Notes OS Open Source Pratique

Grégoire Roumache

Octobre 2020

Table des matières

1 Voir les logs **2**

2 Touver un fichier **2**

3 Révisions **3**
3.1 Manipulation sur l'installation . . . 3
3.2 Rappel sur la configuration réseau . . . 4
3.3 Rappels sur la gestion des utilisateurs et des fichiers . . . 5

4 Administration de base **7**
4.1 Manipulation - DPKG . . . 7
4.2 Manipulation - apt avancé . . . 8
4.3 Manipulation - /proc . . . 8
4.4 Manipulation - Gestion login avancé . . . 9
4.5 Manipulation - Compilation . . . 11

5 Services réseaux de base **12**
5.1 Manipulation - NTP . . . 12
5.2 Manipulation - FTP . . . 12
5.3 Manipulation - SSH . . . 13
5.4 Manipulations avancées . . . 14

6 Gestion des systèmes de fichiers **16**
6.1 Manipulation - Partitionnement . . . 16
6.2 Rappel sur les systèmes de fichiers . . . 17
6.3 Manipulation - Systèmes de fichiers . . . 18
6.4 Manipulation - Montage des partitions . . . 18
6.5 Manipulation - Quotas . . . 18
6.6 Rappel sur le RAID . . . 20
6.7 Manipulation - RAID . . . 20
6.8 Rappel théorique sur le LVM . . . 21
6.9 Manipulation - LVM . . . 22
6.10 Manipulation - RAID + LVM . . . 24
6.11 Manipulation avancées . . . 24

7 Network addressing services **25**
7.1 Manipulation de base – DNS . . . 25
7.2 Autre configuration DNS . . . 28
7.3 Manipulation de base – DHCP . . . 29
7.4 Manipulation avancée . . . 31

8 Démarrage, initialisation, gestion des processus **37**
8.1 Le chargeur de démarrage . . . 37
8.2 Systemd . . . 37
8.3 Gestion des processus . . . 38
8.4 Gestion des modules kernel . . . 40
8.5 PAM . . . 40

9 Serveur WEB **41**
9.1 Concepts . . . 41
9.2 Manipulation . . . 43

10 Files Sharing **46**
10.1 Notions . . . 46
10.2 Manipulation modifiée – samba . . . 47
10.3 Manipulation modifiée – nfs . . . 49

---

1 Voir les logs

Attention ! Quand on a un problème, vérifier les logs.

- **Meilleure option** : `journalctl -xe`
- Voir les logs en continu : `journalctl -fxe`
- Voir le statut d'un service : `systemctl status <service>`
- Voir les derniers logs : `dmesg | less`
- Lister les fichiers logs : `ls /var/log`
- Voir les derniers logs système (sauf auth) : `tail [-n <nb_lignes>] /var/log/syslog`
- Voir les logs en continu : `tail -f /var/log/syslog`

Logs Apache2:

- Les logs sont dans le dossier : `/var/log/apache2/`
- En cas de problème pour relancer apache : `tail /var/log/apache2/error.log`

Commandes cool pour lister les services (ceux qui ont un problème sont en rouge) :

- `systemctl list-units --type service -all`
- `systemctl list-unit-files --type service`

Pour déboguer une configuration dns :

- `sudo named-checkconf -z`

2 Touver un fichier

`find <folder> -name <nom_fichier>`

3 Révisions

3.1 Manipulation sur l'installation

- Lancer l'installation en mode texte en ANGLAIS dans une VM virtualbox avec un DD de taille dynamique de 20 Go.
- Suivre les instructions en configurant correctement le clavier (belge), etc.
- Réaliser manuellement le partitionnement suivant :
  - sda1 : /boot = 256 Mo
  - sda2 : swap = 1 Go
  - sda3 : / = 10 Go
  - sda4 : partition logique
    - sda5 : /home = 2 Go
    - sda6 : /tmp = 2 Go
    - free space = max
- Compte utilisateur : user. Mots de passe : tttttt.
- Choisir comme mot de passe pour root : tttttt.
- N'installer que le système de base.

  > Guide pour l'installation:
  >
  > 1. lancer l'installation *sans* le mode graphique
  > 2. prendre langue: *english*, pays: *other -> europe -> Belgium*
  > 3. prendre pays des paramètres par défaut: *united states*, clavier: *belgian*
  > 4. hostname: *debian*, laisser vide le domain name
  > 5. mot de passe root: *tttttt* ($= 6 \times t$), utilisateur: *user*, mot de passe user: *tttttt*
  > 6. lancer le partitionnement en mode manuel
  > 7. sur *SCI1 (0,0,0) (sda) ...*, taper *enter*, puis sur *yes*
  > 8. sur *free space*, taper *enter*, puis *create a new partition*
  >   (a) 256 MB - primary - beginning - ext4 – /boot – bootable flag = *on*
  >   (b) 1 GB - primary - beginning - swap area
  >   (c) 10 GB - primary - beginning - ext4 - /
  >   (d) 2 GB - logical - beginning - ext4 - /home
  >   (e) 2 GB - logical - beginning - ext4 - /tmp
  > 9. taper *enter* sur *finish partitionning and write changes to disk*
  > 10. dans *software selection*, ne sélectionner que *standard system utilities*
  > 11. installer GRUB

- Si besoin, ajouter un proxy. Dans quelle situation cela peut-il être utile ?

  > En général, un proxy est utilisé pour:
  >
  > - accélérer la navigation
  > - la sécurité du réseau local
  > - le filtrage et l'anonymat

- Prendre le miroir réseau par défaut.
- Installer le boot manager dans le MBR. Où peut-il être encore installé ? Expliquer la différence et comment cela fonctionne.

  - MBR = master boot record = 1er secteur d'un disque dur
  - On peut aussi l'installer sur n'importe quelle partition àpd moment où il y a un flag `boot` (seulement UEFI, pas BIOS).
  - La différence est que avec le MBR, on charge le GRUB directement. Avec l'autre méthode, l'UEFI va chercher la partition avec le boot dans la table de partitions.

- Quel est le boot manager installé ?

  GRUB = grand unified bootloader

- Finir l'installation, se connecter au système, vérifier que tout fonctionne correctement.

  1. Pour avoir accès aux commandes système telles que: `useradd`, `mkfs` ou `fdisk`:
     - `export PATH="$PATH:/sbin"`
  2. Ajouter `user` au sudoers:
     - `su root`
     - `apt install sudo`
     - `usermod -aG sudo user`
     - `exit`

3.2 Rappel sur la configuration réseau

- Vérifier la configuration IP : `ip addr show`
- Vérifier la MAC address : `ip link show <interface>`.
- Vérifier la default gateway : `ip route list`.
- Vérifier les serveurs DNS `cat /etc/resolv.conf` (ou `systemd-resolve -status`).
- Adresse IP statique:
  - Nettoyer : `ip addr flush dev enp0s3`
  - Ajouter : `ip addr add 192.168.50.5/24 dev enp0s3`
  - Supprimer : `ip addr del 192.168.50.5/24 dev enp0s3`
- Adresse IP dynamique :
  - `dhclient [-v ] enp0s3`
  - `pkill dhclient`
- Adresses IP supplémentaires : `ip addr add 192.168.50.50/24 dev enp0s3`
- Nom d'hôte : `hostname [nouveauNom]`
- Passerelle par défaut & Passerelles supplémentaires :
  - `ip route add 10.10.20.0/24 via 192.168.50.100`
  - `ip route del 10.10.20.0/24`
  - `ip route add default via 192.168.50.100`
  - `ip route del default`
- Mac Address : `ip link set dev interface address XX:XX:XX:XX:XX:XX`
- Carte réseau :
  - `ip link set enp0s3 up`

  - `ip link set enp0s3 down`

- Déconfigurer - arrêter / Configurer-activer le réseau une carte :
  - `ifup enp0s3`
  - `ifdown enp0s3`

- Paramètres de base (relancer le service réseau : `systemctl restart networking`) :

```
auto enp0s3
iface enp0s3 inet static
address X.X.X.X
netmask X.X.X.X
gateway X.X.X.X
dns-nameservers X.X.X.X Y.Y.Y.Y
```

- Mac Address : fichier */etc/network/interfaces* : `hwaddress ether 00:01:04:1b:2C:1F`

3.3 Rappels sur la gestion des utilisateurs et des fichiers

- Se connecter en tant qu'utilisateur dans une console : `su user`.
- Devenir root temporairement : `su root`.
- Lister à l'écran les fichiers /etc/passwd et /etc/group : `cat <fichier>`.
- Expliquer la signification des champs pour l'utilisateur root, user, et le groupe user: `man 5 passwd`.
- Créer les groupes grp1, grp2, et grp3. Le GID du groupe 3 doit être 999 : `groupadd [-g <gid>] <groupe>`.
- Créer les utilisateurs user1, user2 et user3 qui doivent avoir respectivement comme groupe principal grp1, grp2, et grp3 :
  - `adduser [--ingroup <group>] <user>`
  - vérification : `groups <user>`
- user2 doit pouvoir accéder aux fichiers des groupes grp2 et grp3, avec les mêmes droits que ces derniers. Vérifier que c'est bien le cas.
- Supprime le groupe 3. Est-ce possible ? Pq ? — Non car grp3 est le groupe primaire de user3.
- Supprime l'utilisateur 3 en gardant sans supprimer le répertoire personnel ni le groupe 3.
- À qui appartient le répertoire /home/user3 désormais ? Pq ? `stat /home/user3`, UID = 1003/unknown.
- Comment retrouver les fichiers qui appartenaient à l'utilisateur 3 ou au groupe 3 ? Comment les supprimer ?
  - `find <dossier> -group <group>`
  - `find <dossier> -gid <user-gid>`
  - `find <dossier> -group <group> | xargs rmdir`
  - `find <dossier> -gid <user-gid> | xargs rmdir`
  - (`sudo find <dossier> -gid <user-gid> | xargs sudo rmdir`)
- Se connecter dans une autre console virtuelle (non graphique !) avec root

- appuyer sur : `ctrl+alt+f2`
- il y a des consoles de `f1` à `f6`

- Modifier les mots de passe des utilisateurs créés précédemment : `sudo passwd <user>`.
- Qui d'autre que root peut modifier un mot de passe ? L'utilisateur lui-même.
- Comment user2 peut-il devenir temporairement user1 ? Comment prendre complètement l'environnement de user1 ? `su user1`
- Se déconnecter des différentes consoles : `exit`.
- Désactiver les comptes user1 et user2 via 2 méthodes différentes et vérifier.

  Méthode 1:

  - `usermod --lock --expiredate 1970-01-02 <username>`

  Méthode 2:

  - `chage -E 0 username`

- Réactiver les comptes.

  Méthode 1:

  - `usermod --unlock --expiredate '' <username>`

  Méthode 2:

  - `chage -E -1 username`

- Créer un nouveau répertoire rep2 contenant un fichier vide fic2 dans /tmp et regarder les droits associés à ces nouveaux fichiers.
  - `mkdir /tmp/rep2`
  - `touch /tmp/rep2/fic2`
  - `namei -mo /tmp/rep2/fic2`

  `drwxr-xr-x` ($\Longrightarrow$ `d rwx r-x r-x`)
  - `d` : c'est un répertoire.
  - `rwx` : le propriétaire peut lire, écrire et exécuter.
  - `r-x` : le groupe peut lire et exécuter le fichier, pas le modifier.
  - `r-x` : le reste peut lire et exécuter le fichier, pas le modifier.

- Changer les droits du fichier fic2 afin que personne ne puisse les modifier.
  - À qui s'applique le changement:
    - u : user = utilisateur
    - g : group = groupe
    - o : others = autres
    - a : all = tous
  - La modification que l'on veut faire:
    - + : ajouter
    - - : supprimer
    - = : affectation
  - Le droit que l'on veut modifier:

    - r : read = lecture
    - w : write = écriture
    - x : execute = exécution

  `chmod a-w /etc/rep2/fic2`

- Essayer de supprimer le fichier /tmp/rep2/fic2. Est-ce possible ? Pourquoi ?

  - `rm /tmp/rep2/fic2`

  Ça marche. Je suppose que c'est parce que je suis le propriétaire.

- Changez le propriétaire de rep2 et fic2 à user2 et le groupe propriétaire à grp3. Essayer de supprimer le fichier avec user2

  - changer le propriétaire (change owner) : `sudo chown <user> <fichier>`
  - changer le groupe (change group) : `sudo chgrp <group> <fichier>`

- Modifier le umask. Est-ce d'application pour les comptes pré existants ? (si non que faire?). Et pour les futurs comptes crées ?

  - afficher l'umask : `umask -S`
  - changer l'umask : `umask u=rwx,,g=rx,o=rx`

- Modifier les droits par défaut à 0700 pour les futurs fichiers des utilisateurs, mais aussi ceux déjà présents : `umask 0700`.
- Créez un lien symbolique dans votre répertoire personnel pointant sur /var/log : `ln -s <fichier> <lien>`.
- Créez un lien symbolique dans votre répertoire personnel pointant sur un fichier texte : `ln -s <fichier> <lien>`.
- Créez un lien physique dans votre répertoire personnel pointant sur le dossier /tmp/votreNom : `ln <fichier> <lien>`.

4 Administration de base

- Installer le logiciel ethtool : `sudo apt install ethtool`.
- Ajouter les paquets contrib et non-free.

  - `sudo nano /etc/apt/sources.list`

    ```
    # ajouter : contrib non-free, à la fin des lignes avec des urls
    deb http://http.deb.debian.org/debian stable main contrib non-free
    ```

  - `sudo apt update`

- Récupérer les informations sur ce paquet : `sudo apt show ethtool`.
- Désinstaller ce paquet complètement : `sudo apt remove ethtool`.
- Mettre à jour votre distribution : `sudo apt full-upgrade`.

4.1 Manipulation - DPKG

- Téléchargez et Installez le paquet iptraf

- chercher le fichier sur le serveur ftp : `http://ftp.de.debian.org/debian/pool/`
- `wget http://ftp.de.debian.org/debian/pool/main/i/iptraf/iptraf_3.0.0-8.1_amd64.deb`
- `dpkg -i iptraf_3.0.0-8.1_amd64.deb` (ou : `ls | grep iptraf | xargs sudo dpkg -i`)

- Vérifier si un paquet est installé correctement ? (Astuce : utiliser grep) : `dpkg -l | grep iptraf`
- Vérifier que le paquet binutils est installé : `binutils`
- Lister les fichiers installés par celui-ci : `dpkg-deb -L <package>`
- Désinstallez (complètement) le paquet iptraf :
  - remove: `dpkg -r <package>`
  - purge: `dpkg -P <package>`

4.2 Manipulation - apt avancé

Configurer le système pour rester en version stable, mais installer samba en unstable.

1. Ajouter le dépôt sid dans */etc/apt/sources.list*.

```
deb http://deb.debian.org/debian/ sid main contrib non-free
deb-src http://deb.debian.org/debian/ sid main contrib non-free
```

2. Créer le fichier */etc/apt/preferences.d/samba*.

```
Package: samba
Pin: release a=unstable
Pin-Priority: 1001

Package: *
Pin: release a=unstable
Pin-Priority: 200
```

Remarque: priorité par défaut = 500. Samba va être installé en instable, pas le reste.

3. Vérification:
   - `sudo apt-cache policy samba`
   - `sudo apt-cache policy bind9`
   - `sudo apt update`
   - `sudo apt list --upgradable`

   Remarque: c'est normal de ne pas réussir à mettre samba à jour (`apt update && apt upgrade`). Les paquets ne restent instables que quelques jours à quelques semaines seulement. Pour tester entièrement cette config, il faut trouver un paquet qui est toujours en instable.

4.3 Manipulation - /proc

- Repérer l'emplacement (dans /proc) ou se trouve le fichier ip_forward.
  - `sudo find .  -name "ip_forward"`
  - `/proc/sys/net/ipv4/ip_forward`

- Ce fichier contient la valeur qui renseigne de l'état d'activation de la redirection. 0 pas de redirection, 1 redirection activée. Quelle est sa valeur ? valeur = 0
- Modifiez la.

  Le fichier appartient à root et il est le seul utilisateur à pouvoir le modifier $\Longrightarrow$ `sudo`.

- Éditer le fichier /etc/sysctl.conf : `sudo nano /etc/sysctl.conf`
- Ajouter ou modifier la ligne : net.ipv4.ip_forward=1
  - si 0, modifiez en 1
  - si 1, modifiez en 0
- Pour activer sans redémarrer : sysctl -p /etc/sysctl.conf ou /etc/init.d/procps.sh restart
- **PAS DANS LE COURS:** configurer la machine debian pour faire du routage.

  Routage NAT:

  - `sudo nano /etc/sysctl.conf`
  - décommenter cette ligne: `net.ipv4.ip_forward=1`
  - reboot, ou remplacer 0 par 1 dans: `/proc/sys/net/ipv4/ip_forward`
  - `sudo iptables --table nat --append POSTROUTING --jump MASQUERADE --out-interface enp0s3`
  - `sudo apt install iptables-persistent` (sélectionner *yes* à *save current ipv4 rules*)
  - `sudo systemctl restart networking`

`iptables`	outil d'administration pour filtrer les paquets et le NAT
`--table nat`	spécifie la table/module à modifier
`--append`	spécifie la règle à modifier
`POSTROUTING`	pour les paquets sur le point de sortir
`FORWARD`	pour les paquets passant par le routeur
`--jump`	spécifie l'objectif de la règle (sans `-j`, `-g`, la règle n'est pas appliquée)
`MASQUERADE`	mascarade l'ip du paquet par l'ip de l'interface
`ACCEPT`	accepte tous les paquets
`--out-interface`	spécifie l'interface de sortie
`--in-interface`	spécifie l'interface d'entrée

4.4 Manipulation - Gestion login avancé

- Dans le fichier login.defs, le paramètre UMASK est utilisé pour définir les droits par défaut sur le répertoire personnel (/home/<user>). Un umask représente une valeur inversée des droits octroyés au dossier. Présenté autrement, l'umask représente les droits non accordés au dossier. Exemple, umask par défaut : 022.

  Pour chercher *umask* dans le fichier:

  1. `sudo nano /etc/login.defs`
  2. taper: `/umask`
  3. navigation: "/" pour le prochain *umask*, "?" pour le *umask* précédent

  Autre possibilité de recherche:

  1. `nl /etc/login.defs | grep UMASK`

2. `sudo nano /etc/login.defs`
3. taper: `ctrl+_`, puis: 136

- Créez un utilisateurs. Rappel : useradd toto -d /home/toto -m.
- Vérifiez les droits sur le répertoire /home/toto : `namei -mo /home/toto`
  rwxr-xr-x : 755
- On a bien les droits en écriture (2) non autorisé pour le groupe et les autres utilisateurs. Modifier la valeur de l'umask pour :
  - propriétaire : tous les droits
  - groupe propriétaire : droits en lecture uniquement
  - autres : aucun droit

  Modification temporaire :
  - `umask u=rwx,g=r,o=`
  - `umask -S` ($\Longrightarrow$ u=rwx,g=r,o=)
  - `umask` ($\Longrightarrow$ 0037)

  Modification permanente:
  - `nl /etc/login.defs | grep UMASK`
  - `sudo nano /etc/login.defs`
  - `ctrl+_`
  - `136` $\Longrightarrow$ `enter`
- Créez un nouvel utilisateur et vérifiez que vous avez bien les bons droits : `sudo useradd <user>`.
  rwxr-----
- Modifiez ces valeurs pour que l'UID et le GID commence à 2000.

  Modifier l'uid et le gid pour 1 utilisateur/groupe :
  - `usermod -u <new_uid> <user>`
  - `groupmod -g <new_gid> <group>`

  Modifier l'uid et le gid minimum:
  - `sudo nano /etc/login.defs`
  - `ctrl+w` (w = where is...)
  - `uid_min` $\Longrightarrow$ `enter`
  - `ctrl+w` (w = where is...)
  - `gid_min` $\Longrightarrow$ `enter`
- Créez un nouvel utilisateur et vérifier l'UID et le GID : `sudo useradd <user>`.
- Connectez-vous à la console Debian avec le compte utilisateur créé pendant l'installation du système (UID:1000).
  - `cat /etc/passwd | grep 1000`
  - `su <user>`
- Réssayer la commande : apt update. Cette commande demande des droits superutilisateurs, ce qui n'est pas le cas de cet utilisateur.

- Connectez-vous maintenant en root : `su root`.
- Installez le paquet sudo (pensez à mettre la jour la liste des paquets avant l'installation) : `apt install sudo`.
- Créez l'utilisateur toto et ajoutez-le au fichier sudoers dans la section "User privilege specification".
  - `sudo useradd toto`
  - `sudo nano /etc/sudoers`
- Connectez-vous en toto et essayez à nouveau la commande apt update en ajoutant sudo devant.
  - `su toto`
  - `sudo apt update`
- Recommencez en créant un nouvel utilisateur mais en l'intégrant au groupe sudo et pas dans le fichier.
  - `usermod -aG sudo <user>`

4.5 Manipulation - Compilation

- Installer `build-essential` : `sudo apt install build-essential`.
- Récupérer le code source :
  `wget https://freefr.dl.sourceforge.net/project/htop/htop/1.0.2/htop-1.0.2.tar.gz`
- Décompressez le code source :
  - `tar zxvf <fichier>`
  - pour ne pas taper le nom du fichier : `ls | grep htop | xargs tar zxvf`
- Exécutez le programme de configuration : `./configure` (dans le dossier décompressé).
- Corrigez les erreurs éventuelles

  Personnellement, j'ai dû installer *libncurses*, avec la commande: `sudo apt install libncurses5-dev`.
- Compilez : `sudo make`
- Installez : `sudo make install`
- Démarrer le logiciel htop : `htop`
- Compilation et installation du programme hello
  1. récupérer le code source du programme hello à l'aide de la commande apt-get source : `sudo apt source hello`
  2. compiler en parallèle sur 2 ou 4 cœurs suivant votre matériel
     (a) `sudo ./configure`
     (b) `sudo make -j <nb_coeurs>` (ne pas en mettre trop)
  3. installez et lancez le programme
     - `sudo make install`
     - `./hello`

5 Services réseaux de base

5.1 Manipulation - NTP

- Purpose of the Network Time Protocol (NTP) = synchronize the local time of network hosts. NTP synchronize time using UTC. So it's important to configure your local time zone. NTP use the udp port 123.
- Installer ntp: `sudo apt install ntp`.
- Donner un serveur NTP pour la config de l'horloge:
  - `sudo nano /etc/ntp.conf`
  - ajouter la ligne: `server ntp.belnet.be`
  - commenter les lignes commençant par: `pool`
  - `sudo systemctl restart ntp`
  - vérification: `ntpq -p`
- Normalement, ntp ajuste le temps de manière progressive et lente. Avec: `sudo ntpd -g -n -q`, on peut le faire d'un seul coup:
  - -g = le premier ajustement peut être grand
  - -n = *do not fork*
  - -q = *set the time and quit* = ntpd ne devient pas un démon et quitte après la première synchro
- Après avoir synchronisé, faire: `ntpq -p`, le paramètre *delay* devrait être à 0.

5.2 Manipulation - FTP

- Installer le démon vsftpd: `sudo apt install vsftpd`.

  Remarque: après l'installation, on peut déjà se connecter en ftp avec: `ftp <ip>`.
- Désactiver l'accès anonyme et autoriser la connection des utilisateurs locaux.
  - `sudo nano /etc/vsftpd.conf`
  - changer:

    ```
    anonymous_enable=NO
    local_enable=YES
    ```

  - `sudo systemctl restart vsftpd`
- Activer l'écriture pour les utilisateurs non-anonymes.
  - `sudo nano /etc/vsftpd.conf`
  - changer: `write_enable=YES`
  - `sudo systemctl restart vsftpd`
- Chrooter un utilisateur. Est-ce une bonne idée ?
  - chrooter un utilisateur[a] = isoler un utilisateur dans une prison sans avoir accès au système
  - bonne idée pour empêcher un utilisateur de modifier des fichiers en dehors de son répertoire
  - serveur FTP cloisonné avec chroot:
    1. Créer un utilisateur pour ftp:
       * `sudo useradd <user> -d /srv/ftp -s /sbin/nologin`
       * `sudo passwd <user>`

2. Si *nologin* n'est pas dans `/etc/shells`: `echo '/sbin/nologin' ≫ /etc/shells`
3. Mettre ces paramètres dans le fichier `/etc/vsftpd.conf`:

   ```
   anonymous_enable=NO
   local_enable=YES
   write_enable=YES
   chroot_local_user=YES
   ```

   puis redémarrer le service: `sudo systemctl restart vsftpd`
4. Créer un dossier que l'utilisateur ftp peut modifier:
   * `sudo mkdir /srv/ftp/upload`
   * `sudo chmod a=rwx /srv/ftp/upload`
5. Connexion ftp: `ftp <ip>`

[a] https://www.vincentliefooghe.net/content/mise-place-dun-serveur-ftp-cloisonn%C3%A9

- Essayer de lire/écrire avec un compte local.
  - `sudo apt install ftp`
  - `ip a`
  - `ftp <ip>`
  - on peut naviguer avec: `ls`, `pwd`, `cd`
  - sélectionner le répertoire pour le téléchargement: `lcd <répertoire>`
  - télécharger un fichier: `get <file>`
  - télécharger plusieurs fichiers avec des wildcards: `mget *.txt`
  - uploader un fichier: `put <fichier>`
  - uploader plusieurs fichiers: `mput *.txt`
  - quitter la connection: `quit`, `exit`, `bye`

5.3 Manipulation - SSH

Tester avec un client linux et un client windows:

- Que se passe-t-il lorsqu'on modifie l'ip du serveur après avoir accepté sa clé rsa/dsa ? Pourquoi ?

  Le pc redemande d'accepter la clé du serveur car le certificat est lié à l'ip.

- `root` n'est pas autorisé à se connecter en ssh directement. Comment se connecter en `root` ? `su`
- À quoi servent les fichiers `nologin`, `hosts.allow`, `hosts.deny` ?

  Ils servent à:
  - interdire la connexion pour certains utilisateurs
  - lister les utilisateurs authorisés/interdits à se connecter en SSH (et d'autres services)

- N'autoriser que les sessions ssh encryptées avec AES.

  Ne fonctionne pas...
  - `man ciphers`
  - `openssl ciphers AES256`
  - modification peut-être: `openssl ciphers -ciphersuites TLS_AES_256_GCM_SHA384`
  - vérification peut-être: `openssl ciphers -s -tls1_3`

> Comment je vérifie:
>
> – `openssl ciphers -v -ciphersuites TLS_AES_256_GCM_SHA384 > f1`
> – `openssl ciphers -v -ciphersuites TLS_AES_256_GCM_SHA384 | grep AES > f2`
> – diff f1 f2
>
> Autre possibilité (côté client) – **fonctionne**:
>
> – `sudo nano /etc/ssh/ssh_config`
> – décommenter la ligne avec *Ciphers*, laisser uniquement les chiffrements avec *aes*
> – `sudo systemctl restart ssh`
>
> Autre possibilité (côté serveur) – **fonctionne**:
>
> – `sudo nano /etc/ssh/sshd_config`
> – ajouter le paramètre: `Ciphers aes256-ctr`
> – `sudo systemctl restart sshd`

- Activer la compression du côté client.

> Utiliser: `ssh` avec la commande: `-C` (en majuscule).

- Que fait la commande `scp` ?

> (Like cp, but trough network, encapsulated in ssl) `scp [<ip>:]<file> [<ip>:]<file>` = *secure copy* = copie des fichiers de manière sécurisée sur un réseau

- Essayer cette commande.

> `scp f1 10.0.2.15:/home/user/scp-f1`

- Peut-on utiliser scp avec windows ?

> – note du prof : en téléchargeant le logiciel Winscp
> – **sinon** : avec la commande `scp`, comme sur linux.

5.4 Manipulations avancées

NTP:

- Empêcher les hôtes de votre LAN de lire d'autres informations que le temps.

> – `man ntp.conf`
> – `sudo nano /etc/ntp.conf`
> – ajouter: `restrict <réseau> mask <masque> nomodify notrap nopeer noquery` (= ignore tous les paquets, sauf pour demander le temps)
> – `sudo systemctl restart ntp`

- Couper tout accès aux autres hôtes.

> Dans le fichier: `/etc/ntp.conf`, ajouter: `restrict default ignore` (= ignore tous les paquets).

- Comment empêcher les machines de changer la configuration excepté pour le localhost ?

Dans le fichier: `/etc/ntp.conf`, ajouter:

```
restrict default ignore # (= ignore tous les paquets)
restrict 127.0.0.1      # (= accès sans aucune restriction)
```

FTP:

- Quel est le meilleur choix de shell pour un utilisateur avec seulement un accès à FTP ? `/sbin/nologin`
- Créer un utilisateur avec seulement un accès à FTP, pas d'accès à la shell, avec l'option: `allow_writeable_chroot = no`
  - `sudo nano /etc/vsftpd.conf`
  - modifier l'option: `allow_writeable_chroot=NO`
  - `sudo systemctl restart vsftpd`
  - `sudo useradd <user> -d /srv/ftp -s /sbin/nologin`
- Comment donner/retirer l'accès à un utilisateur spécifique ? Essayer.
  - En l'ajoutant dans les fichiers: `hosts.allow`, ou: `hosts.deny`.
    (syntaxe = `<démon> : <utilisateur> [: deny/allow]`)
  - Dans un des 2 fichiers (mais de préférence: `hosts.deny`) ajouter:
    `vsftpd : <user> : deny`
- N'autoriser que les connexions FTP sécurisées.
  - `sudo nano /etc/vsftpd.conf`
  - modifier ce paramètre (ligne 151 pour moi): `ssl_enable=YES`
  - **rappel**: ssl = secure socket layer
  - `sudo systemctl restart vsftpd`

SSH:

- Utiliser l'authentication cliente forte avec clés RSA sans mot de passe. Comment autoriser uniquement l'authentication avec certificat (clé) ?

  Modification des paramètres sur le serveur SSH:

  - `sudo nano /etc/ssh/sshd_config`
  - modifier les paramètres:

    ```
    PasswordAuthentication no   # => uniquement avec clé
    PubkeyAuthentication yes
    HostKey /etc/ssh/ssh_host_rsa_key
    ```

  - `sudo systemctl restart sshd`

  Méthode sur le client[a]:

  1. Générer une paire de clé:
     - `ssh-keygen -t <algorithme> -b <nb_bits> -f <fichier>`
     - `ssh-keygen -t rsa -b 2048 -f id_rsa`
  2. Envoyer la clé publique au serveur ssh
     - `ssh-copy-id -i <fichier_clé_pub> <user>@<ip>`
     - `ssh-copy-id -i id_rsa.pub user@192.168.0.2`

3. Se connecter au serveur SSH:
   - `ssh -i <fichier_clé_privée> <user>@<ip>`
   - `ssh -i id_rsa user@192.168.0.2`

[a]https://serverpilot.io/docs/how-to-use-ssh-public-key-authentication/

- Protéger la clé avec un mot de passe. Vous devez être capable de le configurer manuellement.

  ATTENTION ! Utiliser un nom de clé différent.

  1. Générer une paire de clé:
     - `ssh-keygen -t <algorithme> -b <nb_bits> -f <fichier>`
     - `ssh-keygen -t rsa -b 2048 -f new_rsa`
  2. Envoyer la clé publique au serveur ssh
     - `scp -i <clé_privée_actuelle> <nouvelle_clé_publique> <ip>:<dossier_host_key>`
     - `scp -i id_rsa new_rsa.pub 192.168.0.2:/etc/ssh/ssh_host_rsa_key`
  3. Se connecter au serveur SSH:
     - `ssh -i <fichier_clé_privée> <user>@<ip>`
     - `ssh -i id_rsa user@192.168.0.2`
  4. Supprimer l'ancienne clé publique dans le dossier *host key* (par défaut: /etc/ssh/ssh_host_rsa_key/).

6 Gestion des systèmes de fichiers

6.1 Manipulation - Partitionnement

- Pour la manipulation, vous aurez d'une VM Debian sans interface graphique :
  - HDD : 20Go
  - sda1 : /boot – ext4 – 256 Mo
  - sda2 : Swap – 1Go
  - sda3 : Partition étendue
  - sda5 : / - ext4 - 15Go

  Remarques:
  - partition étendue[a] = partition qui contient des partitions (en théorie, on ne peut faire que 4 partition, si on en veut plus, il faut faire une partition étendue qui contiendra les autres partitions)
  - pour avoir un partitionnement comme ça, il faut mettre la racine à la fin de la ROM
  - si c'est impossible de créer une partition étendue à l'installation, on peut la créer avec la commande `sudo cfdisk`

  [a]https://doc.ubuntu-fr.org/partitions

- Lister les partitions actuelles via les commandes fdisk, cfdisk et sfdisk.
  - `sudo fdisk -l`
  - `sudo cfdisk`
  - `sudo sfdisk -l`

- Sous fdisk:
  - Lister les commandes intégrées à fdisk et les expliquer brièvement.

* m = menu = affiche l'aide
* l = list = liste les types de partitions
* F = free = liste les espaces libres
* p = print = affiche la liste de partitions
* v = verify = vérifie la liste de partitions
* i = information = affiche les informations sur une partition
* d = delete = supprime une partition
* n = new = ajoute une nouvelle partition
* t = type = change le type d'une partition
* w = write = enregistre les changements et quitte
* q = quit = quitte sans enregistrer

Remarques:

* toutes les partitions sont répertoriées dans le dossier `/dev`
* taper: `fdisk <partition>` (ex: `fdisk /dev/sda3`), pour lancer des commandes `fdisk`

- Afficher la table des partitions.

  commande = p

- Lister les différents types de partitions existants.

  commande = l

- Quels sont les codes pour les partitions Linux, Linux Swap, LVM, RAID, FAT32?

  * Linux = 93
  * Linux Swap = 82
  * LVM = 8e
  * RAID = fd
  * FAT32 = b

- Via fdisk créer 3 partitions de 300Mo.

  - `sudo fdisk <disque>` (ATTENTION ! mettre un disque pas une partition, `/dev/sda` pas `/dev/sda<nb>`)
  - commande = n
  - à l'option *last sector*, taper: `+300M`

- En cas de manque de place dans la partition étendue, reportez-vous aux manipulations avancées.

6.2 Rappel sur les systèmes de fichiers

- Pour lister le contenu d'une disquette DOS en LINUX, vous pouvez utiliser la séquence de commandes suivantes :
  - mdir a :
  - dir a :
  - mount -t vfat /dev/fd0/mnt/floppy ; ls /mnt/floppy ; unmount /dev/fd0
  - lsdos /mnt/floppy
- Pour créer un système de fichier ext3, ext4 et reiserfs, utilisez les commandes :
  - mkfs.ext2 /dev/...
  - mkfs.ext3 /dev/...
  - mksf.ext4 /dev/...

- mkfs.reiserfs /dev/...

ext3 est une évolution importante d'ext2 qui intègre un système de journalisation.

- Pour modifier la taille d'un système de fichier ext3, ext4 ou reiserfs, vous devez utiliser la commande : `resize2fs`.

6.3 Manipulation - Systèmes de fichiers

- Formater la 1ère partition en ext2 avec des blocs de 1ko avec 4ko **par** inode.
  - `sudo mkfs.ext2 <partition> -b <taille_bloc> -i <taille_par_inode>`
  - `sudo mkfs.ext2 /dev/sda5 -b 1024 -i 4096`
- Formater la 2ème en reiserfs.
  - `sudo apt install reiserfsprogs`
  - `sudo mkfs.reiserfs /dev/sda6`
- Formater la 3ème en ext4.
  - `sudo mkfs.ext4 /dev/sda7`

6.4 Manipulation - Montage des partitions

- Monter la partition en ext2 dans /mnt
  - `sudo mount -t ext2 /dev/sda5 /mnt`
- Vérifier que cela fonctionne bien.
  - `ls /mnt`
  - `ls /dev/sda5`

Remarques:

- montage = rendre une unité de stockage accessible
- point de montage = répertoire utilisé comme point d'accès à l'unité de stockage
- `umount` pour démonter, pas `unmount`

6.5 Manipulation - Quotas

- Quelques liens sur les quotas :
  - https://www.digitalocean.com/community/tutorials/how-to-set-filesystem-quotas-on-debian-10
  - https://web.mit.edu/rhel-doc/5/RHEL-5-manual/Deployment_Guide-en-US/ch-disk-quotas.html
  - https://www.unix.com/red-hat/129215-set-quota-folder.html
  - https://superuser.com/a/720392
- Créer un compte utilisateur toto.
  - `sudo adduser toto`
  - `sudo passwd toto`
- Mettre en place et activer la gestion des quotas pour /home.
  - **Possible uniquement si `/home` est sur une partition séparée (donc différente de /).**
  - Si ce n'est pas le cas, il faut réinstaller.

Activation des quotas sur /.

Installer le package:

- `sudo apt install quota`

Configurer le système de fichier pour les quotas:

- `sudo nano /etc/fstab`
- ajouter les options `usrquota`, `grpquota` comme ceci:
  `UUID=<uuid> / ext4 errors=remount-ro,usrquota,grpquota 0 1`
- remonter le système de fichier: `sudo mount -o remount /`

Activer la gestion des quotas pour `/`:

- créer les fichiers de quotas: `sudo quotacheck -ugm /`
- active les quotas: `sudo quotaon -v /`
  - bonne réponse = `<device> [/]: user quotas turned on`
  - mauvaise réponse = `... on <device> [/]: device or ressource busy`
  - solution: `quotaoff -v /`, puis: `quotaon -v /`

Activation des quotas sur `/home`.

Installer le package:

- `sudo apt install quota`

Configurer le système de fichier pour les quotas:

- `sudo nano /etc/fstab`
- ajouter les options `usrquota`, `grpquota` comme ceci:
  `UUID=<uuid> /home ext4 defaults,usrquota,grpquota 0 2`
- remonter le système de fichier: `sudo mount -o remount /home`

Activer la gestion des quotas pour `/home`:

- créer les fichiers de quotas: `sudo quotacheck -ugm /home`
- active les quotas: `sudo quotaon -v /home`
  - bonne réponse = `<device> [/home]: user quotas turned on`
  - mauvaise réponse = `... on <device> [/home]: device or ressource busy`
  - solution: `quotaoff -v /home`, puis: `quotaon -v /home`

- Choisir pour toto une limite douce de 15mo, hard de 25mo, période de grâce de 2 jours.
  - `sudo edquota -u toto` (modification = `<device> 0 15M 25M 0 0 0`)
  - `sudo edquota -t` (modification = `<device> 2days unset`)
  - `sudo setquota -u toto 15M 25M 0 0 /`
  - `sudo setquota -t 86400 unset /` (86400 = 2 jours en secondes)

  Remarques:
  - c'est impossible de créer une période de grâce par défaut pour un seul utilisateur uniquement
  - c'est possible de modifier la période de grâce pour un utilisateur si il a dépassé sa softlimit
  - même si il est mis que `unset` devrait être accepté, ça ne marche pas pour moi...
- Vérifier que cela fonctionne.

- `sudo quota -vs toto`
- `sudo repquota -s /`

- Faire de même pour les limites en nombre d'inodes.
  - `sudo edquota -u toto` (modification = `<device> 0 15M 25M 0 15M 25M`)
  - `sudo edquota -t` (modification = `<device> 2days 2days`)

- **Remarques**:
  - quota = quantité de mémoire disque allouée à un utilisateur
  - quota soft = limite qui peut être dépassée temporairement (warning)
  - quota hard = limite indépassable
  - Ce n'est pas possible de placer des quotas sur un répertoire particulier.

Commandes pour activer les quotas:

`quotaon`	active les quotas sur un système de fichiers
`edquota`	modifie les quotas hard, soft et les périodes de grâce
`quotacheck`	examine le système de fichier et résoud les incohérences
`quota`	affiche les quotas des utilisateurs

6.6 Rappel sur le RAID

- Rappel: RAID = redundant array of independent disks = techniques de virtualisation du stockage permettant de répartir des données sur plusieurs disques durs[1]
- Types de RAID:
  - raid 0 = aucune redondance
  - raid 1 = données copiées identiquement sur plusieurs DD
  - raid 3 = données copiées non-identiquement sur plusieurs DD (rapide + safe, >= 3 DD)
  - raid 4 = comme raid 3, mais taille des segments variable
  - raid 5 = comme raid 4, mais usure répartie sur chacun des disques
  - raid 6 = comme raid 5, mais double redondance des données de parité (>= 4 DD)
  - raid 01 = raid 0 + raid 1 par dessus
  - raid 10 = raid 1 + raid 0 par dessus
  - raid 50 = raid 5 + raid 0 par dessus
  - raid 0+5: plusieurs raid 0 regroupés en un seul raid 5 (minimum 3 raid 0)
- Mode dégradé:
  - quand un disque est retiré du raid, ses données sont accessibles grâce à la parité des autres
  - cela dégrade les performances, d'où l'appelation *mode dégradé*
  - pour prévenir ceci, on utilise un spare disk en attente pour remplacer le disque défaillant

6.7 Manipulation - RAID

- Créer 3 nouvelles partitions de 100Mo avec le type approprié pour réaliser un raid software.

[1] https://fr.wikipedia.org/wiki/RAID_(informatique)

  - `sudo cfdisk`
  - créer 3 partitions normales

- Créer un raid miroir avec un spare disk à l'aide de mdadm.

  raid miroir = raid 1 avec 2 disques et un spare disk

  - `sudo apt install mdadm`
  - `sudo mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/sda8 /dev/sda9 --spare-devices=1 /dev/sda10`
  - vérification (1): `mdadm --query /dev/md0`
  - vérification (2): `mdadm --detail /dev/md0`

- Monter le raid et le tester en y plaçant des répertoires, fichiers, etc.

  - `sudo mkfs.ext4 /dev/md0`
  - `sudo mount /dev/md0 /mnt`
  - `sudo mkdir /mnt/deleteme`
  - `sudo touch /mnt/deleteme/f1`

- Comment vérifier le statut du raid (2 solutions différentes).

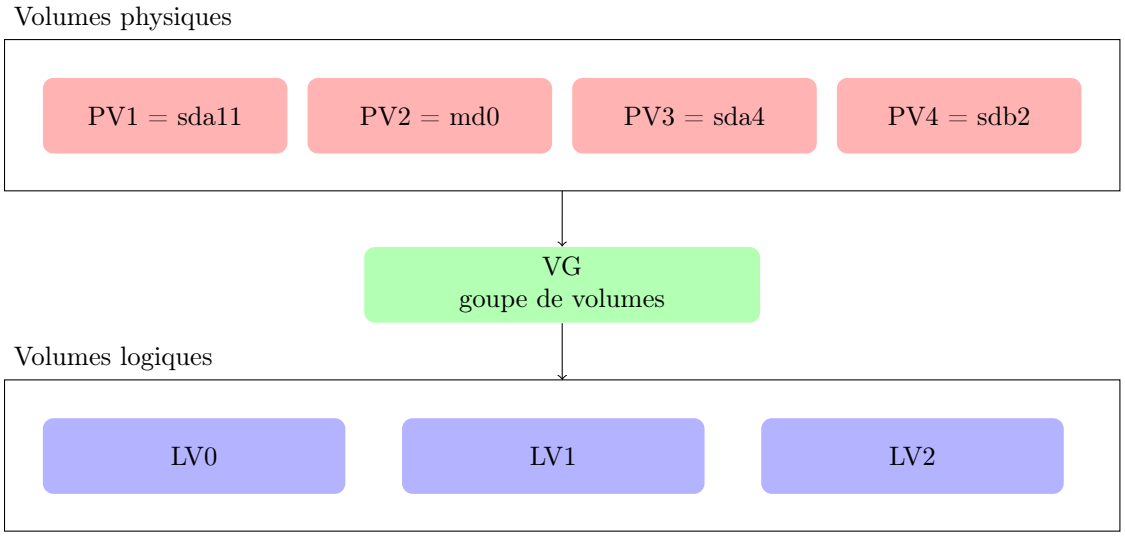
  - `cat /proc/mdstat`
  - `sudo mdadm --detail /dev/md0`

- Simuler la panne d'un des disques pour vérifier que le spare est bien utilisé.

  - `sudo mdadm /dev/md0 --fail /dev/sda8`
  - `sudo mdadm --detail /dev/md0`

6.8 Rappel théorique sur le LVM

- LVM = logical volume manager = gestionnaire de volume logique, permet par exemple de réduire la taille d'une partition pour libérer de l'espace pour en agrandir une autre.
- Avantages:
  - Il n'y a pas de limitation du style partition primaire, étendue, logique.
  - L'emplacement des données sur le disque n'est pas important.
  - On peut varier la taille des volumes très facilement sans risque pour les données contrairement à une modification de partitions standard.
- Inconvénient: Si un disque tombe en panne, tous les volumes qui utilisent ce disque sont perdus. Il faut donc utiliser le RAID par dessus.
- Glossaire
  - PV = physical volume
  - VG = volume group = groupe de PV
  - LV = logical volume = partie de VG
  - PE = physical extension = plus petite unité de VG
  - LE = logical extension = nombre de PE affecté à un volume logique (LV)
- Graphiquement:



- Commandes:
  - lvmdiskscan : recherche les PV présents sur la machine
  - vgscan : analyse tous les périphériques disques supportés dans le système afin de rechercher les volumes physiques LVM et les groupes de volumes. Cela permet de construire le fichier de cache LVM dans /etc/lvm/.cache, qui maintient une liste des périphériques LVM en cours.
  - pvscan : recherche les volumes physiques à travers tous les périphériques blocs LVM supportés dans le système
  - lvscan : scanne tous les VG connus ou tous les PV pris en charge dans le système pour déterminer le LV.
  - pvcreate : Initialisation d'une partition ou d'un disque pour l'utiliser en LVM
  - vgcreate : Création d'un groupe de volume
  - lvcreate : Création d'un volume logique
  - vgextend : Ajout d'une partition à un VG
  - vgreduce : Suppression d'une partition à un VG
  - lvextend : Permet d'augmenter la taille d'un LV
  - lvreduce : Permet de réduire la taille d'un LV
  - pvdisplay : Affiche les propriétés du volume physique
  - vgdisplay : Affiche les propriétés du groupe de volume
  - lvdisplay : Affiche les propriétés du volume logique
  - lvremove : Supprime un LV. Les données sont perdues.
  - vgremove : suppression d'un VG
  - pvremove : Supression d'un PV
  - pvmove : Déplacer des données présentent sur un PV vers un autre

6.9 Manipulation - LVM

- Créer 3 partitions LVM de 80, 160 et 240 Mo respectivement.
  - `sudo apt install lvm2`
  - `sudo cfdisk` (mettre le type en : *linux lvm*)
- Créer les volumes physiques et vérifier leurs propriétés.

- `sudo pvcreate -v /dev/sda{12,13,14}`
- `sudo pvdisplay` : Affiche les propriétés du volume physique

- Créer le groupe de volume VG0 et vérifier ses propriétés.
  - `sudo vgcreate -v <nom_vg> <pv>`
  - `sudo vgcreate -v vg0 /dev/sda{12,13,14}`
  - `sudo vgdisplay` : Affiche les propriétés du groupe de volume

- Créer un volume logique de 120Mo et vérifier ses propriétés.
  - `lvcreate -n <nom_LV> -L <taille> <vg>`
  - `lvcreate -n lv0 -L 120M vg0`
  - `sudo lvdisplay` : Affiche les propriétés du volume logique

- Le formater et le monter dans mnt.
  - `sudo mkfs.ext4 /dev/vg0/lv0`
  - `sudo mount /dev/vg0/lv0 /mnt`

- Vérifier l'espace disponible sur le système de fichier.
  - `df /mnt` (df = disk free)

- Agrandir le volume logique de 50Mo.
  - `lvextend -L +50M /dev/vg0/lv0`

- Est-ce que l'espace disponible sur le système de fichier à changé ?
  Pourquoi ? Si non, bien s'assurer que l'espace disponible est augmenté !
  - `df /mnt` (df = disk free)
  - espace disponible sur le système de fichier = 104750 (inchangé)
  - l'espace disponible sur le système de fichier n'a pas augmenté car on n'a pas utilisé la commande `resize2fs` pour l'agrandir

- Si l'on veut rétrécir un LV, dans quel ordre réaliser les étapes[2] ? Pourquoi ?
  1. démonter le système de fichiers
     - `sudo umount /mnt`
  2. vérifier le système de fichiers
     - `e2fsck -ff /dev/vg0/lv0` (corrige les erreurs avec le journal)
  3. réduire le système de fichiers
     - `resize2fs /dev/vg0/lv0 10M`
  4. réduire le volume logique (LV)
     - `lvreduce -L -10M /dev/vg0/lv0`
  5. re-vérifier le système de fichiers
     - `e2fsck -ff /dev/vg0/lv0` (corrige les erreurs avec le journal)
  6. remonter le système de fichiers

[2] https://www.tecmint.com/extend-and-reduce-lvms-in-linux/

- `sudo mount /dev/vg0/lv0 /mnt`

- Supprimer le volume logique.
  - `sudo umount /mnt`
  - `sudo lvremove vg0/lv0`

6.10 Manipulation - RAID + LVM

- Le principe est identique à l'utilisation de disque simple (non RAID). Il faut au préalable créer le RAID (/dev/md0). Ensuite on l'utilise simplement comme PV : pvcreate /dev/md0.
- Créer et tester un ensemble LVM par-dessus un ensemble raid. Vérifier.
  - `sudo pvcreate -v /dev/md0`
  - `sudo vgcreate -v vg1 /dev/md0`

6.11 Manipulation avancées

- Via fdisk créer 3 partitions de 300Mo
  - `sudo fdisk /dev/sda`
  - commande = n
- Convertir la partition en ext2 (sda6) en ext3 et vérifier à nouveau
  - `sudo tune2fs -j /dev/sda6` (-j = ajouter la journalisation)
  - `sudo mount /dev/sda6 /mnt`
  - `stat -f /mnt`
- Rendre permanent le montage de la partition dans /mnt/test
  - Démonter sda6 : umount /dev/sda6
  - Créer le répertoire test : mkdir /mnt/test
  - Monter sda6 dans test : mount /dev/sda6 /mnt/test
  - `cp /etc/fstab f1`
  - `sudo blkid /dev/sda6 >> f1` (pour obtenir l'uuid)
  - `nano f1`, modifier la dernière ligne: `UUID=<uuid> /mnt/test ext3 defaults 0 0`
  - `sudo cp f1 /etc/fstab`
- Lister les montages actifs
  - `mount`
- Vérifier l'espace libre/occupé sur les partitions montées
  - `df`
- Une fois la partition montée, vérifier si les fichiers présents auparavant sont effacés ou pas.
  - `ls /mnt/test`

  Ils ne sont pas effacés.
- Déplacer le répertoire home sur la partition ext4. Attention, travaillez de manière sécurisée ! Il ne faut pas risquer de perdre des données[3] !

[3] https://www.tecmint.com/move-home-directory-to-new-partition-disk-in-linux/

1. Créer la partition, le système de fichiers et la monter:
   - `sudo cfdisk`, créer une partition de 2 Go
   - `sudo mkfs.ext4 /dev/sda7`
   - `sudo mount /dev/sda7 /mnt`
2. Copier les données de `/home` sur l'autre partition, puis les supprimer:
   - `sudo cp -a /home/* /mnt/` (`-a` = copie récursive aussi proche que possible de l'original[a])
   - `sudo diff -r /home /mnt` (en cas de problème, supprimer puis recopier)
   - `sudo rm -rf /home/*`
3. Démonter la partition puis la remonter sur `/home`:
   - `sudo umount /mnt`
   - `sudo mount /dev/sda7 /home`
4. Modifier l'UUID dans `/etc/fstab`:
   - `cp /etc/fstab f1`
   - `sudo blkid /dev/sda6 >> f1` (pour obtenir l'uuid)
   - `nano f1`
     (a) commenter l'ancienne ligne avec `/home`
     (b) modifier la dernière ligne: `UUID=<uuid> /home ext4 defaults 0 2`
   - `sudo cp f1 /etc/fstab`
5. Vérification finale:
   - `sudo reboot`
   - `df -hl`

[a]Par exemple: `-a` ne change pas la date de modification, contrairement à `-r`. Plus d'informations sur: https://unix.stackexchange.com/a/44981

7 Network addressing services

7.1 Manipulation de base – DNS

- Install bind.
  - `sudo apt install bind9 bind9utils bind9-doc`
- DNS cache server.

  Mettre la machine en NAT, puis changer la configuration dans `/etc/network/interfaces`:

```
allow-hotplug enp0s3
iface enp0s3 inet static
    address 10.0.2.15
    netmask 255.255.255.0
    gateway 10.0.2.2
    dns-nameservers 127.0.0.1
```

  Commenter toutes les lignes dans `/etc/resolv.conf`.

  `bind` est configuré en serveur dns cache par défaut.
  - `nl /etc/bind/named.conf.options`
  - paramètres:

```
options {
    directory "/var/cache/bind";
    dnssec-validation auto;
    listen-on-v6 { any; };
};
```

En cas de modification du fichier (ex: pour ajouter un forwarder):

- `sudo named-checkconf -z` (vérifie la syntaxe des options)
- `sudo systemctl restart bind9`
- `sudo tail -f /var/log/syslog`

Ajouter: `nameserver <ip>`, dans: `/etc/network/interfaces`, la machine client.

- Test bind as a cache server with nslookup.
  - `sudo apt install dnsutils`
  - `nslookup fb.com`
  - `nslookup fb.com` (ça devrait aller plus vite la 2ème fois)
  - `nl /var/cache/bind/named_dump.db` (contient le cache du serveur dns)
  - `dig fb.com` (query time = 190 msec)
  - `dig fb.com` (query time = 0 msec)
- Check and test the server configuration with:
  - `named-checkconf -z`
    Vérifier la syntaxe des fichiers de configuration:
    - `sudo named-checkconf -z`
  - `named-checkzone`
    Vérifie la validité du fichier de zones:
    - `sudo named-checkzone [<fichier_conf> | <domaine.com> <fichier_zone>]`
    - `sudo named-checkzone /etc/bind/named.conf`
  - `nslookup`
  - `dig`
    - `dig fb.com`
  - `host`
    - `host fb.com`
- Create a primary dns zone "myname.linux".
  - Don't forget the reverse zone !
  - Tip : start from the db.empty file !

Fichier `/etc/bind/named.conf.local`:

```
zone "myname.linux" {
    type master;
    file "/etc/bind/db.myname.linux";
};

// 10.0.2.15
```

```
zone "2.0.10.in-addr.arpa" {
    type master;
    file "/etc/bind/db.myname.linux.inv";
};
```

Fichier `/etc/bind/db.myname.linux`:

```
$TTL    86400
@       IN      SOA     myname.linux.    root.myname.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      myname.linux.
        IN      A       10.0.2.15
dns     IN      CNAME   myname.linux.
www     IN      CNAME   myname.linux.
```

Fichier `/etc/bind/db.myname.linux.inv`:

```
$TTL    86400
@       IN      SOA     myname.linux.    root.myname.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      myname.linux.
15      IN      PTR     myname.linux
15      IN      PTR     dns.myname.linux
15      IN      PTR     www.myname.linux
```

Vérifications:

- `sudo named-checkconf -z` (vérifie la syntaxe des options)
- `sudo named-checkzone myname.linux /etc/bind/db.myname.linux`
- `sudo named-checkzone 10.in-addr.arpa /etc/bind/db.myname.linux.inv`
- `sudo systemctl restart bind9`
- `sudo tail /var/log/syslog`
- `nslookup www.myname.linux`
- `nslookup dns.myname.linux`
- `nslookup myname.linux`
- `nslookup 10.0.2.15`

- Add an MX, an A and a CNAME record.
  - Don't forget good practices: ONE A record for each IP! Others → CNAME.
  - MX record is for a zone, not for an host!
  - Don't forget to check and test the server configuration!

Fichier `/etc/bind/db.myname.linux`:

```
$TTL    86400
@       IN      SOA     myname.linux.    root.myname.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      myname.linux.
@       IN      MX      10 myname.linux.
        IN      A       10.0.2.15
dns     IN      CNAME   myname.linux.
www     IN      CNAME   myname.linux.
mail    IN      CNAME   myname.linux.
```

Ça revient à ajouter la ligne avec le MX, et la ligne avec mail en CNAME.

- Create the corresponding reverse zones and PTR records.

Fichier `/etc/bind/db.myname.linux.inv`:

```
$TTL    86400
@       IN      SOA     myname.linux.    root.myname.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      myname.linux.
15      IN      PTR     myname.linux
15      IN      PTR     dns.myname.linux
15      IN      PTR     www.myname.linux
15      IN      PTR     mail.myname.linux
```

- Don't forget to check and test.

Vérifier l'enregistrement MX:

  - `nslookup mail.myname.linux`
  - `nslookup`
  - `set q=MX`
  - `myname.linux`

7.2 Autre configuration DNS

- Fichier `/etc/bind/db.myname.linux`:

```
$TTL    86400
@       IN      SOA     srv.myname.linux.    root.myname.linux. (
                        1           ; Serial
                        604800      ; Refresh
```

```
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      srv.myname.linux.
@       IN      MX      10 srv
        IN      A       10.0.2.15
dns     IN      CNAME   srv
www     IN      CNAME   srv
mail    IN      CNAME   srv
```

- Fichier `/etc/bind/db.myname.linux.inv`:

```
$TTL    86400
@       IN      SOA     srv.myname.linux.    root.myname.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      srv.myname.linux.
15      IN      PTR     srv
15      IN      PTR     srv
15      IN      PTR     srv
15      IN      PTR     srv
```

7.3 Manipulation de base – DHCP

- Install the dhcp server.
  - `sudo apt install isc-dhcp-server`

  Remarques:
  - perso, quand je reboot, le serveur ne fonctionne plus mais ça ne marque rien dans le journal
  - il faut redémarrer le service avec: `systemctl restart isc-dhcp-server`
  - mais avant, il faut que l'interface sur laquelle le service est configuré soit up (`ifup <interface>`)
- Study a few configuration file's included examples.
  - `ls /etc/dhcp`
  - `cat /etc/dhcp/dhcpd.conf`
- By default, the service is not working. Check in the logs why!
  - `journalctl -xe`
  - `tail /var/log/syslog`
  - `tail f /var/log/syslog` (pour afficher en continu)
- Activate the service. Be careful!
  - Do it on the right interface.
  - Configure it to work on the NIC that is not plugged to the iesn network. Why? $\Longrightarrow$ only one DHCP/LAN.

Ajouter l'interface:

- `sudo nano /etc/default/isc-dhcp-server`
- ajouter `enp0s8` aux interfaces en ipv4

Remarque: les interfaces dans virtualbox sont:

- `enp0s3` = en NAT
- `enp0s8` = en réseau interne

Configurer le dhcpd:

- `sudo nano /etc/dhcp/dhcpd.conf`
- modifier:

```
authoritative;

subnet 192.168.0.0 netmask 255.255.255.0 {
    option routers 192.168.0.1;
    range 192.168.0.2 192.168.0.254;
    option broadcast-address 192.168.0.255;
}
```

Tester la config:

- `sudo systemctl restart isc-dhcp-server`
- `sudo tail -f /var/log/syslog`
- lancer la 2ème machine en dhcp dans le réseau interne

Remarque: L'interface enp0s8 doit être configurée en statique car elle ne peut pas recevoir de configuration sinon.

- Check in the file the default lease time.

Ouvrir le fichier:

- `sudo nano /etc/dhcp/dhcpd.conf`

Les configs de lease time sont:

```
default-lease-time 600;
max-lease-time 7200;
```

- Don't forget to configure the right values (choose DNS values from the previous labs) for:
  - the netmask,
  - the IP address range,
  - the name servers,
  - the gateway,
  - the search domain,
  - the ntp server,
  - etc.

```
# /etc/dhcp/dhcpd.conf
subnet 192.168.0.0 netmask 255.255.255.0 {      # netmask
    range 192.168.0.2 192.168.0.254;            # ip range
    option domain-name-servers 192.168.0.1;     # dns
    option routers 192.168.0.1;                 # gateway
    option domain-search "myname.linux";        # search domain
    option ntp-servers 192.168.0.1;             # ntp
    option broadcast-address 192.168.0.255; # pas obligatoire
}
```

Remarque: il faut que le ntp et le dns soient installés.

- dans `/etc/ntp.conf`, ajouter: `restrict 192.168.0.0 mask 255.255.255.0`

- Test it using one of the lab's switches (or a cross cable) and with your neighbour's computer as a client, or with a Live CD VM.

  Vérifications sur la machine cliente:

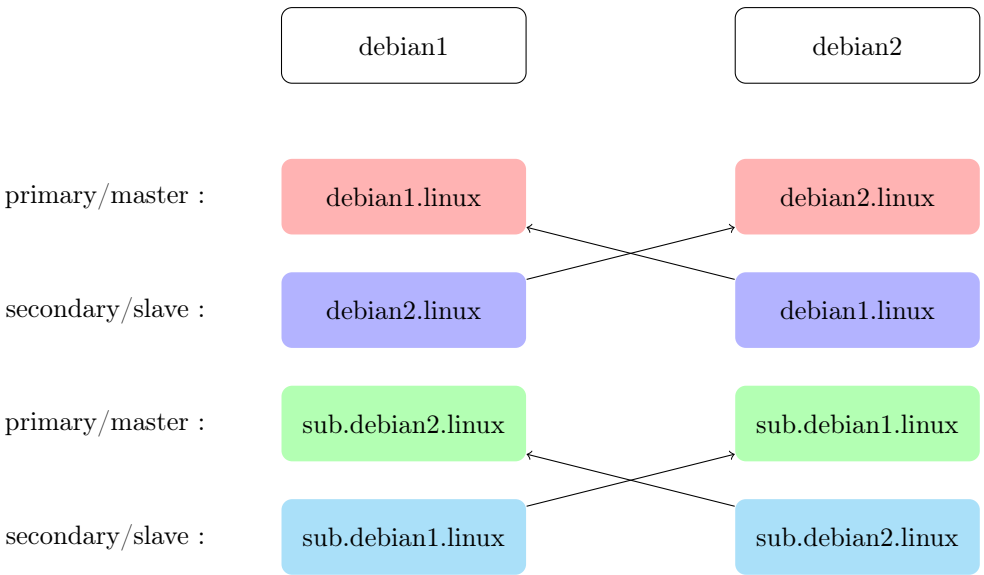
  - `ip a` (ip + netmask)
  - `ip route` (gateway)
  - `cat /etc/resolv.conf` (dns + search domain)
  - `ntptrace` (ntp)

- Configure the server so that it always gives the client the same IP (a fixed lease).

```
host client_fixed_ip {
    hardware ethernet 08:00:07:26:c0:a5;
    fixed-address 192.168.0.2;
}
```

7.4 Manipulation avancée

- Create a secondary dns zone with your neighbour as the primary zone, and reciprocally.



debian1:

- Fichier `/etc/network/interfaces`:

```
allow-hotplug enp0s8
iface enp0s3 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    gateway 192.168.0.1
    dns-nameservers 192.168.0.1 192.168.0.2
```

- Fichier `/etc/bind/named.conf.local`:

```
zone "debian1.linux" {
    type master;
    file "/etc/bind/db.debian1.linux";
};
// 192.168.0.1
zone "1.0.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.debian1.linux.inv";
};

zone "debian2.linux" {
    type slave;
    masters { 192.168.0.2; };
};
// 192.168.0.2
zone "2.0.168.192.in-addr.arpa" {
    type slave;
    masters { 192.168.0.2; };
};
```

- Fichier `/etc/bind/db.debian1.linux`:

```
$TTL    86400
@       IN      SOA     debian1.linux.    root.debian1.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      debian1.linux.
        IN      A       192.168.0.1
www     IN      CNAME   debian1.linux.
```

- Fichier `/etc/bind/db.debian1.linux.inv`:

```
$TTL    86400
@       IN      SOA     debian1.linux.    root.debian1.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
```

```
                            86400       ; Negative Cache TL
                            )

        @       IN      NS      debian1.linux.
                IN      PTR     debian1.linux
                IN      PTR     www.debian1.linux
```

debian2:

- Fichier `/etc/network/interfaces`:

```
allow-hotplug enp0s3
iface enp0s3 inet dhcp
```

- Fichier `/etc/bind/named.conf.local`:

```
zone "debian2.linux" {
    type master;
    file "/etc/bind/db.debian1.linux";
};
// 192.168.0.2
zone "2.0.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.debian2.linux.inv";
};

zone "debian1.linux" {
    type slave;
    masters { 192.168.0.1; };
};
// 192.168.0.1
zone "1.0.168.192.in-addr.arpa" {
    type slave;
    masters { 192.168.0.1; };
};
```

- Fichier `/etc/bind/db.debian2.linux`:

```
$TTL    86400
@       IN      SOA     debian2.linux.    root.debian2.linux. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      debian2.linux.
        IN      A       192.168.0.2
www     IN      CNAME   debian2.linux.
```

- Fichier `/etc/bind/db.debian2.linux.inv`:

```
$TTL    86400
@       IN      SOA     debian2.linux.    root.debian2.linux. (
```

```
                                1           ; Serial
                                604800      ; Refresh
                                86400       ; Retry
                                2419200     ; Expire
                                86400       ; Negative Cache TL
                                )

        @       IN      NS      debian2.linux.
                IN      PTR     debian2.linux
                IN      PTR     www.debian2.linux
```

Attention ! Pas besoin de donner des fichiers dans les zones esclaves.

- Don't forget to check and test.
  - `sudo systemctl restart bind9`
  - `nslookup www.debian1.linux`
  - `nslookup www.debian2.linux`
- Add the secondary DNS to your DHCP's configuration file.

  Fichier `/etc/dhcp/dhcpd.conf`:

  ```
  subnet 192.168.0.0 netmask 255.255.255.0 {
      # modifier cette option:
      option domain-name-server 192.168.0.1, 192.168.0.2;
  }
  ```
- Only your secondary zone is allowed to ask for updates. Only your primary zone is allowed to send updates to your secondary one.
  - allow-notify = slave, by default only master can update
  - allow-transfer = master, by default anybody can take update

  Les modifications doivent être réalisées dans les points suivants.
- Queries are only authorized for your domains.

  Fichier `/etc/bind/named.conf.options`:

  ```
  options {
      // ajouter l'option suivante:
      allow-notify { 192.168.0.0; };
  }
  ```
- Hosts from your lan can queries for external domains too.

  Fichier `/etc/bind/named.conf.options`:

  ```
  options {
      // ajouter l'option suivante:
      allow-transfer { 192.168.0.0; };
  }
  ```

- Create a sub domain called "sub.debian1.linux.". This domain is delegated to a new zone managed by your neighbor's DNS service.

  Debian1:

  – Fichier `/etc/bind/named.conf.local`:

  ```
  zone "sub.debian1.linux" {
      type slave;
      masters { 192.168.0.2; };
  };
  ```

  Debian2:

  – Fichier `/etc/bind/named.conf.local`:

  ```
  zone "sub.debian1.linux" {
      type master;
      file "/etc/bind/db.sub.debian1.linux";
  };
  ```

  – Fichier `/etc/bind/db.sub.debian1.linux`:

  ```
  $TTL    86400
  @       IN      SOA     sub.debian1.linux.    root.sub.debian1.linux. (
                          1           ; Serial
                          604800      ; Refresh
                          86400       ; Retry
                          2419200     ; Expire
                          86400       ; Negative Cache TL
                          )

  @       IN      NS      sub.debian1.linux.
  ; j'ai laissé l'ip de debian1, mais on s'en fiche
          IN      A       192.168.0.1
  www     IN      CNAME   sub.debian1.linux.
  ```

- Create a zone "sub.debian2.linux". This zone manages your neighbor sub domain.

  Debian1:

  – Fichier `/etc/bind/named.conf.local`:

  ```
  zone "sub.debian2.linux" {
      type master;
      file "/etc/bind/db.sub.debian2.linux";
  };
  ```

  – Fichier `/etc/bind/db.sub.debian2.linux`:

  ```
  $TTL    86400
  @       IN      SOA     sub.debian2.linux.    root.sub.debian2.linux. (
                          1           ; Serial
                          604800      ; Refresh
                          86400       ; Retry
                          2419200     ; Expire
  ```

```
                            86400       ; Negative Cache TL
                            )

        @       IN      NS      sub.debian2.linux.
        ; j'ai laissé l'ip de debian2, mais on s'en fiche
                IN      A       192.168.0.2
        www     IN      CNAME   sub.debian2.linux.
```

Debian2:

- Fichier `/etc/bind/named.conf.local`:

```
zone "sub.debian2.linux" {
    type slave;
    masters { 192.168.0.1; };
};
```

LE RESTE DE CETTE MANIP EST OPTIONNEL

- Notes:
  - It's possible to dynamically update the DNS database with the DHCP server (Like in the AD).
  - To do that, it's necessary to modify bind et dhcpd configuration.
- Modify the dns et dhcp server configuration file to activate DynDNS.

  http://www.linux-france.org/prj/edu/archinet/systeme/ch37.html

- Secure the exchange of information between the servers
  - manual key or dnssec-keygen -a HMAC-MD5 -b 128 -r /dev/urandom -n USER DDNS_UPDATE
  - ```
    key DDNS_UPDATE {
        algorithm HMAC-MD5.SIG-ALG.REG.INT;
        secret "<key>";
    };
    ```

  - ```
    include "/etc/bind/ddns.key";

    zone "example.org" {
        type master;
        notify no;
        file "/var/cache/bind/db.example.org";
        allow-update { key DDNS_UPDATE; };
    };

    zone "2.168.192.in-addr.arpa" {
        type master;
        notify no;
        file "/var/cache/bind/db.192.168.2";
        allow-update { key DDNS_UPDATE; };
    };
    ```

  - !!!!! : file permissions for bind to read/updates zone files!

- DHCP:ddns-updates on;

```
ddns-update-style      interim;
ignore                 client-updates;
update-static-leases   on;

include "/etc/dhcp/ddns.key";
zone example.org. {
    primary 127.0.0.1;
    key DDNS_UPDATE;
}

zone 2.168.192.in-addr.arpa. {
    primary 127.0.0.1;
    key DDNS_UPDATE;
}
```

- Your neighbor will be the DHCP and the DNS client.
- Configure the client to use the hostname "my_firstname". sudo vi /etc/dhcp3/dhclient.conf.
- Set hostname as follows: send host-name "smlfkjqmflkjmlkjfqmlkfsdj";

8 Démarrage, initialisation, gestion des processus

8.1 Le chargeur de démarrage

À l'écran du GRUB, passer en mode édition et modifier la ligne linux

1. au démarrage, à l'écran du GRUB, taper sur la touche *e*
2. modifier la 3ème ligne en partant par la fin dans l'éditeur:
   - modifier *ro*, en *rw*
   - ajouter *init=/bin/bash* à la fin de la ligne
3. lancer la machine, en appuyant sur *ctrl+x* ou *f10*
4. modifier le mot de passe (`passwd`, attention ! clavier en qwerty)
5. taper : `sync`, taper : `exec /sbin/init`
6. relancer la machine : `reboot`

Tester le nouveau mot de passe.

8.2 Systemd

- Installez Bind9.
  - `sudo apt install bind9`
- Vérifiez le statut du service bind9.
  - `sudo systemctl status bind9`
- Vérifiez que le service bind9 est une dépendance de multi-user.target.

> – `sudo systemctl list-dependencies --after multi-user.target`

- Rebootez et vérifiez à nouveau le statut de bind9.

> – `reboot`
> – `sudo systemctl list-dependencies --after multi-user.target`

- Supprimez la dépendance de bind9 à multi-user.target.

> – `sudo systemctl disable bind9`

- Rebootez et vérifiez à nouveau le statut de bind9.

> – `reboot`
> – `sudo systemctl status bind9`

- Pouvez vous néanmoins démarrer le service ?

> Si on l'a masqué, non. Sinon je ne sais pas.

- Effectuez les modifications nécessaire pour empêcher le démarrage de bind9.

> – `sudo systemctl mask bind9`

8.3 Gestion des processus

- Lister les processus dont vous êtes le propriétaire qui sont attachés à votre console.

> – `ps --user <user>`

- Lister l'entièreté de vos processus.

> – `ps -e`

- Lister tous les processus en cours d'exécution.

> – `ps -aux`

- Lister les processus sous forme d'un arbre.

> – `ps -e --forest`

- Quelle est la particularité du processus 1?

> C'est toujours `systemd`.

- Si vous lancez 10 fois un programme, est-ce que le PID restera le même?

> Non. Ça change à chaque fois.

- Listez en continu les processus, en les triant par :
  - %CPU
  - %mémoire
  - Temps CPU
  - top

  - `top -o %CPU`
  - `top -o %MEM`
  - `top -o TIME+`
  - `ps -aux --sort=cpu,mem,time`

- Trouver son numéro de processus via les commandes:
  - ps
  - pgrep

  - `ps -aux | egrep bind`

- Lancer Firefox avec la plus faible priorité possible. Vérifier la priorité reçue.

  - `nice -20 firefox &`

- Modifier la priorité de Firefox à la plus petite priorité possible. Vérifier la priorité reçue.

  - `ps -l`
  - `renice -20 <PID_firefox>`

- Consulter sa page de manuel si besoin et le lancer en avant plan, à des fin de "debug" par exemple:
  - man named
  - -d (debug) -g (foreground + messages)

- Le tester avec nslookup.

  - `named -d`
  - `named -g`

- Mettre bind en pause.

  - `jobs -l`
  - `kill -STOP %<job_id>`

- Le relancer en avant plan.

  - `fg <jobid>` (fg = foreground)

- Tuer bind.

  - `kill bind9`

- Quelle est la "procédure" standard pour relancer bind ? Analyser le script qui gère bind.

  - `kill -START %<job_id>`

- À l'aide de cron, faites redémarrer votre machine tout les jours à la même heure.

  - `at <heures>:<minutes>`
  - `commande`
  - *ctrl+d*

  Remarque: vérifier l'heure avec la commande : `date` (au cas où la machine n'est pas à l'heure belge).

- Créer un script qui redémarre votre machine. Lancez-le dans quelques instants à l'aide de at.
  - `at <heures>:<minutes> -f <fichier>`
- Vérifiez que votre script est en attente d'exécution.
  - `atq`

8.4 Gestion des modules kernel

- Lister les modules kernel utilisés.
  - `lsmod`
- Vérifier que le module vfat est désactivé. Sinon le désactiver.
  - `modinfo vfat`
- Monter un système de fichier vfat (partition, clef usb, etc.).
  - `mount -t vfat <partition> <dossier>`
- Lister les modules. Que constatez-vous? Qu'en conclure?
  - `lsmod`
- Récupérer les informations du modules reiserfs.
  - `modinfo reiserfs`
- Activer le module reiserfs et vérifier.
  - `insmod /lib/modules/4.19.0-12-amd64/kernel/fs/reiserfs/reiserfs.ko`
  - `lsmod | grep reiserfs`

8.5 PAM

- Grâce à Pam, permettre la création automatique du répertoire personnel des utilisateurs qui n'en auraient pas. Tester.

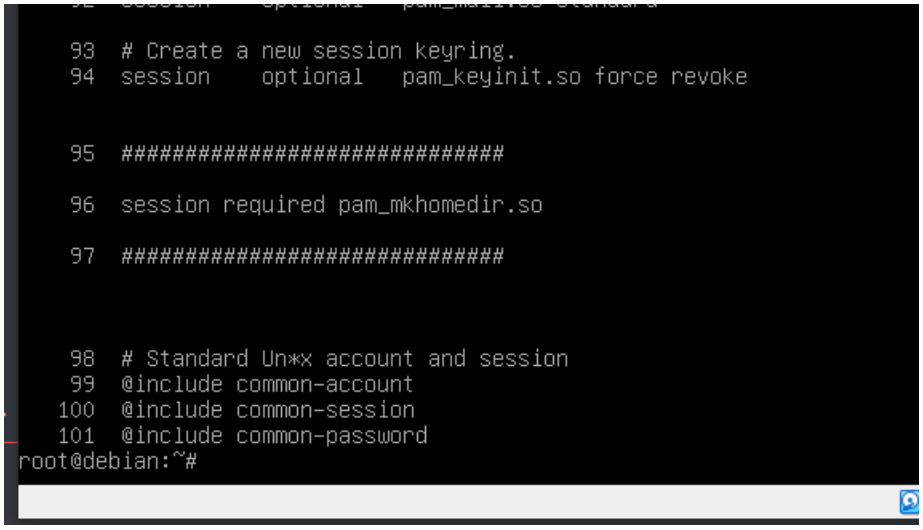
  Ajouter un utilisateur qui n'a pas de répertoire personnel:
  - `useradd test1`
  - `passwd test1`

  Dans le fichier `/etc/pam.d/login`, ajouter:

  ```
  session required pam_mkhomedir.so
  ```

  Relancer la machine et se connecter avec l'utilisateur qui n'a pas de répertoire personnel.

  Remarque: il faut mettre les lignes de session vers la fin, sinon la machine risque de refuser toutes les sessions (impossible de se connecter).



- Modifier le nombre d'essais de saisie d'un nouveau mot de passe à 2. Créer une stratégie de mot de passes qui force une longueur minimale, etc.

  Dans le fichier `/etc/pam.d/login`, ajouter:

  ```
  session required pam_tally.so deny=2
  session required pam_unix.so  minlen=6
  ```

- Créer un utilisateur "pam". Pour ce dernier, interdire la connexion durant une courte tranche horaire. Vérifier.

  - `adduser pam`
  - `nano /etc/pam.d/login`
  - modifier:

    ```
    # interdiction de 10 sec
    session required pam_tally.so deny=2 lock_time=10
    ```

  - se connecter sur le compte avec un mauvais mot de passe à 2 reprises

9 Serveur WEB

9.1 Concepts

- Installer un navigateur en ligne de commande, pratique pour tester nos configurations quand on a pas d'interface graphique:
  - `sudo apt install w3m w3m-img`
- Visiter *www.google.com*:
  - `w3m www.google.com`
- Installer le service http apache:
  - `sudo apt install apache2`
- Tester avec le navigateur w3m en ligne de commande:
  - `w3m 127.0.0.1`

- Tester avec un navigateur graphique.
- Les paramètres généraux d'apache sont stockés dans le fichier */etc/apache2/apache2.conf*.
- Les ports d'écoute sont configurés dans le fichier */etc/apache2/ports.conf*.
  - par défaut, le port 80 est utilisé
  - si il y a plusieurs ip, on peut associer port et ip : `Listen 192.168.0.1:80`
- Chaque site possède son fichier de configuration que l'on peut "activer". Ils se trouvent dans le répertoire : */etc/apache2/sites-available*.
- Ajouter la ligne suivante dans le fichier de configuration principal:

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

  Pour activer un site, il suffit de créer un lien dans le répertoire */etc/apache2/sites-enabled* vers le fichier de configuration dans le répertoire */etc/apache2/sites-available*.
- Exemple configuration */etc/apache2/sites-availbale/apache.local.conf* :

  ```
  <VirtualHost 192.168.0.1:80>
      ServerName www.apache.local
      ServerAdmin webmaster@apache.local
      DocumentRoot /var/www/html/apache.local/
      ErrorLog ${APACHE_LOG_DIR}/apache.local.error.log
      CustomLog ${APACHE_LOG_DIR}/apache.local.access.log combined
  </VirtualHost>
  ```

- Le protocole HTTPS est basé sur deux technologies :
  - le cryptage asymétrique (clé publique / clé privée)
  - la certification numérique X509
- Générer une clé de chiffrement et un certificat:
  - `openssl req -x509 -nodes -newkey rsa:1024 -keyout key.private -out certificat.pem`
    - req : demande un certificat
    - -x509 : on demande un certificat autosigné et pas une demande de signature
    - -nodes : La clé de serveur n'est pas protégée par mot de passe
    - -newkey rsa:taille : on crée une nouvelle clés asymétrique RSA 1024 (bits)
    - -keyout : le fichier de la clé privée
    - -out : le fichier du certificat

  Attention ! Après avoir tapé la commande, on demande quelques informations, à *common name*, donner le nom du dns qui donne l'url d'accès au serveur. Sinon le navigateur affichera une alerte de sécurité.
- Charger le module ssl dans apache:
  - `apachectl -M | grep ssl`, vérifie que le module *ssl_ module* est déjà chargé
  - `a2enmod ssl`, pour charger le module sinon
- Configuration des clés:
  - copier les clés dans le dossier */var/www/keys*
  - dans *ssl.conf*, ajouter:

    ```
    SSLCertificateFile /var/www/keys/certificat.pem
    SSLCertificateKeyFile /var/www/keys/key.private
    ```

- Ouverture du port https dans le fichier ports.conf
  - c'est actif par défaut
  - sinon, ajouter:

```
<IfModule ssl_module>
    Listen 443
</IfModule>
```

- Activer le moteur SSL. Dans le fichier *secure.local.conf*, ajouter:

```
SSLEngine ON
```

9.2 Manipulation

- Vous aurez besoin de 3 machines sous VirtualBox en réseau interne :
  - un serveur DNS : 192.168.20.200/24
  - un serveur Web : 192.168.20.201/24
  - un client graphique : 192.168.20.10/24

Client graphique, fichier */etc/network/interfaces*:

```
auto enp0s3
iface enp0s3 inet static
    address 192.168.20.10
    netmask 255.255.255.0
```

Serveur dns:

- fichier */etc/network/interfaces*:

```
auto enp0s3
iface enp0s3 inet static
    address 192.168.20.200
    netmask 255.255.255.0
```

- fichier */etc/bind/named.conf.local*:

```
zone "apache.local" {
    type master;
    file "/etc/bind/db.apache.local";
};
zone "autre.local" {
    type master;
    file "/etc/bind/db.autre.local";
};
zone "secure.local" {
    type master;
    file "/etc/bind/db.secure.local";
};
// 192.168.20.201
zone "201.20.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/db.201.20.168.192.in-addr.arpa";
};
```

– fichier */etc/bind/db.201.20.168.192.in-addr.arpa*:

```
$TTL    86400
@       IN      SOA     local.    root.local. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      local.
        IN      PTR     apache.local
        IN      PTR     www.apache.local
        IN      PTR     autre.local
        IN      PTR     www.autre.local
        IN      PTR     secure.local
        IN      PTR     www.secure.local
```

– fichier */etc/bind/db.apache.local*:

```
$TTL    86400
@       IN      SOA     apache.local.    root.apache.local. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      apache.local.
        IN      A       192.168.20.201
www     IN      CNAME   apache.local.
```

– fichier */etc/bind/db.autre.local*:

```
$TTL    86400
@       IN      SOA     autre.local.    root.autre.local. (
                        1           ; Serial
                        604800      ; Refresh
                        86400       ; Retry
                        2419200     ; Expire
                        86400       ; Negative Cache TL
                        )

@       IN      NS      autre.local.
        IN      A       192.168.20.201
www     IN      CNAME   autre.local.
```

– fichier */etc/bind/db.secure.local*:

```
$TTL    86400
@       IN      SOA     secure.local.    root.secure.local. (
                        1           ; Serial
```

```
                            604800      ; Refresh
                            86400       ; Retry
                            2419200     ; Expire
                            86400       ; Negative Cache TL
                            )

        @       IN      NS      secure.local.
                IN      A       192.168.20.201
        www     IN      CNAME   secure.local.
```

Serveur web:

- fichier `/etc/network/interfaces`:

```
auto enp0s3
iface enp0s3 inet static
    address 192.168.20.201
    netmask 255.255.255.0
```

Remarque: commenter tout le fichier */etc/resolv.conf* et ajouter : `nameserver 192.168.20.201`.

- Mettre en ligne 2 sites accessibles au client via les URL *www.apache.local*, *www.autre.local*.

Serveur web:

- `sudo mkdir /var/www/apache.local`
- `sudo mkdir /var/www/autre.local`
- `sudo mkdir /var/www/secure.local`
- `sudo mkdir /var/www/keys`
- `sudo chmod a=rwx /var/www/*`
- `echo "welcome on www.apache.local !" > /var/www/apache.local/index.html`
- `echo "welcome on www.autre.local !" > /var/www/autre.local/index.html`
- `echo "welcome on www.secure.local !" > /var/www/secure.local/index.html`
- `cd /etc/apache2/sites-available`
- `sudo cp ./000-default.conf ./apache.local.conf`
- `sudo cp ./000-default.conf ./autre.local.conf`
- `sudo cp ./000-default.conf ./secure.local.conf`
- `sudo nano ./<domaine>.conf`

```
<VirtualHost *:80>
    # ajouter les lignes suivantes:
    ServerName <domaine>
    ServerAlias www.<domaine>

    # modifier cette ligne:
    DocumentRoot /var/www/<domaine>/
</VirtualHost>
```

- `sudo a2ensite apache.local.conf`
- `sudo a2ensite autre.local.conf`

- `sudo a2ensite secure.local.conf`
- `sudo systemctl restart apache2`

- Mettre en ligne un 3ème site utilisant le protocole SSL *www.secure.local*.
  - `cd /var/www/keys`
  - `openssl req -x509 -nodes -newkey rsa:2048 -keyout key.private -out certificat.pem`
  - `cd /etc/apache2/sites-available`
  - `sudo nano ./secure.local.conf` (**peut-être pas besoin de loader le module**)

```
# ajouter la ligne suivante:
LoadModule ssl_module modules/mod_ssl.so

<VirtualHost *:443> # 80 -> 443
    # ajouter les lignes suivantes:
    SSLEngine ON
    SSLCertificateFile /var/www/keys/certificat.pem
    SSLCertificateKeyFile /var/www/keys/key.private
</VirtualHost>
```

  - `sudo a2enmod ssl` (pour charger le module ssl)
  - `sudo systemctl restart apache2`

- Sur le site *apache.local*, le module qui gère le langage PHP doit être activé.
  - `sudo apt install libapache2-mod-php`
  - `sudo a2enmod php7.3` (appuyer sur *tab* pour que la version de php s'auto-complète)

- Créez ensuite une page *index.php* qui sera chargée par défaut.

  Créer le fichier */var/www/apache.local/index.php*:

```
<?php
    echo '<h1> Welcome to www.apache.local ! </h1>';
    echo '<p> Written in index.php </p>';
?>
```

  Modifier le fichier */etc/apache2/sites-available/apache.local.conf*:

```
<VirtualHost *:80>
    # ajouter les lignes suivantes:
    <Directory /var/www/apache.local/>
        DirectoryIndex index.php
    </Directory>
</VirtualHost>
```

10 Files Sharing

10.1 Notions

- NAS and SAN are both network storage systems. The difference is to the type of resources used.
  - NAS : resource stored on a specific device connected to the LAN

  - SAN : resource stored in a data network

- SAMBA uses two daemons: smbd and nmbd. Smbd is responsible of resource sharing, authentication, and permissions.
- There are different modes :
  - user: authentication is requested at the first connection to the server and uses a client account of the server (login / password). The client has access to all resources.
  - share : obsolete share-based authentication, uses a simple password for access to a single resource.
  - domain : the machine is in a domain. It is the domain server that manages authentication. The domain server can be a Windows server or a SAMBA server.
  - ads : make the members of a domain function in native mode. In this case your machine accepts Kerberos tickets.
- Mount a share directory : smbmount //server/e /smb/e
- Management of a SAMBA user
  - smbpasswd -a usersamba add a user
  - smbpasswd -d usersamba disable account
  - smbpasswd -x usersamba delete account
  - smbpasswd -e usersamba enable account
- NFS is the abbreviation for Network File System.

10.2 Manipulation modifiée – samba

J'ai modifié la manip parce que le mode de samba qu'on demandait d'utiliser (share) n'existe plus.

- Créer un share en mode *user*.

  Installer, configurer et créer le share:

  - `sudo apt install samba smbclient`
  - `sudo mkdir /srv/share`
  - `sudo chmod a=rwx /srv/share`
  - `sudo nano /etc/samba/smb.conf`

```
# mode user => mode par défaut
[global]
    # secure = user
# ajouter ceci à la fin du fichier -> ligne 200+
[share]
    comment = This is my share :)
    path = /srv/share
    read only = no
    writeable = yes
```

```
        browseable = yes
```

- `testparm`

Créer des utilisateurs samba (ajoute un utilisateur existant à samba):

- `sudo adduser test1`
- `sudo adduser test2`
- `sudo smbpasswd -a test1`
- `sudo smbpasswd -a test2`
- `sudo pdbedit -w -L` (liste les utilisateurs samba)
- `sudo systemctl restart smbd`
- `smbclient //debian/test1` (devrait ne pas fonctionner)
- `su test1`
- `smbclient //debian/test1`
- `smbclient -U test2 //debian/test2`
- `smbclient //debian/share`
- `smbtree`

- Créer un utilisateur non-activé, sans shell valide, avec seulement un répertoire utilisateur. Pourquoi est-ce utile ? Créer l'utilisateur *tango* suivant ces critères (utiliser `useradd`).

Comme quand on a chrooté un utilisateur pour la manip ftp, c'est une bonne idée pour empêcher un utilisateur de modifier des fichiers en dehors de son répertoire.

- `sudo useradd tango --shell /sbin/nologin`
- `sudo passwd tango`
- `sudo mkdir /home/tango`
- `sudo usermod -d /home/tango tango`
- `sudo smbpasswd -a tango`

- Créer un répertoire pour exporter/modifier les fichiers de configuration.

- `mkdir /srv/config`
- `chmod a=rwx /srv/config`
- `sudo nano /etc/samba/smb.conf`

```
    [config]
        comment = This is a share for config files ^_^
        path = /srv/config
        read only = no
        writeable = yes
        browseable = yes
```

- `testparm`
- `sudo systemctl restart smbd`
- `echo "hello :)" > fichier`
- `chmod a=rwx fichier`
- `sudo smbclient -U tango //debian/config`

- `put fichier` (upload le fichier)

10.3 Manipulation modifiée – nfs

- Configurer un share nfs sur le réseau 10.0.2.0/24.
  - `sudo apt install nfs-kernel-server nfs-common`
  - `sudo mkdir /srv/nfs`
  - `sudo chmod a=rwx /srv/nfs`
  - `sudo nano /etc/exports`

```
# rw = allow read & write
# sync = write to disk before replying to queries
# 10.0.2.0/24 = réseau dans lequel se trouve la machine
/srv/nfs    10.0.2.0/24(rw,sync)
```

  - `sudo systemctl restart nfs-kernel-server`

- Monter le dossier nfs dans le dossier */ahome*.
  - `sudo mkdir /ahome`
  - `sudo chmod a=rwx /ahome`
  - `sudo mount <ip_serveur_nfs>:/srv/nfs /ahome`

- Créer un utilisateur local dont le répertoire personnel est dans */ahome*.
  - `sudo adduser --no-create-home --home /ahome/nfsuser nfsuser`

- Modifier le fichier */etc/fstab* pour faire un montage permanent.
  - `sudo nano /etc/fstab`

```
<ip_serveur_nfs>:/srv/nfs /ahome nfs4 rw 0 0
```

  - `sudo reboot`
  - `mount | grep ahome`

- Configurer le service automount pour monter automatiquement */ahome*.
  - `sudo apt install autofs`
  - démonter tous les montages manuels (a priori, juste */ahome*) : `sudo umount /ahome`
  - `mount | grep ahome`
  - `sudo nano /etc/auto.master`

```
# <point_de_montage>    /etc/auto.<type>
/ahome                  /etc/auto.nfs
```

  - `sudo nano /etc/auto.nfs`

```
# <nom_share>   <options>        <ip_serveur>:<dossier_share>
partage_ahome   -fstype=nfs4,rw  10.0.2.15:/srv/nfs
```

  - `sudo systemctl restart autofs`

- `mount | grep ahome`