

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Administration Réseaux

Pr. Abdelhak LAKHOUAJA

Département d'Informatique
Faculté des Sciences
Université Mohammed Premier
Oujda

SMI-S6

Année universitaire : 2019/2020

- Page web du cours :
<http://lakhouaja.oujda-nlp-team.net/teaching/bachelor-level/administration-reseaux/>
- Page web personnelle :
<http://lakhouaja.oujda-nlp-team.net/>

Chapitre 1

Introduction et rappels

But de l'administration système

- Sécurité et bon fonctionnement du réseau
- Mettre à la disposition des utilisateurs les outils nécessaires
- Veiller à la bonne conduite des utilisateurs

- L'utilisation des outils graphiques ne permet pas de comprendre le comportement des systèmes d'exploitation.
- L'utilisation des **scripts** (bash, perl, powershell (Windows), ...) permet de :
 - créer plusieurs comptes en même temps (voir TP1) ;
 - modifier et sauvegarder plusieurs fichiers ;
 - ...

Utilisation du compte administrateur

- Pour éviter des erreurs graves, il est déconseillé d'utiliser le compte administrateur pour des tâches courantes.
- Sous Linux :
 - le compte administrateur est **root** et son invite est **#**
 - l'invite d'un utilisateur normal est **\$**
 - pour certaines commandes, utilisez des alias :

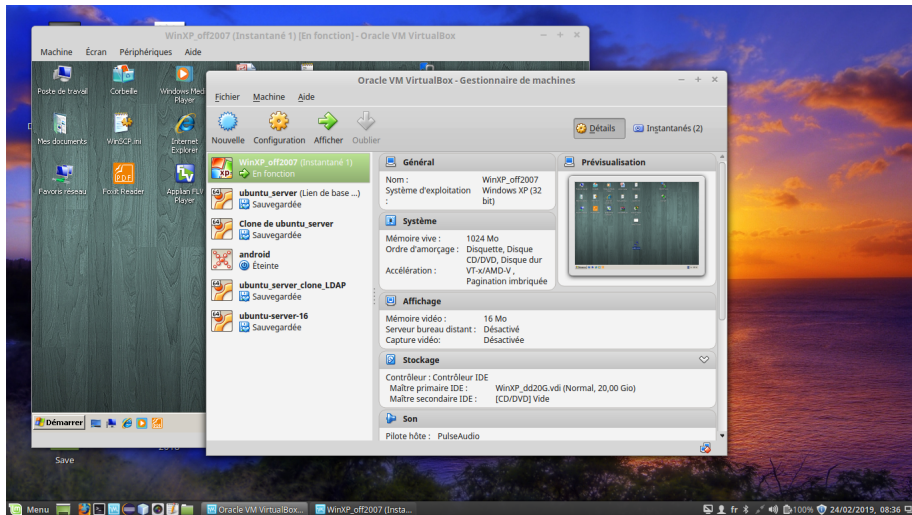
```
alias mv='mv -i '  
alias rm='rm -i '  
alias cp='cp -i '
```

La virtualisation est utilisée pour exécuter plusieurs systèmes d'exploitation en même temps sans avoir à redémarrer la machine. Elle permet par exemple d'exécuter un système Linux sur Windows ou inversement.

Elle permet de tester différents systèmes et d'exploitation. En cas de problème, la machine n'est pas affecté. Elle permet aussi de tester des tâches d'administration sur un système sur lequel on a des droits limités.

Il existe différentes technologies qui permettent la virtualisation : Xen, Qemu, KVM, Vmware (www.vmware.com), VirtualBox (www.virtualbox.org) etc.

VirtualBox (Windows XP qui tourne sous Linux)



- Netkit est un projet Open-Source qui permet de simuler des réseaux informatiques. Il est basé sur des machines Linux.
- Voir TP.

Gestion des utilisateurs

Création d'un nouveau utilisateur

La commande :

adduser smi

permet de créer l'utilisateur **smi**.

Elle a pour effet :

- d'ajouter une ligne à la fin du fichier « **/etc/passwd** », semblable à :
smi:x:1001:1001:,,:/home/smi:/bin/bash
- de créer le répertoire : **/home/smi**.
- d'ajouter le groupe **smi** dans le fichier « **/etc/group** ».
- d'ajouter le mot de passe crypté dans le fichier « **/etc/shadow** ».
Ce fichier est lisible seulement par root.

⇒ Dans un script, il est préférable d'utiliser la commande :
useradd (voir TP).

Gestion des utilisateurs

Suppression d'un utilisateur

- La commande :
userdel smi
permet de supprimer l'utilisateur **smi**.
- Options :
 - -r : supprime le dossier personnel de l'utilisateur (et son contenu !).
 - -f : forcer la suppression des fichiers de l'utilisateur.

Gestion des groupes

- La commande :
groupadd nom_groupe
permet d'ajouter un nouveau groupe.
- La commande :
groupdel nom_groupe
permet de supprimer un groupe.
- La commande :
adduser smi nom_groupe
permet d'ajouter l'utilisateur **smi** au groupe **nom_groupe**.
- La commande :
groups smi
permet d'afficher les groupes d'un utilisateur (informations données par la commande **id**)

Administration des utilisateurs

- **su** : pour devenir administrateur (root).
- **su smi** : pour devenir « **smi** ».
- **who** : affiche les utilisateurs connectés.
- **whoami** (qui suis-je ?) : affiche l'utilisateur en cours.
- **last** : affiche les utilisateurs qui se sont connectés au système.

Éteindre/redémarrer le système

- En mode graphique : dépend de la distribution.
- En mode console :
 - **halt** : éteint le système immédiatement (**shutdown -h now**).
 - **reboot** : redémarre le système immédiatement (**shutdown -r now**).
 - **shutdown -h 10** : éteint le système dans 10 minutes. Les utilisateurs reçoivent un avertissement dans leurs consoles.
 - **shutdown -r 10** : redémarre le système dans 10 minutes.

- **ifconfig/ipconfig** : pour configurer une interface réseau ou afficher les informations concernant les interfaces réseaux (commandes déjà utilisées en S5).
- **route** : pour afficher/modifier la table de routage.
- **ping** : pour tester la connexion entre deux machines.
- **host** : c'est un utilitaire simple de conversion de noms DNS en adresse IP (et vice versa) :

Exemple :

```
alkhalil: $ host www.ump.ma  
www.ump.ma has address 196.200.156.5
```

- **nslookup** : disponible sous Linux et sous Windows, comme la commande **host**, cette commande permet d'afficher l'adresse IP d'un nom DNS.

Exemple :

```
alkhalil:~$ nslookup www.ump.ma
Server: 127.0.1.1
Address: 127.0.1.1#53
```

```
Non-authoritative answer:
Name: www.ump.ma
Address: 196.200.156.5
```

- **netstat**
- **tcpdump/wireshark**

netstat -s | -a | -r | -n

Netstat fournit des statistiques sur les :

- paquets émis ou reçus
- erreurs
- collisions
- protocoles utilisés

- le nom et l'état des interfaces du système

```
netstat -i
```

- le contenu de la table de routage

```
netstat -r | n
```

- ainsi que l'état de tous les sockets

```
netstat -a
```

Wireshark ¹ est un outil d'analyse des réseaux qui permet de capturer et d'analyser les paquets qui circulent sur le réseau. Il peut être utilisé pour capturer les paquets qui circulent sur une interface ou pour visualiser le contenu d'un fichier qui des paquets capturés par un autre utilitaire tel que **tcpdump**. Il est multi-plateforme, il fonctionne sous Linux, Windows, MacOS, ...

1. Site officiel : <https://www.wireshark.org/>

interface de wireshark

The image shows the Wireshark 1.10.6 interface. The title bar reads "exam_capture [Wireshark 1.10.6 (v1.10.6 from master-1.10)]". The menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Tools, Internals, and Help. The toolbar contains various icons for file operations, capture, analysis, and display. The filter bar shows "Filter: Expression... Clear Apply Enregistrer".

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	6e:5f:98:37:0c:07	2e:18:cc:d4:8c:e7	ARP	42	Who has 192.168.1.2? Tell
2	0.000051	2e:18:cc:d4:8c:e7	6e:5f:98:37:0c:07	ARP	42	192.168.1.2 is at 2e:18:cc:
3	3.858131	192.168.1.2	192.168.1.1	TCP	74	54490 > http [SYN] Seq=0 Wi
4	3.858191	192.168.1.1	192.168.1.2	TCP	74	http > 54490 [SYN, ACK] Seq
5	3.858279	192.168.1.2	192.168.1.1	TCP	66	54490 > http [ACK] Seq=1 Ac
6	3.858509	192.168.1.2	192.168.1.1	HTTP	160	GET /xab HTTP/1.0 [Packet s

▶ Frame 7: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
▶ Ethernet II, Src: 6e:5f:98:37:0c:07 (6e:5f:98:37:0c:07), Dst: 2e:18:cc:d4:8c:e7 (2e:18:cc:d4:8c:e7)
▶ Internet Protocol Version 4, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.2 (192.168.1.2)
▶ Transmission Control Protocol, Src Port: http (80), Dst Port: 54490 (54490), Seq: 1, Ack: 95, Len: 0

0000 2e 18 cc d4 8c e7 6e 5f 98 37 0c 07 08 00 45 00n_.7....E.
0010 00 34 e6 64 40 00 40 06 d1 0b c0 a8 01 01 c0 a8 .4.d@.@.
0020 01 02 00 50 d4 da 98 6a d0 d0 98 60 f0 4f 80 10 ...P...j ...`.0..
0030 0b 50 b3 06 00 00 01 01 08 02 00 08 36 b4 00 08 P 6

File: "/home/lakhouaja/SMI_S5/TP2/... Packets: 22 · Display... Profile: Default

Les colonnes se présentent comme suit :

No. : représente le numéro du paquet ;

Time : représente le temps de capture du paquet ;

Source : représente l'adresse IP ou MAC de la source ;

Destination : représente l'adresse IP ou MAC destination ;

Protocol : représente le type du protocole capturé ;

Length : représente la taille du paquet (en octets) ;

Info : représente une brève information concernant le paquet.

sous Linux, comme pour la commande **tcpdump**, **wireshark** ne peut pas être utilisé pour capturer des données en mode simple utilisateur. Pour capturer des données il faut passer en mode administrateur.

Description de l'interface

L'interface est découpée en trois zones :

- 1 supérieure : contient l'ensemble des paquets capturés

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	6e:5f:98:37:0c:07	2e:18:cc:d4:8c:e7	ARP	42	Who has 192.168.1.2? Tell
2	0.000051	2e:18:cc:d4:8c:e7	6e:5f:98:37:0c:07	ARP	42	192.168.1.2 is at 2e:18:cc:
3	3.858131	192.168.1.2	192.168.1.1	TCP	74	54490 > http [SYN] Seq=0 Win
4	3.858191	192.168.1.1	192.168.1.2	TCP	74	http > 54490 [SYN, ACK] Seq=
5	3.858279	192.168.1.2	192.168.1.1	TCP	66	54490 > http [ACK] Seq=1 Ack
6	3.858509	192.168.1.2	192.168.1.1	HTTP	160	GET /xab HTTP/1.0 [Packet s

- 2 centrale : affiche les détails d'un paquet sélectionné sous forme de couches

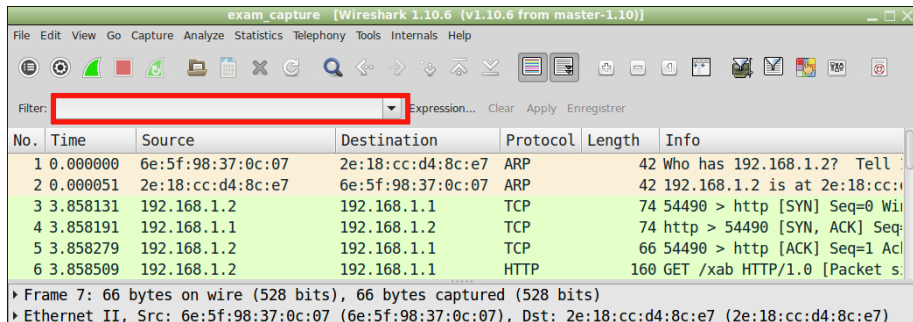
▶ Frame 7: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
▶ Ethernet II, Src: 6e:5f:98:37:0c:07 (6e:5f:98:37:0c:07), Dst: 2e:18:cc:d4:8c:e7 (2e:18:cc:d4:8c:e7)
▶ Internet Protocol Version 4, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.2 (192.168.1.2)
▶ Transmission Control Protocol, Src Port: http (80), Dst Port: 54490 (54490), Seq: 1, Ack: 95, Len: 0

- 3 Zone inférieure : présente le paquet sous format octale et ASCII

0000	2e 18 cc d4 8c e7 6e 5f 98 37 0c 07 08 00 45 00n_ .7....E.
0010	00 34 e6 64 40 00 40 06 d1 0b c0 a8 01 01 c0 a8	.4.d@.@.
0020	01 02 00 50 d4 da 98 6a d0 d0 98 60 f0 4f 80 10	...P...j ...`.0..
0030	0b 50 b3 06 00 00 01 01 08 03 00 08 36 b4 00 08	P 6

Filtres

Il est possible de ne pas afficher tous les paquets en les filtrant. Par exemple, on peut afficher juste les paquets **http**, en tapant **http** dans la zone **Filter** :. Il est possible aussi d'utiliser des expressions.



The screenshot shows the Wireshark 1.10.6 interface. The title bar reads "exam_capture [Wireshark 1.10.6 (v1.10.6 from master-1.10)]". The menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Tools, Internals, and Help. The toolbar contains various icons for file operations, capture control, and analysis. The "Filter:" field is highlighted with a red rectangle and contains the text "http". To the right of the filter field are buttons for "Expression...", "Clear", "Apply", and "Enregistrer". Below the filter field is a table of captured packets.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	6e:5f:98:37:0c:07	2e:18:cc:d4:8c:e7	ARP	42	Who has 192.168.1.2? Tell
2	0.000051	2e:18:cc:d4:8c:e7	6e:5f:98:37:0c:07	ARP	42	192.168.1.2 is at 2e:18:cc:
3	3.858131	192.168.1.2	192.168.1.1	TCP	74	54490 > http [SYN] Seq=0 Wi
4	3.858191	192.168.1.1	192.168.1.2	TCP	74	http > 54490 [SYN, ACK] Seq
5	3.858279	192.168.1.2	192.168.1.1	TCP	66	54490 > http [ACK] Seq=1 Ac
6	3.858509	192.168.1.2	192.168.1.1	HTTP	160	GET /xab HTTP/1.0 [Packet s

Below the table, the packet details pane shows:

- ▶ Frame 7: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
- ▶ Ethernet II, Src: 6e:5f:98:37:0c:07 (6e:5f:98:37:0c:07), Dst: 2e:18:cc:d4:8c:e7 (2e:18:cc:d4:8c:e7)

Exemples de filtres

Filtre/expression	Signification
tcp	afficher seulement les paquets TCP
ip.src==192.168.1.2	afficher seulement les paquets qui sortent de 192.168.1.2
ip.dst==192.168.1.1 && http	afficher les paquets HTTP qui partent vers 192.168.1.1
ip && !udp	afficher les paquets IP mais n'afficher pas les paquets UDP

Chapitre 2

Serveurs web

Protocole HTTP : (HyperText Transfer Protocol)

- Le protocole HTTP est le protocole de transport de données le plus utilisé sur Internet depuis 1990.
- La version 0.9 était uniquement destinée à transférer des données sur Internet (en particulier des pages Web écrites en HTML.
- La version 1.0 du protocole (la plus utilisée) permet désormais de transférer des messages avec des en-têtes décrivant le contenu du message.
- Le but du protocole HTTP est de permettre un transfert des fichiers entre un navigateur (le client) et un serveur Web.

Pourquoi utiliser HTTP ?

HTTP est devenu le protocole de communication de l'Internet. Il :

- est disponible sur toutes les plates-formes ;
- est simple. Ne requière que peu de support pour fonctionner correctement ;
- offre un niveau de sécurité simple et efficace ;
- est utilisable à travers des pare-feu.

HTTP fonctionne selon le schéma classique client/serveur :

- connexion du client vers le serveur ;
- demande d'une information via une **méthode** ;
- renvoi de l'**information** ou une **erreur** ;
- déconnexion.

1xx : Information

2xx : Succès (par exemple : 200 ok).

3xx : Redirection.

4xx : Erreurs (par exemple : 404 Not Found).

5xx : Erreurs venant du serveur HTTP (par exemple : 501 Not Implemented).

On verra en TP plus de codes de retours.

Les méthodes HTTP sont les suivantes :

- GET** : demande de la ressource située à l'URL spécifiée ;
- HEAD** : demande de l'en-tête de la ressource située à l'URL spécifiée ;
- POST** : envoi de données au programme situé à l'URL spécifiée ;
- PUT** : envoi de données à l'URL spécifiée ;
- DELETE** : suppression de la ressource située à l'URL spécifiée.

Requête HTTP

Une requête HTTP est un ensemble de lignes envoyé au serveur par le navigateur. Elle comprend :

- une ligne de requête précise la méthode qui doit être appliquée, et la version du protocole utilisée. La ligne comprend trois éléments séparés par un espace :
 - la méthode ;
 - l'URL ;
 - la version du protocole utilisé par le client (généralement HTTP/1.0) ;

Exemple :

GET / HTTP/1.0

- les champs d'en-tête de la requête : il s'agit d'un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la requête et/ou le client (Navigateur, système d'exploitation, ...).

Une réponse HTTP est un ensemble de lignes envoyées au navigateur par le serveur. Elle comprend :

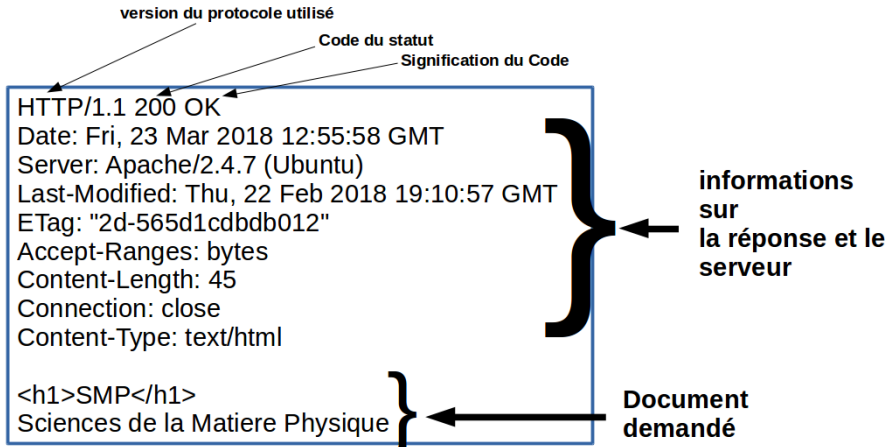
- 1 Une ligne de statut composée de trois éléments séparés par un espace :
 - La version du protocole utilisé
 - Le code de statut
 - La signification du code
- 2 Les champs d'en-tête de la réponse : un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la réponse et/ou le serveur.
- 3 Le corps de la réponse : contient le document demandé

Exemple de requête/réponse HTTP

Requête :

GET / HTTP/1.0

Réponse :

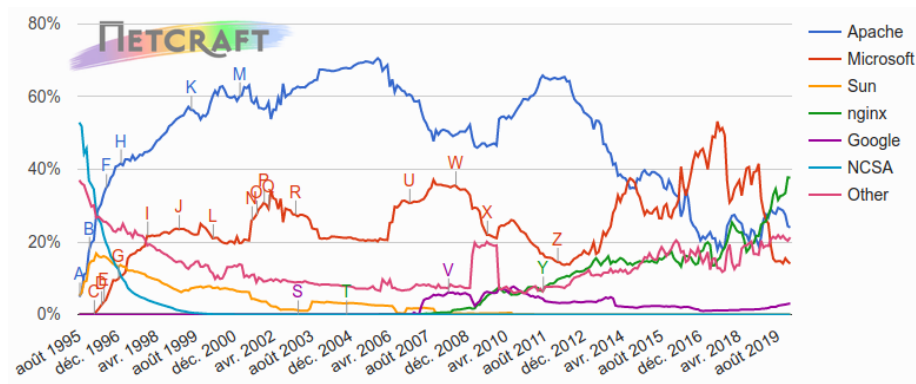


Principaux serveurs :

- Apache
- Microsoft : Internet Information Server (IIS)
- nginx
- gws (Google Web Server)

Serveurs Web en ligne 2020

D'après Netcraft², le serveur apache est le deuxième parmi les serveurs web les plus utilisés.



2. <http://survey.netcraft.com>

Serveurs Web en ligne 2020

Le tableau suivant, montre la part du marché des principaux fournisseurs des serveurs web (selon netcraft).

Développement	12/2019	Pourcentage	01/2020	Pourcentage	Changement
nginx	479 million	37,77%	488 million	37,70%	-0,07
Apache	309 million	24,36%	310 million	23,98%	-0,38
Microsoft	185 million	14,59%	182 million	14,03%	-0.56
Google	37 million	2,94%	39 million	3,02%	0.08

Serveur web apache

Comme on l'a vu, apache³ est parmi les serveurs web les plus populaires. Il est robuste et extensible. Il est distribué sous une licence "Open source" (Licence Apache).

Il est disponible sur plusieurs plateformes (Linux,windows, ...)

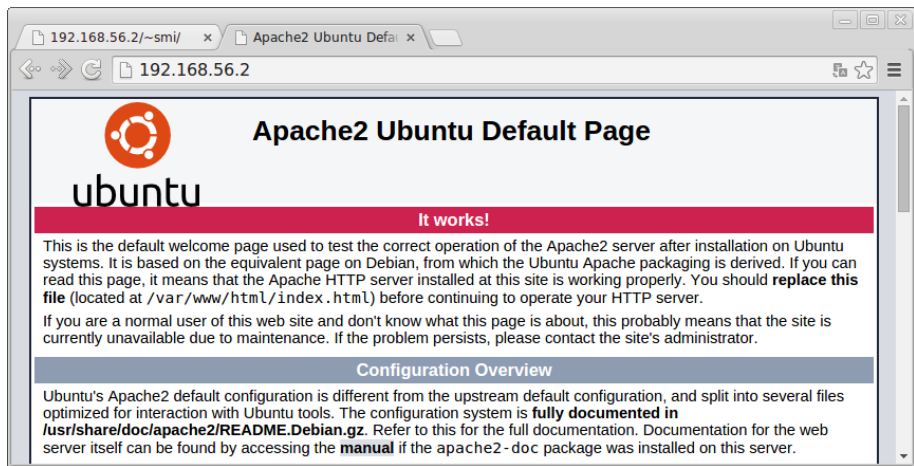
3. Site officiel : <http://httpd.apache.org/>

Une installation minimale peut être faite en ligne de commande de la façon suivante :

```
#sudo apt-get install apache2
```


Vérification de l'installation

Pour vérifier l'installation, il suffit d'utiliser un navigateur web.



Fichiers et répertoires de configuration

Les fichiers et répertoires de configuration d'apache se trouvent dans le répertoire **/etc/apache2** :

apache2.conf : fichier de configuration principale.

envvars : contient les variables d'environnement propres à apache.

conf-available/ : contient des fichiers de configuration additionnels disponibles.

conf-enabled/ : contient des fichiers de configuration activés. Ils sont utilisés dans **apache2.conf** par la ligne :

```
IncludeOptional conf-enabled/*.conf
```

ports.conf : directives de configuration pour les ports et les adresses IP d'écoutes.

Fichiers et répertoires de configuration

mods-available/ : contient une série de fichiers **.load** et **.conf**. Un fichier **.load** contient les paramètres de configuration nécessaires pour charger un module en question. Le fichier **.conf** correspondant, les paramètres de configuration nécessaires pour utiliser le module en question.

mods-enabled/ : pour utiliser un module (activer), il faut mettre un lien symbolique vers le fichier **.load** (et **.conf**, s'il existe) du module associé dans le dossier **mods-available**.

sites-available/ : même chose que **mods-available/**, mais cette fois pour les sites virtuels. Ce n'est pas obligé d'avoir le même nom pour le site et le fichier.

sites-enabled/ : même chose que **mods-enabled/**.

magic : instructions pour déterminer le type **MIME** d'un fichier (**M**ultipurpose **I**nternet **M**ail **E**xtensions - Extensions Multi-usages de la Messagerie par Internet). Par exemple text/html et image/gif.

Remarque :

Par défaut, un seul serveur est disponible (le serveur par défaut). Il est disponible dans **apache2.conf** par la ligne :

```
IncludeOptional sites-enabled/*.conf
```

Activer/désactiver un module

Les commandes **a2enmod** et **a2dismod** sont disponibles pour activer ou désactiver un module.

Exemple : pages web personnelles

Pour permettre aux utilisateurs d'avoir leurs propres pages web disponibles via un lien de type :

`http://NomSite/~utilisateur`

`http://localhost/~smi`

On tape la commande

```
#a2enmod userdir
```

Il faut ensuite redémarrer apache en tapant la commande :

```
#service apache2 restart
```

Exemple d'une page personnelle :

Dans le répertoire personnelle de l'utilisateur **smi**, il faut créer le répertoire **public_html** avec les droits **-rwxr-xr-x** et mettre dedans le fichier **index.html**, avec les droits **-rwxr--r--**.



Activer/désactiver un site

Les commandes **a2ensite** et **a2dissite** sont disponibles pour activer ou désactiver un site. On verra leurs utilisation dans les sections suivantes.

Configuration de base

Avant de commencer la configuration, il faut faire une sauvegarde des fichiers que vous voulez modifier. Par exemple :

```
#cp apache2.conf apache2.conf.save
```

- Port à écouter (ports.conf) :

```
Listen 80
```

- Emplacement par défaut des pages html : **/var/www/html**
mettre les fichiers concernant le site web dans ce répertoire.

- Pages par défaut (mods-enabled/dir.conf) :

```
DirectoryIndex index.html index.cgi index.pl  
index.php index.xhtml index.htm
```

Sites virtuelles

Apache permet de gérer plusieurs sites web. Chaque site est appelé serveur virtuel et possède sa propre configuration.

Il y a deux types de serveurs virtuels :

- 1 Serveurs par nom ; les mêmes sites utilisent la même adresse IP.
Par exemple :

IP	Nom
192.168.56.2	smi.ump.ma
192.168.56.2	sma.ump.ma

- 2 Serveurs par adresse IP ; chaque site utilise sa propre adresse IP.
Par exemple :

IP	Nom
192.168.56.2	smi.ump.ma
192.168.56.10	sma.ump.ma

Configuration de deux sites virtuels par nom

Dans cet exemple, nous allons configurer deux sites virtuels, le premier **smi.ump.ma** et le deuxième **sma.ump.ma**. Les deux sites utilisent la même adresse IP **192.168.56.2**.

Il faut déclarer les deux noms dans le fichier **/etc/hosts** :

192.168.56.2	smi.ump.ma	www.smi.ump.ma
192.168.56.2	sma.ump.ma	www.sma.ump.ma

On pourra utiliser un serveur DNS pour déclarer les noms (voir chapitre concernant le DNS).

Configuration de deux sites virtuels par nom

Il faut créer les répertoires **smi** et **sma** associés dans **/var/www/html** :

```
# mkdir /var/www/html/smi  
# mkdir /var/www/html/sma
```

Dans **/etc/apache2/sites-available/**, il faut créer deux fichiers :
smi.conf et **sma.conf**

Contenu du fichier smi.conf

```
<VirtualHost *:80>  
    DocumentRoot /var/www/html/smi  
    ServerName smi.ump.ma  
    ServerAlias www.smi.ump.ma  
</VirtualHost>
```

Avec :

DocumentRoot : emplacement par défaut des pages html ;

ServerName : nom du serveur virtuel ;

ServerAlias : autre nom (alias) du serveur virtuel.

Contenu du fichier sma.conf

```
<VirtualHost *:80>  
    DocumentRoot /var/www/html/sma  
    ServerName sma.ump.ma  
    ServerAlias www.sma.ump.ma  
</VirtualHost>
```

Activation des deux sites

Il faut activer les deux sites en tapant les commandes :

```
# a2ensite smi  
# a2ensite sma
```

Après l'activation, il faut recharger le serveur apache en tapant la commande

```
# service apache2 reload
```

Les deux sites seront accessibles via les liens :

<http://smi.ump.ma> ou <http://www.smi.ump.ma>

<http://sma.ump.ma> ou <http://www.sma.ump.ma>

Dans cet exemple, nous allons configurer un nouveau site virtuel **smp.ump.ma**, qui utilise une adresse IP différente.

Dans cet exemple, la machine doit être munie, soit de plusieurs interfaces réseaux soit de plusieurs adresses IP associées à la même interface réseau (on parle d'IP aliasing).

IP aliasing

Pour affecter une seconde adresse IP à une interface réseau, il faut exécuter la commande :

```
# ifconfig eth0:0 192.168.56.10 up
```

Remplacez eth0 par une autre interface (par exemple eth1).

L'interface dispose, maintenant, de deux adresses distinctes :

- Adresse IP : 192.168.56.2
- Alias IP : 192.168.56.10

A vérifier avec la commande :

```
# ifconfig
```

Pour rendre la configuration permanente, il faut ajouter les lignes suivantes au fichier **/etc/network/interfaces** :

```
auto eth0:0
iface eth0:0 inet static
address 192.168.56.10
netmask 255.255.255.0
```

Remarque : on peut ajouter autant d'interfaces qu'on veut (eth0:1, eth0:2 ...).

Configuration du site virtuel

Il faut ajouter au fichier **/etc/hosts**, la ligne suivante :

```
192.168.56.10 smp.ump.ma www.smp.  
ump.ma
```

Il faut créer le répertoire **smp** associé dans **/var/www/html** :

```
# mkdir /var/www/html/smp
```

Dans **/etc/apache2/sites-available/**, il faut créer le fichier : **smp.conf**

Contenu du fichier smp.conf

```
<VirtualHost 192.168.56.10:80>  
    DocumentRoot /var/www/html/smp  
    ServerName smp.ump.ma  
    ServerAlias www.smp.ump.ma  
</VirtualHost>
```

Activation du nouveau site

Il faut activer le site en tapant la commande :

```
# a2ensite smp
```

Après l'activation, il faut recharger le serveur apache en tapant la commande

```
# service apache2 reload
```

Le nouveau site sera accessible via les liens :

<http://smp.ump.ma> ou <http://www.smp.ump.ma>

Sécuriser apache

Apache est très modulaire. Dans le chapitre suivant, on verra un module concernant **php**. Dans cette section, on va utiliser un module important dans l'aspect sécurité. Le module **mod_ssl** ajoute la possibilité de crypter les communications entre le client et le serveur.

Le mode **mod_ssl** se trouve dans le package **apache2-common**. Pour l'activer, il faut taper la commande :

```
sudo a2enmod ssl
```

suivie de la commande

```
sudo service apache2 restart
```

Après l'activation, il faut utiliser le préfixe **https://** devant l'adresse du serveur dans la barre du navigateur (par exemple :

<https://192.168.56.2/>).

On a vu un exemple d'utilisation avec **wireshark** (<https://www.wireshark.org>).

Chapitre 3

Php et MySQL

PHP

PHP est un Langage de script interprété (non compilé) spécialement conçu pour le développement d'applications web. Il peut être intégré facilement au HTML⁴.

- Pour installer la version 5 de PHP, il faut exécuter la commande :
`# sudo apt install php5 libapache2-mod-php5`
- Pour installer la version 7.2 de PHP, il faut exécuter la commande :
`#sudo apt install php7.2 libapache2-mod-php7.2`

Il faut ensuite redémarrer apache :

```
# sudo service apache2 restart
```

4. voir php.net

Vérification de l'installation

Dans le répertoire `/var/www/html`, créez le script **info.php** :

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

Dans votre navigateur, tapez l'adresse

<http://localhost/info.php> ; remplacez **localhost** par l'adresse du serveur, par exemple 192.168.56.2, il fournira un ensemble d'informations et de paramètres de configuration (<http://192.168.56.2/info.php>).

Utilisation

L'utilisation de **php** sort du cadre de ce cours, il concerne le cours « Technologie du web ». La documentation complète est disponible sur le site php.net. Il existe aussi plusieurs livres concernant **php**.

MySQL

MySQL est un système de gestion de bases de données relationnel (SGBDR) libre, open-source et gratuit. Il est performant et très populaire. Il est multi-utilisateur.

Installation

Pour l'installer, il faut taper la commande :

```
# sudo apt-get install mysql-server
```

Durant l'installation, vous devez saisir le mot de passe de l'administrateur (**root**) de MySQL. Il a le même nom que l'administrateur Linux (à ne pas confondre les noms !).

Remarque : sur quelques systèmes Linux, le compte **root** de MySQL est désactivé.

Documentation

Pour plus d'informations sur MySQL, veuillez consulter les sites : <http://www.mysql.com/> et <http://dev.mysql.com/doc/>. Il existe aussi plusieurs livres concernant l'utilisation de MySQL. Vous pouvez appliquer ce que vous avez vu dans le cours de « Bases de Données ».

Oubli du mot de passe

Si vous avez oublier le mot de passe de **root** de MySQL, vous pouvez établir un nouveau mot passe en tapant la commande :

```
sudo dpkg-reconfigure mysql-server-5.5
```

Le démon MySQL sera arrêté et vous devez saisir un nouveau mot de passe. Après la saisie, le démon MySQL sera de nouveau démarré.

Utilisation

Dans une console, tapez la commande :

```
mysql -u root -p
```

et tapez votre mot de passe.

Vous arriverez alors sur un prompt de type :

```
mysql>
```

Vous pouvez alors taper des requêtes MySQL. N'oubliez pas le point-virgule à la fin de la requête.

Utilisation

Par exemple, pour créer une base de données qui s'appelle **smi**, tapez la requête :

```
mysql> create database smi;
```

Pour voir les bases de données, tapez la requête :

```
mysql> show databases;
```

Systèmes où **root** est désactivé

Création d'un compte qui a tous les droits

- 1 Connexion à MySQL :

sudo mysql

- 2 Création du compte **admin** :

```
mysql> CREATE USER 'admin'@'localhost'  
-> IDENTIFIED BY 'smi,2019';
```

- 3 Affectation de tous les droits au compte créé :

```
mysql> GRANT ALL ON *.* TO 'admin'@'localhost'  
-> IDENTIFIED BY 'smi,2019';
```

Utilisation de MySQL avec Php

Pour utiliser MySQL avec Php, il faut installer le paquet **php5-mysql**. Pour installer **php5-mysql**, il faut taper la commande :

```
sudo apt-get install php5-mysql
```

Après l'installation de **php5-mysql**, vous pouvez utiliser des applications web qui utilisent **php** comme langage de programmation et peuvent accéder à **MySQL**. Dans la section suivante, nous allons voir **phpmyadmin** qui est une application web écrite en **php** et se connecte à **MySQL**.

PhpMyAdmin

phpMyAdmin est une application écrite en PHP très utile pour l'administration de MySQL. Elle est accessible via un navigateur. Pour l'installer, tapez la commande :

```
sudo apt-get install phpmyadmin
```

Configuration de PhpMyAdmin

Choix du serveur web

Outil de configuration des paquets

Configuration de phpmyadmin

Veuillez choisir le serveur web à reconfigurer automatiquement pour exécuter phpMyAdmin.

Serveur web à reconfigurer automatiquement :

☒ apache2

☐ lighttpd

<Ok>

<Annuler>

Configuration de PhpMyAdmin

Base de données de phpmyadmin

Outil de configuration des paquets

Configuration de phpmyadmin

Le paquet phpmyadmin a besoin d'une base de données installée et configurée avant de pouvoir être utilisé. Ceci peut si nécessaire être géré par dbconfig-common.

Si vous êtes un administrateur de bases de données expérimenté et savez que vous voulez procéder à cette configuration vous-même, ou si votre base de données est déjà installée et configurée, vous pouvez refuser cette option. Des précisions sur la procédure se trouvent dans `/usr/share/doc/phpmyadmin`.

Autrement, vous devriez choisir cette option.

Faut-il configurer la base de données de phpmyadmin avec dbconfig-common ?

<Oui>

<Non>

Configuration de PhpMyAdmin

Affectation d'un mot de passe à l'utilisateur phpmyadmin

Outil de configuration des paquets

Configuration de phpmyadmin

Veuillez indiquer un mot de passe de connexion pour phpmyadmin sur le serveur de bases de données. Si vous laissez ce champ vide, un mot de passe aléatoire sera généré.

Mot de passe de connexion MySQL pour phpmyadmin :

<Ok>

<Annuler>

Configuration de PhpMyAdmin

Confirmation du mot de passe

Outil de configuration des paquets

Configuration de phpmyadmin

Confirmation du mot de passe :

<Ok>

<Annuler>

Utilisation de PhpMyAdmin

Pour son utilisation, saisissez l'adresse

<http://localhost/phpmyadmin>.

Vous pouvez remplacer **localhost** par l'adresse de votre serveur :

<https://192.168.56.2/phpmyadmin>.

Chapitre 4

Serveur DHCP

Dynamic Host Configuration Protocol

Introduction

Une adresse réseau peut être configurée soit de manière statique ou dynamique :

Statique : l'utilisateur configure lui même l'adresse IP de la machine.

Dynamique : la machine obtient l'adresse grâce à un serveur DHCP.

Le serveur DHCP (**D**ynamic **H**ost **C**onfiguration **P**rotocol) est un protocole de configuration dynamique de machines, il permet l'affectation, de façon automatique, des paramètres réseaux à une machine.

En général, le serveur DHCP affecte à un client :

- l'adresse IP ;
- la passerelle par défaut ;
- les adresses IP des serveurs DNS.

Le serveur DHCP peut affecter aussi :

- le nom de la machine ;
- le nom du domaine ;
- le serveur d'impression ;
- le serveur de temps (qui donne le temps à la machine).

Le serveur DHCP attribue les paramètres suivant deux méthodes :

- automatique** : pour une période de temps, il affecte une adresse IP à partir d'un intervalle au client. Si le client n'est pas connecté pour une certaine période de temps, l'adresse peut être affectée à une autre machine ;
- fixe** : en utilisant l'adresse MAC d'une machine, le serveur DHCP affecte toujours la même adresse IP à la machine. Ceci pour assurer qu'une machine avec une adresse MAC, reçoive toujours la même adresse IP.

Remarque : il ne faut pas confondre statique et fixe. Statique veut dire que c'est l'utilisateur qui configure l'adresse IP de sa machine.

Avantages

- ❶ Toute changement dans les paramètres réseaux se fera au niveau du serveur DHCP.
- ❷ Facilité d'ajout de nouvelles machines dans le réseau.

On peut avoir des serveurs DHCP sous Linux et sous Windows-server.

Dans ce qui suit, nous allons utiliser le serveur **isc-dhcp-server**.

Installation du serveur isc-dhcp-server

```
#sudo apt install isc-dhcp-server
```

Vous devez changer la configuration par défaut, en modifiant les deux fichiers :

- 1 **/etc/dhcp/dhcpd.conf**
- 2 **/etc/default/isc-dhcp-server.**

Configuration

Deux cas seront traités :

- 1 adresses dynamiques alloués aux différents machines.
- 2 adresse fixe alloué à la machine **web-smi** ;

On suppose que le serveur dispose de trois interfaces réseaux :

- **eth0** : interface pour se connecter à Internet ; adresse obtenue par dhcp à partir d'un autre serveur DHCP
- **eth1** dont l'adresse IP est : 192.168.1.1
- **eth2** dont l'adresse IP est : 192.168.10.1

Interface(s) d'écoute(s)

- Si vous voulez que le serveur écoute sur certaines interfaces vous devez les spécifier dans le fichier **/etc/default/isc-dhcp-server**.
- Dans notre cas, le fichier doit contenir la ligne :

```
INTERFACES="eth1 eth2"
```

L'écoute se fera sur les interfaces eth1 et eth2.

Configuration du serveur

- La configuration se fait dans le fichier **/etc/dhcp/dhcpd.conf**.
- Les options sont définies de deux façons :
 - 1 **globale**
 - 2 **réseau**.

Dans ce qui suit, nous allons voir un exemple de configuration pour le cas traité.

Options globales

Dans l'exemple suivant, on donnera les options communes aux différents réseaux.

#Nom du domaine DNS

option domain-name "ump.ma";

#Nom(s) de(s) serveur(s) DNS

option domain-name-servers 192.168.100.10,
192.168.10.11;

#Temps de renouvellement des adresses en s (1h)

default-lease-time 3600;

maximum (2h)

max-lease-time 7200;

Options globales (suite)

```
# Mode autoritaire
# Est-ce-que ce serveur est le serveur principal?
authoritative;

# Masque de sous-reseau
option subnet-mask 255.255.255.0;
```

Configuration du réseau 192.168.1.0

```
# declaration du sous reseau 192.168.1.*
subnet 192.168.1.0 netmask 255.255.255.0 {
    # Adresse de diffusion
    option broadcast-address 192.168.1.255;

    # routeur par defaut
    option routers 192.168.1.1;

    # intervalle des adresses
    range 192.168.1.2 192.168.1.100;
}
```

Configuration du réseau 192.168.10.0

```
# declaration du sous reseau 192.168.10.*
subnet 192.168.10.0 netmask 255.255.255.0 {
    # specifier un domaine different de celui par
    #      default :
    option domain-name "fso.ump.ma";

    # Adresse de diffusion
    option broadcast-address 192.168.10.255;

    # routeur par default
    option routers 192.168.10.1;

    # intervalle des adresses
    range 192.168.10.20 192.168.10.200;
}
```

Configuration de la machine « web-smi »

```
host web-smi {  
    # adresse mac de la carte reseau  
    # A remplacer par celle de la machine  
    hardware ethernet 08:00:27:A6:C2:50;  
  
    # adresse attribue  
    fixed-address 192.168.1.200;  
}
```

Remarque :

Si le réseau 192.168.1.0, ne figure pas dans le fichier de configuration, il faut le signaler de la façon suivante :

```
# Ajouter pour comprendre la topologie du reseau  
# Ne fourni aucun service  
subnet 192.168.1.0 netmask 255.255.255.0 {  
}
```


























Redémarrage du serveur DHCP

Après avoir changé les fichiers de configuration, il faut redémarrer le démon **dhcpcd** :

```
sudo service isc-dhcp-server restart
```

Test avec NetKit

Structure du Lab

- ▼  pc1
 - ▼  etc
 - ▼  network
 -  interfaces
- ▼  pc2
 - ▼  etc
 - ▼  network
 -  interfaces
- ▼  servDHCP
 - ▼  etc
 - ▼  default
 -  dhcp3-server
 - ▼  dhcp3
 -  dhcpd.conf
- ▼  serv-web
 - ▼  etc
 - ▼  network
 -  interfaces
-  lab.conf
-  lab.dep
-  pc1.startup
-  pc2.startup
-  servDHCP.startup
-  serv-web.startup

Structure du lab

Fichier **lab.conf**

servDHCP[1]=A

serv-web[0]=A

pc1[0]=A

servDHCP[2]=B

pc2[0]=B

servDHCP[0]=C

Fichier **lab.deb**

pc1 pc2 serv-web : servDHCP

Fichiers **pc1.startup** et **pc2.startup**

/etc/init.d/networking start

Structure du lab (suite)

Fichier **serv-web.startup**

Fixer l'adresse MAC

```
ifconfig eth0 hw ether 08:00:27:A6:C2:50
```

```
/etc/init.d/networking start
```

Fichier **servDHCP.startup**

```
ifconfig eth0 192.168.4.2 up
```

```
ifconfig eth1 192.168.1.1 up
```

```
ifconfig eth2 192.168.10.1 up
```

Démarrage du serveur DHCP

```
/etc/init.d/dhcp3-server start
```

Fichier **/etc/network/interfaces**

Machines **pc1**, **pc2** et **serv-web**

```
# The loopback network interface  
auto lo  
iface lo inet loopback
```

```
# Pour avoir l'adresse de facon automatique  
auto eth0  
iface eth0 inet dhcp
```

Serveur DHCP

Fichiers **/etc/default/dhcp3-server** et **/etc/dhcp3/dhcpd.conf**
(voir contenu plus haut).

Vérification des paramètres réseaux reçus

Au niveau des machine **pc1**, **pc2** et **serv-web** :

- Pour la vérification de l'adresse reçue :
`ifconfig eth0`
- Pour la vérification des serveur des noms et du DNS
`more /etc/resolv.conf`
- Pour la vérification de la table de routage :
`route -n`

Fonctionnement de DHCP

La figure suivante, présente une visualisation par wireshark d'une capture de paquets lors de l'affectation d'une adresse IP à un client DHCP.

No.	Time	Source	Destination	Protocol	Length	Info
3	1.059241	0.0.0.0	255.255.255.255	BOOTP	342	Boot Request from 6e:5f:98:37:0c:07 (6e:5f:98:37:0c:07)
4	1.059960	192.168.1.1	192.168.1.2	ICMP	62	Echo (ping) request id=0xc8c1, seq=0/0, ttl=64
5	1.405486	192.168.1.1	192.168.1.2	BOOTP	342	Boot Reply[Packet size limited during capture]
6	1.406153	0.0.0.0	255.255.255.255	BOOTP	342	Boot Request from 6e:5f:98:37:0c:07 (6e:5f:98:37:0c:07)
7	1.406686	192.168.1.1	192.168.1.2	BOOTP	342	Boot Reply[Packet size limited during capture]
11	6.054863	0e:e5:85:cc:fb:38	6e:5f:98:37:0c:07	ARP	42	Who has 192.168.1.2? Tell 192.168.1.1
12	6.055121	6e:5f:98:37:0c:07	0e:e5:85:cc:fb:38	ARP	42	192.168.1.2 is at 6e:5f:98:37:0c:07

Source	Destination	Protocol	Info
0.0.0.0	255.255.255.255	DHCP	DHCP Discover (utilise UDP) Le client utilise l'adresse 0.0.0.0 (hôte inconnu) et envoie la demande à toutes les machines du réseau.
192.168.1.1	192.168.1.2	ICMP	Echo (ping) request Avant d'affecter l'adresse 192.168.1.2 au client, le serveur DHCP s'assure que cette adresse n'est pas utilisée par une autre machine.

Source	Destination	Protocol	Info
192.168.1.1	192.168.1.2	DHCP	DHCP Offer Le serveur DHCP offre l'adresse 192.168.1.2 au client.
0.0.0.0	255.255.255.255	DHCP	DHCP Request Le client demande l'adresse.
192.168.1.1	192.168.1.2	DHCP	DHCP ACK (acknowledgment - acquittement) Le serveur envoie un accusé d'acceptation.

Source	Destination	Protocol	Info
0e:e5:85:cc:fb:38	6e:5f:98:37:0c:07	ARP	Who has 192.168.1.2? Tell 192.168.1.1 Demande ARP de la part du serveur
6e:5f:98:37:0c:07	0e:e5:85:cc:fb:38	ARP	192.168.1.2 is at 6e:5f:98:37:0c:07 Réponse ARP

Chapitre 5

ftp et ssh

Introduction

FTP (File Transfer Protocol - Protocole pour le Transfert de Fichiers) est un protocole TCP qui permet le téléchargement de fichiers à partir d'un serveur. Ce protocole n'est pas sécurisé du fait que l'envoi des données entre le client et le serveur n'est pas crypté. Pour l'opération inverse (chargement) et pour plus de sécurité, on peut utiliser **ssh** (Secure shell).

Il existe plusieurs serveurs ftp, **fttpd**, **proftpd**, **twoftpd**, ...

ftp permet l'accès de deux façons :

- **anonyme** : l'accès se fera au serveur via le nom d'utilisateur par défaut « anonymous » ou « ftp » ;
- **authentifié** : l'utilisateur doit disposé sur le système distant d'un compte. Cette façon est déconseillé du fait que la connexion au serveur n'est pas sécurisée.

Serveur vsftpd

Dans cette section, nous allons utiliser **vsftpd** qui est facile à installer et à maintenir. Pour l'installer, tapez la commande :

```
sudo apt-get install vsftpd
```


Configuration de la connexion anonyme

Par défaut, **vsftpd** n'est pas configuré pour autoriser la connexion anonyme. Pour l'autoriser, modifiez le fichier **/etc/vsftpd.conf** en changeant la ligne :

anonymous_enable=YES

Par défaut, la valeur était **NO**. Après cette modification, il faut redémarrer le serveur ftp en tapant la commande :

```
sudo restart vsftpd
```

Utilisateur ftp

Durant l'installation, l'utilisateur **ftp** avec le répertoire personnel **/srv/ftp** seront créés. Les fichiers qui seront visibles par connexion ftp anonyme doivent être mises dans ce répertoire.

La commande :

```
tail -n1 /etc/passwd
```

Fournira le résultat :

```
ftp:x:111:119:ftp daemon,,:/srv/ftp:/bin/false
```

Configuration de la connexion authentifiée

Par défaut, **vsftpd** est configuré pour autoriser les utilisateurs authentifiés à télécharger des fichiers. Il n'autorise ni le chargement de fichiers ni la création de répertoires. Pour autoriser le chargement de fichiers et la création de répertoires, il faut éditer le fichier **/etc/vsftpd.conf** et enlever le commentaire à la ligne :

#write_enable=YES

pour devenir :

write_enable=YES

Après, il faut redémarrer le serveur **vsftpd** en tapant la commande :

sudo restart vsftpd

Connexion à partir d'un client

On peut se connecter à un serveur ftp, soit à partir d'un :

- terminal ;
- navigateur.

Connexion à partir d'un terminal

Pour se connecter à partir d'un terminal, il faut taper la commande :

ftp 192.168.56.101

Changez **192.168.56.101** par l'adresse ou le nom de votre serveur.

En validant la commande, vous obtiendrez :

```
Connected to 192.168.56.101.  
220 (vsFTPd 3.0.2)  
Name (192.168.56.101:lakhouaja): ftp  
331 Please specify the password.  
Password:  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp>
```

Connexion

Pour fermer la connexion, tapez **quit** ou **exit** dans l'invite de commandes de ftp. Pour plus de commandes, tapez dans l'invite « help » ou « ? ». Pour l'aide sur une commande, tapez :

? commande

(par exemple : `ftp> ? get`). Vous pouvez aussi utiliser le manuel en ligne de ftp :

`man ftp.`

Connexion

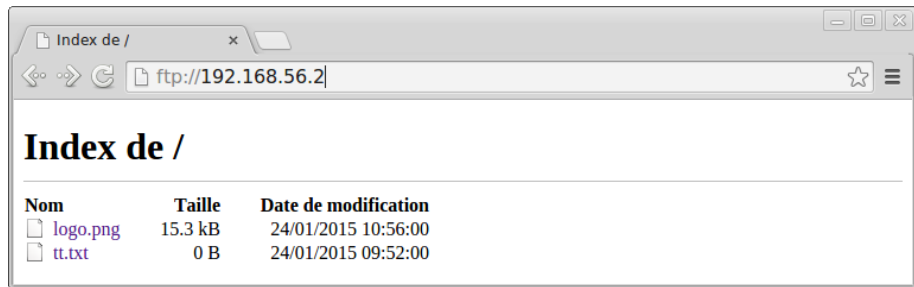
Pour une connexion :

- anonyme, tapez **ftp** ou **anonymous** après : de Name ; pour le mot de passe, il faut juste valider par la touche « Entrée » ;
- authentifié, tapez votre nom de connexion après : de Name et saisissez votre mot de passe.

Connexion à partir d'un navigateur

Vous pouvez vous connecter au serveur **ftp** en utilisant un navigateur.
Pour une connexion :

anonyme : tapez **ftp ://adresse** (par exemple :
`ftp://192.168.56.101`);



Connexion authentifié à partir d'un navigateur

Pour une connexion :

authentifié, tapez `ftp://login@adresse` (par exemple :
`ftp://smi@192.168.56.101`) après validation, saisissez votre mot
de passe.

ssh

Comme nous l'avons signalé au début de ce chapitre, pour le transfert de fichiers en utilisant une connexion sécurisée, il faut utiliser **ssh** (Secure Shell). Sous Linux, le serveur **ssh** disponible de façon libre et gratuite s'appelle **OpenSSH**.

Installation

Pour installer le client, tapez la commande

```
sudo apt-get install openssh-client
```

Pour installer le serveur, tapez la commande :

```
sudo apt-get install openssh-server
```

Connexion à partir d'un client Linux

Pour se connecter à partir d'un client, tapez : `ssh login@adresse`.
Par exemple :

```
ssh smi@192.168.56.101
```

Utilisation de ssh comme ftp sécurisé

Pour utiliser le serveur **ssh** comme serveur **ftp** sécurisé, tapez la commande : `sftp login@adresse`. Par exemple :

```
sftp smi@192.168.56.101
```

Après saisi du mot de passe, vous obtiendrez l'invite de commandes :

```
sftp>
```

Pour fermer la connexion, tapez **quit**, **bye** ou **exit** dans l'invite de commandes. Pour plus de commandes, tapez dans l'invite « help » ou « ? ». Vous pouvez aussi utiliser le manuel en ligne de sftp :

```
man sftp.
```

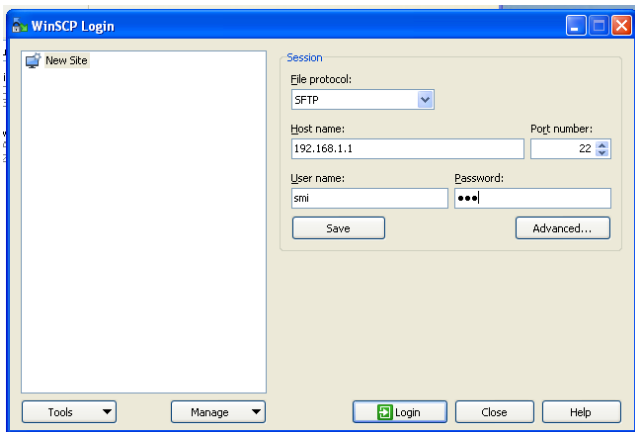
Remarque

Si votre navigateur supporte le protocole **sftp**, vous pouvez taper par exemple :

```
sftp://smi@192.168.56.101/
```

Connexion à partir d'un client Windows

Sous Windows il existe l'application **winscp** disponible en téléchargement à partir du site officiel <http://winscp.net>. Son interface graphique se présente comme suit :



Connexion à partir d'un client Windows

smi - smi@192.168.1.1 - WinSCP

Local Marquer Fichiers Commandes Session Options Distant Aide

C: Disque local

C:\Documents and Settings\Lakhouaja

Nom	Ext	Taille	Type	Date de m...
..			Répertoire parent	29/01/201...
Application Data			Dossier de fichiers	15/01/201...
Bureau			Dossier de fichiers	29/01/201...
Cookies			Dossier de fichiers	23/04/201...
Downloads			Dossier de fichiers	20/03/201...
Favoris			Dossier de fichiers	19/03/201...
IECompatCache			Dossier de fichiers	20/03/201...
IETldCache			Dossier de fichiers	19/03/201...
Local Settings			Dossier de fichiers	15/10/200...
Menu Démarrer			Dossier de fichiers	15/10/200...
Mes documents			Dossier de fichiers	14/10/201...
Modèles			Dossier de fichiers	15/10/200...
PrivacIE			Dossier de fichiers	20/03/201...
Recent			Dossier de fichiers	23/11/201...
SendTo			Dossier de fichiers	29/01/201...
UserData			Dossier de fichiers	15/10/200...
Voisinage d'impression			Dossier de fichiers	15/10/200...
Voisinage réseau			Dossier de fichiers	13/12/201...
WINDOWS			Dossier de fichiers	08/03/201...

0 B de 1 537 KB dans 0 de 22

/home/smi

Nom	Ext	Taille	Date de modif...	Droits	Prop
..			17/01/2015 23:...	rwxt-r-x	root
.cache			26/04/2014 05:...	rw-----	smi
.gfdclient			26/04/2014 17:...	rwxt-r-x	smi
glassfish-4.0			26/04/2014 17:...	rwxt-r-x	smi
public_html			17/01/2015 15:...	rwxt-r-x	smi
test			23/01/2015 15:...	rwxt-r-x	smi
.bash_history		110	24/01/2015 12:...	rw-----	smi
.bash_logout		220	26/04/2014 05:...	rw-r--r--	smi
.bashrc		3 637	26/04/2014 05:...	rw-r--r--	smi
.profile		675	26/04/2014 05:...	rw-r--r--	smi
.viminfo		1 230	17/01/2015 15:...	rw-----	smi
ArabicDictionary.sql		50 715 420	26/04/2014 17:...	rw-r--r--	smi
hello-jaxws.war		3 915	26/04/2014 06:...	rwxt-r-x	smi
logo.png		15 667	24/01/2015 10:...	rw-r--r--	smi
phpmyadmin_4%3a4.0...		6 969 346	26/04/2014 17:...	rw-r--r--	smi
webclient.war		8 136	26/04/2014 06:...	rwxt-r-x	smi

0 B de 56 365 KB dans 0 de 15

F2 Renommer F4 Editor F5 Copier F6 Déplacer F7 Créer un répertoire F8 Effacer F9 Propriétés F10 Quitter

SFTP-3 0:01:49

Copie vers le serveur

Pour copier un fichier ou un répertoire dans le serveur ssh, vous pouvez utiliser la commande **scp** (analogue à la commande **cp** de Linux). Son utilisation est comme suit :

```
scp fichier1 fichier2 ... smi@192.168.56.101:
```

Pour copier un répertoire, il faut simplement ajouter l'option **-r** :

```
scp -r Rep smi@192.168.56.101:
```

Remarque : il ne faut pas oublier **:**, sinon la copie se fera en local (utilisation de **cp**).

Chapitre 6

Partage de dossiers et d'imprimantes

Introduction

Le partage de dossiers et d'imprimantes permet de :

- réduire le coût d'investissement ;
- mutualiser les ressources.

Pour partager des dossiers entre des machines Unix/Linux on utilise le protocole NFS (Network File System).

Pour partager des dossiers et des imprimantes entre des machines Unix/Linux et des machines Windows on utilise le service samba.

Le protocole NFS

Le protocole NFS

C'est un protocole qui fonctionne suivant le modèle client/serveur :

- un serveur met des dossiers à la disposition des machines sur le réseau suivant des droits d'accès.
- d'autres machines peuvent monter ces dossiers. Qui seront vus comme des dossiers locaux.

Côté serveur

Il faut installer le serveur NFS, en tapant la commande :

```
# sudo apt-get install nfs-kernel-server
```

Configuration

Le fichier de configuration est **/etc/exports**. On indique dans ce fichier la liste des répertoires à exporter (partagés) et les noms des machines autorisées à les utiliser.

Exemple :

```
/home machine1(rw, sync, no_subtree_check)
/home *(ro, sync, no_subtree_check)
/TP machine1(rw, sync, no_root_squash, no_subtree_check)
/test *(ro, sync, no_subtree_check)
```

Explication :

- **machine1** peut monter **/home** en lecture/écriture (**rw**) ;
- toutes les autres machines du réseau peuvent monter **/home** en lecture seulement (**ro**) ;
- **machine1** peut monter **/TP** en lecture/écriture (**rw**) ;
- toutes les machines du réseau peuvent monter **/test** en lecture seule (**ro**).

On peut utiliser des noms ou adresses IP pour les machines.

Par exemple :

```
/home 192.10.1.1(rw, sync, no_subtree_check)
/TP smi.ump.ma(ro, sync, no_subtree_check)
```


Configuration

Une fois le fichier **/etc/exports** bien configurer il faut redémarrer (ou relancer) **nfs** :

```
# sudo service nfs-kernel-server restart
```

ou bien

```
# sudo service nfs-kernel-server reload
```

Affichage des répertoires partagés

La commande **exportfs** permet d'afficher les répertoires partagés.

Exemple :

```
# exportfs
```

affichera

```
/etc          192.168.56.1  
/var          192.168.56.1  
/test         <world>
```

Options

Principales options du fichier exports :

Option	Signification
ro	read-only (accès en lecture seule au répertoire exporté)
rw	read-write : le client accède au répertoire en lecture/écriture
root_squash	convertit les UID/GID root, en utilisateur anonyme. L'administrateur de la machine cliente ne peut pas modifier le contenu des répertoires et des fichiers.

Options (suite)

Option	Signification
no_root_squash	désactive la conversion des UID/GID root.
all_squash	convertit tous les UID/GID en utilisateurs anonymes. Utile pour exporter avec NFS des répertoires publics.
sync	ne répondre aux requêtes qu'après l'exécution de tous les changements sur le support réel.
no_subtree_check	annule la vérification des sous-répertoires
fsid	définit le système de fichiers racine (fs : filesystem)

Remarque

Les options doivent être séparées par des virgules, SANS ESPACE.
Pour plus d'options, veuillez consulter le manuel du fichier **exports**
(**man exports**).

Côté client

Pour pouvoir monter des répertoires, il faut installer le package **nfs-common** :

```
#sudo apt-get install nfs-common
```

En tant qu'administrateur du système, pour monter un répertoire distant, il faut utiliser la commande **mount** avec l'option **-t nfs**.

Exemple :

```
#mount -t nfs nom_machine:/home /test
```

ou

```
#mount -t nfs 192.168.56.2:/home /test
```

Remarque

L'option **-t nfs** n'est pas obligatoire :

```
#mount 192.168.56.2:/home /test
```

La commande **df -h** permet de d'afficher les répertoires montés (locaux et distants)

Utilisation de **fsid**

Création des répertoires :

- /partageNFS
- /partageNFS/TP
- /partageNFS/cours
- /partageNFS/documents

Ajouter dans le fichier **/etc/exports** les lignes suivantes :

```
/partageNFS *(ro,fsid=0,no_subtree_check)
/partageNFS/TP 192.168.56.0/24(rw, sync, no_subtree_check)
/partageNFS/cours 192.168.56.0/24(rw, sync, no_subtree_check)
/partageNFS/documents 192.168.56.0/24(rw, sync, no_subtree_check)
```

Pour activer les modifications, il suffit d'utiliser la commande :

sudo exportfs -a

Montage des répertoires

Au lieu d'utiliser la commande

```
#mount 192.168.56.2:/partageNFS/documents /documents
```

Il suffit de faire :

```
#mount 192.168.56.2:/documents /documents
```

puisque, le répertoire **/partageNFS** est défini comme racine.

Montage au démarrage

Pour monter un répertoire au démarrage du système, il suffit d'ajouter les renseignements nécessaire au fichier **/etc/fstab**

Par exemple :

```
nom_machine:/home /home nfs auto,rw,user 0 0
```

NFS et la Sécurité

NFS n'est pas un protocole très sécurisé :

- l'authentification des clients repose uniquement sur le nom de domaine ou l'adresse IP ;
- l'identification des utilisateurs repose sur le « user id » sur le poste client => usurpation possible ;
- le transfert des données est non crypté ;
- utilisation recommandée en intranet isolé, protégé de l'internet par un Firewall.

Le protocole SAMBA

Installation

Samba peut être utilisé pour partager des dossiers et des imprimantes entre des machines Unix/Linux et des machines Windows.

En ligne de commandes, il suffit de taper la commande :

```
#sudo apt-get install samba
```

Configuration

Le fichier principal de configuration de samba est :
/etc/samba/smb.conf.

Avant de modifier le fichier de configuration, il faut le sauvegarder par prudence (`cp smb.conf smb.conf.old`).

Ce fichier est organisé en sections. L'administrateur **root** peut éditer, modifier et ajouter des sections, pour définir de nouvelles ressources à partager.

Section

Une section commence par un mot entre crochets et se termine lorsqu'une autre section commence.

Exemple de sections :

[global]

#ensemble de directives

[homes]

#ensemble de directives

Remarque :

D'une façon générale, les permissions de partage définies dans les sections ne peuvent pas dépasser les permissions des fichiers du serveur hôte.

Pour plus d'informations concernant le fichier `smb.conf`, veuillez consulter le manuel en ligne :

```
man smb.conf
```


Vérification des changements

L'outil **testparm**, permet de tester la syntaxe du fichier de configuration et de détecter les erreurs. Il est recommandé de le lancer systématiquement lors de la modification de **smb.conf**.

Activation des changements

A chaque changement effectué dans **smb.conf**, il faut relancer les démons **smbd** et **nmdbd** :

```
#service smbd restart
```

suivie de :

```
#service nmdbd restart
```

Pour les nouveaux systèmes :

```
systemctl restart smbd
```

suivie de

```
systemctl restart nmdbd
```

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
path =		chemin du répertoire à partager
comment =		texte visible dans le voisinage réseau client
guest ok = yes no	no	permettre l'accès sans authentification
valid users =	tous	liste des utilisateurs autorisés à se connecter à la ressource
printable = true false	false	partage d'un service d'impression et non d'un répertoire.
writable = yes no	no	permet ou non l'écriture sur le répertoire, contraire de read only

Les principaux paramètres de smb.conf

paramètre	valeur par défaut	Description
browseable =	yes	visibilité du partage par tous, même les utilisateurs non autorisés
create mask =	0744	droits maxi accordés à un fichier créé dans la ressource ces droits seront en intersection (and) avec les droits Linux (umask)
directory mask =	0755	droits maxi accordés à un répertoire créé dans la ressource ces droits seront en intersection (and) avec les droits Linux (umask)

Commande umask

L'umask permet d'attribuer des permissions aux fichiers et répertoires créés par l'utilisateur. Il se présente sous la forme de 4 chiffres. La valeur par défaut de l'umask est 0022. Pour obtenir les permissions qui seront utilisées, il faut appliquer la règle suivante :

- pour les fichiers, il faut soustraire le umask de 666.
Par exemple $666 - 0022 = 644$ ce qui donne les droits
rw-r--r--
- pour les répertoires, il faut soustraire le umask de 777.
Par exemple $777 - 0022 = 755$ ce qui donne les droits
rwxr-xr-x

Commande umask (suite)

Si l'utilisateur veut que les nouveaux fichiers soient créés avec les droits **rw-----** et que les nouveaux répertoires soient créés avec les droits **rw-x-----**, il doit utiliser le masque **0077**. Pour cela, il doit taper la commande :

```
umask 0077
```

ou tout simplement :

```
umask 77
```

Remarque

umask accepte les symboles (r, w et x) comme **chmod**

umask 77 peut être utilisée comme suit :

```
umask u=rwx, g=, o=
```

La section globale

```
[global]
```

```
# donner le meme nom de groupe de travail que celui  
des stations Windows (Voisinage reseau/  
identification)
```

```
workgroup = SMI
```

```
# restreindre par sécurité les sous-réseaux autorisés à se connecter  
au serveur
```

```
# ici on se limite aux adresses réseau privé 192.168.1.0 et à  
l'interface "loopback"
```

```
hosts allow = 192.168.1.    127.
```

```
# on peut exclure des machines de l'accès au réseau
```

```
hosts allow = 192.168.1.    EXCEPT 192.168.1.125
```

```
# d'autres possibilités existent : voir le manuel man smb
```

Le répertoire personnel

[homes]

#accès au répertoire personnel de chaque utilisateur.

#la valeur du champ "comment" apparaîtra dans le voisinage réseau

#inutile pour cette section de préciser le path, c'est celui de
l'utilisateur, en fait /home/%u

browseable = no

read only = no

create mode = 0700

valid users = %S

Rendre un répertoire public en lecture seule

Pour rendre un répertoire accessible par tous le monde, il faut tout d'abord le créer ou vérifier qu'il existe.

```
# mkdir /home/partage
```

```
ls -ld /home/partage
```

doit renvoyer les droits par défaut `drwxr-xr-x`, sinon il faut les changer en tapant la commande :

```
chmod 755 /home/partage
```

ou son équivalent

```
chmod u=rwx,go=rx /home/partage
```

pour y ajouter les permissions d'accès et de lecture pour tous.

Rendre un répertoire public en lecture seule

Ensuite, il faut ajouter une nouvelle section « **[partage]** » comme suit :

```
[partage]  
path = /home/partage  
browseable = yes  
read only = yes  
guest ok = yes
```

Rendre un répertoire public en lecture et écriture

Si on veut rendre ce répertoire partagé en écriture aussi, il faut modifier les droits d'accès du répertoire et modifier la section [partage] :

```
chmod 777 /home/partage
```

```
[partage]  
path = /home/partage  
browseable = yes  
writeable = yes  
guest ok = yes  
create mode = 0755
```

Utilitaires SAMBA

- testparm** : permet la validation du fichier de configuration de Samba.
- smbclient** : client Linux/Unix similaire à FTP permettant de se connecter à des partages Samba.
- smbpasswd** : permet à un administrateur de modifier les mots de passe chiffrés utilisés par Samba.
- smbstatus** : dresse l'état des connexions aux partages d'un serveur Samba.

Ajout d'un utilisateur samba

Pour permettre à un utilisateur de se connecter à son répertoire personnel à partir d'autres machines, il faut l'ajouter en tant qu'utilisateur samba en tapant la commande suivante :

smbpasswd -a utilisateur

Par exemple :

smbpasswd -a smi

Problème de connexion avec Windwos

Sous Windows, si on se connecte avec un utilisateur 1 (par exemple **smi**) et on veut se connecter avec un autre utilisateur (par exemple **sma**), la connexion ne réussisse pas. Pour cela, il faut supprimer la connexion à l'utilisateur 1, en tapant la commande (sous un invite de commande) :

```
net use \\nom-partage\utilisateur /delete
```

Par exemple :

```
net use \\Ubuntu\smi /delete
```

Ou **Ubuntu** est le nom de partage et **smi** est le nom de l'utilisateur déjà connecté.

Connexion à partir d'un client Linux

Pour se connecter à partir d'un client Linux en utilisant la commande **smbclient**, il faut taper la commande

```
smbclient //nom-machine/repertoire
```

Par exemple :

```
smbclient //192.168.56.2/partage
```

Pour se connecter en utilisant le compte d'un utilisateur qui s'appelle **smi**, il faut taper la commande :

```
smbclient //192.168.56.2/smi -U smi
```

ensuite, on saisit le mot de passe.

Connexion à partir d'un client Linux

On peut utiliser la commande **mount** pour monter un répertoire partagé (fonctionne sous root) :

```
mount -o username=smi //192.168.56.2/smi Rep
```

Ou **Rep** est le répertoire de montage.

Chapitre 7

Domain Name Service (DNS) Service de Nom de Domaines

Introduction

Domain Name Service (DNS) est un service qui relie les adresses IP et les noms de domaines entre eux. Sous Linux, le DNS est géré par **BIND** (Berkeley Internet Name Domain) « paquet **bind9** ».

Pour les petits réseaux, il suffit d'utiliser le fichier **/etc/hosts** que vous avez vu en semestre 5

Exemple d'un fichier /etc/hosts :

127.0.0.1	localhost	
192.168.56.2	smi.ump.ma	smi
192.168.56.2	www.smi.ump.ma	
192.168.56.2	sma.ump.ma	sma
192.168.56.2	www.sma.ump.ma	

Installation

Dans un terminal, tapez la commande :

```
sudo apt-get install bind9
```

Le fichier de configuration principal de **bind** est **/etc/bind/named.conf**.

Configuration

Bind peut être configuré de plusieurs façons. Il peut être configuré pour être un serveur :

- de cache** : dans ce cas, il sert pour stocker les informations concernant les requêtes sur les noms de domaines ;
- principale** : il lit les données pour une zone à partir d'un fichier stocké localement et il est autoritaire pour cette zone ;
- secondaire** : il obtient les données concernant une zone à partir d'un serveur de noms autoritaire pour cette zone.

Commentaires

Des commentaires peuvent être utilisés :

- `/*` commentaire de type C (peut occuper plusieurs lignes) `*/`
- `//` commentaire de type java, C++
- `#` commentaire de type shell

Configuration comme serveur principale

Dans cette section, nous allons configurer **bind** pour être un serveur principale du domaine **smi6.net**.

Le fichier **/etc/bind/named.conf** contient la ligne :

```
include "/etc/bind/named.conf.local";
```

Cette ligne veut dire, que pour ajouter une zone locale, il faut l'ajouter dans le fichier **/etc/bind/named.conf.local**.

Zone smi6.net

Pour ajouter la zone **smi6.net**, il faut ajouter dans le fichier **/etc/bind/named.conf.local** les lignes suivantes :

```
zone "smi6.net" {  
    type master;  
    file "/etc/bind/db.smi6.net";  
};
```


Zone smi6.net

Le plus simple pour créer le fichier **/etc/bind/db.smi6.net**, est d'utiliser un fichier qui existe déjà. Par exemple :

```
sudo cp /etc/bind/db.local /etc/bind/db.smi6.net
```

Zone smi6.net

Le fichier **/etc/bind/db.smi6.net** contiendra les instructions suivantes :

```
$TTL      86400
@         IN      SOA      smi6.net.      dns.smi6.net. (
250120152      ; Serial
3600         ; Refresh (1 heure)
86400        ; Retry (1 jour)
2419200      ; Expire (28 jours)
86400 )       ; Minimum (1 jour)

IN  A  192.168.1.100

;
@         IN      NS       dns.smi6.net.
dns       IN      A        192.168.1.1
www       IN      A        192.168.1.100
r1        IN      A        192.168.1.1
pc1       IN      A        192.168.1.2
```

Signification des différents champs

- **TTL** (Time To Live) : détermine le temps, en secondes, durant lequel les informations seront conservées dans le cache.
- **SOA** (Start Of Authority) : indique le début d'un enregistrement.
- **NS** (Name Server) : identifie un serveur de nom pour un domaine.
- **A** (internet Address) : adresse internet.
- **@** : désigne le nom du domaine actuel. Il ne faut pas oublier le point (.) après le nom de domaine.
- **;** : commentaire.

Signification des valeurs numériques

- Serial** : numéro de série. Un numéro unique qui identifie la version du fichier de la zone. En général, vous avez cette valeur sous la forme de date de modification du fichier suivie d'un numéro (250120152 : 25/01/2015+2). Dans la plus part des cas, vous trouverez la date sous la forme AAAAMMJJ (250120152 devient : 201501252).
- Refresh** : (rafraîchir) le temps en secondes que doit mettre un serveur DNS secondaire pour vérifier le numéro de série.
- Retry** : le temps en secondes que doit attendre un serveur DNS secondaire après une mauvaise requête de rafraîchissement.
- Expire** : le temps en secondes qu'un serveur DNS secondaire doit utiliser les données avant de faire un rafraîchissement. Cette valeur doit être grande.
- Minimum** : le temps en secondes qui doit être utilisé pour le TTL.

Redémarrage de bind

Une fois les changements effectués, redémarrez le service **bind** en tapant la commande :

```
sudo service bind9 restart
```

Côté client

Pour configurer le client, il faut éditer le fichier de configuration de la résolution de noms **/etc/resolv.conf**.

Exemple :

```
domain smi6.net  
nameserver 192.168.1.1
```

Vérification

A partir d'un client, il suffit d'utiliser la commande **ping** pour vérifier la connexion aux différentes machines. Par exemple :

```
ping r1.smi6.net
```

Utilisation de nslookup

La commande :

```
nslookup pc1.smi6.net
```

fournira le résultat :

```
Server: 192.168.1.1
```

```
Address: 192.168.1.1#53
```

```
Name: pc1.smi6.net
```

```
Address: 192.168.1.2
```

53 correspond au numéro de port utilisé par le serveur DNS (voir fichier **/etc/services**).

Fichier de la zone inverse (Reverse Zone)

Il permet au serveur DNS de faire la résolution d'adresses vers des noms.

Il faut ajouter dans le fichier **/etc/bind/named.conf.local** les lignes suivantes :

```
zone "1.168.192.in-addr.arpa" {  
    type master;  
    file "/etc/bind/db.192";  
};
```

Ensuite, il faut créer le fichier **/etc/bind/db.192** :

```
sudo cp /etc/bind/db.127 /etc/bind/db.192
```

Zone inverse

Il faut éditer le fichier **/etc/bind/db.192** pour devenir comme suit :

```
; Fichier BIND inverse pour le reseau 192.168.1.0
;
$TTL 604800
@      IN      SOA      smi6.net.      dns.smi6.net. (
                    090320152      ; Serial
                    604800      ; Refresh
                    86400      ; Retry
                    2419200      ; Expire
                    604800 )      ;
;
@      IN      NS       dns.
1      IN      PTR      dns.smi6.net.
2      IN      PTR      pc1.smi6.net.
```

Redémarrage de bind9

Une fois le fichier inverse créé, redémarrez le service **bind9** :

```
sudo service bind9 restart
```

Configuration d'un serveur secondaire

Une fois le serveur primaire configuré, il faut avoir un autre serveur secondaire pour que la zone soit toujours disponible même si le serveur principale tombe en panne.

Nous allons supposer que nous disposons d'un autre serveur avec l'adresse **192.168.1.10**.

Activation du transfert

Dans le serveur principale il faut activer le transfert en ajoutant les options **allow-transfer also-notify** au fichier **/etc/bind/named.conf.local** comme suit :

Activation du transfert

```
zone "smi6.net" {  
    type master;  
    file "/etc/bind/db.smi6.net";  
    allow-transfer { 192.168.1.10; };  
    also-notify { 192.168.1.10; }  
};  
zone "1.168.192.in-addr.arpa" {  
    type master;  
    file "/etc/bind/db.192";  
    allow-transfer { 192.168.1.10; };  
    also-notify { 192.168.1.10; }  
};
```

Dans le serveur principal, redémarrez le service **bind** :

```
sudo service bind9 restart
```

Dans le serveur secondaire

Dans le serveur secondaire, il faut installer **bind9** de la même façon que pour le serveur principale. Puis il faut ajouter les déclarations suivantes dans le fichier **/etc/bind/named.conf.local** :

```
zone "smi6.net" {  
    type slave;  
    file "/etc/bind/db.smi6.net";  
    masters { 192.168.1.1; };  
};  
  
zone "1.168.192.in-addr.arpa" {  
    type slave;  
    file "db.192";  
    masters { 192.168.1.1; };  
};
```

Dans le serveur secondaire

Redémarrez le service **bind** :

```
sudo service bind9 restart
```


Chapitre 8

Sécurité

La sécurité doit être prise en considération lors de l'installation et l'utilisation d'un ordinateur.

Les attaques touchent généralement les trois composantes suivantes :

- La couche d'application
- Le système d'exploitation
- La couche réseau

Cependant on distingue différentes attaques au sein d'un réseau dû à la faiblesse des composants :

- faiblesses d'authentification ;
- mauvaises configurations.
- faiblesses d'implémentation ou de bogues ;
- faiblesses liées aux protocoles.

La gestion des utilisateurs est fondamentale dans la sécurité d'un système informatique. De mauvaises privilèges ou un mauvais mot de passe peuvent compromettre la sécurité d'un ordinateur.

Lors de la création d'un nouveau utilisateur avec la commande **adduser** (par exemple **adduser smi**), le répertoire personnel de l'utilisateur **smi** est créé avec les droits `drwxr-xr-x`. Il faut enlever les droits de lecture pour les autres :

```
sudo chmod 750 /home/smi
```

Pour mettre cette valeur par défaut lors de la création d'un nouveau utilisateur avec la commande **adduser**, il faut modifier la valeur de la variable **DIR_MODE** dans le fichier **/etc/adduser.conf** de la façon suivante :

```
DIR_MODE=0750
```

Pour éviter les attaques qui utilisent un dictionnaire, le mot de passe doit être fort. Il doit :

- comporter des lettres minuscules et majuscules, des nombres et d'autres caractères ;
- comporter au moins 8 caractères ;

Il ne doit pas comporter :

- le nom ou le prénom de l'utilisateur ;
- la date de naissance de l'utilisateur ;
- un mot du dictionnaire.

Pour sécuriser un réseau, il faut :

- segmenter le réseau en sous-réseaux ;
- utiliser des filtres ;
- utiliser un pare-feu.

segmenter le réseau en sous-réseaux

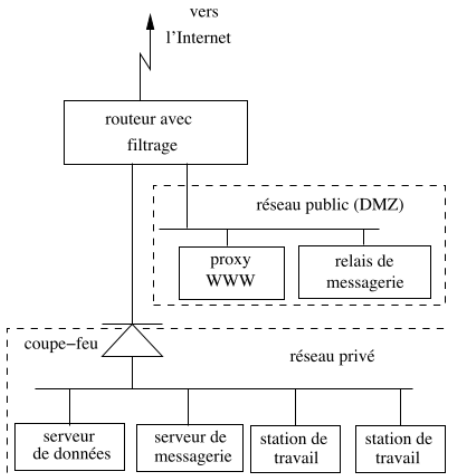
Dans le but de séparer les machines sensibles des autres machines, on peut découper un réseau en plusieurs sous-réseaux, alors que l'ensemble continue à se comporter comme un seul réseau vis-à-vis de l'extérieur.

Les règles d'accès et de trafic appliquées aux réseaux consistent à établir quels sont les type de paquets (en termes de protocole et de numéro de port) autorisés en entrée ou en sortie depuis ou vers tel réseau ou telle adresse particulière.

—> un serveur web pourra recevoir et émettre du trafic HTTP (port 80) mais n'aura aucune raison de recevoir un autre trafic sur le port 22 (protocole SSH). Appliquer ce genre de règles, c'est faire du filtrage par port.

- Le sous-réseau public (souvent appelé « zone démilitarisée » ou DMZ) devra faire l'objet de mesures de sécurité particulièrement strictes. Il est exposé à toutes les attaques en provenance de l'Internet.
- Le principe de base est : tout ce qui n'est pas autorisé est interdit.
- Il est prudent que les serveurs en zone publique contiennent aussi peu de données que possible. Idéalement, ils ne doivent pas contenir de données pour éviter qu'ils soient la cible d'attaques

Filtrage



Tiré du Livre : Sécurité Informatique - Principes Et Méthodes 2ème Edition (Eyrolles).

Les Firewall (Pare Feu)

Un **pare-feu** est un ensemble matériel ou logiciel qui trie les paquets qui circulent par son intermédiaire en provenance ou vers le réseau local, et ne laisse passer que ceux qui vérifient certaines conditions.

C'est un système de protection dédié à la sécurité d'un réseau.

Les noyaux Linux contiennent le système **Netfilter** pour manipuler le trafic réseau. Pour accepter, manipuler ou rejeter un paquet, on utilise **iptables**.

iptables est très utilisé pour mettre en place un pare-feu. Elle utilise 4 ou 5 tables (le nombre dépend du système). Une table permet de définir un comportement précis de **Netfilter**. En fait, c'est un ensemble de chaînes, elles-mêmes composées de règles.

Les tables sont :

- Filter
- NAT
- Mangle
- Raw
- security

C'est la table par défaut. Elle s'utilise sans l'option **-t** et contient les chaînes :

- **INPUT** : pour les paquets destinés aux sockets local ;
- **FORWARD** : pour les paquets routés ;
- **OUTPUT** : pour les paquets générés localement.

Elle est consultée quand un paquet qui crée une nouvelle connexion est rencontré. Elle consiste en trois chaînes :

- **PREROUTING** : pour les paquets qui entrent ;
- **OUTPUT** : pour les paquets générés localement avant le routage ;
- **POSTROUTING** : pour les paquets qui sortent.

sert à modifier d'autres paramètres des paquets IP (notamment le champ ToS — Type Of Service — et les options). Elle consiste en cinq chaînes :

- **PREROUTING** : paquets entrant avant le routage ;
- **OUTPUT** : pour les paquets générés localement avant le routage ;
- **INPUT** : paquets arrivant au système lui-même ;
- **FORWARD** : paquets routés via le système ;
- **POSTROUTING** : pour les paquets qui sortent

La table **Raw** contient les chaînes :

- **PREROUTING** : pour les paquets arrivant de n'importe quelle interface réseau
- **OUTPUT** : pour les paquets générés par les processus locales

Dans les machines virtuelles **netkit**, cette table n'est pas disponible.

Dans ce qui suit nous allons utiliser la table par défaut.

Initialisation des tables

On vide les chaînes au niveau de la table **Filter** :

```
pc1: # iptables -F
```

On supprime les éventuelles chaînes personnelles :

```
pc1: # iptables -X
```

Maintenant faisons pointer par défaut les chaînes de la table **Filter** sur **DROP** (Rejet) :

```
pc1: # iptables -P INPUT DROP
```

```
pc1: # iptables -P OUTPUT DROP
```

```
pc1: # iptables -P FORWARD DROP
```

Les entrées et les sorties sont bloquées.

Un **ping** de **pc1** vers **pc2** donne :

```
pc1: # ping -c 1 192.168.100.2
PING 192.168.100.2 (192.168.100.2) 56(84) bytes of data:
ping: sendmsg: Operation not permitted

--- 192.168.100.2 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, t
```

Les paquets ne sortent pas de **pc1**.

Un **ping** de **pc2** vers **pc1** donne :

```
pc2: # ping -c 1 192.168.100.1
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data:

--- 192.168.100.1 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, t
```

Les paquets arrivent sur **pc1** et sont rejetés. Il suffit de le vérifier avec **tcpdump**.

Test vers la boucle locale

Même un **ping** vers **localhost** est rejeté :

```
pc1: # ping -c 1 localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
ping: sendmsg: Operation not permitted

--- localhost ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, t
```

Examen de la table Filter

Examinons l'état de la table **Filter** :

```
pc1: # iptables -L
```

```
Chain INPUT (policy DROP)
```

```
target          prot opt source                                     destination
```

```
Chain FORWARD (policy DROP)
```

```
target          prot opt source                                     destination
```

```
Chain OUTPUT (policy DROP)
```

```
target          prot opt source                                     destination
```

Autorisation de la boucle locale

On autorise des entrées locales :

```
pc1: # iptables -A INPUT -i lo -j ACCEPT
```

On autorise des sorties locales :

```
pc1: # iptables -A OUTPUT -o lo -j ACCEPT
```

Alors si on fait un **ping** sur la machine elle-même on voit que ça marche :

```
pc2:~# ping -c 1 localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64

--- localhost ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time
rtt min/avg/max/mdev = 0.093/0.093/0.093/0.000 ms
```


Examinons l'état de la table **Filter** :

```
pc1: # iptables -L -v
```

```
Chain INPUT (policy DROP 0 packets, 0 bytes)
```

pkts	bytes	target	prot	opt	in	out	source	dest
14	1176	ACCEPT	all	--	lo	any	anywhere	anyw

```
Chain FORWARD (policy DROP 0 packets, 0 bytes)
```

pkts	bytes	target	prot	opt	in	out	source	dest
------	-------	--------	------	-----	----	-----	--------	------

```
Chain OUTPUT (policy DROP 4 packets, 336 bytes)
```

pkts	bytes	target	prot	opt	in	out	source	dest
14	1176	ACCEPT	all	--	any	lo	anywhere	anyw

L'option -v veut dire verbose (bavard), donne plus de détails.

Autoriser le trafic d'une connexion déjà établie

Pour autoriser une connexion déjà ouverte d'envoyer et de recevoir du trafic :

```
pc1: # iptables -A INPUT -m conntrack -ctstate  
ESTABLISHED -j ACCEPT
```

```
pc1: # iptables -A OUTPUT -m conntrack -ctstate  
ESTABLISHED -j ACCEPT
```

Ouverture de quelques ports/services

Pour autoriser les connexion au serveur **SSH**, il faut :

- autoriser les entrées des requêtes au port **ssh**

```
pc1: # iptables -A INPUT -p tcp -dport ssh -i  
eth0 -j ACCEPT
```

- autoriser les sorties des requêtes utilisant **ssh**

```
pc1: # iptables -A OUTPUT -p tcp -dport ssh -o  
eth0 -j ACCEPT
```

Pour autoriser l'envoi et la réception de messages **ICMP**, il faut :

- autoriser les entrées des requêtes utilisant le protocole **icmp**
`pc1: # iptables -A INPUT -p icmp -i eth0 -j ACCEPT`
- autoriser les sorties des requêtes utilisant le protocole **icmp**
`pc1: # iptables -A OUTPUT -p icmp -o eth0 -j ACCEPT`

Protocoles Sécurisés

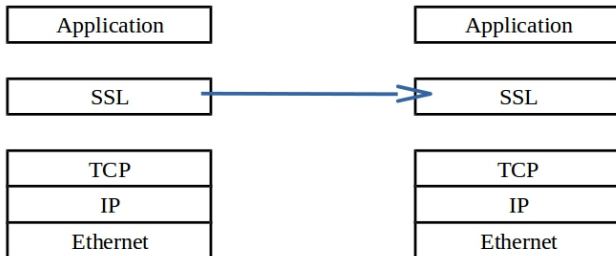
La plupart des protocoles TCP ne sont pas sécurisés. Ce qui signifie que les données transitent en clair sur le réseau.

Pour une sécurité des données qui circulent sur le réseau, des protocoles **sécurisés** ont été développés.

SSL (Secure Sockets Layer)

SSL est un logiciel permettant de sécuriser les communications sous HTTP ou FTP.

Le rôle de SSL est de crypter les messages entre un navigateur et un serveur Web. Le niveau d'architecture où se place SSL est illustré dans la figure suivante. Il s'agit d'un niveau compris entre TCP et les applications.



SSL (Secure Sockets Layer)

Un serveur web qui utilise SSL possède une URL (Uniform Resource Locator) qui commence par **https://** (s : secured - sécurisé).

Établissement de la connexion SSL

L'établissement d'une connexion SSL se présente comme suit :

- ➊ authentification du serveur auprès du client (chiffrement à clé publique) ;
- ➋ choix d'un algorithme de chiffrement pour l'établissement de la connexion sécurisée ;
- ➌ optionnellement, authentification du client auprès du serveur (techniques de chiffrement à clé publique) ;
- ➍ échange des secrets partagés nécessaires à la génération d'une clé secrète (clé de session) pour le chiffrement symétrique ;
- ➎ établissement d'une connexion SSL chiffrée à clé secrète.

TLS (Transport Layer Security) - Couche de Transport Sécurisée

C'est le successeur de SSL. Il ne présente que des différences mineures par rapport à SSL.

IPSec consiste à incorporer les techniques de chiffrement (et d'autres, relatives aussi à la sécurité) au protocole IP lui-même, plutôt que d'avoir recours à des solutions externes.

IPSec utilise 2 protocoles pour implémenter la sécurité sur un réseau IP :

- 1 Entête d'authentification (AH - Authentication Header) permettant d'authentifier les messages.
- 2 Protocole de sécurité encapsulant (ESP - Encapsulating Security Payload) permettant d'authentifier et de crypter les messages.

Avec l'un ou l'autre de ces protocoles, IPSec peut fonctionner en mode transport ou en mode tunnel :

- en mode tunnel chaque paquet IP est encapsulé dans un paquet IPSec lui-même précédé d'un nouvel en-tête IP ;
- en mode transport un en-tête IPSec est intercalé entre l'en-tête IP d'origine et les données du paquet IP.

IPSec : modes de communication

Paquet IP sans IPSec :



- mode transport :



- mode tunnel :



Etablissement d'une connexion IPSec

- ❶ 2 machines doivent s'accorder pour l'utilisation des algorithmes et protocoles à utiliser
- ❷ Une SA (Security Association - Association Sécurisée) est établie pour chaque connexion.
- ❸ Une SA comprend :
 - Un algorithme de chiffrement
 - Une clé de session (Internet Key Exchange)
 - Un algorithme d'authentification (SHA1, MD5)

Les réseaux privés virtuels (VPN : Virtual Private Network)

- Un réseau VPN permet de chiffrer le flux de l'ensemble du trafic sur un ou plusieurs itinéraires donnés.
- Il s'agira d'établir un canal chiffré entre deux nœuds quelconques de l'Internet, ces nœuds pouvant eux-mêmes être des routeurs d'entrée de réseaux.
- Le chiffrement permet aussi d'établir un VPN personnel pour un utilisateur, par exemple entre son ordinateur portable et le réseau local de l'entreprise.

- Permet de créer un tunnel chiffré sur une infrastructure publique entre 2 points.
- Les logiciels de vpn peuvent s'appuyer sur IPSec ou SSL/TLS