Administration Réseau sous Linux
Partie - I

Pr A. FERHAT HAMIDA

- Dans cet atelier, on devait travailler en groupe pour :
  - Installer des distributions Linux : debian et/ou centos sur plusieurs postes.
  - Se familiariser avec un editeur : vi ou vim
  - Rappeler quelques unes des commandes principales de Linux : ls, cd, mkdir, rm, grep, find, sed, awk, more, less, ps, mount, etc.
  - Installer et configurer une carte réseau
  - Configurer le réseau
  - Installer et configurer un serveur web Apache : LAMP-stack (Linux, Apache, MySQL et PHP)
  - Sécuriser le serveur

- Malheureusement, à cause de la pandémie et du confinement qui s'en suivit, vous devez faire certaines étapes en solo.
- <u>Prérequis :</u>
  - Être initié à Linux et familier avec les commandes principales
  - Être familier avec la ligne de commande
  - Être familier avec les réseaux et leurs terminologies : ip, route, dhcp, etc.

- Nous procèderons par plusieurs parties :
  - Installer une carte réseau
  - Configurer l'interface réseau
  - Installer et configurer LAMP
- Toutes les commandes sont exécutées en mode *super utilisateur* (*root*)
- Pour toutes vos questions, et vous devez en avoir, nous pouvons en discuter sur la page facebook (*de préférence*)

  # https://www.facebook.com/groups/235515490878153

  ou par email :

  # ferhat.abdelhak@gmail.com

La carte réseau – commandes disponibles

- Il y a deux groupes de commandes permettant de configurer les cartes réseaux :
  - L'ancien schéma : le package *net-tools* fournit des commandes tels que: `ifconfig, route, nameif, iwconfig, iptunnel, netstat et arp.`

    Beaucoup d'administrateurs réseaux et beaucoup de *texbook* utilisent encore ces commandes qui doivent disparaître un jour ou l'autre et qui sont officiellement obsolètes dans les nouvelles versions des distributions.

La carte réseau – commandes disponibles (suite)

- Il y a deux groupes de commandes permettant de configurer les cartes réseaux :
  - Le nouveau schéma : le package *iproute2* fournit des commandes tels que: `ip et ss`.

    Ces commandes sont plus puissantes que les précédentes et sont recommandées pour toute nouvelle utilisation. La commande *ip* joue le rôle d'un *couteau suisse* ; elle accomplie la majorité des fonctions.
  - Pour chaque commande, on essayera de donner les deux versions.

La carte réseau – nom de l'interface

- Il y a deux façons de nommer les interfaces réseaux:
  - L'ancien schéma :
    - Ethernet cablé : *eth0*, *eth1*, etc.
    - Interface sans fil : *wlan0*, *wlan1*, etc.
  - Le nouveau schéma :
    - Ethernet embarqué : enxxx
    - Ethernet attaché au bus PCI : *pxxxx*
    - Wifi : *wpxxxx*
  - On prendra *eth0* comme nom d'interface. Il peut être le changé selon l'interface disponible.

La carte réseau - méthodes de configuration

- Il y a deux méthodes pour configurer les interfaces réseaux:
  - La méthode classique:
    - On utilise les commandes: `ifconfig, route, ip, etc.`
  - La méthode automatique (facile) :
    - On utilise les utilitaires de configuration : `nmtui ou nmcli`

La carte réseau – détection de la carte

- Au démarrage de Linux, le noyau (*kernel*) devrait détecter les cartes réseau. On est en face de deux cas :
- **Cas 1 : carte détectée :**
  - Pour obtenir la liste des interfaces réseau détectées, on utilise la commande

```
# ifconfig -a
```

≫ ou

```
# ip addr
```

La carte réseau – installation module

- Au démarrage de Linux, le noyau (*kernel*) devrait détecter les cartes réseau. On est en face de deux cas :
- **Cas 2 : carte non détectée :**
  - Si la carte n'est pas détectée, il faut charger le module correspondant

```
# modeprobe <nom du module>
```

  - Pour ce faire, on doit trouver le module.

La carte réseau – installation module – version du noyau

- Les modules disponibles pour le noyau se trouvent dans le dossier

  # /lib/modules/<nom du noyau> /kernel/drivers/net/

  - On doit d'abord connaître : <nom du noyau> en exécutant la commande :

  # uname -r

    - Exemple de réponse :

  # 3.10.0-957.21.3.el7.x86_64

La carte réseau – installation module – dossier des modules

- On a maintenant le dossier :

```
# /lib/modules/3.10.0-957.21.3.el7.x86_64
  /kernel/drivers/net/
```

- On exécute la commande suivante pour obtenir la liste des modules :

```
# ls  /lib/modules/3.10.0-957.21.3.el7.x86_64
  /kernel/drivers/net/
```

La carte réseau – installation module – liste des modules

- De façon plus élégante et plus sûre, la commande suivante fait tout le nécessaire en une seule étape

```
# ls  /lib/modules/`uname -r`/kernel/drivers/net/
```

- Si le module existe, on peut l'installer. Sinon il faut le télécharger, ensuite l'installer.

La carte réseau – installation module – recherche internet

- La meilleure solution pour connaitre le nom du module associé à une carte est de faire une recherche internet, souvent dans le site du constructeur.
- Le noyau offre parfois d'importantes informations sur les cartes réseau via la commande :

```
# dmesg | grep eth0
```

- On obtient la liste des cartes réseaux attachées au bus PCI en exécutant la commande :

```
# lspci |grep Ethernet
```

Configurer l'interface réseau – adresse ip

- On attribue une adresse IP à l'interface :

```
# ifconfig eth0 <adresse ip>
```

  ≫ ou

```
# ip addr add  <adresse ip> dev eth0
```

- Par exemple :

```
# ifconfig  eth0 192.168.1.20
```

  ≫ ou

```
# ip addr add  292.168.1.20 dev eth0
```

Configurer l'interface réseau – mask et test

- On peut spécifier le masque avec l'option netmask (généralement, il est attribuer automatiquement)

  ```
  # ifconfig  192.168.1.20 netmask 255.255.255.128
  ```

  ≫ ou

  ```
  # ip addr add  292.168.1.20/24 dev eth0
  ```

  - 24 représente le nombre de bits masqués

- On teste la configuration par la commande :

  ```
  # ifconfig eth0
  ```

  ≫ ou

  ```
  # ip addr show eth0
  ```

Configurer l'interface réseau - passerelle et test

- Pour ajouter une passerelle (gate way), on utilise la commande route :

```
# route add default gw <adresse ip>
```

  - ou

```
# ip route add default via <adresse ip>
```

- Pour afficher les routes vers les différents réseaux :

```
# route -n
```

  - ou

```
# ip route show
```

Configurer l'interface réseau - test

- Pour vérifier si la carte fonctionne, on essaie de communiquer avec une autre machine. On exécute :

```
# ping <adresse ip 2eme machine>
```

- Pour vérifier l'état de toutes les interfaces, on peut utiliser la commande

```
# netstat -a
```

  - ou

```
# ss -a
```

Configurer l'interface réseau - nom du hôte (hostname)

- La commande *hostname* permet de trouver <le nom du host> (le nom du poste sur lequel on travaille)

  # hostname

  - Exemple de réponse : **mypc.host**

- On peut le changer pour la session en cours (il n'est pas conservé lors du prochain démarrage)

  # hostname <nom du host>

  - Exemple :

  # hostname rt.elec.serv

- On peut le changer définitivement:

  # hostnamectl set-hostname --static "rt.elec.serv"

Configurer l'interface réseau - DNS

- Comme on le sait, pour communiquer entre eux, à chaque ordinateur et à chaque site est assignée une adresse *ip* unique. D'autre part, chaque ordinateur, et chaque site, est connu par un nom (p.ex www.google.com ou rt.elec.serv); question de facilité d'utilisation. On doit connaître la correspondance *ip-nom* pour pouvoir communiquer avec un autre site, ou ordinateur.
- Un serveur DNS garde une telle correspondance. On doit ajouter des serveurs DNS à notre configuration dans des fichiers bien spécifié, comme on le verra juste après.
  - p.ex le serveur publique de google 8.8.8.8

Configurer l'interface réseau – changement permanent

- Les configurations qu'on a vu sont temporaires ; elles sont perdues au redémarrage. Pour devenir permanentes, on doit modifier les fichiers de configuration dont la syntaxe et les emplacements varient selon la distribution et sa version.
- Pour centos, ce sont les fichiers

```
# /etc/sysconfig/network-scripts/ifcfg-eth0
# /etc/sysconfig/network
# /etc/resolv.conf
```

- Pour debian, ce sont les fichiers

```
# /etc/network/interfaces
# /etc/host.conf
# /etc/resolv.conf
```

Configurer l'interface réseau – changement permanent - CentOS7

- Pour centos7, dans le fichier

  # /etc/sysconfig/network-scripts/ifcfg-eth0

  - Ajouter les lignes suivantes

```
IPADDR=192.168.1.20
BOOTPROTO=static
NETMASK=255.255.255.0
TYPE=Ethernet
ONBOOT=yes
GATEWAY=192.168.1.1
```

    - ONBOOT=yes pour que le serveur soit activé automatiquement au démarrage.

Configurer l'interface réseau – changement permanent - CentOS7

- Pour centos7, dans le fichier

  # /etc/sysconfig/network

  - Ajouter les lignes suivantes

```
HOSTNAME=rt.elec.serv
DNS1=192.168.1.1
DNS2=8.8.8.8
```

Configurer l'interface réseau – changement permanent - CentOS6

- Pour centos6, dans le fichier

  # /etc/sysconfig/network-scripts/ifcfg-eth0

  - Ajouter les lignes les suivantes

```
DEVICE=eth0
TYPE=Ethernet
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.1.20
NETMASK=255.255.255.0
ONBOOT=yes
```

    - ONBOOT=yes pour que le serveur soit activé automatiquement au démarrage.

Configurer l'interface réseau – changement permanent - CentOS6

- Pour centos6, dans le fichier

  # /etc/sysconfig/network

  - Ajouter les lignes suivantes

    HOSTNAME=rt.elec.serv
    NETWORKING=yes
    GATEWAY=192.168.1.1

- Dans le fichier

  # /etc/resolv.conf (DNS)

  - Ajouter les lignes suivantes

    nameserver 192.168.1.1
    nameserver 8.8.8.8

Configurer l'interface réseau – changement permanent - debian

- Pour debian, dans le fichier

  # /etc/network/interfaces

  - Ajouter les lignes

```
auto eth0
iface eth0 inet static
        address 192.168.1.20
        netmask 255.255.255.0
        gateway 192.168.1.1
        dns-nameservers 8.8.8.8
```

Configurer l'interface réseau – changement permanent - debian

- Pour debian, dans le fichier

  # /etc/resolv.conf

  - Ajouter les lignes suivantes

    nameserver 192.168.1.1
    nameserver 8.8.8.8

- Dans le fichier,

  # /etc/hosts

  - Ajouter les ips qui ont des aléas, p.ex

    127.0.0.1 rt.elec.serv
    192.168.1.55 mon.smartphone

    - Ce fichier peut exister sur CentOS

Configurer l'interface réseau – changement permanent - nmtui

- Le changement permanent peut être effectué en utilisant l'utilitaire nmtui ou nmcli. Les professionnels préfèrent travailler avec nmcli
- L'utilitaire nmtui est une application en mode texte sous forme de menus facile à suivre et à utiliser. Elle se charge de la majorité de la configuration. Sauf que certaines options ne sont pas configurés comme ONBOOT=yes. Vous devez toujours vérifier les fichiers de configuration.
- Souvent, lorsque Linux *crash* et on démarre en mode texte pour le réparer, on n'a pas accès à internet. On utilise *nmtui* pour établir la connexion.