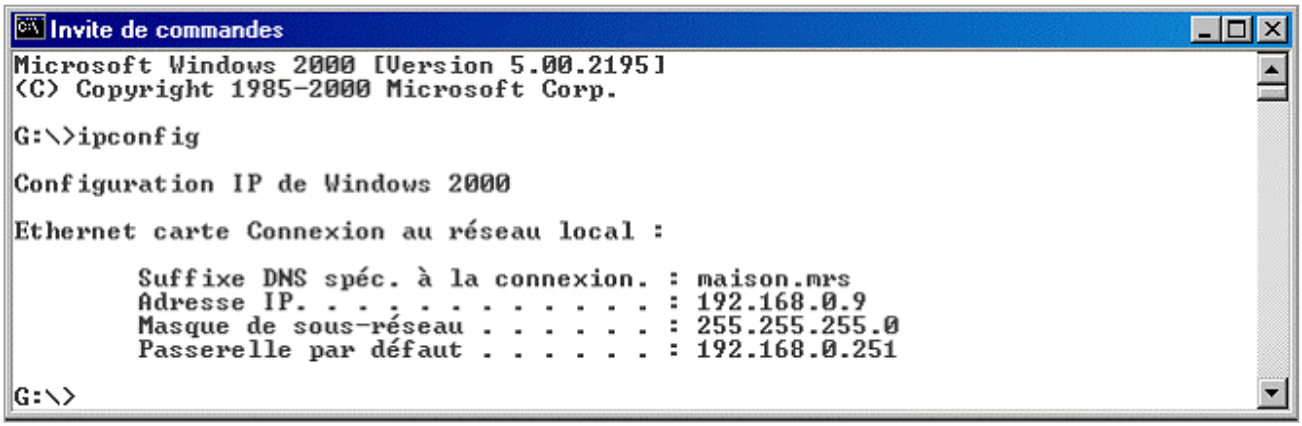
Windows NT4/2000/XP

La configuration

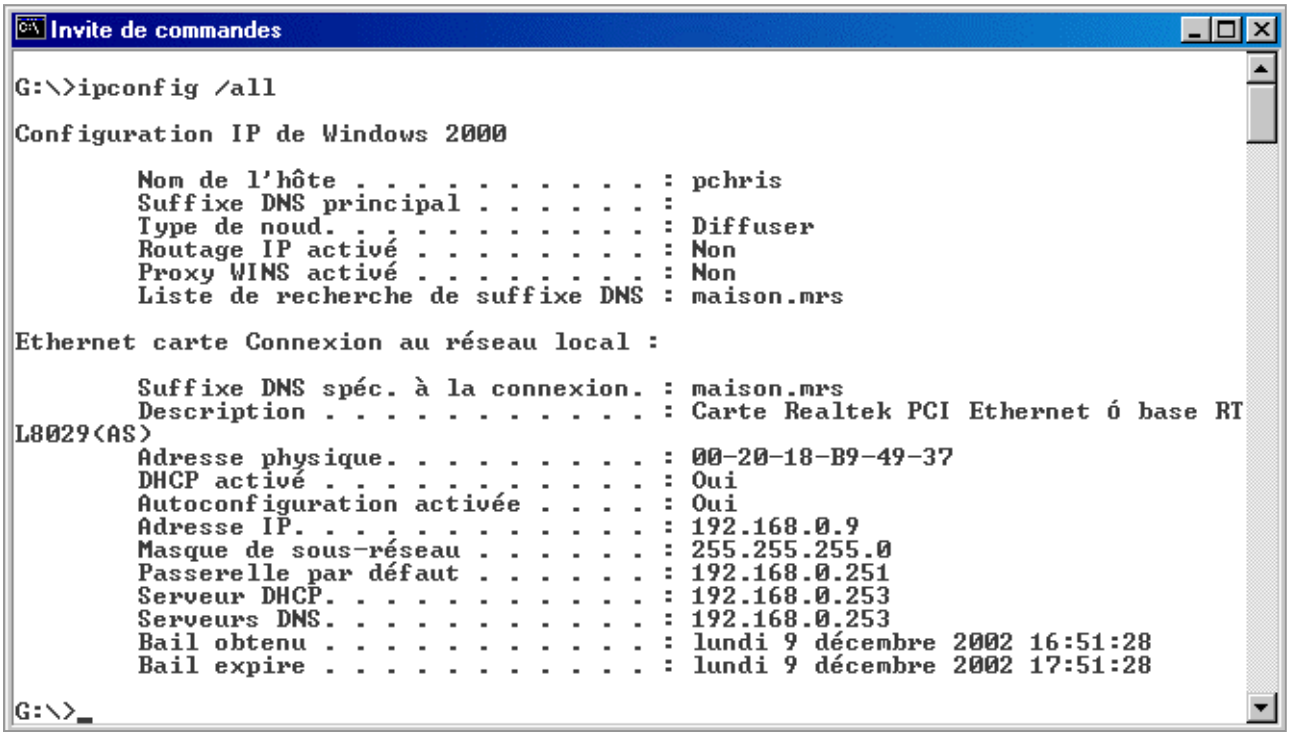
La configuration se fait dans le panneau de configuration, icône "réseau", onglet "protocoles", puis "propriétés" de TCP/IP. Là, vous avez indiqué que la carte doit recevoir une adresse IP dynamiquement.

Vérification

Tapez dans une console, la commande **ipconfig**



Votre adresse doit être affichée. Si vous voulez tous les détails, utilisez la commande "ipconfig /all" :



La commande "**ipconfig**" permet également:

- De résilier le bail: "**ipconfig /release**"
- De renouveler le bail: "**ipconfig /renew**"

C'est cette commande qui est à utiliser pour essayer de récupérer une adresse IP lorsque vous avez des problèmes.

Notes

- Les rubriques "Bail obtenu" et "Expiration du bail" contiennent des valeurs calculées par votre machine. Le serveur DHCP ne donne que la durée.
- La commande en mode graphique "winipcfg" n'existe pas nativement sous Windows NT mais vous pouvez la récupérer dans le kit de ressources techniques (téléchargeable sur le site MS en cherchant bien :-).
  N'essayez pas d'utiliser celle de Windows 95/98, les dll winsock32 utilisées ici ne sont pas compatibles.

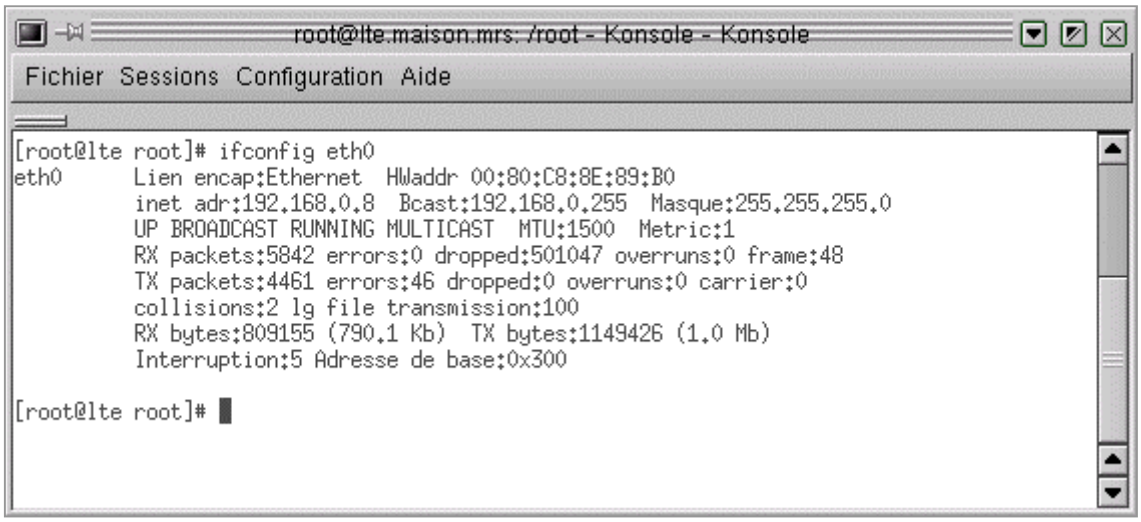
5. Configuration Client Linux

Vérifiez l'état de votre connexion
Dans **/etc/sysconfig/network-scripts**, il y a un fichier intitulé : ifcfg-eth0. Il doit contenir au moins ces lignes :

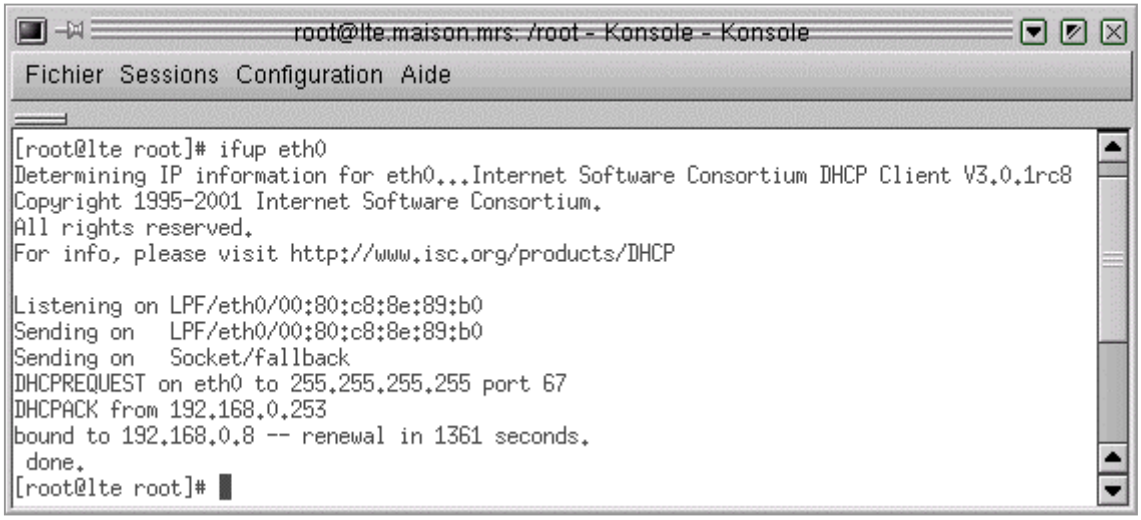
```
DEVICE="eth0"
BOOTPROTO="dhcp"
IPADDR=""
NETMASK=""
ONBOOT="yes"
```

C'est assez parlant pour ne pas nécessiter d'explications particulières.

La commande "**ifconfig eth0**" devrait vous donner quelque chose comme ceci :



Si rien n'apparaît, c'est que votre interface n'est pas activée. Essayez alors ifup eth0 :



Cette commande affiche l'état de Eth0, mais elle ne donne pas toutes les informations que l'on obtient sous Windows avec **winipcfg** ou **ipconfig**. Si vous voulez tout savoir, il faut aller dans le répertoire "**/var/lib/dhcp**" et regarder le fichier **dhclient.leases**. Celui-ci contient l'historique des dialogues DHCP :

Exemple venant d'une copie écran :

```
lease {
interface "eth0";
fixed-address 192.168.0.8;
option subnet-mask 255.255.255.0;
option routers 192.168.0.253;
option dhcp-lease-time 3600;
option dhcp-message-type 5;
option domain-name-servers 192.168.0.253;
option dhcp-server-identifier 192.168.0.253;
option domain-name "maison.mrs";
renew 2 2002/12/10 08:49:42;
rebind 2 2002/12/10 09:14:05;
expire 2 2002/12/10 09:21:35;}
```

Notez que ce fichier peut être beaucoup plus long. Cherchez dedans le dernier bail obtenu. Constatez que vous avez bien la trace de toutes les informations que notre serveur DHCP est capable d'envoyer à ses clients.

Particularités du client DHClient

Grâce aux informations conservées dans ce fichier dhclient.leases, ce client adopte un comportement un peu particulier, que l'on ne retrouve pas dans celui de Microsoft, par exemple.

Lorsqu'un hôte a obtenu un premier bail de la part du DHCP, l'adresse du serveur DHCP est conservée et, même après extinction et redémarrage de l'hôte au bout d'un temps bien supérieur à la durée de son bail, le client commencera par envoyer directement un DHCP request au serveur qu'il connaît. Cette particularité peut dérouter lorsque l'on espionne les dialogues DHCP sur le réseau.

6. Mise a jour automatique du DNS par le DHCP

Attention, cette méthode est expérimentée avec DHCPd 3.0 et BIND 9.2

Il y a en réalité, deux moyens de le faire. Soit c'est le client qui va s'annoncer au DNS, une fois qu'il aura récupéré son bail, ça présente deux inconvénients :

- Tous les clients DHCP ne savent pas le faire,
- ça oblige à ce que tous les hôtes du réseau soient autorisés à effectuer des modifications sur le DNS, ce qui est loin d'être une solution sûre.

Soit, c'est le DHCP qui sera chargé d'effectuer les mises à jour sur DNS, à chaque attribution d'un bail. C'est bien plus sûr, on est certain que ça fonctionnera avec tous les clients, ça augmente juste un peu la charge du serveur. Nous allons choisir cette seconde solution.
Cette méthode, qui est bien entendu très intéressante lorsque l'adressage est dynamique, c'est à dire que l'IP d'un hôte est susceptible de changer dans le temps, l'est moins si l'on a choisi d'attribuer une IP fixe à un ou plusieurs hôtes. D'ailleurs, par défaut, la mise à jour du DNS ne s'effectuera pas dans ces cas. Il y a cependant une clause qui permet de forcer cette mise à jour et nous allons l'utiliser.

6.1 Du côté de BIND

Il faut lui indiquer que les zones de notre domaine peuvent être mise à jour par le serveur DHCP. Il existe une méthode sécurisée consistant à utiliser des clés MD5 pour l'authentification, nous ne l'utiliserons pas ici, mais suivant le cas de figure, ça peut être très vivement conseillé.

Nous allons juste signaler l'adresse IP nécessaire : 127.0.0.1, puisque les deux services tournent sur la même machine.
Nous allons modifier le fichier /**etc**/**named.conf** comme suit :

```
...
# La zone directe du domaine
zone "axius-1.fr" {
        notify no ;
        type master;
        file "axius-ally.fr";
        allow-update {
                127.0.0.1;
                };
        };
# La zone de recherche inverse
zone "11.16.172.in-addr.arpa" {
        notify no ;
        type master;
        file "axius-ally.fr.reverse";
        allow-update {
                127.0.0.1;
                };
        ...
        }
```

Si certaines machines avaient une configuration fixe et étaient référencées dans le DNS, détruisez leurs enregistrements aussi bien dans la zone directe que dans la zone inverse, sinon, la mise à jour dynamique échouera pour ces noms d'hôtes.

Côté Bind, c'est tout ce qu'il y a à faire, dans notre cas. Il ne faut bien entendu pas oublier de redémarrer le service.

6.2 Du côté de DHCPd

Là, il y a plus de travail. Il faut modifier le fichier /etc/dhcpd.conf de la manière suivante :

```
# méthode de mise à jour du DNS :
ddns-update-style interim;

# mise à jour autorisée
ddns-update on;

# ici, on force la mise à jour par le serveur DHCP
ignore client-updates;

# on force également la mise à jour des IP fixes
update-static-leases on;
```

Bien que ça puisse parfois fonctionner sans, il vaut tout de même mieux prendre la précaution d'ajouter en fin de fichier, ceci afin de définir clairement quel DNS doit être mis à jour pour ces zones :

```
zone axius-1.fr. {
   primary 172.16.11.9;
   }

zone 11.16.172.in-addr.arpa. {
   primary 172.16.11.9;
   }
```

A aménager, bien entendu, en fonction de votre propre configuration. Faites bien attention à la syntaxe. N'oubliez aucun point dans les noms des zones, refermez les accolades et finissez vos directives par un point-virgule.
Relancez le service DHCPd, ça devrait maintenant fonctionner.

6.3 Mise en Garde

La mise à jour dynamique de DNS nécessite de connaître le nom de l'hôte qui vient de récupérer un bail, surtout si vous voulez conserver une cohérence entre les noms d'hôtes attribués localement et les noms DNS.

Il faut savoir que si le client DHCP de Windows envoie le nom d'hôte lors de la requête DHCP, les clients Linux comme dhcp client et même dhcpcd ne le font pas par défaut. Si vous n'y prenez garde, vos machines recevront bien leur bail, mais la mise à jour DNS ne s'effectuera pas.

Avec dhcp client, il faut créer un fichier /**etc/dhclient.conf** qui contiendra au moins la ligne :

```
send host-name "lenomdelamachine" ;
```

Consultez la doc de dhcp client pour savoir tout ce que l'on peut configurer par l'entremise de ce fichier,

6.4 Vérifications

Dans /**var**/**named**, à la première attribution d'un nouveau bail, vous devez voir apparaître deux nouveaux fichiers de zone, avec le même nom que les zones de votre domaine, mais avec un suffixe .jnl. Ces fichiers constituent la preuve que ça fonctionne, ce sont des journaux. N'essayez pas de les lire, ils sont en mode binaire. Beaucoup plus tard, vous pourrez constater que les fichiers de zone ont eux aussi été modifiés. De nouveaux enregistrements A sont apparus, suivis d'un enregistrement TXT. Ne modifiez plus ces enregistrements, surtout, n'enlevez pas l'enregistrement TXT, il permet d'indiquer si le champ précédent est issu d'une mise à jour dynamique ou non, et son utilité est primordiale pour les mises à jour futures.

Les outils classiques, **host** sous Linux, **nslookup** sous Windows 2000/XP vous permettront de vérifier les réponses de votre DNS.

6.5 Remarques

Il faudrait étudier avec soin toute la documentation de bind et de dhcpd pour maîtriser parfaitement le mécanisme de mise à jour dynamique, j'avoue ne pas encore avoir eu le courage de le faire.

Vous risquez des ennuis si vous faites une mise à jour de la partie statique de votre zone. Après redémarrage de bind, il se peut que la zone ne fonctionne plus. Observez le journal /var/log/messages, vous aurez probablement une alerte vous indiquant que les journaux ne sont plus exploitables. Dans ce cas, détruisez les fichiers jnl et relancez named. Bien entendu, vous aurez sans doute perdu quelques mises à jour dynamiques, mais ça devrait rentrer dans l'ordre lorsque les baux seront renouvelés.

Mise en œuvre des serveurs Linux et rédaction des procédures :

GROUPE TSSI 01

BOUTIN Eric, KREMER Didier, MOLLET Patrick, RICHARTE Olivier, RICHOZ Philippe

Réalisation du site en Flash puis intégration des pages en HTML :

BOUTIN Eric, MOLLET Patrick

Réalisation des logos « TSSI 2005 » et « Axius Groupe 01 » :

MOLLET Patrick

La Valette, Juillet 2005