Hard Disk Upgrade Mini How-To

Yves Bellefeuille, *yan@ottawa.com*
traduction Eric Cano, *Eric.Cano@cern.ch* Version 1.0, 31 janvier 1998, traduction mars 1998

Comment copier un système **Linux** d'un disque dur à un autre

1 Introduction

Récemment, j'ai remplacé mon petit disque dur de 249 Mo par un disque plus grand. Je voulais transférer mon **Linux** entier, `LILO` inclus, de l'ancien au nouveau disque. Voici comment je l'ai fait.

Dans les explications qui suivent, j'utilise "`/dev/hda`" pour désigner mon "ancien" disque, "`/dev/hda1`" désigne mon ancienne partition **Linux**. "`/dev/hdb`" désigne le nouveau disque, et "`/dev/hdb1`" la nouvelle partition **Linux**.

Je suppose donc que **Linux** est sur la première partition du premier disque. Adaptez ceci à votre propre configuration.

Ce document est basé sur mon propre système, qui fait tourner une **Red Hat 4.2**, et j'ai testé toutes les commandes qui suivent sous cette distribution. Je les ai aussi testées sous une **Debian 1.3.1** et une **Slackware 3.3**, et j'indique quelques différences à prendre en compte si vous utilisez ces distributions.

Si ces commandes ne marchent pas correctement sous votre système, faites le moi - *NdT l'auteur, pas le traducteur* - savoir, en indiquant la version de **Linux** que vous utilisez.

2 Installez les deux disques dans votre système

Comme les systèmes modernes peuvent accepter quatre périphériques "EIDE" sur le contrôleur de disque dur, vous ne devriez pas avoir de problème pour installer en même temps les deux disques sur votre système, même si vous avez d'autres périphériques EIDE. Les disques durs et les lecteurs de CD-ROM sont des périphériques EIDE typiques. Les lecteurs de disquettes et de bandes sont connectés le plus souvent sur le contrôleur de disquettes plutôt que sur le contrôleur de disque dur.

Les contrôleurs SCSI sont plus souples et peuvent accepter sept périphériques. Si vous êtes assez chanceux (et riche) pour avoir un contrôleur SCSI, vous savez sans doute déjà cela, et vous savez probablement lesquels, parmi vos périphériques, sont SCSI ! Pour plus d'information, voyez le

SCSI How-To `http://www.freenix.fr/linux/HOWTO/SCSI-HOWTO.html` .

Même les systèmes les plus vieux peuvent accepter deux périphériques sur le contrôleur de disque dur, donc vous pouvez toujours installer deux disques en même temps. Toutefois, si vous avez un autre périphérique installé en plus de votre disque dur, (par exemple, si vous avez un disque dur et un CD-ROM) vous devrez retirer l'autre périphérique pour pouvoir installer le nouveau et l'ancien disque dur en même temps.

Vous devez configurer les disques comme "maître" ou "esclave" en plaçant les cavaliers de façon appropriée. Vous trouverez souvent des informations sur leur configuration sur les disques eux mêmes ; sinon, consultez les manuels ou les fabricants de vos disques.

Vous devez aussi informer le BIOS de la présence des disques et de leur "géométries". En général vous entrez dans le Setup du BIOS en pressant une touche durant le démarrage du système. Voilà la marche à suivre avec quelques BIOS courants :

American Megatrends (AMI)

Touche *Suppr* pendant l'autotest de démarrage.

Award

Ctrl-Alt-Echap

Compaq

Touche *F10* après que le carré soit apparu dans le coin en haut à droite de l'écran à la mise en route.

Dell

Ctrl-Alt-Entrée

DTK

Touche *Echap* pendant l'autotest de démarrage.

IBM PS/2

Ctrl-Alt-Suppr, puis *Ctrl-Alt-Inser* quand le curseur est dans le coin en haut à droite de l'écran.

Phoenix

Ctrl-Alt-Echap, ou *Ctrl-Alt-S*, ou *Ctrl-Alt-Entrée*

Autres...

De nombreux autres systèmes ont besoin d'une disquette d'*installation* ou de *référence*

(Je - *NdT : l'auteur* - suis intéressé par des informations sur les autres BIOS pour les inclure dans cette liste.)

Redémarrez le système et loguez-vous `root`.

3 Démontez les partitions non-Linux

Certains aiment monter les partitions d'autres systèmes d'exploitation (**DOS**, **Windows**, **OS/2**, etc.) pour pouvoir les utiliser sous **Linux**. Ces partitions devraient être créées et copiées sous leur propre système d'exploitation, et vous devriez les démonter avant de copier votre partition **Linux**. Par exemple, si vous avez une partition montée en `/dos`, vous devez la démonter avec cette commande :

```
umount /dos
```

Notez que la commande est "`umount`", avec le premier "n" qui manque par rapport au mot "*unmount*" - *NdT "démonter" en anglais.*

4 Partitionnez le nouveau disque

Utilisez la commande :

```
fdisk /dev/hdb
```

pour partitionner le nouveau disque.

Pour plus d'informations sur le partitionnement, voyez l' *Installation How-To* `http://www.freenix.fr/linux/HOWTO/Installation-HOWTO.html` et le *Partitionning Mini How-To* `http://www.freenix.fr/linux/HOWTO-vo/mini/Partition` .

Si votre disque a plus de 1024 cylindres, voyez le

Large Disk Mini How-To `http://www.freenix.fr/linux/HOWTO-vo/mini/Large-Disk` . En bref, vous devriez installer tous les fichiers requis pour démarrer **Linux** dans les 1024 premiers cylindres. Une façon de s'en assurer est de créer une petite partition (1Mo ou 2Mo) juste pour le répertoire `/boot` au début du disque. (**spécificité Slackware** : comme le noyau est plutôt en `/vmlinuz` qu'en `/boot/vmlinuz`, vous devez mettre les répertoires `/` et `/boot` dans cette partition.)

Les partitions pour les systèmes autres que **Linux** devraient être créées en utilisant leur propre `fdisk` plutôt que celui de **Linux**.

5 Formatez le nouveau disque.

Utilisez la commande suivante pour formater le nouveau disque :

```
mkfs.ext2 /dev/hdb1
```

Pour rechercher les mauvais blocs (défauts physiques) sur le disque, ajoutez l'option `-c` juste avant `"/dev/hdb1"`.

(*Note :* Contrairement à ce qu'indiquent les pages de manuel, la commande `"mkfs -t ext2 /dev/hdb1"` ne recherche pas les blocs défectueux, sous aucune de la **Red Hat**, **Debian** ou **Slackware**.)

6 Montez le nouveau disque

Créez un répertoire là où vous monterez le nouveau disque, par exemple `/new-disk`, et montez le :

```
mkdir /new-disk
mount -t ext2 /dev/hdb1 /new-disk
```

7 Copiez les fichiers de l'ancien disque au nouveau

Il faut reproduire complètement la structure du disque, liens inclus.

Toutefois, il ne faut *pas* copier le répertoire `/new-disk`, puisque ça reviendrait à recopier le nouveau disque sur lui-même !

De plus, il faut copier le répertoire `/proc` sur le nouveau disque, mais pas son contenu : `"/proc"` est un système de fichiers "virtuel" qui ne contient pas de vrais fichiers, mais plutôt des informations sur les processus qui tournent sur le système.

Voici quatre façons de copier un vieux disque sur un nouveau. Ça peut prendre du temps, d'autant plus que le disque est gros ou la mémoire peu importante. Vous pouvez vous attendre à copier 10Mo par minute, ou même plus.

Vous pouvez suivre la progression de la copie en utilisant la commande `"df"` depuis un autre terminal. Si vous êtes aussi bon public que moi, essayez `"watch df"` ou `"watch ls -l /new-disk"` pour voir un rapport mis à jour toutes les deux secondes ; utilisez *Ctrl-C* pour arrêter l'affichage. Soyez conscient que la commande "`watch`" va ralentir la copie.

1.
```
cp -ax / /new-disk
```

Ceci est la méthode la plus simple, mais ne fonctionnera que si votre système **Linux** est sur une seule partition. L'option `-a` préserve autant que faire se peut le système original. L'option `-x` restreint `cp` à un seul système de fichiers. Ceci est nécessaire pour éviter de copier les répertoires `/proc` et `/new-disk`.

2.
```
cd / && cp -a '/bin/ls -1A | egrep -v "^new-disk$|^proc$"' /newdisk
```

Ceci va à la racine puis copie tous les fichiers et répertoires sauf `/proc` et `/new-disk`. Notez que la première option après ls est le chiffre "1", et non la lettre "L" !

Cette commande devrait fonctionner en toutes circonstances.

3.
```
(cd / && tar cpf - . --exclude new-disk --exclude proc) | (cd
/new-disk && tar xpf -)
```

(Ecrire cette commande sur une seule ligne)

Ceci va dans le répertoire racine, "archive" tout sauf `/proc` et `/new-disk`, va dans `/new-disk` et "désarchive" tout là. Notez qu'il ne doit pas y avoir de slash ("/") avant ou après les noms de répertoire dans les options `-exclude`.

(*Note* : L'option `-l` ne marche pas ici, puisque `tar` recréerait les répertoires `/proc` et `/new-disk` même s'il ne copie pas leurs contenus. C'est pourquoi l'option `-l` de `tar` n'a pas le même comportement que l'option `-x` de `cp`.)

Cette méthode est quelque peu plus lente que les autres.

4.
```
cp -a /bin /boot /dev /etc /home /lib /lost+found /mnt /root /sbin
/tmp /usr /var /new-disk
```

(Ecrire la commande sur une seule ligne)

Le dernier répertoire, `/new-disk`, est la destination pour la commande `cp`. Tous les autres répertoires sont les sources. C'est pourquoi je copie tous les répertoires listés dans `/new-disk`.

Avec cette méthode, vous faites simplement une liste des répertoires que vous voulez copier. Ici j'ai indiqué tous mes répertoires à l'exception de `/proc` et `/new-disk`. Si vous ne pouvez utiliser aucune des méthodes pour une raison quelconque, vous pouvez toujours utiliser cette commande pour spécifier manuellement les répertoires que vous voulez copier.

Avec cette méthode seulement, s'il y a des fichiers dans le répertoire racine lui-même, vous avez besoin d'une autre commande pour les copier. En particulier, ceci est requis avec les **Debian** et **Slackware**, car ces distributions placent des fichiers dans le répertoire racine :

```
cp -dp /* /.* /new-disk
```

Après avoir utilisé une méthode parmi les quatre, vous devez aussi créer le répertoire `/proc` sur le nouveau disque :

```
mkdir /new-disk/proc
```

A ce point, vous pouvez, si vous le voulez, vérifier la structure des fichiers sur le nouveau disque :

```
umount /new-disk
fsck.ext2 -f /dev/hdb1
mount -t ext2 /dev/hdb1 /new-disk
```

Vous pouvez aussi utiliser le script suivant pour comparer les deux disques, et vous assurer que les fichiers ont été copiés correctement.

```
#!/bin/sh
cd /
for file in ‘/bin/ls -1A | egrep -v '^new-disk$|^proc$'‘
do
    find $file -xtype f -exec cmp \{\} /new-disk/\{\} \;
done
```

(**Spécificité Slackware** : une installation de base n'inclut pas les commandes "`cmp`" et "`diff`", alors vous ne pourrez pas lancer ces scripts si vous avez seulement installé les fichiers de base.)

Ceci compare seulement les fichiers normaux, et non les fichiers spéciaux associés à des périphériques (dans le répertoire `/dev`, les sockets, etc., car la commande cmp ne fonctionne pas correctement avec ceux-ci. Je - *NdT l'auteur* - serais intéressé par toute suggestion sur le moyen de vérifier ces fichiers "spéciaux").

8 Mettez à jour "/etc/fstab"

Si votre nouveau disque n'a pas les même partitions ou la même organisation que l'ancien, modifiez le fichier `/etc/fstab` en conséquence. Souvenez vous que ce fichier se trouve actuellement en `/new-disk/etc/fstab`.

Assurez vous que les partitions de disque dans la première colonne correspondent à l'organisation que vous aurez sur le nouveau disque, une fois que l'ancien disque aura été enlevé, et que vous ne monterez plus qu'une partition en "/", comme indiqué dans la seconde colonne.

9 Préparez LILO pour démarrer le nouveau disque

C'est le point compliqué. Je suppose que `LILO` est installé sur le bloc principal de démarrage (*master boot record, MBR*); ceci semble être la configuration la plus courante.

Vous voulez installer `LILO` sur ce qui est actuellement le second disque dur. Il est clair que `LILO` ne peut pas être *lancé* depuis le second disque dur; toutefois, la documentation de `LILO` anticipe le fait que vous vouliez *installer* `LILO` sur le second disque dur, par exemple si le premier disque dur doit être enlevé :

```
LILO ne peut etre stocke sur aucun des emplacements suivants :

- sur le second disque dur. (A moins que, pour des besoins de
sauvegarde, si le premier disque dur va etre retire ou desactive,
ou si un autre lanceur, qui est capable de charger les secteurs de
demarrage d'autres disques, est installe.)
```

Toutefois, la documentation n'explique pas la façon d'installer LILO sur le second disque dur si le premier va être retiré, et j'ai déduit après de nombreux essais qu'il n'est pas possible d'installer LILO sur le MBR du second disque dur et de le faire marcher du premier coup.

A la place, je suggère d'utiliser une disquette de démarrage pour démarrer sur le nouveau disque dur la première fois.

Insérez une disquette vide, formatez la, créez y un système de fichier et montez la :

```
fdformat /dev/fd0H1440
mkfs.ext2 /dev/fd0
mount -t ext2 /dev/fd0 /mnt
```

(**Debian seulement** : La commande "`fdformat`" n'est pas incluse dans l'installation de base chez **Debian**. Si vous n'avez pas cette commande, vous pouvez l'ignorer si la disquette est déjà formatée. Dans ce cas, vous devez rechercher les secteurs défectueux sur la disquette en ajoutant l'option "`-c`" après la commande "`mkfs.ext2`".)

(**Debian et Slackware seulement** : utilisez la commande "`fdformat /dev/fd0h1440`" avec un "h" minuscule)

copiez tous les fichiers de `/boot` sur la disquette :

```
cp -dp /boot/* /mnt
```

(**Slackware seulement** : Copiez le fichier `/vmlinuz` sur la disquette de boot; utilisez la commande "`cp /vmlinuz /mnt`".)

Créez un nouveau fichier `/mnt/lilo.conf` comme suit :

```
boot=/dev/fd0           # Installe LILO sur la disquette.
map=/mnt/map            # Emplacement du fichier de correspondance.
install=/mnt/boot.b     # Fichier a copier sur le secteur d'amorce.
prompt                  # Faire afficher l'invite "LILO boot:" par LILO.
timeout=50              # Lancer le systeme par defaut apres 5 secondes.
                        # (La valeur est en dixiemes de seconde.)
image=/mnt/vmlinuz      # Emplacement du noyau Linux sur la disquette.
    label=linux         # Etiquette du systeme Linux.
    root=/dev/hda1      # Emplacement de la racine sur le nouveau disque
                        # dur. Mettre en accord avec votre systeme.
                        # Notez que vous devez utiliser le nom du futur
                        # emplacement, une fois que l'ancien disque aura
                        # ete retire.
```

(**Debian seulement** : Sur la ligne "image", utilisez le vrai nom du noyau **Linux**. Par exemple avec la **Debian** 1.3.1, utilisez "`/mnt/vmlinuz-2.0.29`".)

Installez LILO sur la disquette de démarrage :

```
/sbin/lilo -C /mnt/lilo.conf
```

L'option "`-C`" dit à "`/sbin/lilo`" quel fichier utiliser.

Démontez la disquette :

```
umount /mnt
```

et lancez la procédure d'extinction du système.

10 Retirez l'ancien disque

Après avoir retiré l'ancien disque, n'oubliez pas de modifier les cavaliers de configuration du disque, et l'information du BIOS en rapport avec les changements.

11 Redémarrez le système, installez `LILO` sur le nouveau disque

Redémarrez le système depuis la disquette que vous venez de créer. Pour cela, vous devez modifier la séquence de démarrage de votre BIOS en "`A: C:`"

Faites tous les changement nécessaires dans le fichier `/etc/lilo.conf` et lancez `/sbin/lilo` pour installer `LILO` sur le nouveau disque. Avec une **Debian**, assurez vous que la ligne "boot" indique "`/dev/hda`" plutôt que "`/dev/hda1`" ou quelque chose de similaire si vous voulez installer `LILO` sur le secteur d'amorce du disque dur.

Vous pouvez essayer de redémarrer votre système depuis votre nouveau disque dur pour vérifier que tout marche bien. Si vous avez des problèmes, vous pouvez toujours utiliser la disquette pour démarrer votre système.

12 Remerciements

Je tiens à remercier tout spécialement le *Dr Konrad Hinsen* de l'Institut de Biologie structurale de Grenoble, France, qui a gentiment agit comme mon gourou **Linux** personnel. Merci aussi à *Frank Damgaard*, *Paul Koning*, et *Josh Rabinowitz*, ainsi qu'à *Scott Christensen* pour avoir attiré mon attention sur quelques particularités de la **Slackware**.