



ci-dessous un rappel de ce que nous avons vu en TD/TP.

Sommaire

1.	structure de l'arbo Linux	1
2.	Configuration réseau	2
3.	dépannage réseau	4
4.	References	4

1. Les consoles

Travailler sous LINUX implique une connexion au système : une session n'est jamais anonyme : login/password obligatoire

Une session utilisateur se matérialise par un programme "interpréteur de commandes" qui s'appelle le shell (une "coquille", qui enveloppe le noyau) dans une console.

6 consoles « virtuelles » : équivaut à se connecter 6 fois.

Basculer en console :

Alt+fn (sous console) ou ctrl+Alt+fn (sous X-windows) avec n de 1 à 6

Alt+f7 pour revenir sous X

2. structure de l'arbo Linux

REGLE DE BASE Sous LINUX, TOUT EST FICHER, organisé en ARBRE (dont la racine est nommée / et dont le propriétaire est root)

Il s'agit d'une arborescence logique, indépendante de la position physique des sous-répertoires (peut s'étendre sur plusieurs partitions, sur plusieurs disques, sur des disques réseaux).

```

/           La racine
/etc        Tous les fichiers de configurations du système
/bin        binaires: commandes executables accessibles à tous
/sbin       binaires systemes : commandes reservées à root
/dev        devices = fichiers spéciaux pour communiquer avec les périphériques
/tmp        temporaire (accessible en écriture à tous)
/var        fichiers variables en taille au cours du temps (traces, mails..)
/home       répertoires personnels des utilisateurs (variables aussi)
/usr        externes / les suppléments - accessibles à tout utilisateur
           sa structure reproduit celle de la racine :
/usr/etc
/usr/bin
/usr/sbin
/usr/local/etc
/usr/local/bin
/usr/local/sbin

```

Répertoires plus spécifiques (peut varier d'un Unix à l'autre)

```

/proc       ce pseudo-répertoire contient une "image" du système
           Essais : cat /proc/cpuinfo, cat /proc/net/arp, cat /proc/net/dev
/root       répertoire personnel du super-utilisateur root

```

```
/opt      applications supplémentaires (comme starOffice, java ..)
/lib      les bibliothèques (équivalents DLL) et les modules du noyau
/boot     le noyau vmlinux et les fichiers de démarrage (grub)
/mnt      points de montage des périphériques
/media    comme /mnt, sauf que dans un Linux bien configuré crée un icône sur le bureau
```

Rappel : séparer le /home et /var au minimum pour un serveur

```
TP
1/ en tant qu'utilisateur lambda, essayer de créer un fichier dans /etc /bin /dev /tmp
/var /home /home/autre_user /usr /proc
2/ sous 2 consoles, essayer la commande « tty »(donne Le nom du device du terminal). Es-
sayer ensuite d'envoyer un message avec « echo toto > /dev/ttyx » (x à remplacer)
3/de meme sous X-windows : ouvrir 2 xterm, « tty » devrait donner "dev/pts/x"
essayer echo "toto" > /dev/pts/1 depuis /dev/pts/0 par exemple
4/ idem en basculant en console (ctrl+alt+F1)
```

3. Configuration réseau

Fichiers à voir et à connaître :

- /etc/network/interfaces
- resolv.conf et hosts
- nsswitch.conf

Les drivers pour le réseau sont dans le noyau, car le réseau est géré dans le noyau sous linux.

Forme du nom des carte réseau (on dit interface) : Eth0..x et lo (localhost)

Configurer les interfaces de manière permanente

Voir aussi ifconfig. Format du fichier /etc/network/interfaces

Mode DHCP (IP automatique)

```
iface eth0 inet dhcp
```

ou

Mode Manuel

```
iface eth0 inet static
address 192.168.1.10
netmask 255.255.255.0
gateway 192.168.1.254
```

options qu'on peut ajouter :

```
allow-hotplug eth0 → montage à chaud de l'interface (activer/desactiver la carte)
auto eth0 → initialisation au démarrage du système de l'interface
```

redémarrer le support du réseau pour prendre en charge les modifs :

```
/etc/init.d/networking restart
```

Configurer le dns

/etc/resolv.conf

Résolution de nom avec un serveur DNS

```
search noms.. (liste de domaines à essayer, si le nom est incomplet)
nameserver liste IP (si nom il faut dans hosts!)
```

/etc/hosts

table de correspondance des IP des machines du sous-réseau et de leur noms

peut compléter ou remplacer le DNS

```
format      IP      NOM
```

/etc/hostname

nom réseau de la machine : commande associée « hostname »

commandes

ifconfig

- Parametres: <nom de la carte> <adresse IP>
- Sans parametres: status sur toutes les cartes réseaux du système
- Permet de configurer une adresse IP à la volée (avec effet immédiat) mais configuration perdue au prochain reboot

Ifup/ifdown: fonctionne avec /etc/network/interfaces

- Ex: ifup eth0 ou ifup -a (tous)
- Ex: ifdown eth0 ou ifdown -a

Route : voir la table de routage

- route add default gw IP
- route del default

Voir aussi routef: affichage plus détaillé

Arp : utilisé moins souvent (correspondance IP/MAC)

Ping <ip> : ;-)

Traceroute/tracert <ip>: affiche le chemin IP jusqu'à l'IP donné en parametre (nslookup/dig)

Dhclient <ethx> : lance une requete DHCP client pour l'interface donnée en parametre

Particularité de VirtualBox

la carte réseau est configuré en NAT sous les paramètres de machine de virtualbox.

Pour Voir les devices actifs : cat /proc/net/dev

Fixer le numéro d'interface réseau en fonction de l'adresse MAC

sous VirtualBox en particulier, rien n'est sur par rapport à l'interface qui sera choisi par default. Cela peut etre eth0, eth8..

La solution vient via le fichier /etc/iftab qui permet de forcer l'association d'une interface réseau avec une adresse MAC physique donnée.

Dans les anciennes versions de Debian/Ubuntu, c'était le package ifrename qui créait ce fichier mais depuis la version 6.06, plus besoin d'installer ce paquet. Il suffit de créer le fichier /etc/iftab manuellement - celui-ci sera utilisé par udev au démarrage de la machine.

Exemple de fichier iftab :

```
eth0 mac 00:13:d4:52:40:b3
eth1 mac 00:12:f0:d7:be:92
```

nsswitch.conf

name system switch conf

format :

```
passwd: files ldap      (users)
shadow: files ldap     (users)
group: files ldap      (users)
hosts: files dns       (machines)
```

on peut y mettre principalement:

```
ldap          Utiliser LDAP
nis+, nis ou yp  Utiliser NIS ou NIS+ appelé aussi YP
dns           Utiliser DNS
files         Utiliser les fichiers locaux sous /etc/
```

```
TP
1/ ifconfig: changer son IP à la volée
2/ ajout/delete de gw

3/ nsswitch avec les noms de machines
ping www.google.fr
supprimer dns dans nsswitch.conf
ping www.google.fr
remettre dns dans nsswitch.conf

mettre dans /etc/hosts une IP bidon pour google.fr
nsswithc.conf mettre "files dns"
ping www.google.fr
donne l'adresse bidon
nsswithc.conf mettre "dns files"
ping www.google.fr
donne la bonne adresse
```

Aides sur Commandes

Aide sur les commandes :

- cmd -h
- cmd --help
- man cmd

passer root dans une console :

```
sudo su
sudo -i
```

4. dépannage réseau

```
1 vérifier le cable
2 loupiotte verte sur la carte
3 ifconfig -a donne eth0 et lo
4 sinon cat /proc/net/dev
4 ping 1 ip sur le réseau local
4 ping ip de la passerelle (gateway)
5 ping 1 ip en dehors du réseau
6 route -> default gw
  route del default
  route add default gw ip
7 dns
  /etc/resolv.conf
```

5. References

Pour approfondir, ci-dessous les pointeurs dont je vous ai parlé en cours

sites génériques sur Linux

<http://lea-linux.org>

<http://www.aful.org>

<http://www.traduc.org/> liste de howto en français (basé sur <http://tldp.org/> The Linux Documentation Project)