


Partitions

- Objet : Partitions
- Suivi :
 - Création par  [smolski](#) le 26/01/2010
- Commentaires sur le forum : [C'est ici^{1\)}](#)

Le processus de partitionnement d'un disque est la création d'une partition mais celle-ci ne peut être écrite que sur un système qui lui est compatible. Un disque doit donc d'abord être compatible avec l'OS d'un système afin que les outils propres à ce système le partitionnent ensuite.

Exemple : Sur la commande `lsblk` :

```
lsblk
```

le retour :

```
sda      8:0    0 223,6G  0 disk
├─sda1   8:1    0 207,6G  0 part /
├─sda2   8:2    0      1K  0 part
└─sda5   8:5    0   16G   0 part [SWAP]
sdb      8:16   0   1,8T   0 disk
sdc      8:32   0 298,1G  0 disk
└─sdc1   8:33   0 298,1G  0 part /media/spagnol/RAPIDE DRIV
```

Indique les disques `sda` `sdb` et `sdc`, ainsi que leurs partitionnements respectifs avec un chiffre en plus pour chacune de ces partitions.

C'est dû à la structure de la table de partition "DOS/MBR" : - 4 entrées de partitions dites "principales" - un nombre variable de partitions "logiques" Les partitions logiques sont créées dans une partition principale dite "partition étendue".

Les numéros 1 à 4 sont réservés aux 4 entrées de partitions principales (dont la partition étendue), même si elles ne sont pas toutes occupées. Les numéros 5 et plus sont réservés aux partitions logiques.

Cette distinction entre les partitions principales et logiques n'existe pas dans le format GPT. Toutes les partitions sont équivalentes à des partitions principales.

Voir sur le forum :

- <https://debian-facile.org/viewtopic.php?pid=256435#p256435>
- <https://debian-facile.org/viewtopic.php?pid=277103#p277103>

Partition étendue

Pourquoi une partition étendue "ça pue" ?
Parce que c'est compliqué et fragile.

Les partitions logiques sont définies comme une liste chaînée. En fait, pour chaque partition logique, il y a une partition étendue cachée emboîtée dans la précédente. Il arrive que la chaîne soit rompue (→ partitions perdues), ou fasse une boucle (→ nombre de partition infini).



Toutes ces tables de partition étendues, visibles ou emboîtées/cachées, ont leurs propres tables de partition, dispersées un peu partout sur le disque.

Pour sauvegarder la table de partition complète, on ne peut pas se contenter de sauvegarder le contenu du MBR du disque, ni même l'EBR de la première partition étendue visible lorsqu'il y a plus d'une partition logique.

Aussi, la création de partitions logiques cause l'écriture de tables de partitions étendues un peu n'importe où sur le disque, ce qui peut écraser des données qu'on chercherait à récupérer ou à l'inverse rend plus difficile de les retrouver.

Alors qu'avec des partitions primaires ou le format GPT, il y a une seule table de partition située dans à un emplacement bien défini.

La numérotation des partitions logiques n'est pas stable et peut changer quand on crée ou supprime une partition logique.

Ce n'est normalement pas gênant si on utilise les [UUID](#) ou [LABEL](#) pour référencer les partitions dans [/etc/fstab](#), mais cela peut l'être pour [GRUB](#) lorsqu'il référence la partition contenant [/boot/grub](#).

Merci à **raleur** d'avoir détaillé cet aspect sur le forum là :

- <https://debian-facile.org/viewtopic.php?pid=290090#p290090>

Important

Pour obtenir les meilleurs résultats, vous devriez toujours utiliser un programme de partitionnement qui respecte les spécifications de la table des partitions et celles des systèmes de fichiers que vous voulez manipuler.

Par exemple, vous devriez créer/modifier :

- des partitions **ntfs** avec les outils que prévoit Microsoft à cet effet ;
- des partitions **hfs** et **hfs+** sur une table des partitions utilisée par MacOS X, avec les outils de MacOS X ;
- des partitions GNU/Linux avec [cfdisk](#) pour créer les tables, puis les outils des paquets [e2fsprogs](#) et [util-linux](#) pour les formater en **ext2**, **ext3**, **ext4**

Un brin d'explications

Avant d'aller plus loin, genre "*Un pont trop loin*"... :

- [Partitions - Détail](#). Le nommage des partitions et leurs fonctions.
- [Partition - LVM](#) Partitionnement adaptable.
- [Commande tune2fs](#) Ajustement des paramètres pour les systèmes de fichiers ext2/ext3/ext4.
- [Commande e2label](#) Modification du label d'un disk ext2/ext3/ext4.

Table de partitions

- [Commande Linux cfdisk](#)
- [Commande Linux sfdisk](#)
- [Commande Linux fdisk](#)
- [Commande Linux gdisk](#)

Formatage

Formater graphiquement

- [Utilitaire de disque](#)
- [gparted](#)

Formater en ligne de commande

- [Commande Linux mkfs](#)

Réparer une partition

- [fsck](#)

Disques et données

- [Sauvegarde ou Récupération des données](#)

BTRFS

- [BTRFS:Présentation et bases](#)
- [Installation de Debian dans un subvolume BTRFS](#)
- [Gestion des sauvegardes de subvolume BTRFS](#)

Sécurité des données

Voici un exemple appliqué pour augmenter au possible la sécurité de la pérennité des données.

Côté disque dur, choisir des numéros de série très différents.

Si tu n'as qu'un revendeur, tu étales tes achats, sinon tu en prends un différent sur plusieurs sites d'achat (quoique ton revendeur unique peut le faire à ta place).

Sinon il faut faire en sorte d'avoir un temps de fonctionnement différent pour les disques. Genre j'ai les mêmes hd sur du raid1 mais j'ai débranché temporairement l'un des 2 disques pour leur mettre

~1000h de décalage.

Raid

Le RAID avec redondance (dont pas RAID0, linear) fournit de la disponibilité.

- En cas de défaillance physique d'un disque, le système peut continuer à tourner et les données restent accessibles.
 - Mais il ne protège pas contre l'effacement accidentel ou malveillant, la corruption de données suite à un plantage ou un arrêt sauvage... bref les erreurs logiques. Contre cela, la seule solution est une sauvegarde.
- En revanche une sauvegarde ne fournit pas de disponibilité puisqu'il faut restaurer les données pour qu'elles soient à nouveau disponibles.
- On ne peut pas faire du RAID avec un NAS. Un NAS exporte des systèmes de fichiers en réseau, alors que le RAID utilise des périphériques de stockage.
- Le RAID pourrait utiliser du SAN, qui exporte des périphériques de stockage en réseau.
- Par contre un NAS peut faire du RAID avec ses propres disques, ou des disque réseau en SAN.

Depuis **raleur** sur le forum là :

- <https://debian-facile.org/viewtopic.php?pid=249599#p249599> 😊

Détail augmenté

1. Découpage du disque en plusieurs partitions et
2. réassemblage au niveau LVM pour minimiser le temps de reconstruction du RAID en cas de perte d'une patoche.

En plus ça permet d'utiliser un disque de secours²⁾ de taille plus petite que le disque initial.

Sur un 2To je fais 4 patoches de 500Go.

Par ailleurs j'ai un disque de secours de 1To que je découpe en 2x500Go, ce qui permet de remplacer une patoche du 2To par une du 1To en cas de pépin et c'est plus rapide de synchroniser 500Go que 2To en cas de problème RAID.

Le disque de secours est débranché. Je le branche de temps à autre une journée pour ne pas qu'il reste trop longtemps sans tourner.



C'est juste une solution d'attente le temps de commander un autre disque et pouvoir remplacer celui qui commence à déconner (parce que même si ça ne concerne initialement qu'une partition, je ne parierai pas que le problème reste longtemps confiné à une patoche.

Source : **m31z0nyx** 😊

Lien utile

- [fstab](#)

Lien vers le forum

- [partition racine - réduire, déplacer et/ou supprimer les partitions suivantes pour libérer de l'espace disque](#)
- [transférer le contenu d'un disque dans un autre disque plus grand](#)

¹⁾

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

²⁾

disque supplémentaire, non utilisé par défaut

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/doc:systeme:partitionnement>

Last update: **19/04/2023 18:59**

