ASK KLAUS!

Klaus Knopper is the creator of Knoppix and co-founder of the LinuxTag expo. He currently works as a teacher, programmer, and consultant. If you have a configuration problem, or if you just want to learn more about how Linux works, send your questions to:

klaus@linux-magazine.com

Migrating?

? I currently know little to nothing about Linux, other than what I have read. I am interested in installing Ubuntu Linux as my operating system.

I have many systems (laptops, desktops), all of which run XP in different variations (e.g., 32- and 64-bit; standard hard-drive installation and some in RAID configurations).

Is it possible to run Linux simultaneously on these systems, normal or RAID? If so, will I have the option on which OS to boot from, or do I need to install another hard drive? (Which isn't a problem if that is the recommended solution.) Are all or some Windows-based pro-

grams functional in Linux? I do a lot of video editing.

What are the alternatives for programs such as Word/Powerpoint/Exel, or do they function in the Linux environment? I also play around with overclocking – do Pi, Memtest, Prime, etc. function in Linux?

My ultimate goal is to migrate completely away from Windows, but to do it cautiously and learn as I go.

Because you are not familiar with Linux yet, you could start by running a Linux distribution on your current operating systems inside a virtualization, such as QEMU or VMware, which does not change your running system other than adding a few files for virtual hard disks to hold the installation.

On the other hand, it may be more efficient to dedicate one of your computers to your experiments, first running a few Live GNU/Linux distros from CD or DVD and then doing a hard disk installation when you start feeling comfortable with the look and feel of one of the tested distributions.

Also, you can install Linux in parallel with Windows on the same computer,

though it is somewhat risky if you have never repartitioned a hard disk before on your own. Then you would resize the Windows partition(s) and free enough space for a GNU/Linux installation, start the installation from DVD, and select "keep installed operating system," which is also called "Dual Boot." After you restart the computer, you will get a boot menu from which you can select which operating system you want to work with in this session.

Although it is possible to run Windows applications – even games [1] – on Linux using Wine or its derivates, native Linux applications can do the same things and then some.

For video editing, I recommend Kino [2], which records from a DV camera, lets you cut and prepare scenes, and has a section with nice video effects and labeling. Because it works on-disk, it does not require a lot of memory, and it can create output formats suitable for SVCD or DVD. Also, Cinelerra [3] is a full movie-editing solution.

OpenOffice, the standard office suite that includes text processing, spreadsheet, drawing, and presentation applications, is available for many platforms and is included in every GNU/Linux dis-
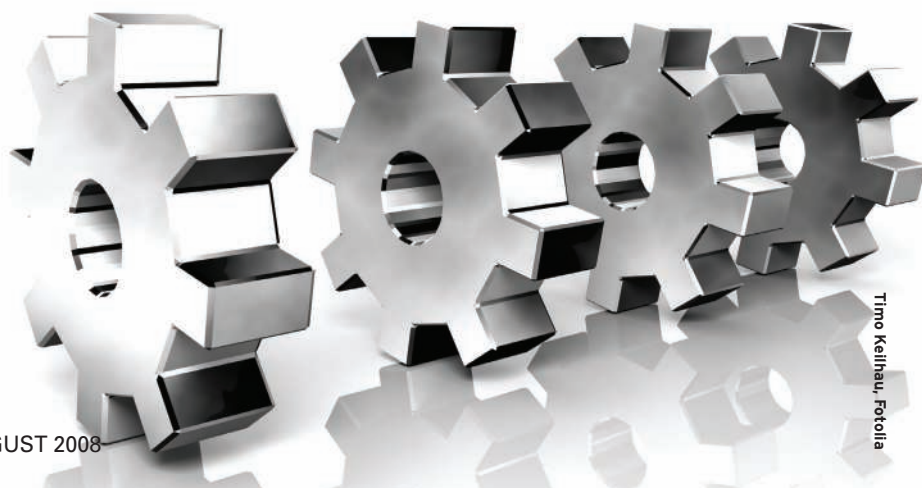
Timo Keilhau, Fotolia

AUGUST 2008

tribution I know. Also, the program can read old/proprietary formats of other programs.

RAID in various levels is also supported by Linux – software as well as hardware RAID.

Be aware that some controllers sold as "RAID" are really only "Softraid" controllers (i.e., they need special drivers that ultimately do nothing more than software-based RAID would do).

Windows uses different filesystems. Linux can read and write FAT32 and NTFS without problems, whereas Windows can't access Linux filesystems directly. However, a few drivers for Windows allow you to copy data from ext2 or ReiserFS.

Memtest, lm_sensors (for watching board and CPU temperature), cpuburn, stress, super pi, and other testing suites are available for Linux.

Also, a few forums discuss Linux and overclocking. Personally, I'm not very fond of this because it shortens the lifetime of your hardware. (Note the warning on the cpuburn homepage [4].)

ftp Disparity

I'm baffled with an effect I'm seeing when transferring video files. The machines are:

- Windows XP, P4 3.0, 2GB RAM, 500GB SATA drive, on-board gig ether (five years old). Ftp transfer to Linux is 600 + MB/sec.
- SUSE 10.3, C2Duo 2.66, 2GB RAM, two 500GB SATAs in Raid 0, dual on-board gig ether (but I haven't config-

ured mating the dual jacks yet).

Ftp transfer to XP is 20MB/sec. Copying anything from this Linux machine to, for example, a USB drive – or even simply duplicating a file – results in a maximum transfer rate of around 20-28MB/sec. Both computers are attached to a gig switch, and I'm getting these readings from KDE's transfer dialog.

I cannot account for this enormous difference in transfer rate. I would appreciate any help you can provide.

It's difficult to say what causes the difference because many components are involved in your benchmark: network cards and drivers, ftp servers (on both sides?), client ftp program, desktops with on-disk caching.

For perhaps a more reliable benchmark, I would recommend a few console tools on the Linux side to rule out any desktop configuration issues.

First of all, I would check on the network card configurations with

```
sudo ethtool eth0
```

which should report something like that shown in Listing 1.

In Listing 1, you can see which data transfer modes are supported and that the card currently runs at 10MB/sec, al-

though it is capable of doing 100MB/sec. This might be related to autonegotiation, the switch determining the speed considered to be most reliable. You can switch off autonegotiation and set the speed to 100MB/sec with:

```
sudo ethtool -s ⤶
eth0 autoneg off speed 100
```

In your setup, you should try the *speed 1000* setting. If you lose connectivity, just re-enable autonegotiation.

To measure the raw data transfer speed, you can use the "bing" tool:

```
sudo bing ⤶
local_machine_ip_address ⤶
remote_machine_ip_address
```

This sends out data and measures reply time until you hit Ctrl + C, then it gives you statistics about data throughput. The numbers shown are the maximum speed you could get at this card configuration, without application data protocol overhead.

For measuring ftp speed, you can use *wget*, which is also the fastest tool for transferring data via http and ftp, and it also supports transfer of partial data from an interrupted previous download. The command:

```
wget -c ftp://⤶
remote_machine_ip_address⤶
/filename
```

Send your Linux questions to klaus@linux-magazine.com.

Listing 1: Network Card Configuration

```
01 Settings for eth0:
02        Supported ports: [ TP MII ]
03        Supported link modes:   10baseT/Half 10baseT/Full
04                                100baseT/Half 100baseT/Full
05        Supports auto-negotiation: Yes
06        Advertised link modes:  10baseT/Half 10baseT/Full
07                                100baseT/Half 100baseT/Full
08        Advertised auto-negotiation: Yes
09        Speed: 10Mb/s
10        Duplex: Half
11        Port: MII
12        PHYAD: 32
13        Transceiver: internal
14        Auto-negotiation: on
```

chris3d, Fotolia

downloads the remote file to the local directory and displays transfer speed while download is in progress.

When you copy to disk (you said that in general, disk writes seem slow on your Linux system), please be aware that Linux and Windows use disk caching in different ways.

Copying of small files "looks" like data is written immediately, when in fact the file data is cached in memory first and written only if cache memory is low, the device is unmounted (Linux) or "safely removed" (Windows), or a sync (write cache data to disk) is forced by a program. When copying data larger than your computer's RAM, you should start seeing the raw disk performance, possibly in peaks because data is still first cached in RAM and written to disk asynchronously.

Although 20-28MB/sec might be an expected result for raw disk throughput on USB 2.0, it's too slow for a recent IDE or SATA disk. If disk writes on your main system disk are slow, please check to see whether DMA is enabled with

```
sudo hdparm /dev/hda
```

which should output something like this:

```
/dev/hda:
 multcount    = 16 (on)
 IO_support   = 1 (32-bit)
 unmaskirq    = 1 (on)
 using_dma    = 1 (on)
 keepsettings = 0 (off)
 readonly     = 0 (off)
 readahead    = 256 (on)
 geometry     = 19457/255/63, ⤶
 sectors = 312581808, start = 0
```

The *using_dma = 1* is relevant here because it tells you that the disk can transfer data to an application without much interaction from the CPU, which can mean a speed improvement by a factor of 5 or more. A few (broken) IDE controllers on mostly older boards can't handle DMA, but for the rest, you can enable it with:

```
sudo hdparm -d1 /dev/hda
```

If your computer freezes after this command, or other strange things happen, you have a controller that cannot handle DMA and probably should reach for the

reset switch. A quick disk read-only benchmark can be done with

```
sudo hdparm -tT /dev/hda
```

which displays something like:

```
/dev/hda: Timing cached reads: ⤶
 572 MB in 2.00 seconds ⤶
= 285.80 MB/sec Timing buffered ⤶
disk reads: 138 MB in ⤶
3.04 seconds = 45.40 MB/sec
```

Forty-five MB/sec is fine for an IDE disk with DMA enabled, but for a disk attached via USB 2.0 on my notebook, I only get about 12MB/sec.

Writing to USB flash disks is very slow. Again, it only "looks" fast when writing small files, before you tell the operating system to disconnect the device and therefore flush data to the chip.

Because network transfer between your two computers generally seems to be fast, it might be that the receiving side is just slow when acknowledging received data, or for some reason, it tries to do hostname resolving on local IP addresses that are not present in the local hosts file (*/etc/hosts* on Linux). Have you tried uploading files in both directions, or is the ftp server just running on one machine?

RAID Drive

I have a comment about your answer on Linux RAID in the June 2008 issue.

In a fast-developing system such as Linux, it is not advisable to install RAID 1 to the / partition. Every time you change your distribution or update your system, you face the danger of reformatting your / partition. And this means you might lose RAID 1 information on your device lists!

Instead, I recommend the following:
1. Have two 1MB */boot* devices on two different disks
2. Equally reserve two 12MB / devices.
3. Use the rest of your disk space for RAID 1 (or EVM).
4. Make your installation on one */boot* and / partition.
5. Create your RAID 1 devices with a mount point to the / device.
6. Boot your machine with a Knoppix disk and mount all of your */boot* and / devices to this system.

7. Make a backup of your */boot* and / devices. Simply use *sudo cp -a source_directory dest_directory* for this purpose.
8. Edit the */boot/grub/menu.lst* and */etc/fstab* files to suit respective devices. Now you have two useful distros, and you can experiment with new versions on one while keeping the other as a working backup. RAID 1 devices remain intact in these procedures.
9. Occasionally, repeat steps 7 and 8 for backup purposes.

After all, RAID 1 is good for data mirroring, but it is not that good for program and system mirroring.

Although your proposed setup for */boot* and / absolutely makes sense when frequently updating kernels, having installed several distributions in parallel on the same disks and having a copy of the raid configuration files available on separate partitions that do not require a raid-enabled kernel or distro for mounting, I have to disagree with your statement about RAID 1 not being useful for program and system mirroring.

The Linux MD code implements read balancing, so when you have two separate disks on different controller slots, throughput when reading files – especially in a busy system with a lot of disk activity – will also be faster because reading of files can be parallelized across the disks.

While the reading process idles when waiting for one disk to complete physical data transfer, it can transmit data from the second disk. This is not exactly doubling the transfer rate, but it does make your programs load noticeably faster.

Writing to disk, on the other hand, is somewhat slower with RAID 1. Also, in theory, with current RAID/EVM meta information, the kernel should be able to reassemble an array from information stored inside the RAID partitions on its own, though it's easier when you keep a copy of raidtab or mdadm.conf on a non-RAID partition. ■

INFO

[1] Wine HQ: *http://www.winehq.org/*

[2] Kino: *http://www.kinodv.org/*

[3] Cinelerra: *http://cinelerra.org/*

[4] Cpuburn home page: *http://pages.sbcglobal.net/redelm/*