

## 0.1 Cloner et Transférer son Linux

Le but de ce document est d'expliquer comment transférer son système d'un disque dur à un autre. Exemple d'applications :

- clonage rapide d'un système existant.
- migration cause disque dur plus moderne.

### 0.1.1 1. Pré-requis :

- Partitionner le disque cible, avec au moins une partition /
- Formater la partition /
- Un espace disque suffisant, au moins le double de la taille réellement occupée par le système actuel + la taille de l'archive du système actuel. Concrètement, ma partition source, fait 1.1 Go, dont 760 Mo sont utilisés, d'après cfdisk ( ou fdisk, on s'en fout ). Je dois faire une partition d'au moins 760 Mo (3 Go dans mon cas) et avoir un espace libre où disposer 760 Mo de données compressées par gunzip.
- Pour être sûr qu'aucun service n'accède au disque dur pendant qu'on migre , démarrez en mode single user, en tapant : init 1 au démarrage de LILO ou en modifiant /etc/inittab

### 0.1.2 2. Explications préalables :

Dans ce document j'explique comment j'ai migré un système de /dev/hda1 , un petit disque IDE , vers /dev/sda1 et /dev/sda3, sur un disque SCSI. J'ai bien entendu intégré le support SCSI en dur (option Y) dans le noyau avant migration. Ainsi que le support Reiserfs, étant donné que c'est le système de fichier que j'utilise sur le DD cible. Pour les détails sur la compilation du noyau, se reporter aux tutos suivants<sup>1</sup>. Dans ce tuto, on va archiver à la volée le / du système source (/dev/hda1) et le detarrer dans la cible (/dev/sda1 et /dev/sda3). Pour ce faire, la cible devra être montée dans un répertoire accessible depuis le système source. Ceci suppose qu'on puisse tourner en boucle jusqu'à remplissage du disque , ou du moindre Ctl+C salvateur (pas Adamo). En gros : si on copie bêtement / vers /mnt/disque\_cible , on va aussi copier /mnt/disque\_cible vers /mnt\_disque\_cible/mnt/disque\_cible/mnt/disque\_cible ....

### 0.1.3 3. Mise en oeuvre sur un exemple :

J'ai créé les partions suivantes :

```
sda1 = / 3 Go
sda2 = swap 150 Mo
sda3 = /home le_reste Go
```

Petit swap, la machine est un K6 450 avec 250 Mo de ram. Si votre système cible, devra tenir sur une unique partition / (pas de partition spéciale /home ou /usr), vous pouvez ignorer les parties traitants de /home. Si votre partition annexe n'est pas /home mais /usr, remplacez simplement dans la suite /home par /usr. Même logique pour les autres cas de figure. J'ai formaté les partitions par :

---

<sup>1</sup> [../?id=35](http://www.truonme.net/?id=35)

```
mkreiserfs /dev/sda1
mkswap /dev/sda2
mkreiserfs /dev/sda3
```

J'ai monté les partitions cibles dans /mnt/destination, comme suite :

```
minibox :~# mkdir /mnt/destination
minibox :~# mount -t reiserfs /dev/sda1 /mnt/destination
minibox :~# mkdir /mnt/destination/home
minibox :~# mount -t reiserfs /dev/sda3 /mnt/destination/home
minibox :~# df -h
Filesystem Size Used Avail Use% Mounted on
/dev/hda1 1.1G 758M 386M 67% /
/dev/sda1 2.8G 316M 2.4G 12% /mnt/destination
/dev/sda1 2.8G 0 2.8G 0% /mnt/destination/home
```

Pour comprendre la suite , il est préférable de lire toute ma prose avant d'appliquer bêtement. Commencez par vérifier ce qui est monté. Cela peut varier d'une distrib à une autre , notamment à cause de /proc/usb , /dev/pts et /dev pour les possesseurs de Mandriva qui utilisent devfs par défaut. Concrètement, tapez :

```
minibox :~# mount
/dev/hda1 on / type reiserfs (rw)
proc on /proc type proc (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/sda1 on /mnt/destination type reiserfs (rw)
/dev/sda3 on /mnt/destination/home type reiserfs (rw)
```

/proc et /dev/pts sont des systèmes de fichiers virtuels , on ne va pas les copier . La règle est : on ne copie QUE les disques hdX ou sdX. La copie peut commencer, à l'exclusion des répertoires que vous ne voulez pas (à adapter à votre cas , mais vous devez exclure /mnt ) :

```
minibox :~# cd / && ( cd / && tar cvf - . --exclude proc --exclude dev/pts --exclude
mnt ) | ( cd /mnt/destination && tar xvf - )
minibox :~# mkdir /mnt/destination/proc
minibox :~# mkdir /mnt/destination/mnt
```

Ca va durer un temps aussi variable que la température du fut du canon. A la fin , on a un système transféré , mais pas bootable pour autant. Pour le rendre bootable, Il vous suffit de mettre à jour le chargeur de boot sur le nouveau système . Pour cela allez sur le système en question :

```
minibox :~# chroot /mnt/destination
```

Avec votre éditeur de texte préféré ( vi , mc , wine notepad.exe ), adaptez /etc/lilo.conf , /etc/fstab a votre goût , et rendez le disque bootable. A ce niveau , chaque cas est particulier , il importe de bien comprendre les mécanismes , et le but de ce tuto n'est pas d'expliquer les nuances de lilo . Dans mon cas , je veux rendre /dev/sda1 bootable , je verrai avec le bios pour indiquer scsi en 1er démarrage. Donc je change dans /etc/lilo.conf :

```
boot=/dev/hda en boot=/dev/sda
root=/dev/hda1 en root=/dev/sda1
```

J'installe lilo sur le disque :

```
minibox :~# lilo
Added Linux*
Added LinuxOLD
```

Le fichier `/etc/mtab` contient la liste des périphériques montés. Avant de redémarrer , il vaut le vider : `vi /etc/mtab` puis, "`dd`" "`dd`" "`dd`" , ou autre selon votre éditeur de texte. Dans mon cas , passant d'un disque ide a un scsi , j'ai du modifier `/etc/fstab` , les noms des partitions ne sont pas les mêmes . En gros , remplacer `hda` par `sda`. Dans le cas de 2 ide , plus courant , il n'y a priori rien à faire sauf à adapter les numéros des partitions, qui peuvent avoir varié. Grand moment ... Reboot ... Ou halt , sur le nouveau disque en `/dev/sda`. A priori , ça doit marcher . Mais lilo est parfois si caractériel. Après toute manipulation des partitions , il vaut mieux rebooter. En effet le système d'exploitation lit ses données depuis le bios. Ne pas rebooter n'empêche pas de continuer la manip , mais il n'est pas évident que les données seront effectivement accessibles. J'ai testé .... C' est une limitation de l'IDE et des i386, pas de Linux . Alors reboot. Bonne chance . Documentations connexes : `man lilo.conf`, `man fstab`, `man tar`, `compil noyau` éventuellement ( cas particulier scsi ). **Remerciements :** M. Alan Cox lui-même pour le tar magique