
100 Linux Tips and Tricks

by Patrick Lambert

Table of Contents

<u>Introduction</u>	1
Copyright	1
About the author	1
Acknowledgements	1
Audience	1
Organization	2
Web resources	2
<u>Installation</u>	4
What this chapter covers	4
Tip 1: Which distribution is good for you	5
Tip 2: How to find a Linux CD-ROM at low cost	6
Tip 3: Multiple operating systems	7
Tip 4: Installing with no CD-ROM drive or modem	8
Tip 5: Swap and memory	9
Tip 6: More swap with a swap file	10
Tip 7: Kernel size and modules	11
Tip 8: The boot prompt	12
Tip 9: Wrong memory size found	13
Tip 10: Master boot record and LILO	14
Tip 11: LILO can't find a kernel on a big drive	15
Tip 12: X Window configuration options	16
Tip 13: Allowing users to mount drives	17
Tip 14: Allowing users to run root programs	18
Tip 15: Linux and NT booting	19
Tip 16: Annoying boot messages	20
Tip 17: Programs on CD-ROM	21
Tip 18: International console	22
Tip 19: Multiple kernels choices	23
Tip 20: Default file permissions	24
Tip 21: Default boot mode	25
Tip 22: More information from usenet	26
Tip 23: Bytes per inodes	27
Tip 24: LILO and boot problems	28
Tip 25: Making CD-ROM images	29
Tip 26: FTP access restrictions	30
<u>Hardware</u>	32
What this chapter covers	32
Tip 1: Detecting 2 ethernet cards	33
Tip 2: Everything on sound cards	34
Tip 3: Non-PostScript printers	35
Tip 4: Use Windows special keys in Linux	36
Tip 5: Added processors	37
Tip 6: Detecting an ISA device	38
Tip 7: Find hardware information	39
Tip 8: Blinking leds on the keyboard	40

Table of Contents

Tip 9: Reading a foreign file system	41
Tip 10: Can't mount root fs	42
Tip 11: Linux on a 286?	43
Tip 12: Linux without a hard drive	44
Tip 13: Shutdown and power off	45
Tip 14: LPD started but no device found	46
Tip 15: Read files from FAT32 drives	47
Tip 16: TV on Linux	48
Tip 17: Device drivers	49
Tip 18: Mouse problems	50
Tip 19: International keyboards	51
Software	53
What this chapter covers	53
Tip 1: Background image in X Window	54
Tip 2: Customize Netscape Communicator	55
Tip 3: POP3 in Pine	56
Tip 4: Multiple accounts in Pine	57
Tip 5: Running Java programs	58
Tip 6: Virtual hosts in Apache	59
Tip 7: Libc versus Glibc	60
Tip 8: Aliases with Omail	61
Tip 9: Samba with Windows 98 or NT 4	62
Tip 10: KDE drag and drop icons	63
Tip 11: Find files	64
Tip 12: asm or linux include files not found	65
Tip 13: ICQ on Linux	66
Tip 14: Reading foreign documents	67
Tip 15: Scanning with Linux	68
Tip 16: Real audio and video	69
Tip 17: Emulation	70
Tip 18: Shared library not found	71
Tip 19: Hard to erase files	72
Tip 20: Files permissions	73
Tip 21: Changing file permissions	74
Tip 22: An international background	75
Tip 23: Powerful file transfer system	76
Tip 24: Editing in text editors	77
Tip 25: Documentation and manual	78
Networking	80
What this chapter covers	80
Tip 1: Easy PPP dialup	81
Tip 2: Internet for your LAN	82
Tip 3: Domains to search in	83
Tip 4: Display IP rather than hostname	84
Tip 5: Is my modem a winmodem?	85

Table of Contents

Tip 6: Sharing files from a Windows system	86
Tip 7: Sorry but this host is not in my list	87
Tip 8: Access to various networks	88
Tip 9: Accessing remote file systems	89
Tip 10: Secure Web server	90
Tip 11: Secure alternative to telnet	91
Tip 12: Speed problems on a PPP connection	92
Tip 13: Names and name servers	93
Tip 14: Who owns this port	94
Tip 15: Network printers	95
Development	97
What this chapter covers	97
Tip 1: Graphical messages to the world	98
Tip 2: Code reuse	99
Tip 3: Makefile don't equal C	100
Tip 4: Parsing the command line in BASH	101
Tip 5: Don't grep grep	102
Tip 6: Move a text into upper case letters	103
Tip 7: Using PASCAL on Linux	104
Tip 8: Segmentation fault	105
Tip 9: Who is online?	106
Tip 10: Graphical toolkits	107
Tip 11: IDE and visual interfaces	108
Tip 12: Free software and copyleft	109
Tip 13: Talking to the terminal	110
Tip 14: Internet technologies	111
Tip 15: Library types	112

Introduction

Copyright

This book is copyright by Patrick Lambert. It is provided free of charge in the hope that it will be useful. You may copy, distribute and print this book. You may not modify it without prior written consent from the author.

The tips in this book are given AS-IS. This means that I shall not be responsible for any damage that may occur from their use. You use them at your own risks.

Linux is a trademark of Linus Torvalds
RedHat is a trademark of RedHat Software Inc.
Windows and DOS are trademarks of Microsoft Corp.
Sound Blaster is a trademark of Creative Labs
PostScript is a trademark of Adobe
Other trademarks and copyrights may apply.

About the author

Patrick Lambert is currently a student in Computer Science at the University of Montreal in Quebec, Canada. He is the author of various Web sites for the Linux community, and of various software packages including [GXedit](#)

Although he does everything from systems administration to software programming, he spends most of his time working on Web sites for the Linux community. You can contact Patrick at drow@darkelf.net

Aknowledgements

I would like to thank Tuomas Kuosmanen for the logo and the images on the Web site.

Audience

This book was written for anyone using Linux, from new users to experts who want to explore this wonderful operating system. The tips and tricks in this book were discovered by myself over years of experience using Linux, and learning about it. Some are very basic tips to make your computing life easier, others are advanced tricks that can save you days of work.

I tried to cover all distributions of Linux in this book. I personally use Slackware and RedHat on PC systems. If you find any error in the book, feel free to contact me so a future second edition could correct them.

Organization

This book is divided into 5 chapters. Each chapter covers a specific topic:

- [Chapter 2](#) covers installation of Linux. These are tips and tricks useful when installing Linux itself or any new program. Some tips will cover new means of installing Linux on non-typical hardware, others will explain how to take Linux distributions from an FTP server and make your own CD-ROM with them, or where to find Linux CD-ROMs for as little as \$2.
- [Chapter 3](#) covers hardware related matters. You will learn tips there on how to get your non-PostScript compatible printer to work, or how to get a sound card detected.
- [Chapter 4](#) covers software. You will find tips there about all kinds of Linux software, including where to find and how to install the Java Development Kit port, and everything about the Pine mail and news program.
- [Chapter 5](#) covers networking in all its forms. There you will see how to setup a PPP connection quickly, without editing all of the configuration files yourself, as well as some nice programs that were made to ease dialup procedures. You will also see tricks on how to make your local LAN network without unexpected problems.
- [Chapter 6](#) is the last chapter but covers an important part of Linux: development. Here you will find a lot of tips on how to write powerful scripts to make your system easier to handle, and a full overview of what to do and what you don't want to do in C to avoid problems like memory leaks, and how to allow easy scalability.

Web resources

This book has a sister Web site at <http://tipoftheweek.darkelf.net> where some of the tips from this book can be found, and where you can submit your own tips to the site, to help the Linux community.

Installation

What this chapter covers

Installation is a very important part of any operating system. This is why I cover this topic first. The next most important thing is installation of programs and software to get your system to do useful tasks. This chapter covers both of these aspects.

Tip 1: Which distribution is good for you

They are all good. But that's not a real tip. What you should be looking for is which distribution you feel the most comfortable with. RedHat has the reputation of being very easy to install. They provide special tools to make the configuration easier. Debian also has some tools, but will usually require you to go on the command line more often to configure the system. If you want to be on your own, and really learn how to edit configuration files then Slackware is for you. The Web site <http://www.linux.org> lists all the available distributions.

In the end, the best person to decide which distribution you like, is yourself.

Tip 2: How to find a Linux CD-ROM at low cost

The Linux market started from a few distributions available only from FTP servers, to full feature commercial distributions available in stores and online including a printed manual and phone support.

Here are the main choices you have when looking for a Linux distribution:

- You can download any Linux distribution from its FTP server. To take a few examples, RedHat can be downloaded from <ftp://ftp.redhat.com>, Slackware from <ftp://ftp.cdrom.com> and Debian from <ftp://ftp.debian.org>. That method is free, but requires you to have a fast Internet connection. Downloading a full Linux distribution over a 56Kbps modem will take you quite a few hours.
- An other way is to buy a full distribution. RedHat, for example, can be bought online for about \$50. This will include a box, a CD-ROM, a boot diskette, a manual and support from RedHat.
- The last way is to buy only the CD-ROM. There are a few places selling CD-ROMs of various distributions for \$2. One of them is <http://www.cheapbytes.com>. You will only get the CD-ROM, but this is all you need to install Linux if you are comfortable with the fact that you don't get a printed manual or free support. You can find the manual and other documentation on the CD-ROM.

Tip 3: Multiple operating systems

A computer only needs one operating system to work. But what if you just want to try out a new system? Do you need to forget about the old one and erase your hard drive? No, you can have as many operating systems on your computer as you wish.

Linux requires 2 partitions to work. Partitions are sections of the hard drive. When you install Linux, it will provide a program called fdisk or disk druid allowing you to create the needed partitions. The main problems people have is that they don't have empty partitions to use for Linux, and they don't want to erase the current Windows or DOS partition. The trick is to resize your current partition to create empty space. Then you will be able to make the partitions needed by Linux to install properly.

Fdisk doesn't allow you to resize a partition. You will need to use another program to do the job, before using fdisk to create the Linux partitions. A very popular commercial product to do this is Partition Magic from <http://www.powerquest.com>.

Let's see step by step what is needed to resize an existing partition to allow the creation of a new one for Linux:

- Buy Partition Magic, or get any other tool that can safely resize partitions.
- Make sure you have at least 150 megs free on your main partition, the required amount for Linux.
- Resize the partitions so you have at least 150 megs free, outside of any current partition.
- Reboot and launch the Linux installation.
- Run fdisk or any partitioning program that comes with the Linux distribution, and follow the installation instructions to make the required Linux partitions.

Tip 4: Installing with no CD-ROM drive or modem

Most Linux distributions come on a CD-ROM. You can also download them from an FTP site, but that requires an Internet connection. What if you have a system with no CD-ROM drive or Internet connection, like an old 486 laptop? The trick here is to have another desktop system with a CD-ROM drive, and a null-modem serial cable.

I will show you how to do it with Slackware. It is also possible with most other Linux distributions. Insert the Linux CD-ROM in the drive on the desktop and copy the A (base) and N (networking) packages on diskettes. You need at least those in order to use a serial cable to transfer the rest of the packages.

Now you need to enable NFS networking on the desktop, and allow the laptop to connect. You can give a temporary IP address to the laptop, like 192.168.1.11 that you need to add to your `/etc/exports` file on your desktop.

To link the two systems together, this is what you need to type on the laptop:

```
/usr/sbin/pppd -detach crtscts lock 192.168.1.11:192.168.1.10 /dev/ttyS1 115200
```

And this on the PC:

```
/usr/sbin/pppd -detach crtscts lock 192.168.1.10:192.168.1.11 /dev/ttyS1 115200
```

This is assuming the cable is linked to `ttyS1` (COM2) on both systems.

With NFS, you can mount the CD-ROM drive remotely and tell the installation program to use a specific path to install the remaining packages. Mount the CD-ROM with a command like this:

```
mount -tnfs 192.168.1.10:/cdrom /mnt
```

Then run the installation program:

```
setup
```

and enter the new path for the packages files.

Tip 5: Swap and memory

One important setting in any protected mode operating system like Linux is the swap space. In the installation, you will need to create a swap partition. A common question is what size should the partition be?

The proper size depends on 2 things: The size of your hard drive and the size of your RAM memory. The less RAM you have, the more swap you will need. Usually you will want to set your swap space size to be twice the RAM size, with a maximum of 128 megs. This of course requires you to have a hard drive with enough free space to create such a partition.

If you have 16 megs of RAM, making the swap space 32 megs or even 64 megs is very important. You will need it. If you have 128 megs of RAM on the other hand, you won't need much swap because the system will already have 128 megs to fill before using swap space. So a swap partition of 128 megs or even 32 megs could be enough.

If you don't select enough swap, you may add more later.

Tip 6: More swap with a swap file

You installed a new Linux system, but forgot to set enough swap space for your needs. Do you need to repartition and reinstall? No, the swap utilities on Linux allow you to make a real file and use it as swap space.

The trick is to make a file and then tell the swapon program to use it. Here's how to create, for example, a 64 megs swap file on your root partition (of course make sure you have at least 64 megs free):

```
dd if=/dev/zero of=/swapfile bs=1024 count=65536
```

This will make a 64 megs (about 67 millions bytes) file on your hard drive. You now need to initialize it:

```
mkswap /swapfile 65536  
sync
```

And you can then add it to your swap pool:

```
swapon /swapfile
```

With that you have 64 megs of swap added. Don't forget to add the swapon command to your startup files so the command will be repeated at each reboot.

Tip 7: Kernel size and modules

To configure Linux to detect a new hardware part, especially on a new kernel, you may need to recompile the kernel. If you add too many devices in the kernel configuration, you may get an error message telling you that the kernel is too big. The trick is to enable modules.

The kernel itself must be a certain size because it needs to be loaded in a fixed memory size. This is one reason why modules can be very handy. If you enable modules, you will need to make them:

```
make modules
```

and install them:

```
make modules_install
```

Then using the modprobe utility you can load selected modules on bootup. This way the kernel will be smaller and will compile with no error.

Tip 8: The boot prompt

The Linux system uses a program called LILO to boot itself. This is the LInux LOader, and will load a kernel and can pass various parameters. This is what the "boot:" prompt is for.

At the "boot:" prompt, you can enter a lot of parameters. You can send parameters to drivers like the ethernet driver, telling it at which IRQ the ethernet card is located, or you can pass parameters to the kernel, like memory size or what to do in a panic. Reading the LILO manual will tell you all of the nice things LILO can be used for.

Note that for device drivers compiled as modules, you need to pass values when you load these drivers, and not on the "boot:" prompt.

Tip 9: Wrong memory size found

The Linux kernel will detect various settings from your computer configuration. This includes the size of memory you have. In some cases, it will find the wrong size. For example, it could find only 64 megs of memory when in fact you have 128 megs.

The trick here is to specify the amount of RAM memory you have with the "mem=" parameter. Here is what you would type when your system boots if you have 128 megs of memory:

```
LILO boot: linux mem=128M
```

This will tell LILO to load the linux kernel with 128 megs of memory.

Tip 10: Master boot record and LILO

What is the master boot record (MBR) and why does LILO erase the old boot loader? Every hard drive has a top space called the MBR where the BIOS will try to load an operating system. Every system has its own loader. DOS has DOS-MBR, Windows NT has the NTLDR and Linux has LILO.

When you install LILO, you can install it in the MBR or in a boot record for the Linux partition. If you want to keep your current boot loader, you can select the Linux partition, and make sure it is the active partition in fdisk. This way you will be able to boot to LILO, and then boot the old loader from the MBR.

If you plan on only using Linux on your system, you can tell LILO to boot right into Linux and not display a "boot:" prompt, and you can install it in the MBR.

Tip 11: LILO can't find a kernel on a big drive

On some big hard drives, LILO can have problems loading your kernel. The problem is because the hard drive has more than 1024 cylinders.

The trick is to make sure your kernel is in the first 1024 cylinders so LILO can find it. The way to do this is to make a small /boot partition at the beginning of the drive, and make sure the kernel is in the /boot directory. You can set the partitions in fdisk, and select the right path for the kernel in /etc/lilo.conf so LILO knows where it is. When you compile your kernel, simply move the new kernel in that directory so LILO can load it.

Tip 12: X Window configuration options

Each Linux distribution has its own X Window configuration program. XFree86 also has a text-based configuration program which is complex to use. But what if both the distribution program and `xf86config`, the text-based configuration for XFree86, do not seem to do what you need? XFree86 also comes with a graphical configuration tool.

The name of the graphical program is `XF86Setup`. This will launch a graphical window and allow you to configure the X Window Server. So if you don't like the console configuration programs, you can use this one:



Tip 13: Allowing users to mount drives

By default, Linux will not allow users to mount drives. Only root can do it, and making the mount binary `suid` root is not a good idea. With a special command in the `/etc/fstab` file, you can change that.

This is a typical line for the `fd0` (A:) drive in `/etc/fstab`:

```
/dev/fd0          /mnt              auto              noauto,user 1    1
```

The keywords here are `noauto` and `user`. `Noauto` tells mount not to try to mount a diskette on boot, and `user` allows any user to mount the drive into `/mnt`. The `auto` keyword is also interesting. It tells mount to try to find out which file system is on the diskette. You could also use `msdos` or `ext2`.

Tip 14: Allowing users to run root programs

When a user starts a command, it runs with the permissions of that user. What if you want to allow them to run some commands with root permissions? You can, and that's called `suid`.

You can set a command to be `suid` root with the `chmod` command. This will make it run as root even if a user starts it. Here is how to set `mybin` `suid` root:

```
chmod +s mybin
```

Note that you must be very careful with this option. If the command has any security hole, or allows the user to access other files or programs, the user could take over the root account and the whole system.

Tip 15: Linux and NT booting

Some people choose to have both Windows NT and Linux on the same system. Windows NT has its own boot loader called NTLDR and Linux has LILO. Which should go on the MBR?

The safest way is to install Windows NT first, and give it the MBR. Then, when you install Linux, tell LILO to install on the Linux partition. Also set the Linux partition as the active partition. When the system boots, LILO will be loaded, and if you want to boot Windows NT, then LILO can load the MBR with NTLDR in it. There is a mini HOWTO text covering this subject available at <http://metalab.unc.edu/pub/Linux/docs/HOWTO>.

Tip 16: Annoying boot messages

When recompiling your kernel, you might end up seeing strange messages on bootup like:

```
modprobe: cannot find net-pf-5
modprobe: cannot find char-major-14
```

These are messages from the modules loader telling you that he can't find specific modules. This usually happens when you compile modules, but modprobe tries to load modules that were not compiled and it can't find them. The way to remove those messages is to set the modules to off. In the file `/etc/conf.modules` you may want to add:

```
alias net-pf-5 off
alias char-major-14 off
```

This will stop modprobe from trying to load them. Of course you could also try to resolve the problem by compiling the modules and make sure modprobe knows where they are.

- [**Bike Fit: Optimise Your Bike Position for High Performance and Injury Avoidance pdf, azw \(kindle\), epub**](#)
- [click Understanding Oil Prices: A Guide to What Drives the Price of Oil in Today's Markets \(The Wiley Finance Series\)](#)
- [read Learning Internet of Things](#)
- [download this I know: Sarah's Confession pdf, azw \(kindle\), epub](#)
- [read online The Rise of the Roman Empire \(Penguin Classics\) online](#)

- <http://dadhoc.com/lib/Why-Europe---The-Medieval-Origins-of-Its-Special-Path.pdf>
- <http://transtrade.cz/?ebooks/Eon--Eon--Book-1-.pdf>
- <http://growingsomeroots.com/ebooks/Learning-Internet-of-Things.pdf>
- <http://patrickvincitore.com/?ebooks/this-I-know--Sarah-s-Confession.pdf>
- <http://jaythebody.com/freebooks/Big-Bob-Gibson-s-BBQ-Book--Recipes-and-Secrets-from-a-Legendary-Barbecue-Joint.pdf>