Partition

- Objet : Les partitions
- Niveau requis :
  débutant, avisé
- Commentaires : *Les partitions en détail - Les systèmes de fichier usuels sous GNU/Linux.*
- Débutant, à savoir : Utiliser GNU/Linux en ligne de commande, tout commence là !
- Les Droits. Pour connaître en détail les droits sur les fichiers et les répertoires...
- Le disque dur en détail
- Suivi :
  à-compléter
  - Création par smolski le 14/05/2010
  - Testé par smolski le 20/09/2013
- Commentaires sur le forum : C'est ici[1)]

Préléminaire

Une unité de stockage (que ce soit un disque dur interne ou externe, une clé USB, un CD-ROM ou DVD-ROM, une disquette, etc.), pour être utilisée par un ordinateur, doit être formatée.

Cela signifie qu'on doit lui assigner un système de fichier[2)], indiquant la structure et le nom des répertoires et fichiers et éventuellement les droits correspondants à chacun.

Système de fichier

Il existe plusieurs types de FS[3)].

Nota

Dans cette partie vous lirez entre crochets [...] la façon de saisir le nom du système de fichier à écrire dans le fichier /etc/fstab.

Le Ext2FS [ext2]

Le système Ext2FS (pour second extented filesystem) a été le standard de facto sous GNU/Linux. Il est désuet à ce-jour ! Il s'est caractérisé par une très faible fragmentation, une bonne tenue à la charge et il a été très fiable. Il permettait une gestion des droits en lecture, en écriture et en exécution par utilisateur et par groupe d'utilisateurs.

Le Ext3FS [ext3]

Ce système est une reprise de l'ext2fs [ext2] avec un historique de journalisation. Il est très fiable

également.

Le Ext4FS [ext4]

Ce système bien plus ***rapide*** dans son utilisation est installé par défaut.

Le BTRFS

Ce système est l'avenir et il est pressentie pour remplacer l'ext4. Il est en cours de développement. Bien que le format du système de fichier soit considéré comme stable par les développeurs [4)], les outils qui permettent de le manipuler sont eux en développement.

- BTRFS:Présentation et bases
- Installation de Debian dans un subvolume BTRFS
- Gestion des sauvegardes de subvolume BTRFS

Le FAT 16 [fat]

Le système FAT 16 (pour File Allocation Table)[fat] est le système de fichiers standard de Microsoft™ utilisé dans ses systèmes d'exploitation MS-DOS et Windows® jusqu'à Windows 95. Il ne permet pas de gestion des droits, il n'accepte pas de fichiers qui ont une taille supérieure à 2 Go et est caractérisé par une fragmentation importante. Il est reconnu presque partout et est souvent présent sur les clefs USB.

Le FAT 32 [vfat]

C'est le système de fichiers développé par Microsoft™ pour ses systèmes d'exploitation. Dans ce cas-ci, la limitation pour la taille des fichiers est de 4 Go.
Il ne possède pas lui non plus de gestion des droits. Sa fragmentation est aussi importante. Il est très courant également.

Bien que le système vfat ne permette malheureusement pas de gérer les droits d'accès aux fichiers, il a l'avantage d'être lisible (presque) partout.
C'est donc un bon choix si l'on souhaite rendre des données accessibles sous plusieurs systèmes (partition de partage entre systèmes).

Pour installer les outils **fat** permettant de le modifier, créer, changer le label, etc... :

Installer dosfstools :

```
apt-get install dosfstools
```

Le SWAP [swap]

Le SWAP est la partition sur le disque système qui sera sollicitée par les applications actives pour

sauvegarder les données les plus anciennes ou les moins demandées lorsque la RAM[5] est saturée. L'OS s'en charge totalement.

De ce fait, comme un programme ne peut travailler qu'avec les données en mémoire vive[6], si celles sauvegardées dans le SWAP sont requises, elles devront d'abord retourner dans la RAM par le chemin inverse et cela ralenti la machine !

Préambule

La RAM est principalement utilisée pour faire du cache et lors de l'hibernation, le cache n'est pas sauvegardé.
Toutefois, lors de l'hibernation, la RAM peut être compressée, donc on peut moduler la préconisation qui suit en fonction de ces éléments.

A partir de 2Go de RAM, il suffit d'avoir un SWAP légèrement plus grand que la taille de la RAM.

Le SWAP n'est utilisé par le système qu'en dernier recours :

1. quand il n'y a plus de RAM disponible
2. et lorsque vous mettez votre PC en hibernation.

Toutefois, avec la valeur de vm.swappiness par défaut (60) : même s'il y a assez de RAM disponible le noyau peut décider de swapper des pages inactives et de les réaffecter par exemple au cache disque qui en fera un usage plus efficace. Voir sur le forum : https://debian-facile.org/viewtopic.php?pid=167091#p167091 les indications portées par raleur.

D'où la nécessité d'avoir toujours un SWAP de taille supérieure à la quantité de RAM, sinon méchant plantage de l'hibernation

D'où aussi certaines recommandations de carrément supprimer le SWAP si tu n'utilises pas cette option.

Le total d'espace alloué au SWAP peut donc être diminué par rapport à l'utilisation propre de son pc et des applications misent en chantier habituellement.

Nota :

À partir de 2-3 Go de RAM, le SWAP ne sert plus de fichier d'échange mais uniquement (pour une utilisation normale) de fichier d'hibernation.
Ainsi, dans des conditions normales d'utilisation, l'espace nécessaire en SWAP pour l'hibernation est très faible, c'est une notion à anticiper qui va varier avec le temps...

Recommandations

1. Si le PC a **plus de 3 Go de RAM**, il est excessif d'allouer d'avantage de SWAP que de RAM.
2. Si le PC a **moins de 2 Go**, allouer 2Go de RAM peut être une sécurité, par exemple si le PC a entre 2 et 3 Go de RAM, allouer 2 Go de SWAP semble suffisant pour une utilisation habituelle.
3. **À partir de 3 Go le SWAP = 2/3 ram** devrait en gros suffire dans la plupart des cas, sauf pour le gars qui hiberne après avoir scanné une photo de sa coquine en 12800000×102400000.

Dieu nous garde de ne jamais oser hiberner ainsi coquin ou coquine, même sur son PC, sans blague !

Lien utile sur le forum

swap : de son intérêt et de son usage :

- https://debian-facile.org/viewtopic.php?id=22395

Remerciements

Merci à **Blacksad** et **Herbert west** d'avoir partagé cette indication sur le forum là :

- Forum debian-facile

Et à la vigilance du **captnfab** *qui du haut de son timon nous apporte les précisions supplémentaires.*

swap - Comment ça marche ?

Nsyo a écrit :
j'ai toujours lu et entendu que le swap avait pour rôle d'être une extension de la ram au cas ou celle-ci serait pleine.

raleur :
C'est une simplification grossière. D'autre part chaque système d'exploitation gère le swap d'une façon qui lui est propre. Le swap n'est pas une extension de la RAM. Il faut bien comprendre que instructions exécutées par les processus et les données manipulées par ces instructions à un instant donné doivent être présentes en RAM.

Nsyo a écrit :
Auquel cas c'est sur le disque qu'écrit le noyau.

raleur :
Pas forcément. Le swap peut être en mémoire. Quel intérêt ? Il peut être compressé (via zram).

Nsyo a écrit :
vm.swappiness par défaut est à 60 chez Debian, donc écriture sur le disque à partir de 40% d'occupation de la ram.

raleur :
60 est la valeur par défaut du noyau Linux, elle n'est pas spécifique à Debian. Je répète que cette valeur ne correspond pas du tout à un seuil d'occupation de la mémoire. Il suffit d'examiner la sortie de la commande **free** pour le voir. Cela n'aurait aucun sens de laisser 60% par défaut de la mémoire libre (donc inutilisée), cela reviendrait à avoir 60% de mémoire en moins !

La mémoire est faite pour être utilisée. TOUTE la mémoire. La mémoire libre est de la mémoire gaspillée.

La mémoire est divisée en pages. En simplifiant à l'extrême, les pages mémoire se répartissent en deux catégories :

1. les pages “anonymes” contenant les données des processus,
2. les pages de cache contenant des données liées au système de fichiers, qui se répartissent elles-même en plusieurs catégories (pagecache, dentries, inodes) mais je n'entrerai pas dans les détails.

Toutes les données lues ou écrites sur le système de fichiers sont mises en cache en mémoire. En lecture, cela permet d'accèder plus rapidement aux données déjà en cache sans avoir besoin de les recharger depuis le disque. En écriture, cela permet de ne pas attendre que les données soient effectivement écrites sur le disque.

Quand le noyau a besoin d'allouer de la mémoire et que la mémoire libre est en deçà d'un certains seuil (qui n'a rien à voir avec vm.swappiness, je le répète), il va lancer une procédure “d'éviction” pour libérer de la mémoire. L'éviction consiste à décharger des données de la mémoire. Les pages candidates à l'éviction sont dans les deux catégories citées précédemment : les pages anonymes et les pages de cache. Les pages de cache contiennent des données qui sont déjà stockées sur disque ou qui ont vocation à y être stockées (écritures en attente). Les données déjà stockées sur disque peuvent être immédiatement supprimées, et celles en attente d'écriture sur disque peuvent être écrites puis supprimées.

Les pages anonymes, en revanche, ne sont pas associées au système de fichiers. Avant de les décharger de la mémoire, elles doivent être écrites dans un espace du disque appelé “espace d'échange” ou swap. La encore, les pages qui ont été écrites dans le swap puis rechargées et utilisées sans être modifiées peuvent être immédiatement supprimées, alors que les pages qui n'ont jamais été écrites dans le swap ou qui ont été modifiées depuis leur précédente écriture doivent être écrites sur disque avant d'être supprimées.

On peut constater une similitude des mécanismes d'échange entre les pages anonymes et le swap d'une part, et les pages de cache et le système de fichiers d'autre part. D'une certaine façon on pourrait dire que les pages anonymes sont le cache du swap.

La valeur de vm.swappiness n'influe que sur la préférence du noyau à évincer des pages de cache (si valeur basse) ou anonymes (si valeur élevée). Mais ce n'est pas le seul critère qui entre en compte. Le noyau peut préférer swapper une page anonyme qui n'a pas été utilisée depuis longtemps plutôt qu'une page de cache qui a été utilisée récemment et a donc une plus grande probabilité d'être utilisée à nouveau.

Lien sur le forum :

- https://debian-facile.org/viewtopic.php?pid=277412#p277412

Merci à **Nsyo** et **raleur** pour la qualité de ces échanges.

A voir aussi :

- Le système GNU/Linux

Le NTFS [ntfs]

Le NTFS (pour New Technology FileSystem) est le système de fichier standard dans les systèmes d'exploitation de Microsoft qui a été introduit avec Windows® NT pour les milieux d'entreprises.

Il est aussi devenu le standard pour nos ordinateurs personnels depuis Windows XP.

Il est caractérisé par une gestion des droits, une fragmentation moins importante que pour les systèmes FAT 16 et 32 et introduit une journalisation partielle et rudimentaire.

La dernière version du projet **ntfs-3g** ouvre la possibilité d'écrire et de lire sans danger sur une partition **NTFS** à partir d'un système GNU/Linux.

Par défaut le montage d'un système de fichiers NTFS avec ntfs-3g se fait avec l'option umask=0, qui équivaut à des permissions 777, soit accès complet pour tout le monde.

Pour installer ntfs-3g :

```
apt-get install ntfs-3g
```

Tout simplement...

Pour monter un disque dur ntfs dans le `/etc/fstab`, il faut indiquer le type **ntfs-3g** au lieu de **ntfs** seul.

Exemple :

```
# Montage disque ntfs
UUID=xxxxxxxxxxxxxxx       /media/ntfs           ntfs-3g     defaults     0
0
```

Le XFS [xfs] et le ReiserFS [reiserfs]

Ce sont des systèmes de fichiers journalisés, comme l'Ext3FS [ext3]. Aujourd'hui réputés robustes (i.e. utilisables sur des systèmes en production), autorisent la gestion des droits. Très faible fragmentation.

Le ReiserFS

[reiserfs] a des meilleures performances que les autres systèmes de fichier lorsqu'il gère une multitude de fichiers de faible taille.

CDROM

C'est en général l'ISO 9660, mais d'autres systèmes utilisent des formats particuliers : HFS pour le Mac, par exemple.

Dans une installation standard, un démon (gnome-volume-manager) doit assurer le montage automatique des systèmes de fichier.

Dès qu'un disque dur externe en USB (par exemple) est détecté, Nautilus, le gestionnaire de fichier, ouvre une fenêtre affichant le contenu de cette nouvelle unité de stockage. Une icône permettant l'ouverture de cette fenêtre devrait également apparaître sur votre bureau.

Si ce n'est pas le cas, utiliser la commande mount.

Le système voit les périphériques au travers d'une arborescence située dans le répertoire /dev/*.

Les blocks

La taille des blocks est définie automatiquement selon

1. le système de fichier
2. le CPU (architecture)

Ensuite, elle est modifiable en fonction de l'utilisation mais bien souvent il vaut mieux laisser le *mode automatique* faire le bon choix du paramétrage du système de fichier.

Par exemple avec Ext3, la taille des blocks représente la plus petite segmentation du disque géré par le FS.

Cette valeur peut varier de 1024, 2048 à 4096 octets.

De ce fait plus la taille des blocks est grande, plus il y aura de pertes d'espace disque, mais meilleurs seront les performances moins d'I/O[7)].

Ext4 quand à lui se rapproche du système xfs, qui fonctionne différemment au niveau de la gestion de ses blocks , il prend en charge la notion de zone étendue (extent).
Un extent est une zone du disque contiguë qui peut être de très grande capacité (Go).
L'utilisation d'extent réduit la fragmentation et améliore les performances lors de l'utilisation de gros fichiers.
Enfin sa taille des blocks est comprise entre 512 octets et 64 Ko (limité à 4Ko sur x86).

Chaque FS posséde ses avantages et inconvénients, à voir donc selon l'utilisation.
C'est un domaine passionnant qui risque d'évoluer encore rapidement avec l'arrivée de SSD.

Depuis **stopher** sur le forum ici :

- Les blocks facilement.

UUID

L'UUID est l'identification d'une partition formatée sur un disque dur.

Il est d'usage maintenant d'identifier les partitions individuellement par leur **UUID** plutôt que par leur emplacement sur le disque.
L'avantage immédiat est qu'en cas de modification des emplacements matériels des disques (ou de crash de l'un d'eux) l'**UUID** des partitions des autres disques resteront accessibles !

Voir :

- fstab
- les UUID des partitions

Liste des partitions

Pour avoir la liste des partitions d'un disque dur, vous pouvez utiliser en root[8)] l'outil fdisk ou mieux blkid.

Formater

UTILISEZ prioritairement :

- **les outils libres** pour formater les systèmes de fichier utilisés par le libre.
- **les outils privateurs** pour formater les systèmes de fichier utilisés par le privatif,

Dans un terminal

- mkfs. Formater avec une ligne de commande.

En graphique, sur le bureau

- gparted. Formater sur le bureau avec Gparted.
- gnome-disk-utility. Formater sur le bureau Gnome avec Utilitaire disque.

fiche : *Pour effacer complètement les données, un formatage ou la suppression de la partition Windows avec Gparted suffit-il ?*
Mieux vaut-il utiliser la commande dd avec :

```
dd if=/dev/zero of=/dev/sdX bs=512 status=progress
```

bs=512 compte tenu du résultat de la commande fdisk -l

```
fdisk -l
```

```
Unités : secteur de 1 × 512 = 512 octets
Taille de secteur (logique / physique) : 512 octets / 512 octets
```

La commande :

```
wipefs -a /dev/sdX
```

est-elle suffisante avant de créer une nouvelle table de partition ?

raleur : Le formatage normal ne réinitialise que les méta-données du système de fichiers, sans effacer le contenu des fichiers. On peut le récupérer avec des outils comme photorec.

- La suppression de la partition n'efface même pas les méta-données du système de fichiers, donc il est encore plus facile de récupérer les données en retrouvant la position de la partition avec des outils comme testdisk ou gpart.
- wipefs sur le disque entier ne fait qu'effacer les quelques octets de “signature” identifiant la table de partition, laissant intactes les méta-données et les données.
- dd efface réellement les données. Par contre il vaut mieux spécifier une taille de bloc d'au moins 4K car une taille de 512 peut affecter négativement la vitesse.

fiche : *Donc, il faut utiliser cette commande :*

```
dd if=/dev/zero of=/dev/sdX bs=4096 status=progress
```

Elle sera efficace sur n'importe quel disque dur (ancien, récent) ?

raleur : Oui.
On peut utiliser une taille de bloc encore plus grande : 128K, 1M…

Lien au forum

- https://debian-facile.org/viewtopic.php?id=24964

Merci à **fiche** et **raleur** pour ces explications pratiques et détaillées.

Et maintenant

- **Montage** automatique des partitions : le fichier fstab *et quelques ficelles...*
- **Réparer** une partition : * fsck
- **Créer** les tables où seront implantées les partitions : cfdisk

Liens sur le forum

swap en dualboot

Voir sur le forum ici :

- https://debian-facile.org/profile.php?id=4112

1)

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

2)

FS

3)

systèmes de fichier

4)

https://btrfs.wiki.kernel.org/index.php/Main_Page#Stability_status

5)

mémoire vive

6)

la RAM

7)

écriture/lecture

8)

su