

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
 HAYSTACK OBSERVATORY
 WESTFORD, MASSACHUSETTS 01886

11 August 2010

Telephone: 781-981-5400
 Fax: 781-981-0590

To: Mark 5 Development Group

From: Dan L. Smythe

Subject: Cloning system disks

I describe here how I cloned a 250-GB system disk to a 128-GB solid state disk, by reducing the size of the original `sda1` partition to fit on the new 128-GB disk, and copying the root partition (`sda1`) to the new disk. I am sure there are other, perhaps better, ways to do this; but this way worked for me. Note that in general it is not safe to use `parted` or `dd` on a mounted file system, so you will need to boot from a CD to make the clone.

Here is a general outline of the procedure:

- 1) Use `fdisk` or `sfdisk` to determine the size of the new disk.
- 2) Use `parted` to resize the partitions on the old disk to fit the new disk.
- 3) Use `sfdisk` to create identical partitions on the new disk
- 4) Use `dd` to copy the root partition from the old disk to the new disk.
- 5) Use `grub-install` to install grub on the new disk.

It is a good idea to update the OS before making the clone. If the Field System is installed, you can use `/etc/cron.weekly/apt-get` or `/usr2/fs/etc_cron.weekly_apt-get` to download and install Debian security updates.

Power down the system and install the new disk as `SATA1`, where the original OS disk is `SATA0`.

Use `fdisk -lu` to get the sizes of the existing partitions, and the size of the new disk. For example:

```
# fdisk -lu /dev/sda
```

```
Disk /dev/sda: 250.0 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders, total 488397168 sectors
Units = sectors of 1 * 512 = 512 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	63	483090614	241545276	83	Linux
/dev/sda2		483090615	488392064	2650725	5	Extended
/dev/sda5		483090678	488392064	2650693+	82	Linux swap / Solaris

```
# fdisk -lu /dev/sdb
```

```
Disk /dev/sdb: 128.0 GB, 128035676160 bytes
255 heads, 63 sectors/track, 15566 cylinders, total 250069680 sectors
Units = sectors of 1 * 512 = 512 bytes
```

Use `df -h` to confirm that the space Used on the original disk will fit on the new disk. If there's not enough space, you'll need to erase some files to make it fit.

Boot from a CD before using `parted` or `dd` to insure the original system disk is not mounted (inactive) while resizing or copying.

Use `parted` to reduce the size of the `sda1` partition to fit the new disk; it will take a long time if the new disk is a lot smaller than the old one. Don't forget to allow space for the `sda5` swap partition on the new disk. For example

```
parted /dev/sda print
parted /dev/sda resize 1 .0323 125000
```

Note that for `parted`, the units are MB.

Use `fdisk -lu` to check the new partition size.

Use `sfdisk -d` to save a copy of the partition layout:

```
# sfdisk -d /dev/sda > ~/backup-sda.sf
# cat ~/backup-sda.sf
# partition table of /dev/sda
unit: sectors

/dev/sda1 : start=      63, size=244139742, Id=83, bootable
/dev/sda2 : start=483090615, size=  5301450, Id= 5
/dev/sda3 : start=      0, size=      0, Id= 0
/dev/sda4 : start=      0, size=      0, Id= 0
/dev/sda5 : start=483090678, size=  5301387, Id=82
```

Note the gap between the end of `sda1` and the beginning of `sda2` that was created when you used `parted` to resize `sda1`.

Now make a copy of this partition layout file. This new layout file will be edited and used to partition the new disk.

```
# cp backup-sda.sf new-sdb.sf
```

Edit `new-sdb.sf` to make the partitions fit on the new disk. Don't forget to change all the `sdas` to `sdb` [`:%s/sda/sdb`]:

```
# cat ~/new-sdb.sf
# partition table of /dev/sdb
unit: sectors

/dev/sdb1 : start=      63, size=244139742, Id=83, bootable
/dev/sdb2 : start=244139805, size=  5301450, Id= 5
/dev/sdb3 : start=      0, size=      0, Id= 0
/dev/sdb4 : start=      0, size=      0, Id= 0
/dev/sdb5 : start=244139868, size=  5301387, Id=82
```

Note the 63-sector difference between the start of `sdb2` and `sdb5`, to make room for the partition table of the extended partition. Note also that the size of `sdb5` must be 63 sectors smaller than `sdb2`.

Use `sfdisk` to partition the new disk according to the new layout:

```
# sfdisk /dev/sdb < new-sdb.sf
```

Be careful; if you type `sda` instead of `sdb`, you will corrupt your old system disk.

Verify the new partitions on the new disk (`sdb`):

```
# fdisk -lu /dev/sdb
```

```
Disk /dev/sdb: 128.0 GB, 128035676160 bytes
255 heads, 63 sectors/track, 15566 cylinders, total 250069680 sectors
Units = sectors of 1 * 512 = 512 bytes
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1	*	63	244766339	122069871	83	Linux
/dev/sdb2		244139805	249441254	2650725	5	Extended
/dev/sdb5		244139868	249441254	2650693+	82	Linux swap / Solaris

Warning: Do not try to boot the new disk before installing `grub` on the new disk with the `grub-install` step below. More than once, I destroyed my partition table, and had to start over, before I figured out what corrupted the partition table.

Use `dd` to copy the root partition from the old system disk to the new disk, for example:

```
dd if=/dev/sda1 of=/dev/sdb1 bs=64k
```

The block size is not critical; it just affects the speed of the `dd` copy. Note that it will take a long time to copy the root partition. It took about 45 minutes to copy a 122-GB partition (50 MB/s).

The final step is to install `grub` on the new disk. Warning: Do not try to boot the new disk before completing this step. More than once, I destroyed my partition table, and had to start over, before I figured out what corrupted the partition table.

Run `halt` and power down the system, remove the old OS disk, and replace the old OS disk with the new one, and restore power with the CD still in the drive.

The purpose of the 2 commands below is simply to make the `grub-install` command in the `/sbin/` directory on the `sda1` system image available for execution.

```
# mkdir /mnt/sysimage
# mount /dev/sda1 /mnt/sysimage
```

Now make your newly mounted directory the root (or parent) directory. This you do by running the `chroot` command as follows:

```
# chroot /mnt/sysimage
```

To install the GRUB boot loader, execute the following command:

```
# grub-install --recheck /dev/sda
```

Now, finally, it is safe to enter `reboot` to reboot the new system disk.