

---

# TP 1

## Gestion du stockage des données

---

### Objectif

---

Etudier les différentes solutions de stockage pour assurer l'évolutivité d'un système informatique.

### Contexte

---

La société GSB dispose d'un serveur de fichiers sous Linux Ubuntu. Ce dernier contient les fichiers pour les dossiers projets de la société.

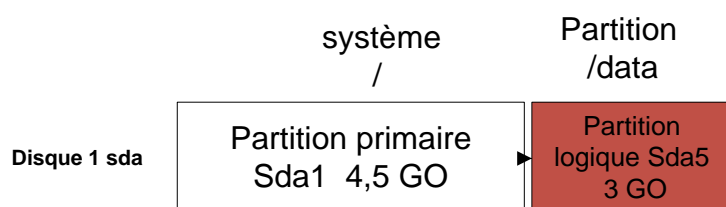
Nous allons dans un premier temps installer ce serveur.

### Observations

---

- ➔ Prendre connaissance du document « *ressource – Disque-partition et système de fichiers V2* » et réaliser, par groupe de deux, une synthèse avec XMIND
- ➔ Vous allez installer le serveur Ubuntu LTS dans une VM en respectant le plan de partitionnement indiqué dans le tableau ci-dessous :

Disque SCSI de 8 GO



- **Disque sda**

Partition primaire			
Point de montage	partition	Taille	Type de FS
/ (racine)	Sda1 Primaire	4,5 GO	Ext4

	Partition logiques		
/data	Sda5	3GO	Ext4
Swap	Sda6	500 MO	Swap

Avant de lancer le partitionnement, faire valider configuration par le professeur !!!!!

## Observer l'état du système de fichiers

Les commandes de manipulation du système de fichiers Linux sont présentées dans « Les commandes de gestion des systèmes de fichiers »

Pour visualiser les partitions sur le serveur *# fdisk -l*

```
/dev/sda1 *    2048    7813119    3905536    83 Linux
/dev/sda2      7815166    11532287    1858561     5 Étendue
/dev/sda5      7815168    9766911    975872    83 Linux
/dev/sda6      9768960    11532287    881664    82 partition d'échange Linux / Solaris
```

Pour visualiser l'espace disque disponible sur le serveur *:# df -h*

```
root@ubuntu:~# df -h
```

```
/dev/sda1      3,7G   945M  2,5G  27% /
.....
.....
/dev/sda5      922M   1,2M  874M   1% /data
```

Vous constatez que la partition de /dev/sda5 permettant de stocker les données, utilisateurs est presque pleine. ;-)) En réalité non ;-)), mais nous allons faire tout comme !

Il faut donc faire évoluer le serveur pour permettre de continuer à stocker les données utilisateurs dans cette partition de stockage.

Nous allons étudier 2 solutions pour résoudre le problème :

- ➔ La première solution consiste à ajouter d'un nouveau disque de 3 GO pour qui servira au stockage des données.
- ➔ La seconde solution consiste à utiliser la technologie Logical volume Manager – LVM

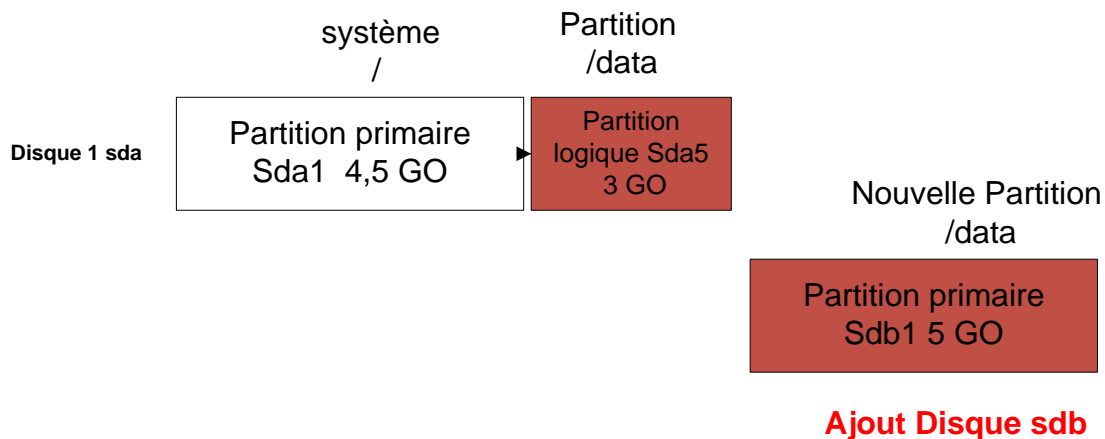
## Première solution : Ajout d'un disque physique

---

### Installation du disque

---

Nous allons stocker les données sur le nouveau disque sdb1 et donc abandonner la partition sda5 comme représenté ci-dessous.



Il faut tout d'abord ajouter un disque dur au serveur Linux. Nous ajouterons un disque de 5 GO de type SCSI.

Cette opération peut être réalisée à chaud pour les machines supportant la technologie « hot plug ».

Ajouter sur votre VM un disque dur de 5GO.

### Configuration du disque pour son utilisation dans linux

---

3 étapes sont nécessaires pour configurer le disque

- **Partitionnement** : permet de segmenter le disque
- **Formatage** : permet d'attribuer un système de fichier
- **Montage** du disque : Permet de rendre le disque accessible

Pour réaliser la configuration du disque utiliser la documentation d'installation suivante :

### Disque dur où es-tu ?

---

La première étape consiste tout simplement à trouver notre disque, en s'étant auparavant assuré qu'il est bien branché et connecté (dans le cas dans la machine virtuelle)

Sous la majorité des distributions Linux on trouvera une convention de nommage en "hd" ou "sd". Le premier correspondant aux disques IDE qui seront donc "hda" pour le premier disque, "hdb" pour le second disque et respectivement "hda1" pour la première partition primaire du disque, "hda2" pour la seconde pour finir avec

“hda5” et suivantes pour les partitions secondaires. En revanche pour les disques durs SATA ou SCSI, le système les note “sd” en suivant la même nomenclature de lettrage et de chiffrement que pour les disques durs IDE. C’est une note importante car elle permet de retrouver facilement ces périphériques parmi cela :

```
#ls /dev
```

```
op3      midi      pts      sg0      stdout   t
op4      net       random   sg1      tty      t
op5      network_latency root     sg2      tty0     t
op6      network_throughput rtc      shm      tty1     t
op7      null      rtc0     snapshot tty10    t
op-control oldmem    sda      snd      tty11    t
KEDEV    port      sda1     sndstat  tty12    t
ppper    ppp       sda2     sr0      tty13    t
elog     psaux     sda5     stderr   tty14    t
m        ptmx      sdb      stdin    tty15    t
```

On remarque donc ici sda qui est mon disque principale (SATA ou SCSI) qui contient trois partitions (deux primaires et une logique) puis un second disque “sdb” qui en l’occurrence est le disque que je viens d’ajouter.

On pourra aussi visualiser avec la commande `#fdisk -l`

### Partitionnement

On va à présent présenter la partition du disque dur, on va ainsi pouvoir découper des zones de notre disque afin d’effectuer, par exemple, plusieurs points de montage ou plusieurs systèmes de fichiers. On utilise pour cela fdisk avec lequel on va pointer le disque dur en question :

```
1 fdisk /dev/sdb
```

```
root@VLAD-IV:~# fdisk /dev/sdb
Le périphérique ne contient pas une table de partitions DOS ou Sun, SGI, OSF valable
Création d'une nouvelle étiquette DOS avec id de disque 0xcc210c69.
Les modifications restent en mémoire jusqu'à ce que vous les écriviez.
Après quoi, bien sûr, le contenu précédent sera irrécupérable.

Avertissement : indicateur 0x0000 incorrect dans la table de partitions 4, sera corrigé par w (écriture)
```

On peut ici afficher les différentes options qui s’offrent à nous via la saisie de “m” :

```

Commande (m pour l'aide): m
Commande d'action
  a basculer l'indicateur d'amorçage
  b éditer l'étiquette BSD du disque
  c basculer l'indicateur de compatibilité DOS
  d supprimer la partition
  l lister les types de partitions connues
  m afficher ce menu
  n ajouter une nouvelle partition
  o créer une nouvelle table vide de partitions DOS
  p afficher la table de partitions
  q quitter sans enregistrer les changements
  s créer une nouvelle étiquette vide pour disque de type Sun
  t modifier l'identifiant de système de fichiers d'une partition
  u modifier les unités d'affichage/saisie
  v vérifier la table de partitions
  w écrire la table sur le disque et quitter
  x fonctions avancées (pour experts seulement)

```

On va donc saisir “n” pour ajouter une nouvelle partition au disque ciblé :

```

Commande (m pour l'aide): n
Type de partition :
  p primaire (0 primaire(s), 0 étendue(s), 4 libre(s))
  e étendue

```

On saisit “p” pour ajouter une partition primaire (4 maximum), on saisit ensuite le numéro de cette partition (entre 1 et 4 pour la partition primaire, au-dessus de 5 pour une partition étendue/logique) et ensuite les secteurs de début/fin de ladite partition. Dans mon cas, je construis une partition unique prenant la totalité du disque étant donné que ce sera un simple ajout de disque sur un système existant :

```

Sélection (p par défaut) : p
Numéro de partition (1-4, par défaut 1): 1
Premier secteur (2048-10485759, par défaut 2048):
Utilisation de la valeur par défaut 2048
Dernier secteur, +secteurs or +taille{K,M,G} (2048-10485759, par défaut 10485759):
Utilisation de la valeur par défaut 10485759

```

On saisit ensuite “w” qui signifie “write” qui va donc écrire les changements que nous venons d’ordonner sur le disque en question. Attention, si des données existent sur le disque en question, elles seront supprimées :

```

Commande (m pour l'aide): w
La table de partitions a été altérée.

Appel de ioctl() pour relire la table de partitions.
Synchronisation des disques.

```

On sera alors de retour sur notre terminal où nous pourrons à nous regarder le contenu de /dev (“ls /dev”). Nous verrons alors un “sdb1” qui correspond donc à la partition 1 du second disque “sdb” SATA ou SCSI. C’est ce que nous venons de créer :

```
pts      sdb1
random   sg0
root     sg1
rtc      sg2
rtc0     shm
sda      snapshot
sda1     snd
sda2     sndstat
sda5     sr0
sdb      stderr
```

## Formatage

Nous avons créé une partition exploitable mais celle-ci n'a pas été préparée à être traitée avec un système de fichier spécifique, NTFS, FAT, ext3 ? Nous allons donc occuper à présent de formater notre partition. Nous utiliserons pour cela l'utilitaire `mkfs.ext3` disponible nativement sur Debian 7. On va donc pointer avec cet utilitaire notre partition à formater :

```
1 mkfs.ext3 /dev/sdb1
```

```
root@VLAD-IV:~# mkfs.ext3 /dev/sdb1
mke2fs 1.42.5 (29-Jul-2012)
Étiquette de système de fichiers=
Type de système d'exploitation : Linux
Taille de bloc=4096 (log=2)
Taille de fragment=4096 (log=2)
« Stride » = 0 blocs, « Stripe width » = 0 blocs
327680 i-noeuds, 1310464 blocs
65523 blocs (5.00%) réservés pour le super utilisateur
Premier bloc de données=0
Nombre maximum de blocs du système de fichiers=1342177280
40 groupes de blocs
32768 blocs par groupe, 32768 fragments par groupe
8192 i-noeuds par groupe
Superblocs de secours stockés sur les blocs :
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocation des tables de groupe : complété
Écriture des tables d'i-noeuds : complété
Création du journal (32768 blocs) : complété
Écriture des superblocs et de l'information de comptabilité du système de
fichiers : complété
```

Voilà, il ne nous reste plus qu'à monter notre partition pour que celle-ci soit utilisable. On utilise pour cela le fichier dédié à cette tâche `/etc/fstab`. Il nous suffit ici de rajouter une simple ligne (qui peut se complexifier pour des cas d'utilisation ou des besoins spécifiques) qui va se charger de monter notre disque. Pour information, mettre une ligne dans ce fichier permet de monter le disque dur au démarrage car ce fichier est lu et exécuté au démarrage de la machine. On pourra également exécuter une simple ligne de commande pour monter le disque temporairement (jusqu'au prochain redémarrage). Par exemple, si je veux monter mon disque dur sur `/data`, je crée mon répertoire avec :

```
1 mkdir /data
```

Puis j'ajoute cette ligne dans `/etc/fstab`

**Pensez à supprimer l'ancienne ligne dans fstab qui assure le montage de data sur sda5 en commentant la ligne :**

```
« #UUID=69650936-d17d-4fd6-9e12-bf01f886046c /data      ext4  defaults    0
```

Votre fichier doit ressembler à celui-ci.

```
# /etc/fstab: static file system information.
# <file system> <mount point> <type> <options>    <dump> <pass>
proc      /proc      proc nodev,noexec,nosuid 0    0
# / was on /dev/sda1 during installation
UUID=c2d2dc4c-b591-46a8-bb2a-c43e13aa4193 /      ext4  errors=remount-ro 0    $
# /data was on /dev/sda5
#UUID=69650936-d17d-4fd6-9e12-bf01f886046c /data      ext4  defaults    0    $ ←commentez
# swap was on /dev/sda6 during installation
UUID=4a600360-a6cd-474e-9879-12dbd5004af7 none    swap  sw          0    0
/dev/sdb1  /data  ext3  defaults ← nouvelle ligne
```

On utilisera ensuite la commande qui va nous permettre de relire et de réaffecter le contenu du fichier /etc/fstab :

```
1 mount -a
```

On pourra alors vérifier la présence de notre nouveau disque en listant les montages de notre système

```
1 mount
```

```
/dev/sdb1 on /data type ext3 (rw,relatime,errors=continue,user_xattr,acl,barrier=1,data=ordered)
```

Vous constatez maintenant que votre répertoire data est monté sur le nouveau disque et dispose de 3GO.

```
root@SRVFIC:~# df -h
Sys. de fichiers Taille Utilisé Dispo Uti% Monté sur
/dev/sda1      3,7G  945M  2,5G  28% /
udev           235M   4,0K  235M   1% /dev
tmpfs          98M   312K   98M   1% /run
none           5,0M    0 5,0M   0% /run/lock
none           244M    0 244M   0% /run/shm
/dev/sdb1      3,0G   69M  2,8G   3% /data
```

Dans la pratique il faudrait recopier les données de l'ancienne partition /dev/sda5 vers cette nouvelle partition /dev/sdb1.

## Analyse

L'ajout d'un nouveau disque résout le problème temporairement, mais deux questions se posent :

- Si ce nouveau disque est plein il faudra recommencer les opérations et modifier encore les paramètres du système ?

L'ajout du second disque a provoqué la perte de la partition sda5. ?

On déduit donc que :

**Le système de partitionnement actuel ne permet pas un stockage évolutif.** Tout ajout de disque ou partition nécessite de reconfigurer le système de fichiers

**Le système de partitionnement actuel n'est pas optimisé,** Il n'est pas possible de fusionner 2 partitions, par exemple sda5 et sdb1 pour stocker les données utilisateurs, car un dossier ou un point de montage est lié à une seule partition !

Nous allons maintenant utiliser une seconde solution pour la gestion des disques qui répond à ces problèmes.

**Note : Pensez à mettre à jour votre WIKI avec les commandes linux apprises**

---