

Administration des systèmes Unix

TP : Gestion des disques

L'atelier concerne:

- L'ajout d'un disque dur sur une machine
- Partitionnement du nouveau disque dur
- Initialisation du système de fichiers
- Création d'un point de montage

Configuration initiale :

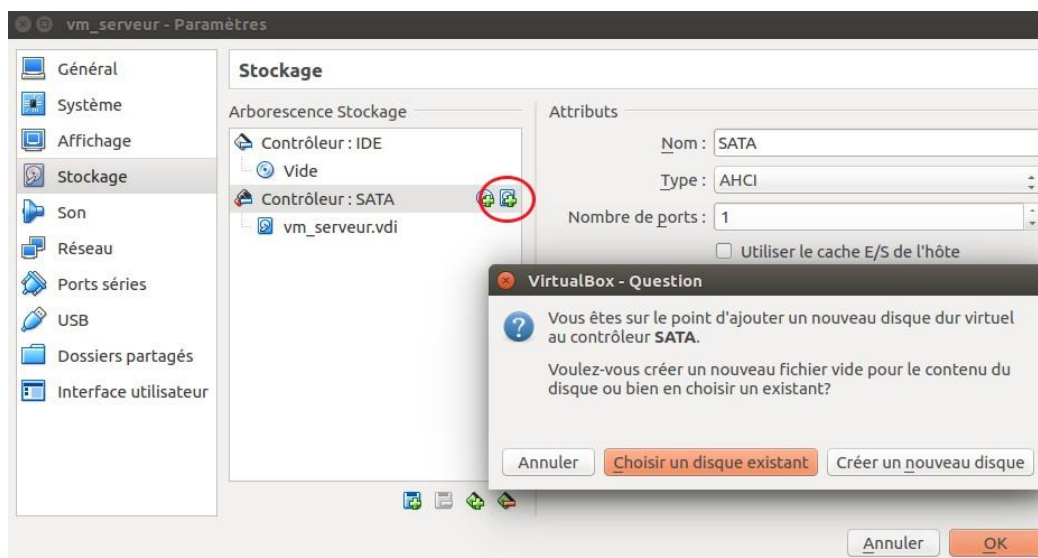
- Ce TP est à réaliser avec une station sous linux ubuntu

Ce TP concerne la gestion des disques durs. On va créer un nouveau disque virtuel et créer une partition dedans.

1. Ajouter un disque dur à la machine virtuelle

Éteignez la machine virtuelle et dans VirtualBox, allez dans "configuration".

Dans la section "Stockage", cliquez sur "Ajouter un disque dur" au niveau de votre Contrôleur SATA puis sur "Créer un nouveau disque". Laissez le type de fichier par défaut, choisissez un fichier de taille fixe, donnez-lui un nom et allouez-lui une taille de 1Go.



Maintenant que votre nouveau disque dur est "physiquement" créé, vous pouvez redémarrer votre machine virtuelle .

Pour pouvoir utiliser votre nouveau disque, il vous reste à :

- créer au moins une partition sur ce disque
- formater cette partition
- monter cette partition

2. Partitionner le nouveau disque dur

Sous Linux les périphériques systèmes sont représentés par des fichiers dans **/dev**. En particulier, les disques durs SCSI (et en fait maintenant presque tous les disques sont gérés comme des disques SCSI) sont représentés par des fichiers de type **/dev/sda**, **/dev/sdb**, etc. La dernière lettre correspond à l'ordre dans lequel les disques sont reconnus.

il suffit de taper la commande pour avoir des informations sur les disques durs:

```
$ ls -l /dev/sd*
```

On voit le disque système **/dev/sda** sur lequel l'installation par défaut a créé une partition **/dev/sda1**. Nous voyons également le nouveau disque **/dev/sdb**. Sans surprise, on ne trouve pas encore de **/dev/sdb1** car nous n'avons pas encore créé de partition sur ce disque. C'est ce que nous allons faire maintenant grâce à l'utilitaire **gdisk**.

Commençons par obtenir des informations sur nos disques avec l'option **-l** suivie du fichier périphérique à examiner. Comme vous avez pu le remarquer, pour avoir accès aux fichiers périphériques, il nous faut utiliser la commande **sudo**.

```
$ sudo gdisk -l /dev/sda
```

```
$ sudo gdisk -l /dev/sdb
```

Maintenant que vous êtes sûr que le disque additionnel dédié aux données est bien **/dev/sdb**, créez une partition sur ce disque.

Sans option, **gdisk** est un programme interactif. En tapant **?** puis "Entrée", nous voyons la liste des commandes disponibles. Il est indiqué que pour ajouter une partition, il nous faut taper **n**.

On peut alors accepter toutes les valeurs par défaut, ce qui nous amène à créer une partition numéro 1 qui commence au premier secteur disponible et termine au dernier secteur disponible (tout le disque donc). Un code indique alors que le système de fichier de la partition sera un “système de fichiers Linux”, ce qui nous va très bien.

N’hésitez pas à vérifier que votre nouvelle partition correspond bien à ce que nous voulons par la commande **p** puis enregistrez les changements sur le disque avec **w**.

Notez que votre système a tout de suite remarqué la création de cette nouvelle partition et a créé un fichier associé **/dev/sdb1**.

À ce stade, vous avez juste écrit une table de partition au début de votre disque de données qui indique que ce disque contient une partition. Il vous faut maintenant la formater.

3. Formater le nouveau disque dur (Initialisation du système de fichiers)

Nous allons formater la nouvelle partition au format **ext4** qui est actuellement le format par défaut sous Ubuntu et qui est un très bon choix pour la plupart des usages sous Linux. Pour cela, il suffit de taper la commande :

```
$ sudo mkfs -t ext4 /dev/sdb1
```

La commande **mkfs** permet de créer un système de fichier. Avec **-t ext4**, on précise qu’on veut un système ext4 et on donne en argument le fichier device **/dev/sdb1** qu’on veut formater.

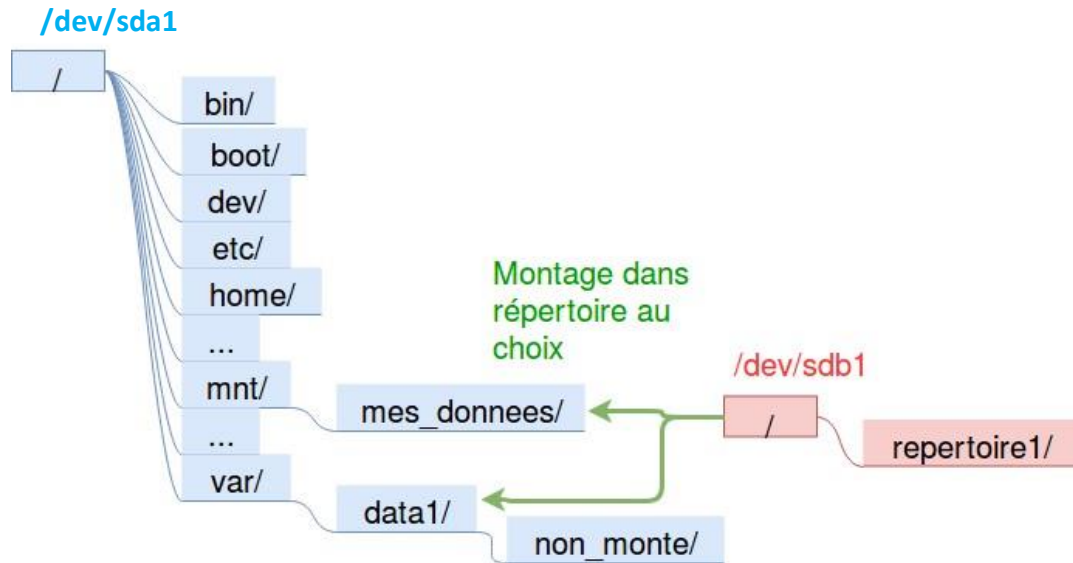
Votre nouveau disque dur est prêt. Pour l’utiliser, il ne reste plus qu’à le “monter”.

4. Monter le nouveau disque

Pour rappel, sous Windows, les disques durs, lecteur cd et autres périphériques de stockage sont identifiés par une lettre. Sous Linux, la racine de notre arborescence se nomme **/** puis on trouve des dossiers, des sous-dossiers, des sous-sous-dossiers, etc.

En fait, chaque partition/système de fichier possède sa propre racine **/** mais pour pouvoir accéder à cette partition, il vous faut “**monter**”, c’est à dire rattacher cette racine à un dossier de votre arborescence principale.

Dans le cadre de ce TP, nous allons définir le point de montage du nouveau disque comme illustré sur la figure ci-dessous.



Premièrement nous allons créer deux répertoires dont nous nous servons de point de montage : **/var/data1/non_monte** et **/mnt/mes_donnees**

```
# mkdir -p /var/data1/non_monte
```

```
# mkdir -p /mnt/mes_donnees/
```

Avec la commande **mount** , nous allons monter la partition **/dev/sdb1** dans le répertoire **/mnt/mes_donnees** en précisant que le système de fichier est de type **ext4**.

```
# mount -t ext4 /dev/sdb1 /mnt/mes_donnees
```

Vous pouvez ensuite créer le répertoire **repertoire1** à la racine de votre nouveau disque dur (premier niveau après le point de montage).

```
# mkdir -p /mnt/mes_donnees/repertoire1
```

Vérifiez la présence du répertoire crée par la commande :

```
# ls /mnt/mes_donnees
```

Pour démonter le disque de données, on utilise la commande **umount**. Comme le fichier de partition/dev/sdb1 et le répertoire de montage /mnt/mes_donnees sont maintenant “associés” dans la table de montage, nous n’avons besoin de passer qu’un des deux en argument pour indiquer le lien à défaire.

```
# umount /dev/sdb1
```

On vérifie que le répertoire /mnt/mes_donnees est maintenant vide car le répertoire repertoire1 est sur le disque de données.

```
# ls /mnt/mes_donnees
```

Nous allons maintenat monter /dev/sdb1 dans le répertoire /var/data1

```
# mount -t ext4 /dev/sdb1 /var/data1
```

En listant le contenu du répertoire **/var/data1** on va retrouver le répertoire repertoire1 (et un répertoire lost+found) mais le répertoire **non_monte** qui était dans **/var/data1** a disparu. Ce répertoire n'est pas perdu, mais il est inaccessible car "caché" par le montage que nous venons de réaliser par-dessus. Vous pourrez d'ailleurs retrouver ce répertoire dès que vous démonterez **/dev/sdb1**.

Vous savez maintenant comment accéder à votre nouveau disque dur mais il serait quand même bien pratique qu'il puisse être monté automatiquement au démarrage du système. Pour cela, vous pouvez rajouter la ligne suivante à la fin du fichier **/etc/fstab** :

/dev/sdb1 /var/data1 ext4 defaults 0 0

Chaque ligne du fichier **/etc/fstab** comporte six champs séparés par des espaces ou des tabulations et qui indiquent dans l'ordre :

- le périphérique à monter
- le point de montage
- le type de système de fichiers
- les options de montage séparées par des virgules (par défaut pour nous)

Au démarrage, le système va lancer la commande **mount -a** qui montera tous les périphériques indiqués dans le fichier **/etc/fstab**.