

Move Linux onto a new hard disk

norman.krebs@gns-systems.de

Tue Mar 24 2009 09:59:06

1 Preliminary

I have a box with a Linux installation (Debian/Lenny, kernel 2.6.25.8 x86_64) on a parallel ATA drive (**PD1**) and two serial ATA drives for data (**SD2+SD3**). The PATA drive is a little small now and I don't want to use it until it dies and I want to replace it by one of the data drives and for more data I add a third SATA drive (**SD4**).

old	...	new
PD1	PD1	SD2
SD2	SD2	SD4
SD3	SD3	SD3
	SD4	

Table 1: Hard drives

old	new	mountpoint
/dev/hda1	/dev/sda1	/
/dev/hda2	/dev/sda2	swap
/dev/hda5	/dev/sda5	/usr/local
/dev/sdb1	/dev/sdb1	/usr2
/dev/sdc1	/dev/sdc1	/usr3

Table 2: Partitions

It's obvious that the point is to move the existing linux installation from the PATA drive to the SATA drive. (/dev/hda1 -> /dev/sda1). By the way I want to resize the system partition (/dev/sda1).

2 Boot a live medium

I use the grml-CD (www.grml.org). If your Linux installation is a 32-bit installation you need a 32-bit life CD; for 64-bit Linuxe - a 64-bit live CD. Boot the CD.

3 Prepare the disks

We are going to destroy all data on the old data disk **SD2** to make it the new system disk. Copy the data onto the new data disk **SD4** before. Attach the new data disk **SD4** to the machine (p.e.: `/dev/sdd`), partition it with one primary partition and copy the data:

```
# fdisk /dev/sdd
# mkfs.ext3 /dev/sdd1
# mount /dev/sda1 /mnt/sda1
# mount /dev/sdd1 /mnt/sdd1
# rsync -Havx /mnt/sda1/* /mnt/sdd1/
# umount /dev/sda1 ; umount /dev/sdd1
```

Partition the new system disk:

```
# fdisk /dev/sda
```

Create two primary partitions (`/dev/sda1`, `/dev/sda2`) one extended partition (`/dev/sda3`) acquiring the rest of the disk space and a logical drive in the extended partition (`/dev/sda5`)

action	command
create a new Partition	"n" (size: "+40000M"==40GB)
set partition type	"t" (83=Linux, 82=swap)
show partitioning	"p"
write changes	"w"
quit fdisk	"q"

Table 3: fdisk commands

Make file systems on the new drive:

```
# mkfs.ext3 /dev/sda1
# mkfs.ext3 /dev/sda5
# mkswap /dev/sda2
```

4 Relocation

4.1 Mounting

Create the mount points if needed and mount the old and new partitions:

```
# mount /dev/hda1 /mnt/hda1 ; mount /dev/hda5 /mnt/hda5
# mount /dev/sda1 /mnt/sda1 ; mount /dev/sda5 /mnt/sda5
```

4.2 Copy data

Copy all the crap from old to the new place:

```
# rsync -Havx /mnt/hda1/* /mnt/sda1/
# rsync -Havx /mnt/hda5/* /mnt/sda5/
```

4.3 Config files

4.3.1 menu.lst

Change /mnt/sda1/boot/grub/menu.lst (hda1 -> sda1):

```
title          Debian GNU/Linux, kernel 2.6.25.8
root           (hd0,0)
kernel        /boot/vmlinuz-2.6.25.8 root=/dev/sda1 ro
initrd        /boot/initrd.img-2.6.25.8
```

4.3.2 fstab

Change /mnt/sda1/etc/fstab:

old:

```
/dev/hda1      /                ext3    errors=remount-ro 0    1
/dev/hda2      none             swap    sw                0    0
/dev/hda5      /usr/local       ext3    defaults          0    0
/dev/sdb1      /usr2            ext3    defaults          0    0
/dev/sda1      /usr3            ext3    defaults          0    0
...
```

new:

```
/dev/sda1      /                ext3    errors=remount-ro 0    1
/dev/sda2      none             swap    sw                0    0
/dev/sda5      /usr/local       ext3    defaults          0    0
/dev/sdb1      /usr2            ext3    defaults          0    0
/dev/sdc1      /usr3            ext3    defaults          0    0
...
```

4.4 Chroot

Mount necessary things and make a chroot:

```
# mount -o bind /dev /mnt/sda1/dev
# mount -t proc /proc /mnt/sda1/proc
# chroot /mnt/sda1
```

4.5 Initrd

Rebuild the initial ram disk (initrd):

```
# mkinitramfs -o /boot/initrd.img-2.6.25.8 2.6.25.8
```

4.6 Grub

Reinstall grub:

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

4.7 Finally

Quit the chroot-environment and reboot into the new system.