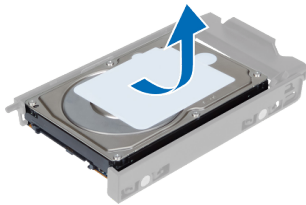


Same PC, new HD

Serge Y. Stroobandt

Copyright 2016, licensed under [Creative Commons BY-NC-SA](#)

Introduction



Here are my notes about how to change the system hard disk on your GNU/Linux computer. This comes handy when you want to upgrade your «*spin-nig rust*» boot drive to a much quicker **solid-state drive (SSD)**.

Hardware



With the PC **powered down**, install and connect the new hard drive. For the time being, **the old hard drive remains in place** to allow the copying of its partitions to the new drive.

However, you may be dealing with a laptop computer, and/or have only one hard drive bay or SATA port available. If this is the case, use a SATA to USB adaptor to temporarily connect the new hard drive.



Figure 1: SATA to USB3 adapter

Use a live DVD or USB



Instead of meddling with chroot, I opted for the more straightforward technique of booting from either a **live DVD** or a **live USB**.

At start-up, you most probably will have to instruct your computer's **BIOS** to boot from the live DVD or USB instead of any HD.

Copy partitions with GParted



With the live session running, carefully identify the old and new hard drive using the `lsblk` command.

```
$ lsblk
NAME MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda   8:0    0 279.5G  0 disk
├─sda1 8:1    0    1K  0 part
├─sda5 8:5    0   512M  0 part /boot
├─sda6 8:6    0    4.5G  0 part [SWAP]
├─sda7 8:7    0    64G  0 part /
└─sda8 8:8    0 210.5G  0 part /home
sdb   8:80   0 298.1G  0 disk
sr0   11:0   1  1024M  0 rom
```

From the Settings menu, launch **GParted**. A cool feature of GParted is that it allows for copying partitions from one drive to another. This works for all partitions, except for the [SWAP] partition which is currently in use by the live session. Its contents is of no importance to us, so again **GParted is used to create an equally sized swap file system on the new hard disk.**

Partitions copied to the new drive can be made bigger, but not smaller. If you are copying partitions to a smaller sized SSD, you will need to **reduce the existing partions on the old drive first.**

After copying the partitions with GParted, the list of block devices may look similar to this. Note that on the old `sda` drive I had implemented **logical volume management (LVM)**; on the new `sdb` drive I chose not to do so any more.

```
$ lsblk
NAME MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda   8:0    0 279.5G  0 disk
├─sda1 8:1    0    1K  0 part
├─sda5 8:5    0   512M  0 part /boot
├─sda6 8:6    0    4.5G  0 part [SWAP]
├─sda7 8:7    0    64G  0 part /
└─sda8 8:8    0 210.5G  0 part /home
sdb   8:80   0 298.1G  0 disk
├─sdb1 8:81   0   512M  0 part /boot
├─sdb2 8:82   0    4.5G  0 part
├─sdb3 8:83   0    44G  0 part /
└─sdb4 8:84   0 249.1G  0 part /home
sr0   11:0    1  1024M  0 rom
```

Reinstall GRUB



Now, mount the partition of the new drive which would normally be used for booting (here /dev/sdb1).

```
$ sudo mount /dev/sdb1 /mnt
```

Finally, reinstall GRUB on the new drive:

```
$ sudo grub-install --boot-directory=/mnt /dev/sdb
```

GRUB FlexNet warning

Whilst doing the above, GRUB reinstalled but not without issuing the following warning:

```
/usr/sbin/grub-install:
warning: Sector 32 is already in use by the program `FlexNet';
avoiding it.
This software may cause boot or other problems in future.
Please ask its authors not to store data in the boot track.
```

As usual, you are not the only one dealing with this issue on the Internet. What happened is that the new drive is actually a decommissioned drive that previously held commercially licensed software under Microsoft Windows™. *FlexNet* appears to be a digital contraption that keeps record of one's licenses by writing data to a sector in the extended boot record (EBR). This is certainly not the kind of digital parasite we would like to see living in our EBR!

Removing FlexNet

Still within the same live DVD/USB session or in a new one, remove *FlexNet* from the affected drive (here /dev/sdb), using the sector number reported by GRUB (here sector 32):

```
$ sudo dd if=/dev/zero of=/dev/sdb bs=512 count=1 seek=32
```

There is no need to backup this usually empty part of the **extended boot record (EBR)**.

Reinstalling GRUB after FlexNet removal



Remaining within the live session. If not done so earlier, mount the partition of the affected drive which is normally used for booting (here /dev/sdb1).

```
$ sudo mount /dev/sdb1 /mnt
```

Finally, reinstall GRUB on the affected drive with:

```
$ sudo grub-install --boot-directory=/mnt /dev/sdb
```

GRUB should now install without any warnings. You are now safe to reboot from the disinfected hard drive.



This work is licensed under a **Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License**.

Other licensing available on request.

Unattended **CSS** typesetting with **Prince**.

This work is published at http://hamwaves.com/same_pc.new_hd/en/.

Last update: Thursday, August 25, 2016.