Configuring software RAID

SOME LIKE IT SOFT

BERNHARD KUHN

High performance hardware RAID controllers are expensive. But there is a cheaper alternative. Software RAID is also available for Linux. However, before you can use it the hurdle of installation and configuration has to be overcome.

Software RAID is not an option favoured by every server administrator. It steals valuable processor time from the server. However, nowadays even inexpensive computers such as you can buy at your local superstore have computing power that would make the super-computer of a decade ago look puny. At the same time, small workgroup or intranet servers often only have to serve a couple of web pages or files via the bottleneck of Fast Ethernet, and perhaps distribute the occasional email. These tasks on their own are not enough to make the processor break out into a sweat. In this situation it is possible to bypass the usual hardware solution and save a tidy bit of money by using a software alternative instead.

SoftRAID isn't the right solution if you are considering a heavyweight multiprocessor server with lots of capacity and hot-swap capable components. In that situation the cost of a true RAID controller will add an insignificant amount to the cost. The simpler commissioning and maintenance of the hardware disk jugglers also has advantages in mission-critical domains. Where the demand for peak performance is vital a hardware-based system is the only possible choice. But if your demands are more modest and you are prepared to compromise on commissioning and maintenance you can manage perfectly well with this free software solution.

Obtaining the software

Since the official release of the first SoftRAID implementation in Linux kernel 2.0 a lot has happened. Originally only RAID Level 0 was supported. Also, anyone who wanted to install a bootable root file system on it had to patch the kernel and grapple with the Initial RAM disk. However, from kernel version 2.2 hard disk configurations using RAID 0, 1 and 5 can be recognised automatically by the (patched) kernel. Using an (also modified) LILO one can also boot up without any problems from RAID 1.

No need for backups?

Among home computer users SoftRAID is enjoying increasing popularity. Home users don't tend to be very diligent about making backups and are reluctant to fork out for an expensive but rarely used tape drive. Writeable CD-ROMs aren't the answer, since a full backup won't usually fit the storage medium. For this reason it often seems a good idea to back up by making a copy of the system on a second hard disk. Hard disks are relatively inexpensive and markedly faster than either CD-RW or tape.

Using a hard disk for backup isn't very practical if you need to install and then remove the drive from the system each time. But if the second drive is fitted permanently inside the computer it is no trouble at all. In that case, as long as the two drives are identical, they can easily be run as a software RAID 1 array to make the most of the potential increase in performance (see the article "Raid Basics", also in this issue.)

If you consider it important to keep your backup separate from the computer you can still do so. To perform a backup all you need to do is connect the disk and wait for synchronisation to occur in the background (see the box "Background Rebuild") After an hour or so the mirror disk will contain a copy of the system and can be removed from the computer and put back in a safe place. If your main drive fails this will save you the extremely tedious business of restoring from a backup. Instead, you simply replace the failed drive with the backup drive and switch on.

This may sound too good to be true. And there is a small catch: if a voltage spike or some other disaster were to zap both drives during the synchronisation process it is likely that all your valuable data would be lost. However, the chances of that happening are, as you can imagine, very slight. Nevertheless, for anyone who installs SoftRAID to increase failsafety and improve transfer performance in a commercial environment a traditional backup strategy remains an absolute must.

Info

RAID-Patches
http://people.redhat.com/mingo/raid-patches/

Software-RAID HOWTO
http://www.linux.org/help/ldp/howto/Software-RAID-HOWTO.html

Readers who are put off by the description of installation and configuration that follows would be well advised to consider the latest Red Hat distribution. From version 6.1 on, the graphical installer supports the option of bootable RAID 1 and thereby saves a great deal of challenging work.

Configuration

After installation comes the configuration of the RAID array. You will be spoilt for choice here as to which RAID level is best suited to your needs. (To help you choose, see the article "Raid Basics" in this issue.) Configuration is equally easy whichever variant you choose. In the file /etc/raidtab the RAID drives must be defined and then initialised just once with mkraid. Later, the kernel starts the RAID configuration automatically, so any leftovers in /etc/rc.d/boot.local should be removed.

Listing 1 shows a simple RAID 1 configuration. The options are largely self-explanatory: this is a level 1 RAID device */dev/md0* consisting of two partitions (*/dev/sda4* and */dev/sdb4*) using a chunk size of 8 KByte. The statement "1" in *persistent-superblock* is needed so that the RAID configuration is automatically recognisable by the kernel right from boot-up. All RAID partitions must also possess the ID "fd" using fdisk ("Linux raid autodetect"). *md* stands for "Multiple Device". With this type of device, besides a RAID array hard disks can be arranged in a linear fashion with respect to each other so as to make what looks like one big hard disk.

Listing 1: RAID 1

```
raiddev /dev/md0
  raid-level              1
  nr-raid-disks           2
  persistent-superblock   1
  chunk-size              8

  device         /dev/sda4
  raid-disk      0
  device         /dev/sdb4
  raid-disk      1
```

Listing 2 shows a RAID 5 configuration consisting of three hard disks. Defective disks are removed

Background Rebuild

Defective or replaced hard disks are not allowed into the SoftRAID array until they have been manually integrated using raidhotadd. After that, background reconstruction can commence. The operating system and the applications running on it are largely unaffected by this: recovery of data will take place only when no other I/O transfers are pending.

```
[root@bee /root]# cat /proc/mdstat
Personalities : [linear] [raid0] [raid1] [raid5] [translucent]
read_ahead 1024 sectors
md0 : active raid1 hda2[0] 2297216 blocks [2/1] [U_]
unused devices: <none>

[root@bee /root]# raidhotadd /dev/md0 /dev/hda3

[root@bee /root]# cat /proc/mdstat # disk reconstruction
Personalities : [linear] [raid0] [raid1] [raid5] [translucent]
read_ahead 1024 sectors
md0 : active raid1 hda3[2] hda2[0] 2297216 blocks [2/1] [U_]
recovery=6% finish=23.3min
unused devices: <none>
```

RAID kernel installation

The following installation instructions have been tested with Red Hat 6.1 and Kernel-2.2.14. But the process should work just as well with other distributions and more recent kernel versions. A normal version 2.2 Linux kernel does in fact already know about RAID levels 0, 1 and 5, but it has trouble automatically recognising the partitions being handled by them. Because of this, some assistance in the form of a kernel patch is required. Also, it won't hurt to install the latest RAID tools. Both of these can be found on the Red Hat web site. In addition, the original kernel sources will be needed.

```
cd /tmp
wget http://people.redhat.com/mingo/raid-patches/raid-2.2.14-B1
wget http://people.redhat.com/mingo/raid-patches/raidtools-dangerous-0.90-20000116.tar.gz
wget ftp://ftp.uk.kernel.org/pub/linux/kernel/v2.2/linux-2.2.14.tar.gz
```

In the first step of installation, the kernel has to be unpacked, patched, configured and installed. When performing the last two steps you may find the manual that came with your Linux distribution to be helpful. RAID-specific options in the kernel configuration are used, and all RAID levels are compiled in.

```
cd /usr/src && mv linux linux.old
tar -xzf /tmp/linux-2.2.14.tar.gz
cd linux
patch -p1 < /tmp/raid-2.2.14-B1
make menuconfig
make clean && make dep && make bzImage
make modules && make modules_install
cp arch/i386/boot/bzImage /boot/zImage-raid
vi /etc/lilo.conf
lilo && reboot
```

After rebooting, the command cat /proc/mdstat will shows whether the previous procedure was successful. Now the latest RAID tools should be installed. Then the RAID configuration can begin.

```
cd /usr/src
tar -xzf /tmp/raidtools-dangerous-0.90-20000116.tar.gz
cd raidtools-0.90 && ./configure && make && make install
```

Table 1: RAID-Tools 0.90

mkraid	one-off installation and boot up of a SoftRAID configuration
raidhotadd	insert replacement or spare disk
raidhotremove	remove defective disk from the group
raidstart	start multiple device — only needed in exceptional cases
raidstop	stop multiple device

from the RAID system using *raidhotremove* and *raidhotadd* and after the swap, reconnected. If removable cradles are used, defective SCSI hard disks can even be swapped while operations continue or "hot spare" disks added later. The kernel interface is used for this:

```
echo "scsi [add-single-device|remove-single-device] <controller> <bus> <target> <lun>" > /proc/scsi/scsi
```

In this way, SCSI disks can be entered into or removed from the device table. (This can be used to simulate crashes.) Unfortunately hot-swapping does not work with IDE drives yet. Also, with IDE-based RAID it is also only possible to connect a maximum of eight devices. When an IDE channel is occupied by two hard disks the transfer rates sometimes reduces considerably. However, the performance is still adequate for many applications.

configuration 2: RAID 5

```
raiddev /dev/md0
  raid-level              5
  nr-raid-disks           4
  nr-spare-disks          0
  parity-algorithm        left-symmetric
  persistent-superblock   1
  chunk-size              32

  device                 /dev/sda4
  raid-disk              0
  device                 /dev/sdb4
  raid-disk              1
  device                 /dev/sdc4
  raid-disk              2
  device                 /dev/sdd4
  raid-disk              3
```

RAID 10 can be obtained simply by creating a RAID 1 array consisting of two RAID 0 configurations. For *device* the corresponding */dev/md** is entered instead of an actual partition.

Booting up ReiserFS-RAID

At present, SoftRAID only works well with LILO using RAID 1. For all other variants it is still necessary to struggle with initrd and other tedious matters. In the meantime RAID works with Root-ReiserFS which not long ago would not have been possible:

```
...
[test@lab1 test]$ mount
/dev/md0 on / type reiserfs (rw)
...
```

When formatting an ext2 partition it is also possible to tell the file system driver by means of the "stride" parameter what chunk sizes the device under it prefers to process. The choice of chunk size for the file system and multiple device has a significant effect on the performance of the SoftRAID system.

Kernel 2.4

Unfortunately the RAID implementation in the upcoming 2.4 kernel series is not yet ready to go. But since the new kernel is still awaited (what's more, everyone is cordially invited to help out!), we can hope that the RAID support will be completed by the time it comes out.

More productive

Linux Software RAID is already bomb-proof and has been in use for some time in mission-critical applications. For example, the web server of Linux New Media AG (publisher of "Linux-Magazin" and "Linux-User" in Germany) has been running a RAID 1 configuration without any problems since it was set up almost a year ago. ■