Chapitre 4
Les disques et le système de fichiers

1. Représentation des disques

Note préalable : les unités de mesure de stockage utilisées dans ce chapitre et dans l'ensemble de ce livre utilisent la représentation de l'usage traditionnel en kilo-octets, selon la règle 1 ko = 1024 octets (2^{10}), sauf indication contraire. Cette représentation se nomme théoriquement Kio (kibioctet).

1.1 Nomenclature

Ceci est un petit rappel des points déjà rencontrés dans le chapitre Présentation de Linux. Suivant le type de contrôleur et d'interface sur lesquels les disques sont connectés, Linux donne des noms différents aux fichiers spéciaux des périphériques disques.

Chaque disque est représenté par un fichier spécial de type bloc. Chaque partition aussi.

1.1.1 IDE

Cette section est conservée pour des raisons historiques, la norme SATA ayant remplacé la norme IDE sur la quasi-totalité des ordinateurs de bureau et portables depuis plus de dix ans. Les disques reliés à des contrôleurs IDE (appelés aussi PATA (*Parallel ATA*) ou ATAPI) se nomment hdX :

- hda : IDE0, Master
- hdb : IDE0, Slave

– hdc : IDE1, Master
– hdd : IDE1, Slave
– etc.

Contrairement aux idées reçues, il n'y a pas de limites au nombre de contrôleurs IDE, sauf le nombre de ports d'extension de la machine (slots PCI). De nombreuses cartes additionnelles et convertisseurs existent permettant de lire d'anciens disques IDE. Au-delà de quatre disques ou lecteurs, les fichiers se nomment hde, hdf, hdg, etc.

Les lecteurs CD-Rom, DVD et graveurs de type IDE/ATAPI sont vus comme des disques IDE et respectent cette nomenclature.

Les noyaux Linux utilisent maintenant par défaut une API appelée libata pour accéder à l'ensemble des disques IDE, SCSI, USB, Firewire, etc. La nomenclature reprend celle des disques SCSI, abordée au point suivant.

1.1.2 SCSI, SATA, USB, FIREWIRE, etc.

Les disques reliés à des contrôleurs SCSI, SCA, SAS, FibreChannel, USB, Firewire, thunderbolt (et probablement d'autres interfaces exotiques) se nomment sdX. L'énumération des disques reprend l'ordre de détection des cartes SCSI et des adaptateurs (hosts) associés, puis l'ajout et la suppression manuelle des autres via hotplug ou udev.

– sda : premier disque SCSI
– sdb : deuxième disque SCSI
– sdc : troisième disque SCSI
– etc.

La norme SCSI fait une différence entre les divers supports. Aussi les lecteurs CD-Rom, DVD, HD-DVD, Blu-ray et les graveurs associés n'ont pas le même nom. Les lecteurs et graveurs sont en srX (sr0, sr1, etc.). Vous pouvez aussi trouver scd0, scd1, etc. mais ce sont généralement des liens symboliques vers sr0, sr1, etc.

Au-delà de sdz, l'énumération redémarre à sdaa, sdab, etc.

La commande **lsscsi** permet d'énumérer les périphériques SCSI. Notez que les disques sont bien en sdX, tandis que le lecteur dvd est en srX.

```
$ lsscsi
[4:0:0:0]    disk    ATA      ST380011A         8.01    /dev/sda
[5:0:0:0]    cd/dvd  LITE-ON  COMBO SOHC-4836V  S9C1    /dev/sr0
[31:0:0:0]   disk    USB2.0   Mobile Disk       1.00    /dev/sdb
```

1.2 Cas spéciaux

1.2.1 Contrôleurs spécifiques

Certains contrôleurs ne suivent pas cette nomenclature. C'est par exemple le cas de certains contrôleurs RAID matériels. C'est du cas par cas. Un contrôleur Smart Array sur un serveur HP, utilisant le pilote cciss, place ses fichiers de périphériques dans /dev/cciss sous les noms cXdYpZ, où X est le slot, Y le disque et Z la partition. Les nouveaux contrôleurs utilisent le pilote hpsa, exploitant la couche SCSI du noyau et donc un nommage standard des périphériques.

1.2.2 Virtualisation

La représentation des disques des systèmes invités (*guests*) virtualisés dépend du type de contrôleur simulé. La plupart sont de type IDE ou SCSI, et dans les deux cas bien souvent avec la libata ils sont vus comme du SCSI. Cependant certains systèmes comme KVM ou XEN (ainsi que les environnements cloud les utilisant, comme AWS) proposant de la paravirtualisation offrent un contrôleur spécifique présentant les disques sous le nom vdX (virtual disk x), ou xvdX :

– vda : premier disque virtualisé, ou xvda,
– vdb : deuxième disque virtualisé, ou xvdb,
– etc.

1.2.3 SAN, iSCSI, multipathing

Les disques raccordés via un SAN (*Storage Area Network*, généralement en fibre optique) ou par iSCSI sont vus comme des disques SCSI et conservent cette nomenclature. Cependant les systèmes de gestion des chemins multiples (*multipathing*) se plaçant par-dessus fournissent d'autres noms. Powerpath nommera les disques emcpowerx (emcpowera, emcpowerb, etc.) tandis que le système par défaut de Linux appelé multipath les nommera mpathx (mpath0, mpath1, etc.) ou tout autre nom choisi par l'administrateur.

2. Manipulations de bas niveau

2.1 Informations

La commande **hdparm** permet d'effectuer un grand nombre de manipulations directement sur les périphériques disques gérés par la bibliothèque libata, c'est-à-dire tous les disques SATA, ATA (IDE) et SAS. La commande **sdparm** peut faire à peu près la même chose pour les disques SCSI. Notez que bien que les noms de périphériques de la libata soient identiques à ceux du SCSI, il est fort probable que de nombreuses options de configuration de hdparm ne fonctionnent pas sur des disques SCSI, la réciproque étant vraie pour sdparm avec les disques SATA ou IDE. La suite se base sur hdparm.

Pour obtenir des informations complètes sur un disque, utilisez les paramètres `-i` ou `-I`. Le premier récupère les informations depuis le noyau et obtenues au moment du boot, le second interroge directement le disque. Préférez le `-I` qui donne des informations très détaillées.

```
# hdparm -I /dev/sda

/dev/sda:

ATA device, with non-removable media
    Model Number:       VBOX HARDDISK
    Serial Number:      VB91a2e953-933cdc65
    Firmware Revision:  1.0
Standards:
    Used: ATA/ATAPI-6 published, ANSI INCITS 361-2002
    Supported: 6 5 4
Configuration:
    Logical        max      current
    cylinders      16383    16383
    heads          16       16
    sectors/track  63       63
    --
    CHS current addressable sectors:    16514064
    LBA    user addressable sectors:    63152320
    LBA48  user addressable sectors:    63152320
    Logical/Physical Sector size:           512 bytes
    device size with M = 1024*1024:        30836 MBytes
    device size with M = 1000*1000:        32333 MBytes (32 GB)
    cache/buffer size  = 256 KBytes (type=DualPortCache)
Capabilities:
    LBA, IORDY(cannot be disabled)
    Queue depth: 32
```

```
    Standby timer values: spec'd by Vendor, no device specific minimum
    R/W multiple sector transfer: Max = 128        Current = 128
    DMA: mdma0 mdma1 mdma2 udma0 udma1 udma2 udma3 udma4 udma5 *udma6
         Cycle time: min=120ns recommended=120ns
    PIO: pio0 pio1 pio2 pio3 pio4
         Cycle time: no flow control=120ns  IORDY flow control=120ns
Commands/features:
    Enabled    Supported:
        *    Power Management feature set
        *    Write cache
        *    Look-ahead
        *    48-bit Address feature set
        *    Mandatory FLUSH_CACHE
        *    FLUSH_CACHE_EXT
        *    Gen2 signaling speed (3.0Gb/s)
        *    Native Command Queueing (NCQ)
Checksum: correct
```

2.2 Modification des valeurs

Plusieurs paramètres des disques peuvent être modifiés. Attention cependant ! Certaines options de hdparm peuvent se révéler être dangereuses tant pour les données contenues sur le disque que pour le disque lui-même. La plupart des paramètres sont en lecture et écriture. Si aucune valeur n'est précisée hdparm affiche l'état du disque (ou du bus) pour cette commande. Voici quelques exemples d'options intéressantes.

- **-c** : largeur du bus de transfert EIDE sur 16 ou 32 bits. 0=16, 1=32, 3=32 compatible.
- **-d** : utilisation du DMA. 0=pas de DMA, 1=DMA activé.
- **-X** : modifie le mode DMA (mdma0 mdma1 mdma2 udma0 udma1 udma2 udma3 udma4 udma5). Vous pouvez utiliser l'un des modes précédents ou des valeurs numériques : 32+n pour les modes mdma (n variant de 0 à 2) et 64+n pour les modes udma.
- **-C** : statut de l'économie d'énergie sur le disque (unknown, active/idle, standby, sleeping). L'état peut être modifié avec -S, -y, -Y et -Z.
- **-g** : affiche la géométrie du disque.
- **-M** : indique ou modifie l'état du Automatic Acoustic Management (AAM). 0=off, 128=quiet et 254=fast. Tous les disques ne le supportent pas.
- **-r** : passe le disque en lecture seule.

– **-T** : bench de lecture du cache disque, idéal pour tester les performances de transfert entre Linux et le cache du disque. Il faut relancer la commande deux ou trois fois.
– **-t** : bench de lecture du disque, hors cache. Mêmes remarques que l'option précédente.

Ainsi la commande suivante passe le bus de transfert en 32 bits, active le mode DMA en mode Ultra DMA 5 pour le disque sda :

```
# hdparm -c1 -d3 -X udma5 /dev/sda
```

Voici quelques autres exemples :

```
# hdparm -c /dev/sda

/dev/sda:
 IO_support = 0 (default 16-bit)

# hdparm -C /dev/sda

/dev/sda:
 drive state is: active/idle

# hdparm -g /dev/sda

/dev/sda:
 geometry = 3931/255/63, sectors = 63152320, start = 0

# hdparm -T /dev/sda

/dev/sda:
 Timing cached reads: 23868 MB in 2.00 seconds = 11950.45 MB/sec
# hdparm -t /dev/sda

/dev/sda:
 Timing buffered disk reads: 308 MB in 3.02 seconds = 101.87 MB/sec
```

3. Choisir un système de fichiers

3.1 Principe

3.1.1 Définition

L'action de « formater » un disque, une clé ou tout support de données consiste uniquement à créer sur un support de mémoire secondaire (volume de stockage) l'organisation logique permettant d'y placer des données. Le mot « formatage » sous Linux est utilisé pour décrire la création d'un système de fichiers. On parle donc de système de fichiers qui est à la fois l'organisation logique des supports au niveau le plus bas comme au niveau de l'utilisateur.

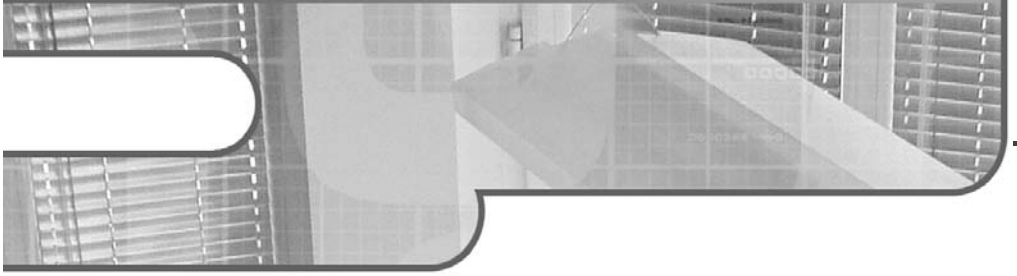
Les informations ne sont pas écrites n'importe comment sur les disques. Une organisation est nécessaire pour y placer tant les informations sur les fichiers qui y sont stockés que les données. Ce sont le système de fichiers et les pilotes associés qui définissent cette organisation. Si les principes de base sont souvent les mêmes entre les divers systèmes présents sous Linux, les implémentations et les organisations logiques des données sur le disque varient fortement. Aussi il n'existe pas un type de système de fichiers, mais plusieurs, au choix de l'utilisateur ou de l'administrateur système.

Tous les systèmes de fichiers Linux doivent respecter les normes POSIX. Comme POSIX définit un ensemble de règles de base, un système de fichiers peut aller au-delà de cette norme en proposant des extensions. La plupart de celles-ci concernent des éléments de sécurité, comme les ACL ou selinux.

Le principe de base d'un système de fichiers est d'associer un nom de fichier à son contenu et d'y permettre l'accès : création, modification, suppression, déplacement, ouverture, lecture, écriture, fermeture. Suivant ce principe, le système de fichiers doit gérer ce qui en découle : mécanismes de protection des accès (les permissions, les propriétaires), les accès concurrents, etc.

3.1.2 Représentation

Outre l'organisation et le stockage des informations et des données sur les fichiers, le système de fichiers doit fournir à l'utilisateur une vision structurée de ses données, permettant de les distinguer, de les retrouver, de les traiter et de les manipuler sous forme de fichiers au sein d'une arborescence de répertoires avec les commandes associées. De même, chaque système de fichiers doit fournir le nécessaire pour que les programmes puissent y accéder.



Chapitre 3
Gestion des périphériques de stockage

1. Gestion des périphériques de stockage

Ce chapitre traite de la gestion des disques RAID, de la configuration bas-niveau des disques, des disques réseau iSCSI, ainsi que du gestionnaire de volumes logiques LVM.

1.1 Configuration des disques RAID

L'objectif de cette section est de vous apprendre à :

– configurer et implémenter du RAID logiciel. Cela inclut les niveaux de RAID 0, 1 et 5.

1.1.1 Compétences principales

– Fichiers de configuration et utilitaires de gestion du RAID logiciel.

1.1.2 Éléments mis en œuvre

– `mdadm.conf`
– `mdadm`
– `/proc/mdstat`
– Partition de type `0xFD`

1.2 Optimiser l'accès aux périphériques de stockage

L'objectif de cette section est de vous apprendre à :

– configurer le noyau pour gérer différents types de disques ;
– connaître les outils logiciels pour lister et modifier le paramétrage des périphériques iSCSI.

1.2.1 Compétences principales

– Outils et commandes pour configurer le DMA pour des périphériques IDE, y compris ATAPI et SATA.
– Outils et commandes pour configurer des disques SSD (*Solid State Drive*), y compris AHCI et NVMe.
– Outils et commandes pour configurer ou analyser les ressources système (par exemple les interruptions).
– Connaissances de base de la commande `sdparm` et de son utilisation.
– Outils et commandes de gestion des périphériques iSCSI.
– Connaissances de base du SAN, y compris les protocoles spécifiques (AoE, FCoE).

1.2.2 Éléments mis en œuvre

– `hdparm, sdparm`
– `nvme`
– `tune2fs`
– `fstrim`
– `sysctl`
– `/dev/hd*, /dev/sd*, /dev/nvme*`
– `iscsiadm, scsi_id, iscsid` et `iscsid.conf`
– `WWID, WWN, n° LUN`

1.3 Logical Volume Manager

L'objectif de cette section est de vous apprendre à :

– créer et supprimer des volumes logiques, des groupes de volumes et des volumes physiques. Cet objectif inclut les instantanés (*snapshots*) et le redimensionnement des volumes logiques.

1.3.1 Compétences principales

- Outils de la suite LVM.
- Redimensionner, renommer, créer, supprimer des volumes logiques, des groupes de volumes et des volumes physiques.
- Créer et maintenir des instantanés (*snapshots*).
- Activer des groupes de volumes.

1.3.2 Éléments mis en œuvre

- `/sbin/pv*`
- `/sbin/lv*`
- `/sbin/vg*`
- `mount`
- `/dev/mapper/`
- `lvm.conf`

2. Configuration des disques RAID

La technologie RAID (*Redundant Array of Independent Disks*) permet de combiner différents périphériques pour qu'ils soient vus comme un seul espace de stockage par les applications. On peut ainsi améliorer les temps d'accès et/ou la fiabilité des périphériques de stockage. Les différentes techniques mises en œuvre sont définies par leur niveau de RAID. Les niveaux les plus courants sont RAID 0, 1 et 5.

Le RAID peut être géré de façon matérielle, par des contrôleurs de disques spécialisés, ou logicielle, au niveau du système d'exploitation.

Linux implémente un pilote de gestion logicielle du RAID, le pilote `md` (*Multiple Device driver*), qui gère les niveaux les plus courants de RAID, 0, 1 et 5.

Remarque

D'autres solutions peuvent être mises en place pour gérer du RAID logiciel sur Linux : RAID LVM ou RAID directement pris en charge par le gestionnaire de système de fichiers ZFS ou Btrfs.

2.1 Les principaux niveaux de RAID

2.1.1 Le RAID 0

Le RAID 0 (agrégat par bandes, *striping*) combine plusieurs disques en un seul ensemble. Les blocs de données sont répartis sur des bandes de taille identique réparties uniformément sur les différents disques. Les opérations d'entrées-sorties peuvent donc être très rapides, car effectuées simultanément par les différents contrôleurs disques.

En revanche, la fiabilité de l'ensemble est fortement diminuée, puisqu'il suffit de perdre un disque pour perdre l'ensemble des données. Il n'y a aucune redondance des données stockées, et la cohérence des volumes logiques est détruite en cas de défaillance d'un disque.

L'espace de stockage utile d'un ensemble RAID 0 est égal à la capacité utile du plus petit des disques, multipliée par le nombre de disques qui le composent, puisqu'il n'y a pas de redondance des données et que les bandes de données sont réparties uniformément sur les disques (chaque disque doit avoir le même nombre de bandes).

Avantages :

– Rapidité de lecture et d'écriture des ensembles de blocs.
– Utilisation optimale de l'espace disque, si les disques sont de même taille.

Inconvénients :

– Pas de redondance des données, donc pas de tolérance de panne.
– La perte d'un disque compromet l'ensemble des données stockées, la fiabilité de l'ensemble est égale à la fiabilité du moins fiable des disques utilisés.

2.1.2 Le RAID 1

Le RAID 1 (disques miroirs, *mirroring*) combine plusieurs disques en un seul ensemble. Chaque bloc de données utile est écrit sur chacun des disques. Cette redondance assure une excellente fiabilité à l'ensemble, d'autant plus grande qu'il y a davantage de disques. Tant qu'il reste un disque opérationnel, les données sont intactes, et tant que le contrôleur de ce disque fonctionne, elles restent accessibles.

Les opérations de lecture peuvent être plus rapides, car elles peuvent être effectuées simultanément par les différents contrôleurs disques.

L'espace de stockage utile d'un ensemble RAID 1 est égal à la capacité utile du plus petit des disques.

Avantages :

– Excellente tolérance de panne, proportionnelle au nombre de disques combinés (et au nombre de contrôleurs de disques pour l'accessibilité).
– Bonnes performances en lecture.

Inconvénients :

– L'espace disque nécessaire est au moins deux fois la taille de l'espace disque utile.
– Les performances en écriture peuvent être impactées, même si en général les écritures se font simultanément sur les différents disques.

2.1.3 Le RAID 5

Le RAID 5 (agrégat par bandes avec parité) combine au moins trois disques en un seul ensemble. Les blocs de données sont répartis sur des bandes de taille identique réparties uniformément sur les différents disques sauf un. Pour chaque ensemble de bandes, une bande de parité est calculée et écrite sur le disque restant. L'emplacement de la bande de parité est réparti à tour de rôle sur les disques.

En cas de perte d'une bande de données, la bande de parité permet de la reconstituer, assurant ainsi la tolérance de panne. Mais ce mécanisme n'est efficace que pour un seul disque inaccessible, la défaillance de deux disques ou plus entraîne la perte des données de l'ensemble des disques. Tant qu'un disque n'est pas opérationnel, il n'y a plus de tolérance de panne pour les nouvelles écritures. C'est pourquoi les ensembles de disques en RAID 5 intègrent généralement un disque de secours (*spare disk*), qui n'est utilisé que pour remplacer un disque défaillant.

Une fois le disque défaillant réparé ou remplacé, il faut reconstruire l'ensemble RAID 5 en reconstituant les données et les bandes de parité pour les écrire sur le disque remplaçant.

Les opérations de lecture peuvent être très rapides, car effectuées simultanément par les différents contrôleurs disques. Les écritures peuvent être ralenties, à cause du calcul et de l'écriture de la bande de parité.

L'espace de stockage utile d'un ensemble RAID 5 est égal à la capacité du plus petit des disques, multipliée par le nombre de disques qui le composent, moins 1 à cause des bandes de parité et moins 2 s'il y a un disque de secours (*spare*).

Avantages :

– Tolérance de panne, limitée à un disque. Pendant la défaillance du disque, il n'y a plus de tolérance de panne, sauf avec un disque de secours.

Inconvénients :

– Une partie de l'espace disque n'est pas utilisable pour les données.
– Les performances en écriture peuvent être impactées, à cause du calcul de la parité.

2.2 Configuration du RAID

Le pilote md est un module du noyau qui prend en charge le RAID logiciel sur un ensemble de périphériques de stockage, disques durs complets et/ou partitions de disques durs.

La commande `mdadm` permet de configurer des volumes RAID et de les gérer. Elle fait partie du paquet `mdadm`.

2.2.1 Création d'un volume RAID

Un volume RAID est composé de plusieurs espaces de stockage, qui peuvent être des disques durs entiers ou des partitions de disque dur.

La création d'un volume RAID se fait par l'option `-C` de la commande `mdadm`. Il faut spécifier le nom du nouveau volume ou son numéro, le niveau de RAID à mettre en œuvre et la liste des espaces de stockage à lui allouer.

Remarque

Le fichier de configuration de la commande, généralement `/etc/mdadm/mdadm.conf`, est devenu facultatif dans les versions récentes et n'est pas créé à l'installation du paquet.

Syntaxe

```
mdadm -C FicSpecVol -l|--level=Niveau -n|--raid-devices=NbDevRaid
[ -x|--spare-devices=NbSecours ] FicSpec1 ... FicSpecN
```

Principaux paramètres

`-C FicSpecVol`	Fichier spécial du volume RAID créé.
`-l|--level=Niveau`	Niveau de RAID.
`-n|--raid-devices=NbDevRaid`	Nombre d'espaces de stockage actifs.
`-x|--spare-devices=NbSecours`	Nombre d'espaces de stockage de secours.
`FicSpec1 ... FicSpecN`	Espaces de stockage.

Description

L'option `-C` crée un nouveau volume RAID. Le fichier spécial qui lui sera associé, `FicSpecVol`, est généralement de la forme `/dev/mdX`, `X` étant un chiffre, mais ce n'est pas obligatoire.

L'option `-n` spécifie le nombre d'espaces de stockage à utiliser pour le volume. Il doit être égal ou supérieur au nombre d'éléments de la liste `FicSpec1 ...`, `FicSpecN` moins le nombre d'espaces de stockage de secours, indiqué par l'option `-x`.

Une fois créé, le volume RAID peut être immédiatement utilisé. Il est vu comme un périphérique en mode bloc, on peut donc y créer un système de fichiers ou en faire un volume physique LVM.

Exemples

On utilise deux partitions de disques durs, `/dev/sda4` et `/dev/sdd1`, pour créer un volume `RAID` de niveau `1` (miroir). Comme les partitions contiennent des systèmes de fichiers et sont de tailles différentes, la commande affiche un avertissement et demande une confirmation :

```
mdadm -C /dev/md0 -l 1 -n 2 /dev/sda4 /dev/sdd1
mdadm: /dev/sda4 appears to contain an ext2fs file system
       size=5237760K  mtime=Thu Jan  1 01:00:00 1970
mdadm: Note: this array has metadata at the start and
   may not be suitable as a boot device.  If you plan to
   store '/boot' on this device please ensure that
   your boot-loader understands md/v1.x metadata, or use
   --metadata=0.90
mdadm: largest drive (/dev/sdd1) exceeds size (5232640K) by more than 1%
Continue creating array? y
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
```

Le volume RAID est créé :

```
ls -l /dev/md0
brw-rw----. 1 root disk 9, 0 10 mars  18:08 /dev/md0
Le fichier /dev/md0 est un fichier spécial bloc.
```

La commande `blkid` donne des informations sur les deux espaces de stockage composant le volume RAID :

```
blkid /dev/sda4 /dev/sdd1
/dev/sda4: UUID="760b8ab1-9041-5cd1-7e1e-1308fa7e75fd" UUID_SUB="679cb515-
2f5c-6078-6829-4b9f0f928907" LABEL="beta:0" TYPE="linux_raid_member"
PARTUUID="12deb3a0-04"
/dev/sdd1: UUID="760b8ab1-9041-5cd1-7e1e-1308fa7e75fd" UUID_SUB="e77f5f03-
448a-1b48-feb6-9bedc62e40b5" LABEL="beta:0" TYPE="linux_raid_member"
PARTUUID="c3072e18-01"
```

Les deux partitions sont de type RAID Linux et ont reçu un label `beta:0`, `beta` étant le nom de la machine.