

Mandrake Linux 8.2

Server Reference Manual

MandrakeSoft

March 2002

<http://www.mandrakelinux.com/>

Mandrake Linux 8.2 : Server Reference Manual
by MandrakeSoft

Copyright © 1999-2002 by **MandrakeSoft**

Table of Contents

Preface	i
1. Legal Notice	i
2. About Mandrake Linux	i
2.1. Contact Mandrake Community	i
2.2. Support Mandrake	ii
2.3. Purchasing Mandrake Products	ii
3. About This Server Reference Manual	ii
4. Authors And Translators	iii
5. Tools Used in The Making of This Manual	iii
6. Note From The Editor	iv
7. Conventions Used in This Book	iv
7.1. Typing Conventions	iv
7.2. General Conventions	v
I. Common Services Configuration Wizards	1
1. Server Configuration Wizards	1
1.1. Foreword	1
1.2. Network Configuration Wizard	2
1.3. DHCP Server	6
1.4. Domain Name Server	7
1.5. Adding a DNS Entry	8
1.6. Postfix server configuration	9
1.7. Samba Server Configuration	10
1.8. Firewalling configuration	12
1.9. Web Server Configuration	14
1.10. FTP server configuration	15
1.11. News Server configuration	16
1.12. Proxy Server Configuration	17
1.13. Time Configuration	20
2. Configuring Masqueraded Clients	23
2.1. Linux Box	23
2.2. Windows XP Box	25
2.3. Windows 95 or Windows 98 Box	25
2.4. Windows NT or Windows 2000 Box	28
2.5. DOS Box Using The NCSA Telnet Package	32
2.6. Windows For Workgroup 3.11	33
2.7. MacOS Box	33
2.8. OS/2 Warp Box	35
II. In-Depth Configuration of Common Services	41
3. Internet/Intranet Web Server	41
3.1. Installation	41
3.2. Step-by-Step Configuration Example	41
3.3. Advanced Configuration	44
4. Postfix Mail Server	47
4.1. Installation	47
4.2. Step-by-Step Configuration Example	47
4.3. Advanced Configuration	48
5. Incoming Mail Server: POP And IMAP	51
5.1. Foreword	51
5.2. Installation	51
5.3. Step-by-Step Configuration Example	51
5.4. Advanced Configuration	53
6. Resource Sharing	55
6.1. Resource Sharing: Samba	55
6.2. Resource Sharing: FTP	57
6.3. Resource Sharing: NFS	59
7. MySQL Database Server	63
7.1. Getting Started	63
7.2. Creating a User For The Database	63
7.3. Creating a Database	63

7.4. Creating a Table	64
7.5. Managing Data in a Table	65
8. NIS Client And Server	67
8.1. Installation	67
8.2. Step-by-Step Configuration	67
8.3. Client Advanced Configuration	69
9. BIND DNS Server	71
9.1. Installation	71
9.2. Step-by-Step Configuration Example	71
9.3. Advanced Configuration	74
III. Applied Theory	81
10. Security Under GNU/Linux	81
10.1. Preamble	81
10.2. Overview	81
10.3. Physical Security	84
10.4. Local Security	89
10.5. Files and File-System Security	90
10.6. Password Security and Encryption	95
10.7. Kernel Security	100
10.8. Network Security	103
10.9. Security Preparation (before you go on-line)	109
10.10. What To Do During and After a Breaking	111
10.11. Security Sources	112
10.12. Frequently Asked Questions	114
10.13. Conclusion	116
Security-related terms	116
11. Networking Overview	119
11.1. Copyright	119
11.2. How to Use This Chapter	119
11.3. General Information About Linux Networking	120
11.4. Generic Network Configuration Information	121
11.5. Ethernet Information	125
11.6. IP-Related Information	128
11.7. Using Common PC Hardware	128
11.8. Other Network Technologies	130
11.9. Cables And Cabling	139
A. GNU Free Documentation License	143
A.1. GNU Free Documentation License	143
0. PREAMBLE	143
1. APPLICABILITY AND DEFINITIONS	143
2. VERBATIM COPYING	144
3. COPYING IN QUANTITY	144
4. MODIFICATIONS	144
5. COMBINING DOCUMENTS	145
6. COLLECTIONS OF DOCUMENTS	146
7. AGGREGATION WITH INDEPENDENT WORKS	146
8. TRANSLATION	146
9. TERMINATION	146
10. FUTURE REVISIONS OF THIS LICENSE	146
A.2. How to use this License for your documents	146
Glossary	149

List of Tables

List of Figures

1-1. An Example of an Internal Network.....	1
1-2. Accessing the Wizards through the Control Center.....	2
1-3. Wizdrake Warning.....	2
1-4. Enter the Host Name for this Machine.....	3
1-5. Select the Device Connected to your Internal Network.....	3
1-6. Enter the Network Address for your Private Network.....	4
1-7. Enter the IP Address for your Server.....	4
1-8. Choose the Network Device and IP Address of the Gateway to the Internet.....	5
1-9. Confirm the Configuration.....	6
1-10. Choose the Range of Addresses Available via your DHCP Server.....	6
1-11. Enter the Addresses for the Name Servers.....	7
1-12. specify the name to associate to a static IP address.....	8
1-13. Enter your mail domain name.....	9
1-14. Enter the name of the SMTP server.....	10
1-15. Share files and printers?.....	10
1-16. Choose the work group for your shares.....	11
1-17. What name for your <i>Samba</i> server?.....	11
1-18. How tight do you want your firewall to be?.....	12
1-19. Which network interface is the gateway to the Internet?.....	13
1-20. Where should your web server be visible from?.....	14
1-21. Where should your FTP server be visible from?.....	15
1-22. Which server do you want to fetch newsgroups from?.....	16
1-23. How often do you want to check for news?.....	16
1-24. Choose the Proxy Port.....	17
1-25. Choose the Cache Sizes.....	17
1-26. Select Access Control Policy.....	18
1-27. Restrict access to a particular subnetwork.....	19
1-28. Use an upper level proxy?.....	19
1-29. What method do you want for time synchronization?.....	20
1-30. Choose your time servers.....	20
2-1. Reconfiguring The Local Network With draknet.....	24
2-2. Setting up The Gateway With draknet.....	24
2-3. Setting up The Gateway With Windows XP.....	25
2-4. The Network Icon Under Windows 95.....	25
2-5. The Network Configuration Panel Under Windows 95.....	25
2-6. The TCP/IP Configuration Panel Under Windows 95.....	26
2-7. The Gateway Configuration Panel Under Windows 95.....	27
2-12. Accessing The TCP/IP Control Panel.....	33
2-13. Automatic Configuration of Internet Access For MacOS.....	34
2-14. Manual Configuration of Internet Access For MacOS.....	34
3-1. Webmin's Main Apache Module Screen.....	41
3-2. Apache' Default Server Configuration Screen.....	41
3-3. Document Options Section.....	42
3-4. Alias And Redirection Section.....	43
3-5. SSL Options Section.....	44
3-6. The Configuration Screen of Apache Processes.....	44
3-7. Directory Limitations Using .htaccess.....	45
4-1. Postfix Module's Start-Up Screen.....	47
4-2. Postfix's Main Configuration Screen.....	47
5-1. xinetd Module's Start-Up Screen.....	51
5-2. POP3 Configuration Module.....	52
6-1. The Samba Module's Main Window.....	55
6-2. Configuring The Common Networking Options.....	55
6-3. Setting The Authentication Method.....	56
6-4. Configuring Your Sharing Entries.....	56
6-5. WU-FTP's Main Configuration Page.....	57
6-6. Wu-FTP Banner And Messages.....	58

6-7. Anonymous FTP Configuration Page	58
6-8. Starting The NFS Configuration.....	60
6-9. Creating NFS Export.....	60
6-10. Creating NFS Mount Points.....	61
6-11. Configuring NFS Mount Point.....	61
7-1. Creating a MySQL User.....	63
7-2. Creating a MySQL Database.....	63
7-3. Creating a MySQL Table.....	64
7-4. Modifying a MySQL Table.....	64
7-5. Managing Your Data.....	65
8-1. NIS Server.....	67
8-2. NIS Client.....	68
9-1. Files And Directories.....	71
9-2. Creating a Forward Master Zone.....	71
9-3. Creating a Reverse Master Zone.....	72
9-4. Adding Machine Names.....	72
9-5. Starting Bind.....	73
9-6. Apply Changes to Bind.....	73
9-7. Configuring The Client.....	73
9-8. The BIND 9 Administrator Reference Manual Through Webmin.....	75
11-1. A Dynamic Routing Example.....	124
11-2. The NULL-Modem Cabling.....	139
11-3. 10base2 Ethernet Cabling.....	140

Preface

1. Legal Notice

This manual is protected under **MandrakeSoft** intellectual property rights. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with the invariant sections being *About Mandrake Linux*, page i, with the front-cover texts being listed below, and with no Back-Cover Texts. A copy of the license is included in the section *GNU Free Documentation License*, page 143.

Front-cover texts:

MandrakeSoft March 2002
<http://www.mandrakesoft.com/>
Copyright © 1999,2000,2001,2002 by MandrakeSoft S.A. and MandrakeSoft Inc.

“Mandrake”, “Mandrake Linux” and “MandrakeSoft” are registered trademarks of **MandrakeSoft S.A.**; Linux is a registered trademark of Linus Torvalds; *UNIX* is a registered trademark of The Open Group in the United States and other countries. All other trademarks and copyrights are the property of their respective owners.

2. About Mandrake Linux

Mandrake Linux is a *GNU/Linux* distribution supported by **MandrakeSoft S.A.** **MandrakeSoft** was born in the Internet in 1998 with the main goal to provide an easy-to-use and friendly *GNU/Linux* system. The two pillars of **MandrakeSoft** are open-source and collaborative work.

2.1. Contact Mandrake Community

Following are various Internet links pointing you to various **Mandrake Linux** related sources. If you wish to know more about the **MandrakeSoft** company, connect to its web site (<http://www.mandrakesoft.com/>). There is also the **Mandrake Linux** distribution (<http://www.mandrakelinux.com/>) web site and all its derivatives.

First of all, **MandrakeSoft** is proud to present its new open help platform. MandrakeExpert (<http://www.mandrakeexpert.com/>) isn't just another web site where people help others with their computer problems in exchange for up-front fees, payable regardless of the quality of the service received. It offers a new experience based on trust and the pleasure of rewarding others for their contributions.

In addition, MandrakeCampus (<http://mandrakecampus.com/>) provides the *GNU/Linux* community with open education and training courses on all open-software-related technologies and issues. It also gives teachers, tutors and learners a place where they can share knowledge.

There is a site for the “mandrakeholic” called Mandrake Forum (<http://www.mandrakeforum.com/>): a primary site for **Mandrake Linux** related tips, tricks, rumors, pre-announcements, semi-official news, and more. This is also the only interactive web site hosted by **MandrakeSoft**, so if you have something to tell us, or something you want to share with other users, search no longer: this is a place to do it!

In the philosophy of open source, **MandrakeSoft** is offering many means of support (<http://www.mandrakelinux.com/en/ffreesup.php3>) for the **Mandrake Linux** distributions. You are invited in particular to participate in the various Mailing lists (<http://www.mandrakelinux.com/en/flists.php3>), where the **Mandrake Linux** community demonstrates its vivacity and keenness.

Finally, do not forget to connect to MandrakeSecure (<http://www.mandrakesecure.net/>). This site gathers all security related material about **Mandrake Linux** distributions. You'll notably find there security and bug advisories, as well as security and privacy-related articles. A must for any server administrator or user concerned about security.

2.2. Support Mandrake

By popular request, **MandrakeSoft** proposes that its happy customers make a donation (<http://www.mandrakelinux.com/donations/>) to support the forth-coming developments of the **Mandrake Linux** system. Your contribution will help **MandrakeSoft** provide its users with an ever better distribution, ever safer, easier, up-to-date, and with more supported languages.

For the many talented, your skills will be very useful for one of the many tasks required in the making of a **Mandrake Linux** system:

- Packaging: a *GNU/Linux* system is mainly made of programs picked up on the Internet. These programs have to be packaged so that they will hopefully work together.
- Programming: there are many many projects directly supported by **MandrakeSoft**: find the one that most appeals to you, and offer your help to the main developer.
- Internationalization: translation of the web pages, programs and their respective documentation.
- Documentation: last but not least, the book you are currently reading requires a lot of effort to stay up-to-date with the rapid evolution of the system.

Consult the contributors page (<http://www.mandrakesoft.com/labs/>) to learn more about the way you can contribute to the evolution of **Mandrake Linux**.

On August 3rd 2001, after having established itself as one of the world leaders in Open Source and *GNU/Linux* software, **MandrakeSoft** became the first *Linux* company listed on a European stock market. Whether you're already a **MandrakeSoft** shareholder or wish to become one, our Investor pages (<http://www.mandrakesoft.com/company/investors>) provide the best financial information related to the company.

2.3. Purchasing Mandrake Products

For **Mandrake Linux** fans wishing to benefit from the ease of on-line purchasing, **MandrakeSoft** now sells its products worldwide from its MandrakeStore (<http://www.mandrakestore.com/>) e-commerce web site. You will find not only **Mandrake Linux** software — operating systems and network tools (Single Network Firewall), but also special subscription offers, support, third party software and licenses, training documentation, *GNU/Linux* related books, as well as other goodies related to **MandrakeSoft**.

3. About This Server Reference Manual

Welcome, and thank you for using **Mandrake Linux**! This book is aimed at those of you who wish to use their **Mandrake Linux** system as a server. This book is divided into 3 parts:

- *Common Services Configuration Wizards*: an introduction to the **Mandrake Linux**-specific server wizards, which will help you to configure different servers such as DNS, SMTP, web, and FTP. Then we tackle the configuration of masquerading clients.

The first chapter is an in-depth look at the different services you can configure with your **Mandrake Linux** box through the *Mandrake Control Center*. After you are finished with this chapter, you should be able to configure and fine-tune services such as DHCP, DNS or *Postfix*.

Next, we cover the configuration of masqueraded clients through a **Mandrake Linux** box, allowing to work in interconnected networks using many platforms such as **Microsoft DOS**, *Windows 9x* and *Windows NT*, Novell Netware, SCO OpenServer and Solaris. In order for this chapter to be useful, you need a well configured LAN since we focus on the gateway, not DNS or connection problems.

- *In-Depth Configuration of Common Services*: we explore the different *Webmin* modules which will help you to configure available services:
 - the “Internet/Intranet Web Server”, page 41 chapter discusses the *Apache* server;
 - the “Postfix Mail Server”, page 47 chapter explains how to configure a *Postfix* server to send e-mails through the SMTP protocol;
 - the POP and IMAP protocols used to retrieve mail are explored in the “Incoming Mail Server: POP And IMAP”, page 51 chapter;

- in “*Resource Sharing*”, page 55, we focus on the *Samba* and NFS protocols to share files in a multiple platform environment and *GNU/Linux*-only network, respectively. The usage of *WU-FTP* is also detailed;
 - next, “*MySQL Database Server*”, page 63 explores solely the configuration of a *MySQL* database server: creating, modifying and managing data in tables;
 - the “*NIS Client And Server*”, page 67 chapter discusses remote user management and explains the configuration of both the server and client side of NIS;
 - in the last chapter of this part, “*BIND DNS Server*”, page 71, we expose the DNS’s strong points, detailing *BIND* as a name server.
- *Applied Theory*: finally, we discuss two topics in the last part: security and networking:
- the “*Security Under GNU/Linux*”, page 81 chapter, based on a *HOWTO* by Kevin Fenzi and Dave Wreski, gives many pointers to system administrators on how to better secure their networks. With security being one of the main focus of our Internet-driven world, this is a **mandatory** reading.
- at last, the “*Networking Overview*”, page 119 chapter is based on a *HOWTO* by Joshua D. Drake (aka POET) and gives many resources to sort out your networking needs. It pinpoints *GNU/Linux*-compatible hardware and explains fundamental networking services such as DHCP, DNS, etc.

Have fun and start that coffee machine!

4. Authors And Translators

The following people contributed to the making of the **Mandrake Linux** manuals:

- Yves Bailly
- Camille Bégnis
- Marco De Vitis
- Francis Galiègue
- Hinrich Göhlmann
- Carsten Heiming
- Fabian Mandelbaum
- Joël Pomerleau
- Peter Rait
- Roberto Rosselli Del Turco
- Christian Roy
- Stefan Siegel

These people also participated at various degrees: Philippe Ambon, Jay Beale, Hoyt Duff, Joël Flores-Carpio, Giuseppe Ghibò, Till Kampetter, Alexander Sasha Kirillov, Damien Dams Krotkine, Robert Kulagowski, Kevin Lecouvey, François Pons, Guillaume Poulin, Pascal Pixel Rigaux, John Rye and Laurence Tricon.

5. Tools Used in The Making of This Manual

This manual was written in *DocBook*. *perl* and *GNU make* were used to manage the sets of files involved. The XML source files were processed by *openjade* and *jadetex* using custom Norman Walsh’s stylesheets. Screen-shots were taken using *xwd* or *GIMP* and converted with *convert* (from the *ImageMagick* package). All this software is available on your **Mandrake Linux** distribution, and all parts of it are free software.

6. Note From The Editor

As you may notice while you go from one chapter to another, this book is a composite document from various authors. Even though much care has been taken in insuring the technical and vocabulary consistency, the style of each author is obviously preserved.

Some of the authors write in English even though it is not their native language. Therefore, you may notice strange sentence constructions; do not hesitate to let us know if something is not clear to you.

In the open-source philosophy, contributors are always welcomed! You may provide help to this documentation project by many different means. If you have a lot of time, you can write a whole chapter. If you speak a foreign language, you can help with the internationalization of this book. If you have ideas on how to improve the content, let us know - even advice on typos is welcomed!

For any information about the **Mandrake Linux** documentation project, please contact the documentation administrator (<mailto:documentation@mandrakesoft.com>).

7. Conventions Used in This Book

7.1. Typing Conventions

In order to clearly differentiate special words from the text flow, the documentation team uses different renderings. The following table shows an example of each special word or group of words with its actual rendering and what this means.

Formatted Example	Meaning
<i>inode</i>	This formatting is used to stress a technical term, explained in the <i>Glossary</i> .
<code>ls -lta</code>	Indicates commands or arguments to a command. This formatting is applied to commands, options and file names. Also see the section about “ <i>Commands Synopsis</i> , page v”.
<code>ls(1)</code>	Reference to a man page. To get the page in a <i>shell</i> (or command line), simply type <code>man 1 ls</code> .
<code>\$ ls *.pid imwheel.pid</code>	The documentation team uses this formatting for text snapshots of what you may see on your screen. It includes computer interactions, program listings, etc.
<code>localhost</code>	This is literal data that does not generally fit in with any of the previously defined categories. For example, a key word taken from a configuration file.
<i>Apache</i>	This is used for application names. The example used is not a command name but, in particular contexts, the application and command name may be the same but formatted in different ways.
<u>Files</u>	This is used for menu entries or graphical interface labels in general. The underlined letter indicates the keyboard shortcut, if applicable.
<i>SCSI-Bus</i>	It denotes a computer part or a computer itself.
<i>Le petit chaperon rouge</i>	This formatting identifies foreign language words.
Warning!	Of course, this is reserved for special warnings in order to stress the importance of words; read out loud :-)



This icon highlights a note. Generally, it is a remark in the current context, giving additional information.



This icon represents a tip. It can be a general advice on how to perform a specific action, or a nice feature that can make your life easier.



Be very careful when you see this icon. It always means that very important information about a specific subject will be dealt with.

7.2. General Conventions

7.2.1. Commands Synopsis

The example below shows you the symbols you will find when the writer describes the arguments of a command:

```
command <non literal argument>
[--option={arg1,arg2,arg3}] [optional arg. ...]
```

These conventions are standard and you may find them at other places such as the man pages.

The “<” (lesser than) and “>” (greater than) symbols denote a **mandatory** argument not to be copied verbatim, but to be replaced according to your needs. For example, <filename> refers to the actual name of a file. If this name is `foo.txt`, you should type `foo.txt`, and not <foo.txt> or <filename>.

The square brackets “[]” denote optional arguments, which you may or may not include in the command.

The ellipsis “...” mean an arbitrary number of items can be included.

The curly brackets “{ }” contain the arguments authorized at this specific place. One of them is to be placed here.

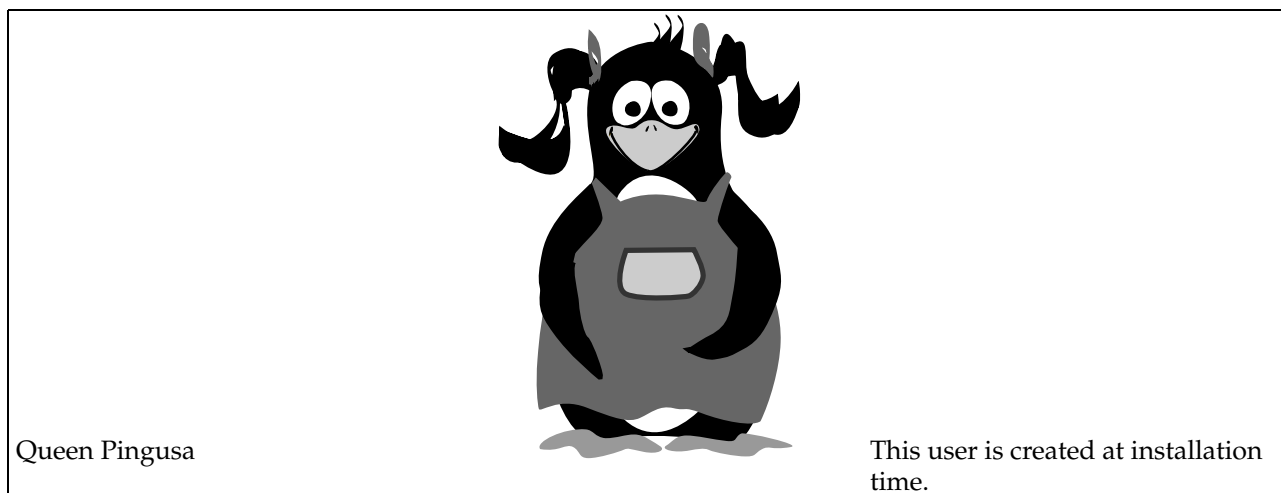
7.2.2. Special Notations

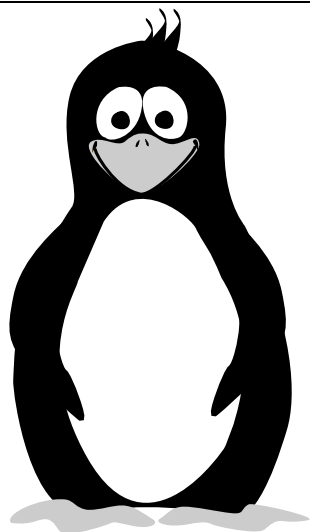
From time to time, you will be directed to press, for example, the keys Ctrl+R, which means you need to press and hold the Ctrl and tap the R key as well. The same applies for the Alt and Shift keys.

Also about menus, going to menu item **File→Reload user config (Ctrl+R)** means: click on the **File** text displayed on the menu (generally horizontal on the top of the window). Then in the pull-down menu, click on the **Reload user config** item. Additionally, you are informed that you can use the key combination Ctrl+R, as described above, to achieve the same result.

7.2.3. System Generic Users

Whenever possible, we used two generic users in our examples:





Peter Pingus

This user is created afterwards by the system administrator.

I. Common Services Configuration Wizards

Introduction to Server Wizards And Masqueraded Clients

This part is divided into two chapters: the first one details the **Mandrake Linux** server wizards, while the second goes deep into the configuration of masqueraded clients.

1. Introducing Server Wizards

Through wizards, the *“Server Configuration Wizards”*, page 1 chapter will help you to configure servers such as DNS (Domain Name Server), DHCP (Dynamic Host Configuration Protocol), *Samba*, web, FTP, etc.

2. Masquerading Clients

In the *“Configuring Masqueraded Clients”*, page 23 chapter, we will show you how to use **Mandrake Linux** with masquerading set up as a gateway to the outside world, in a multiple operating system network. The information in that chapter covers platforms such as Macintosh, **Microsoft DOS**, *Windows 9x* and *Windows NT*, Novell Netware, SCO OpenServer, Sun Solaris, and more.

Chapter 1. Server Configuration Wizards

1.1. Foreword

The configuration wizards, which come with **Mandrake Linux**, are made to configure a server located between a local network and the Internet. They give you the ability to make configurations quickly and efficiently for most common services in a local network as well as Internet Web and FTP services. In this chapter, we will suppose that your network is as shown in figure 1-1, and that **Mandrake Linux** is installed on the server. Configuring and bringing up the Internet connection (if you have one) is beyond the scope of this chapter.

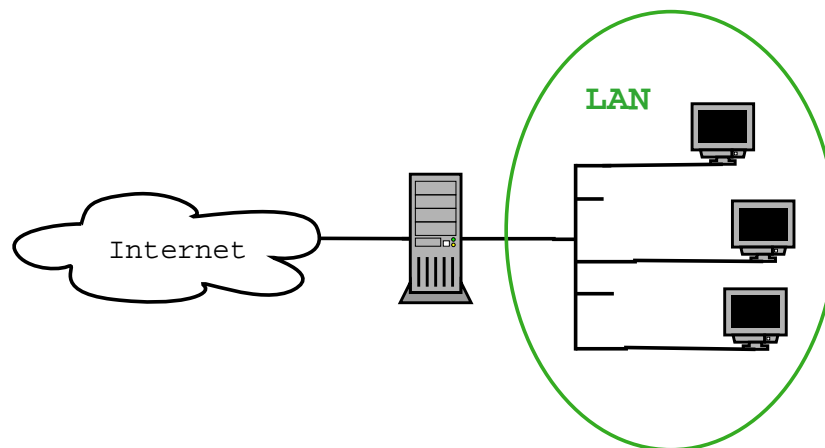


Figure 1-1. An Example of an Internal Network

Wizards can help you configure the following:

- *Network Configuration Wizard*, page 2: setting up the IP address of your network adapter (the one connected to the internal network), and the name of the server;
- *Domain Name Server*, page 7: configuring name resolutions for machines outside the private network;
- *Adding a DNS Entry*, page 8: identifying static machines names and IPs inside your local network;
- *Firewalling configuration*, page 12: as your server presumably acts as the gateway to the outside world, you will be able to set the firewalling policy here;
- *DHCP Server*, page 6: your server will be able to dynamically assign IP addresses to new machines on the network;
- *Samba Server Configuration*, page 10: if the server is to act as a file or print server for *Windows* machines, this wizard will help you setup public shared files and printers, and announce their names into the *Windows* network;
- *Web Server Configuration*, page 13: here you will be able to specify whether your web server will be reachable from the outside network, or from the internal network, or both;
- *News Server configuration*, page 15: you can make your server act as a local mirror of an external news server;
- *FTP server configuration*, page 14: as for the web server configuration, you will be able to specify from where your FTP server should be reachable;
- *Postfix server configuration*, page 9: configuring your mail domain for sending and receiving mail from the outside;
- *Proxy Server Configuration*, page 17: configuring your server to act as a Web proxy cache. This speeds up Web browsing while limiting the bandwidth on the Internet;
- *Time Configuration*, page 20: your machine can also give time to other machines using the NTP protocol (*Network Time Protocol*); this wizard will help you configure this service.



For experienced users: wizards are limited to configure only C class networks, and only the basic configuration is handled for each service. This should be enough for most situations, but if you wish for a more fine-tuned configuration, you will have to edit the configuration files by hand.

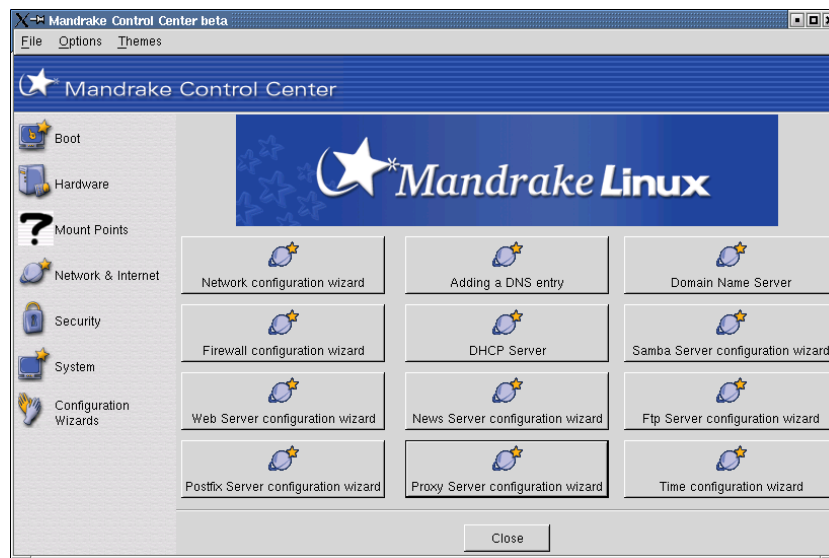


Figure 1-2. Accessing the Wizards through the Control Center

The server configuration wizards are available through the *Control Center*. When the *wizdrake* package is installed, a new menu entry appears in the **Mandrake Control Center** menu (figure 1-2).

You will be able to access wizards individually by clicking on the corresponding button. In this chapter wizards are described in no particular order, but the first one is required to be run first before other wizards.

1.2. Network Configuration Wizard

Normally you would have already configured networking during installation, but maybe you want to reconfigure it. For example, you have more than one network card, and the one configured during installation may not be the one connected to the internal network. This wizard will help you to configure this interface card and give a name to the machine.

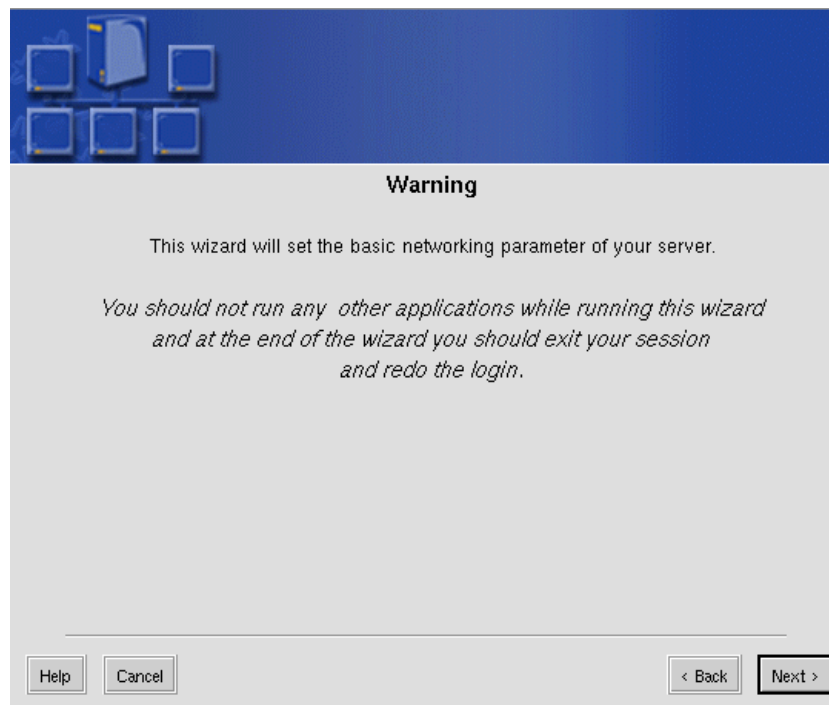


Figure 1-3. Wizrake Warning

If you need to reconfigure anything, you will have to restart your X session after all configuration steps are completed: as the machine will have changed its name and base address, the X authority mechanism will not let you run any more programs until you restart it. The wizard tells you about this, as shown in figure 1-3.

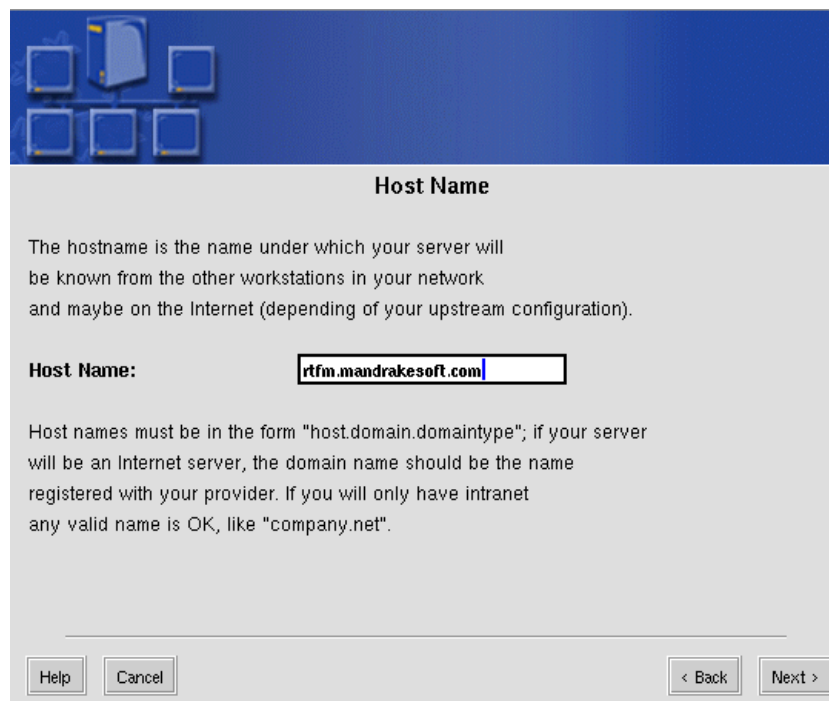


Figure 1-4. Enter the Host Name for this Machine

First, you have to enter the fully qualified domain name for this machine, if your network is directly connected to the Internet and you have a registered domain name. The server name should be part of this domain (figure 1-4).

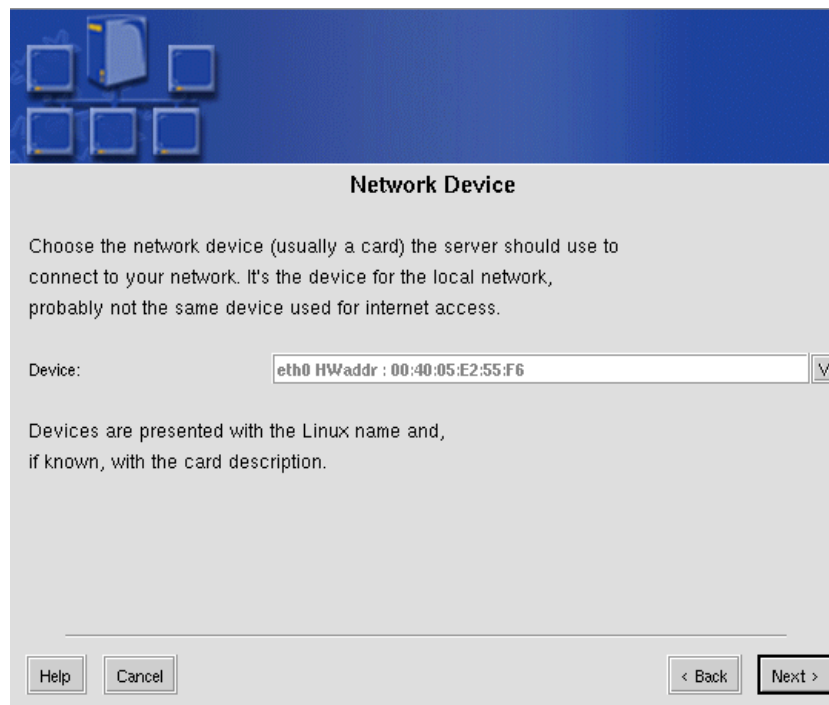


Figure 1-5. Select the Device Connected to your Internal Network

You have then to specify which network interface card is connected to your local network - **not** the interface card which gives you access to the outside! If you have several network interface cards in your machine, be sure to choose the correct one (figure 1-5).

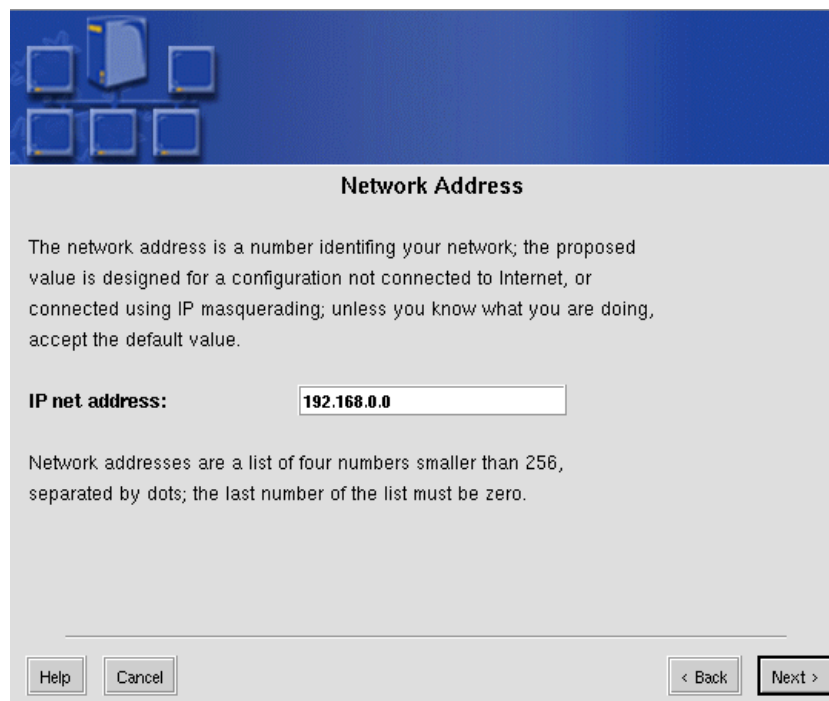
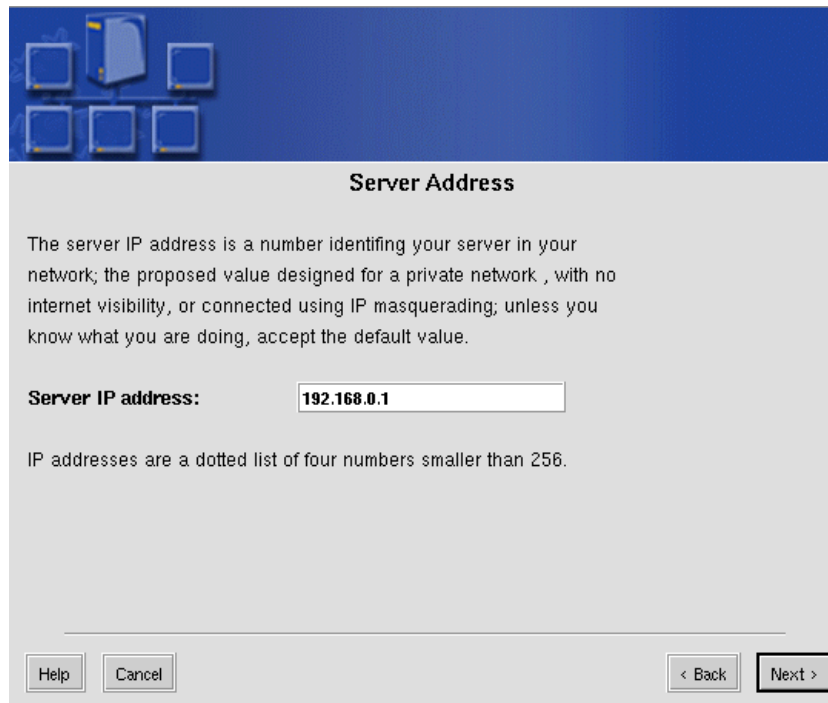


Figure 1-6. Enter the Network Address for your Private Network

Then you have to enter the network address: the wizards configures a C class network, this will always be of the form x.y.z.0. In the example shown in figure 1-6, the network address is private. Do not choose such a network address if your network is directly connected to the Internet!



Server Address

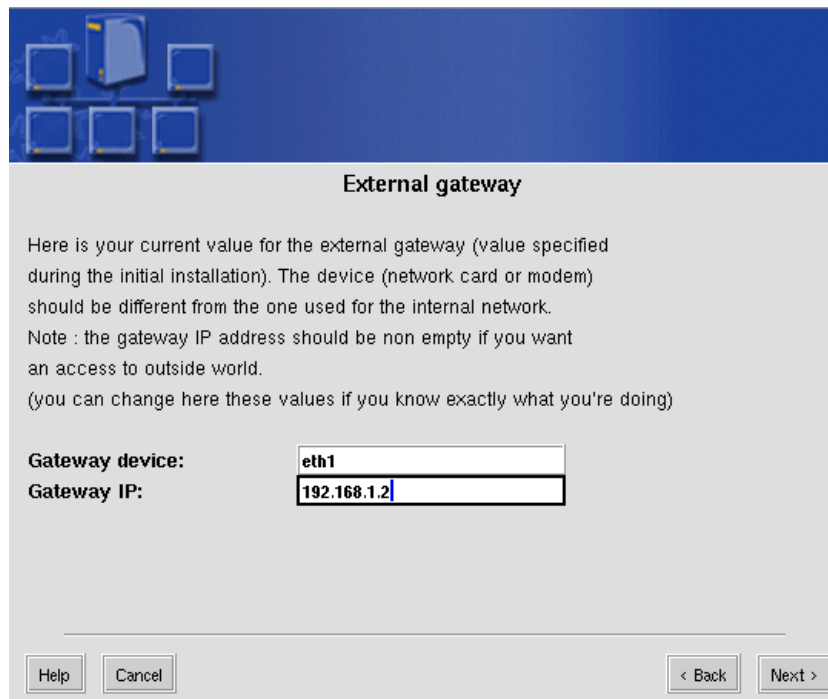
The server IP address is a number identifying your server in your network; the proposed value designed for a private network, with no internet visibility, or connected using IP masquerading; unless you know what you are doing, accept the default value.

Server IP address:

IP addresses are a dotted list of four numbers smaller than 256.

Figure 1-7. Enter the IP Address for your Server

You then have to enter the IP address for your server. Of course, this address must agree with the network address you have entered before. Also double check that the name you entered at the first step actually matches with the address you will enter here (figure 1-7).



External gateway

Here is your current value for the external gateway (value specified during the initial installation). The device (network card or modem) should be different from the one used for the internal network.

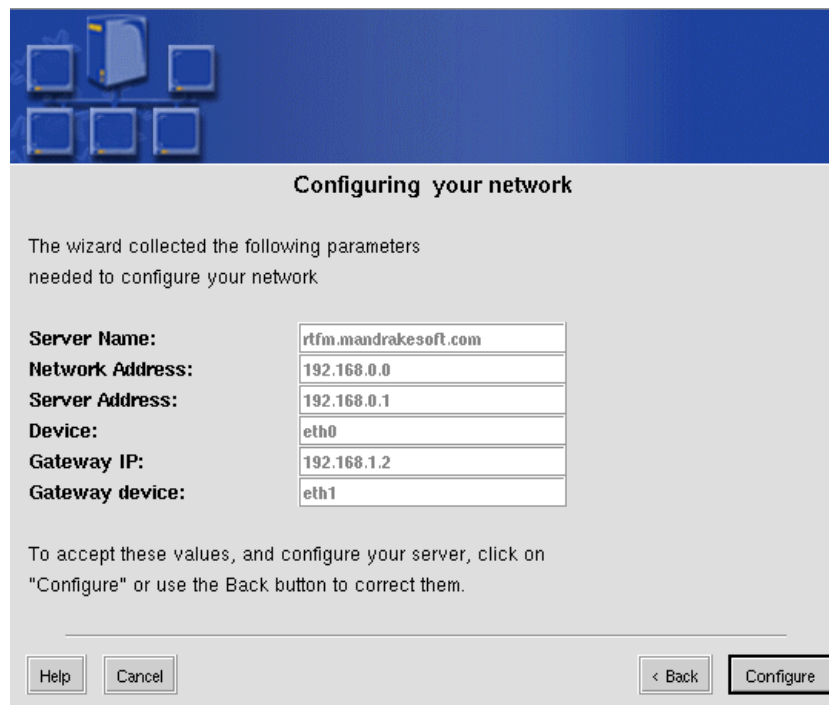
Note : the gateway IP address should be non empty if you want an access to outside world.
(you can change here these values if you know exactly what you're doing)

Gateway device:

Gateway IP:

Figure 1-8. Choose the Network Device and IP Address of the Gateway to the Internet

Finally, if you have a gateway, you need to specify the device and IP address of that gateway; which is the machine that gives you access to the Internet. The IP address field is only relevant if the gateway to the Internet has a fixed address: if you use a dialup connection by modem, for example, the device will be `ppp0` and you will not specify the IP address as it is dynamically assigned at connection time (figure 1-8). If you have no Internet access, just leave both fields blank.



Configuring your network

The wizard collected the following parameters needed to configure your network

Server Name:	rtfm.mandrakesoft.com
Network Address:	192.168.0.0
Server Address:	192.168.0.1
Device:	eth0
Gateway IP:	192.168.1.2
Gateway device:	eth1

To accept these values, and configure your server, click on "Configure" or use the Back button to correct them.

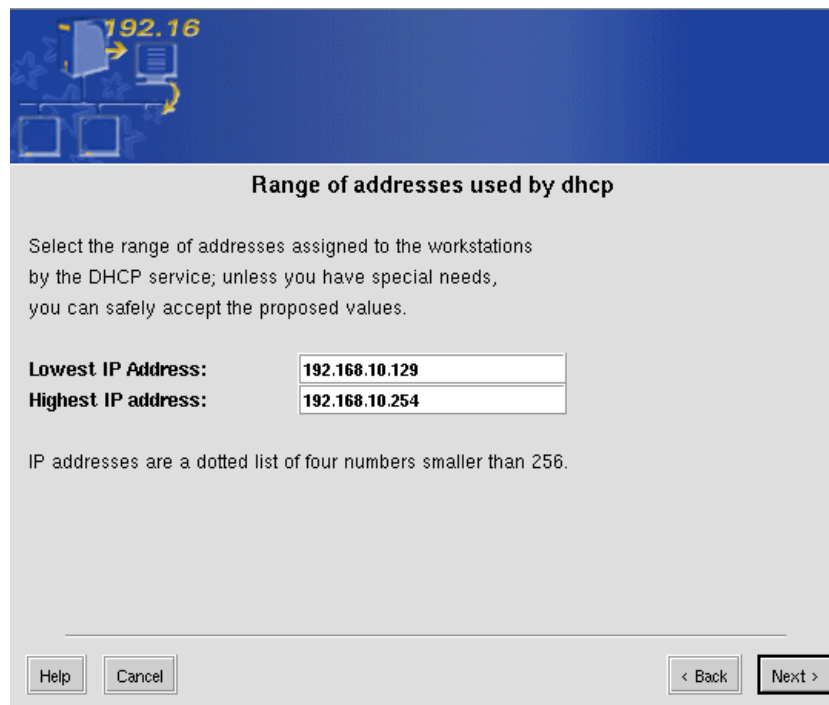
Help Cancel < Back **Configure**

Figure 1-9. Confirm the Configuration

You are now done with the basic configuration. The wizard will present you with all the information that you have entered, and you now need to select **Configure** if you are satisfied with the configuration, or you can select **Back** to go back and modify the configuration, as many times as you like (figure 1-9).

1.3. DHCP Server

DHCP stands for *Dynamic Host Configuration Protocol*. This protocol allows for new machines connecting to your local network to be automatically assigned an IP address, get the addresses of the name servers and the address of the gateway when relevant.



Range of addresses used by dhcp

Select the range of addresses assigned to the workstations by the DHCP service; unless you have special needs, you can safely accept the proposed values.

Lowest IP Address: 192.168.10.129

Highest IP address: 192.168.10.254

IP addresses are a dotted list of four numbers smaller than 256.

Help Cancel < Back Next >

Figure 1-10. Choose the Range of Addresses Available via your DHCP Server

All you have to do is specify the range of addresses that you want to have available via DHCP, as shown in figure 1-10.

1.4. Domain Name Server

DNS stands for *Domain Name System*. DNS is what allows you to specify a machine by its name instead of its IP address. This wizard, however, will not let you configure a DNS server (if you wish to do so, you will have to do it by hand), instead it will let you specify which external name servers you want to use.



→ www.mand
← 63.209.80

DNS Server Addresses

DNS will allow your network to communicate with the Internet using standard internet host names. In order to configure DNS, you must provide the IP address of primary and secondary DNS server; usually this address are given by your Internet provider.

Primary DNS Address: 192.168.10.11
Secondary DNS Address: 192.168.10.14

IP addresses are a dotted list of four numbers smaller than 256.

Help Cancel < Back Next >

Figure 1-11. Enter the Addresses for the Name Servers

The wizard asks you for two addresses (not names!) to two DNS servers (figure 1-11), but if you do not have any secondary name server just leave that field blank.



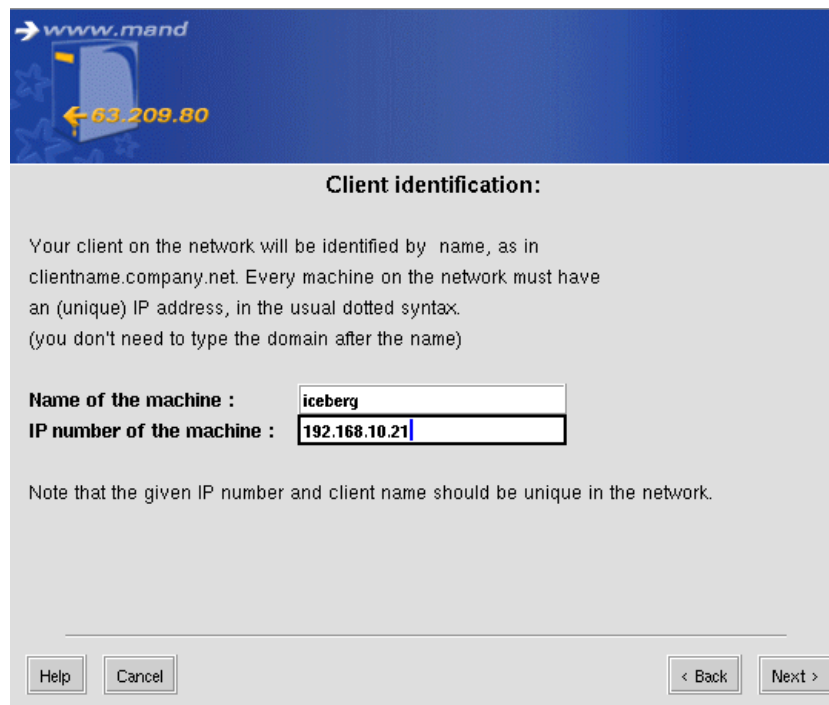
The DNS domain for the local network is automatically extracted from the domain name specified during the local server configuration during first step: *Network Configuration Wizard*, page 2

This step will also automatically configure a caching DNS server, speeding up Internet DNS requests made from the local network.

1.5. Adding a DNS Entry

To access a machine on your local network, it is generally easier to do it through a name than an IP address (just because it is easier to remember). For that it is enough to tell your name server which name match which IP.

All you have to do is specify for each machine its static IP (by opposition to dynamic DHCP IP), and the associated name, as shown in figure 1-12.



Client identification:

Your client on the network will be identified by name, as in clientname.company.net. Every machine on the network must have an (unique) IP address, in the usual dotted syntax. (you don't need to type the domain after the name)

Name of the machine :

IP number of the machine :

Note that the given IP number and client name should be unique in the network.

Help Cancel < Back Next >

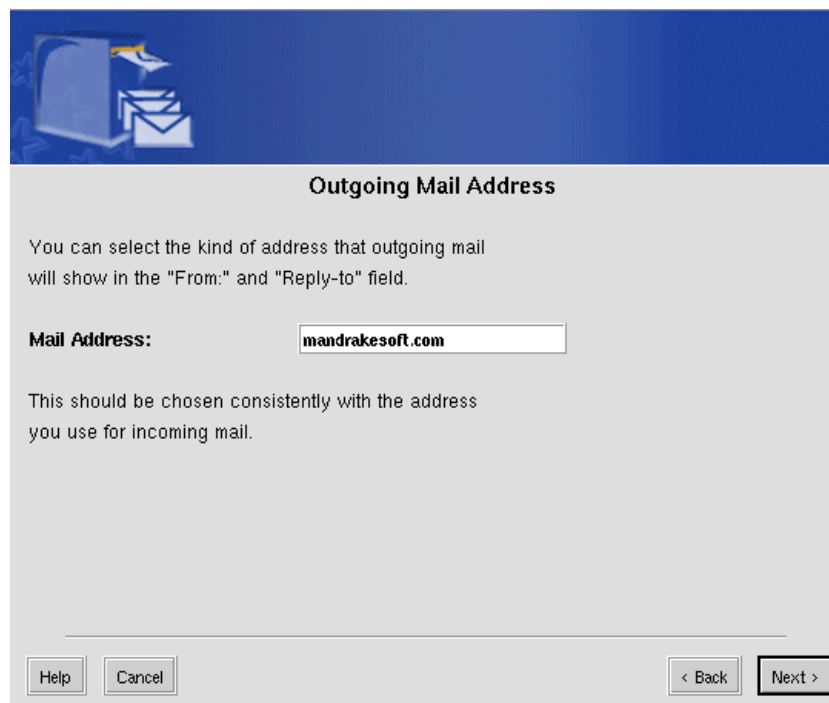
Figure 1-12. specify the name to associate to a static IP address

Launch the wizard as many times as needed for each machine you want to associate a formal name to.



The domain name for those clients is the one defined during previous step.

1.6. Postfix server configuration



Outgoing Mail Address

You can select the kind of address that outgoing mail will show in the "From:" and "Reply-to:" field.

Mail Address:

This should be chosen consistently with the address you use for incoming mail.

Help Cancel < Back Next >

Figure 1-13. Enter your mail domain name

This wizard will help you configure your incoming and outgoing mail. Your Internet service provider will normally have given you a mail domain. The first thing you will have to do is enter this mail domain name as shown in figure 1-13.



Mail addresses will be of the form <user>@<Mail server Domain>



Figure 1-14. Enter the name of the SMTP server

Then, you will have to enter the name of the mail server which will be responsible for delivering mail. Usually, this will be your provider's SMTP server (*Simple Mail Transfer Protocol*). You will enter this server name in the relevant field, as shown in figure 1-14.

1.7. Samba Server Configuration

Samba is a software package which allows *GNU/Linux* to act as a file and/or printer server for *Windows* machines. This wizard will only help you configure public shares, not private shares (if you wish to do so, refer to the documentation in the *Samba* package).

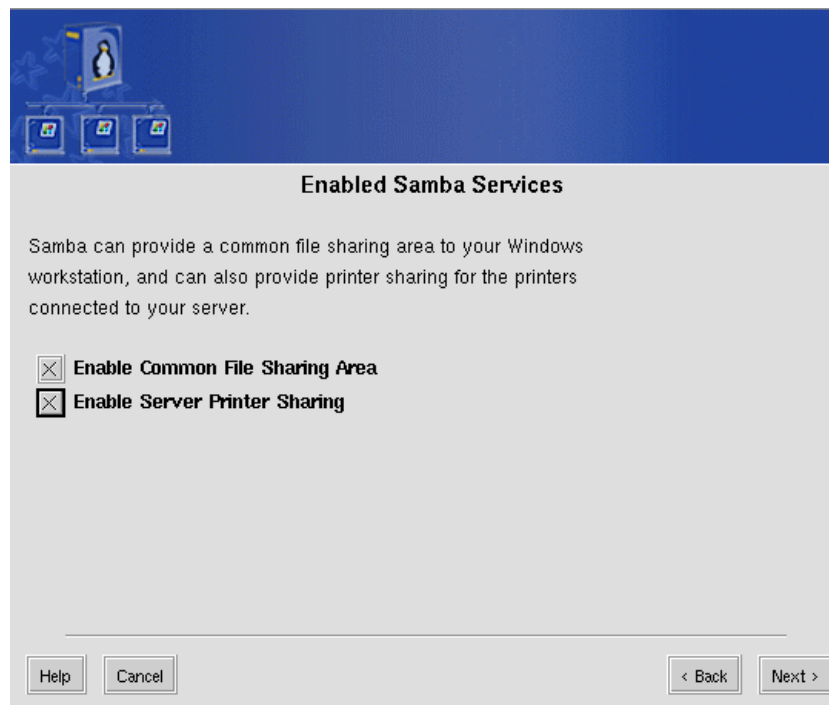


Figure 1-15. Share files and printers?

There are three steps in configuring *Samba*. The first will be to decide whether you want to enable files and printer sharing, as shown in figure 1-15.

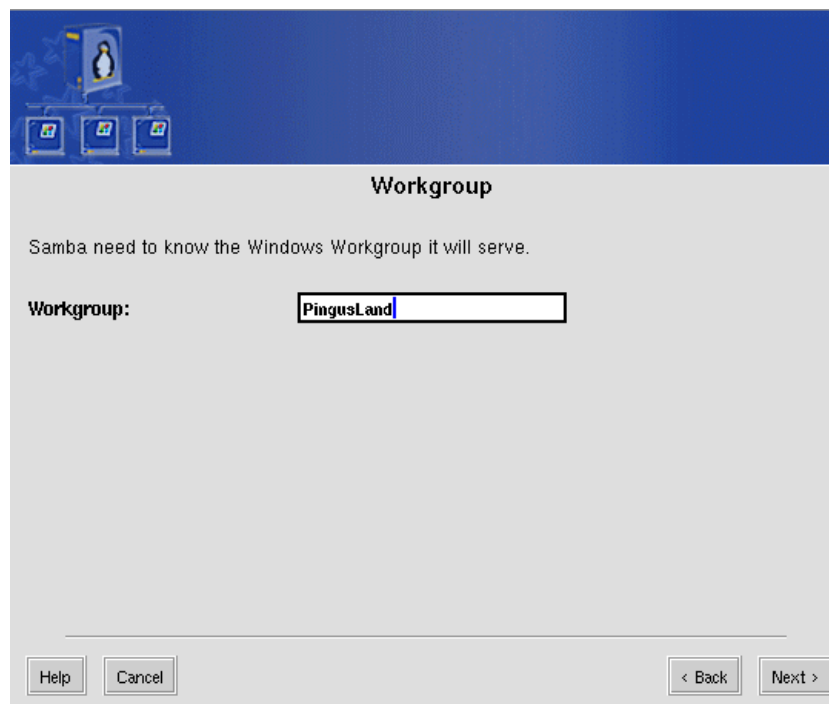


Figure 1-16. Choose the work group for your shares

Then you must enter the work group for which these shares will be available (figure 1-16). You can either create a new work group or choose an existing one, but if you don't know what to do, please refer to your system administrator.

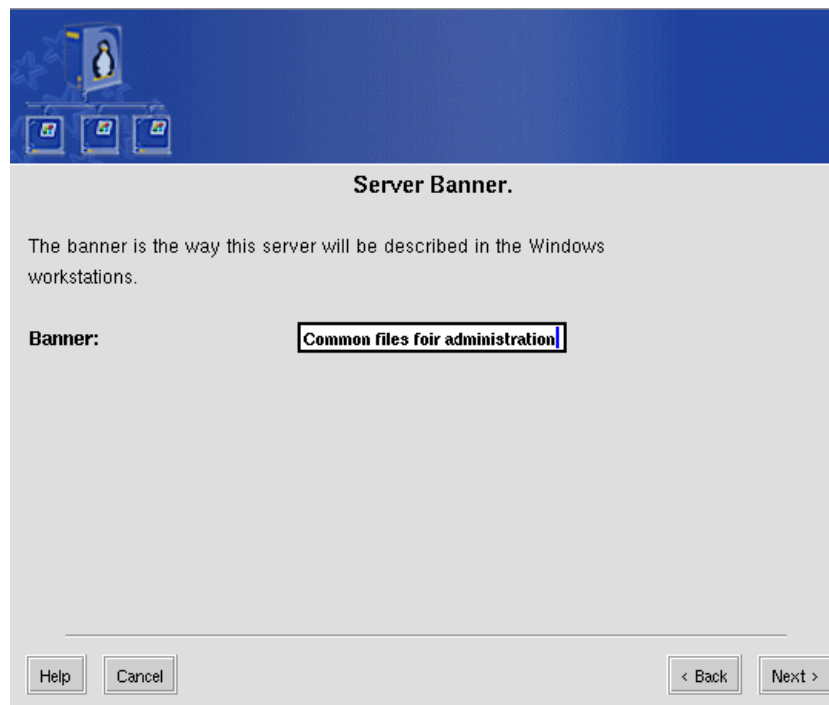


Figure 1-17. What name for your Samba server?

Finally, you will have to specify the name by which your **Mandrake Linux** server will be known to *Windows* machines, as shown in figure 1-17. You may choose whatever name you want.

1.8. Firewalling configuration



Figure 1-18. How tight do you want your firewall to be?

This wizard will help you to configure basic firewall rules. It should be enough for most configurations, but if this is not enough for you, you will have to configure it by hand. The different set of rules are shown in figure 1-18, choose the one that best fits your needs.

None

All ports are open, NAT is not activated (this level is not recommended if the server is to be connected to the Internet).

Low

All ports are open, NAT is activated

Medium

Opens only Internet ports corresponding to services configured through this wizard. NAT is activated.

Strong

All ports closed but: outgoing HTTP and mail traffic, plus ssh traffic in both directions. NAT is not activated.

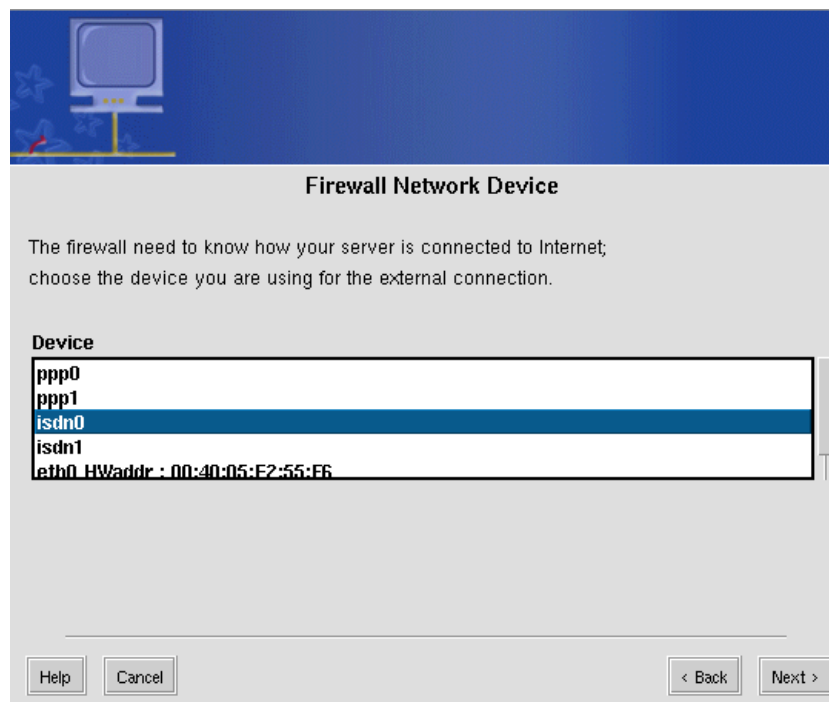


Figure 1-19. Which network interface is the gateway to the Internet?

Then, it is very important to tell the wizard which network interface card that is used for the gateway to the Internet (figure 1-19). If you specify the wrong device, your firewall will be useless!

1.9. Web Server Configuration

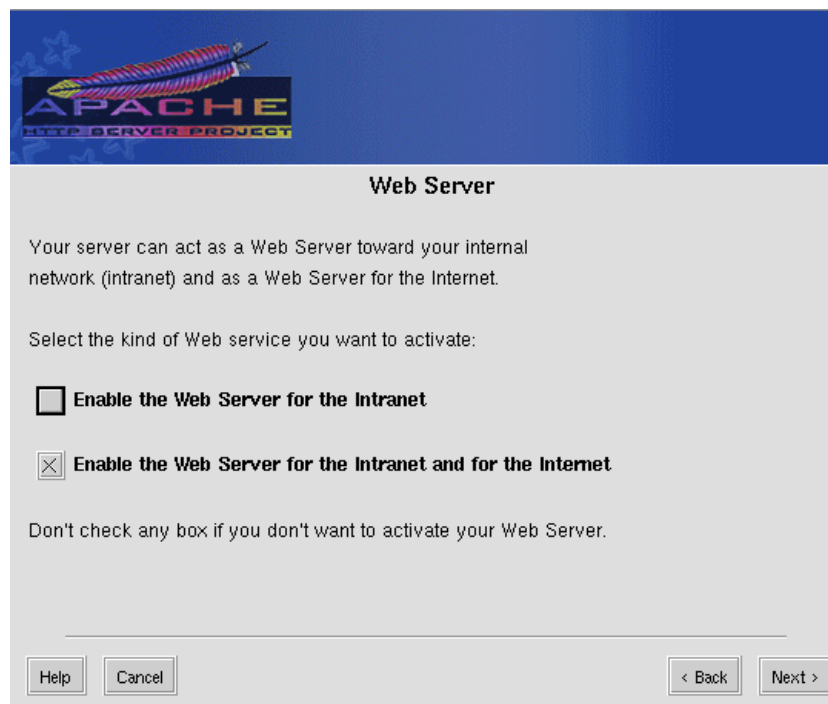


Figure 1-20. Where should your web server be visible from?

This wizard will simply let you specify if your web server will be disabled, visible from the local network only, or visible by anyone from both the local network and the external network (generally the Internet). Check the appropriate box as shown in figure 1-20.



To begin populating your Web site, simply put the files in the `/var/www/html/` directory. You can connect to your web site as soon as the wizard is finished through the URL: `http://localhost`

1.10. FTP server configuration

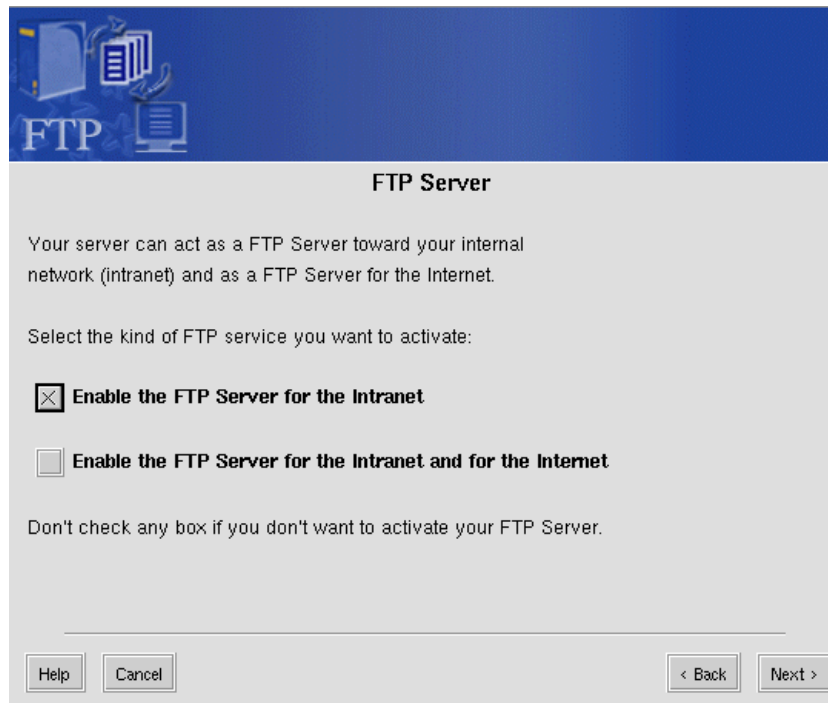


Figure 1-21. Where should your FTP server be visible from?

This wizard resembles the one used to configure a web server: it will let you specify whether FTP should be disabled, visible from the local network only, or visible from both the local network and the external network. A sample window is shown in figure 1-21.



To begin populating your anonymous FTP server, simply put the files in the `/var/ftp/pub/` directory. You can connect to your FTP server as soon as the wizard is finished through the URL:`ftp://localhost/pub`. Home directories are also accessible by default with local passwords authentication. If queen wants to access his home repository she has to use the URL:`ftp://queen@localhost`.

1.11. News Server configuration

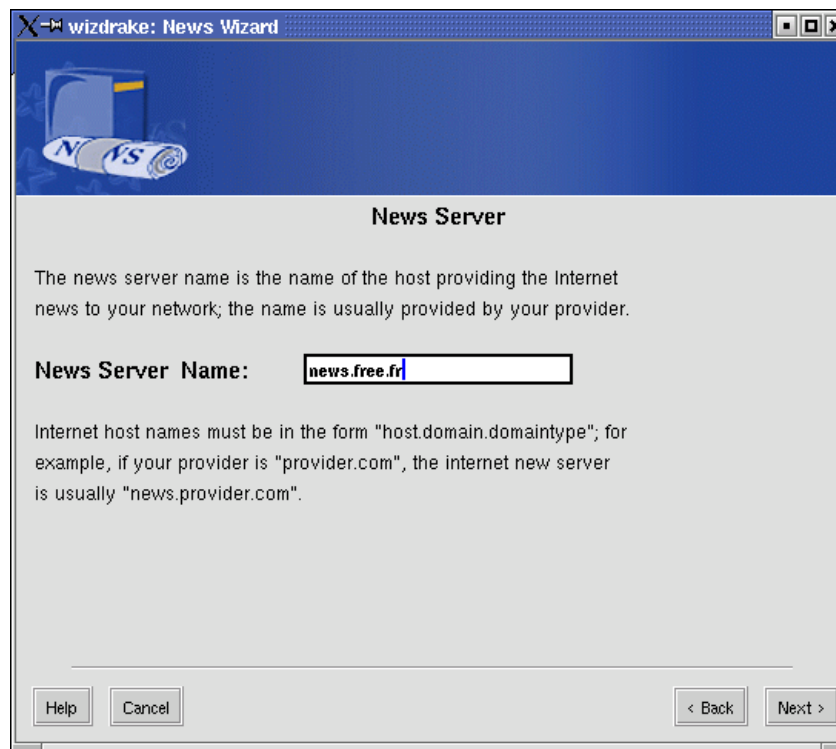


Figure 1-22. Which server do you want to fetch newsgroups from?

This wizard will configure a news gateway: your server will be able to fetch newsgroups from an external news server (usually, the one of your service provider) and make them visible to your internal network. Therefore, the first step is to specify which external news server you want to use, as shown in figure 1-22.

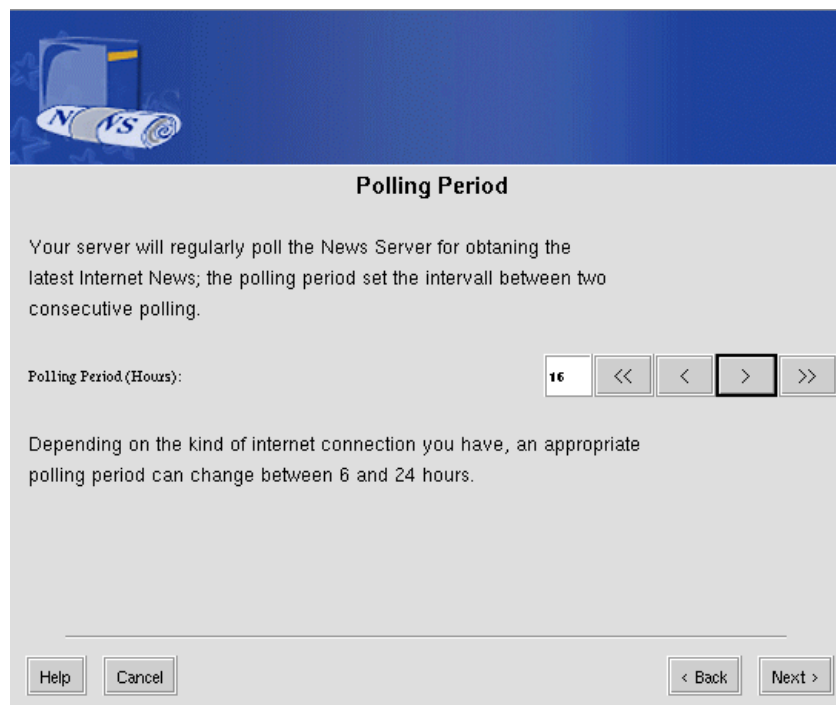
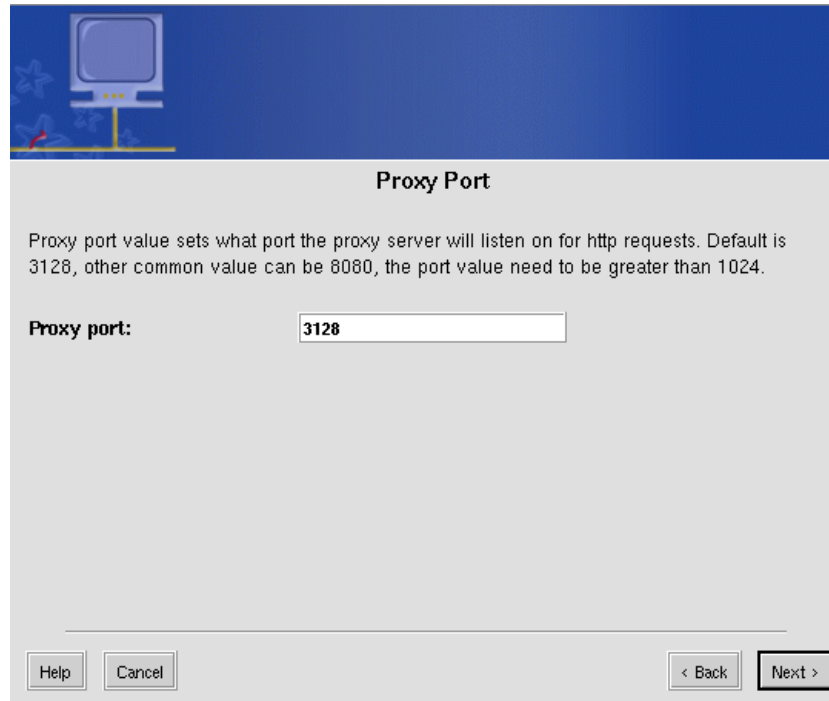


Figure 1-23. How often do you want to check for news?

Then you need to specify the interval (in hours) between every refresh (figure 1-23). Do not specify a too high interval: news evolves rapidly, as pretty much everything else on the Internet for that matter...

1.12. Proxy Server Configuration

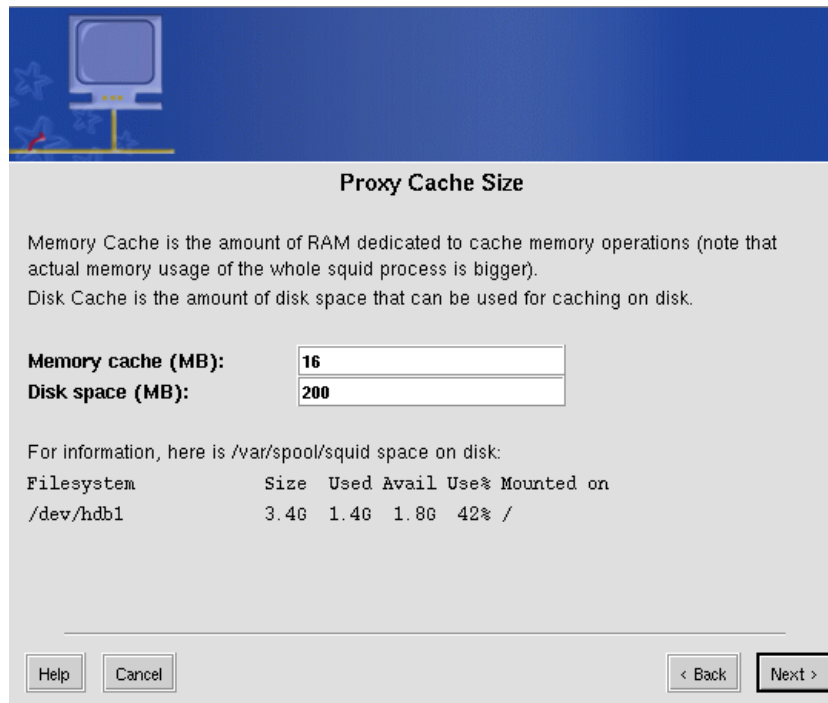
The *squid* proxy server is very useful for a local network accessing a lot of Web pages through a slow, or relatively slow connection. It maintains a cache of most visited pages so that they don't need to be retrieved twice from the Internet if requested by two different users.



The screenshot shows a configuration window titled "Proxy Port". The window has a blue header with a computer icon. Below the header, the title "Proxy Port" is centered. A paragraph of text explains: "Proxy port value sets what port the proxy server will listen on for http requests. Default is 3128, other common value can be 8080, the port value need to be greater than 1024." Below this text, there is a label "Proxy port:" followed by a text input field containing the value "3128". At the bottom of the window, there are four buttons: "Help", "Cancel", "< Back", and "Next >". The "Next >" button is highlighted with a black border.

Figure 1-24. Choose the Proxy Port

First of all you need to choose a port for the proxy to listen requests on. Users will have to configure their Web browsers to use this port as proxy port and your server name as proxy server.



Proxy Cache Size

Memory Cache is the amount of RAM dedicated to cache memory operations (note that actual memory usage of the whole squid process is bigger).
Disk Cache is the amount of disk space that can be used for caching on disk.

Memory cache (MB):

Disk space (MB):

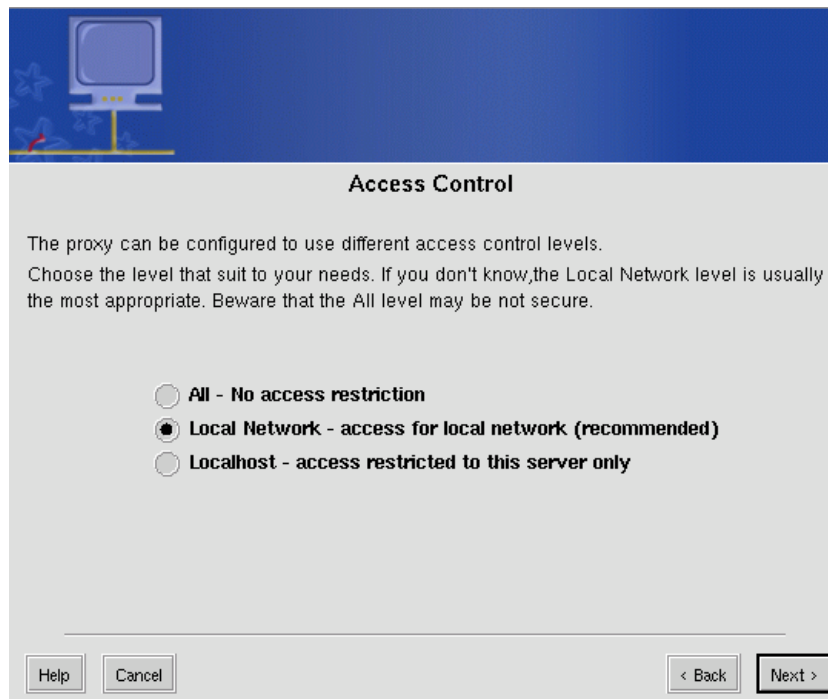
For information, here is /var/spool/squid space on disk:

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/hdb1	3.4G	1.4G	1.8G	42%	/

Buttons: Help, Cancel, < Back, Next >

Figure 1-25. Choose the Cache Sizes

Depending on your memory you can allocate more or less to the Proxy. The more memory cache, the less disks access on the server. Depending on your available disk size you can allocate more or less room for cached pages. The more place, the less accesses to the Internet.



Access Control

The proxy can be configured to use different access control levels.
Choose the level that suit to your needs. If you don't know, the Local Network level is usually the most appropriate. Beware that the All level may be not secure.

☐ **All - No access restriction**
☒ **Local Network - access for local network (recommended)**
☐ **Localhost - access restricted to this server only**

Buttons: Help, Cancel, < Back, Next >

Figure 1-26. Select Access Control Policy

Three access levels are possible for clients wishing to use the proxy:

- **All.** There is no restriction, all computers are granted access to the cache;
- **Local Network.** Only machines on the local network can have access to the proxy;
- **localhost.** Only the local machine, the server, can access its own proxy.

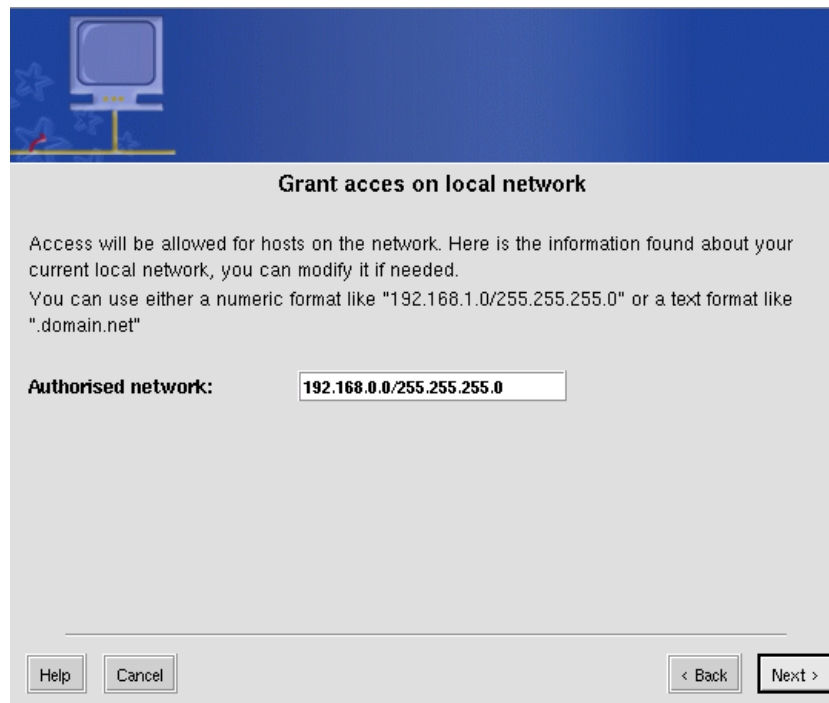


Figure 1-27. Restrict access to a particular subnetwork

If you have previously chosen the **Local Network** access policy, you can here choose to restrict even more the access to a particular subnetwork or domain. Enter your choice following the notation proposed.

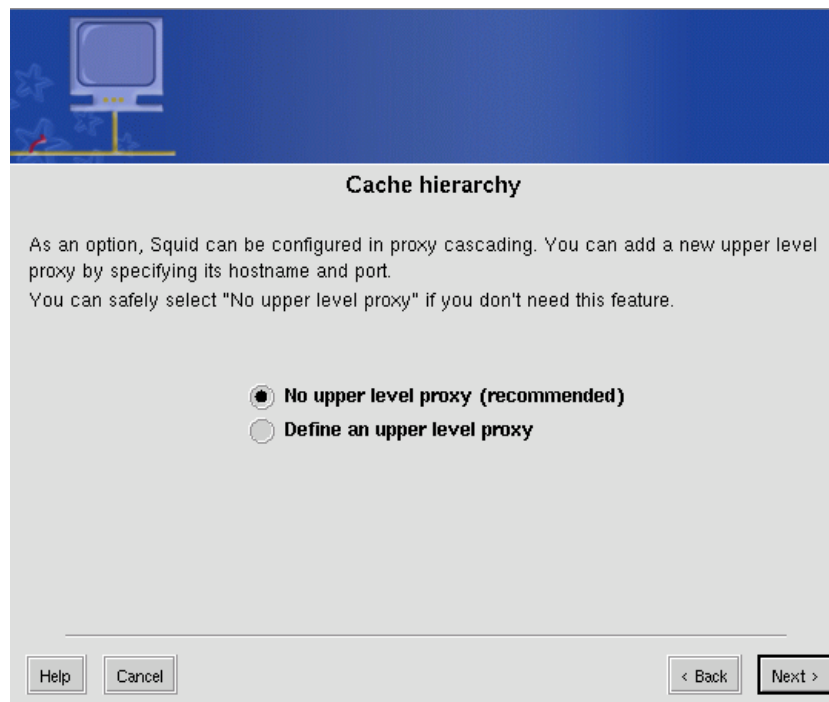


Figure 1-28. Use an upper level proxy?

If your server itself has access to another bigger proxy connected to the Internet, you can choose here to **Define an upper level proxy** to which requests will be forwarded. If so the next step will ask you for the name of that server.

1.13. Time Configuration

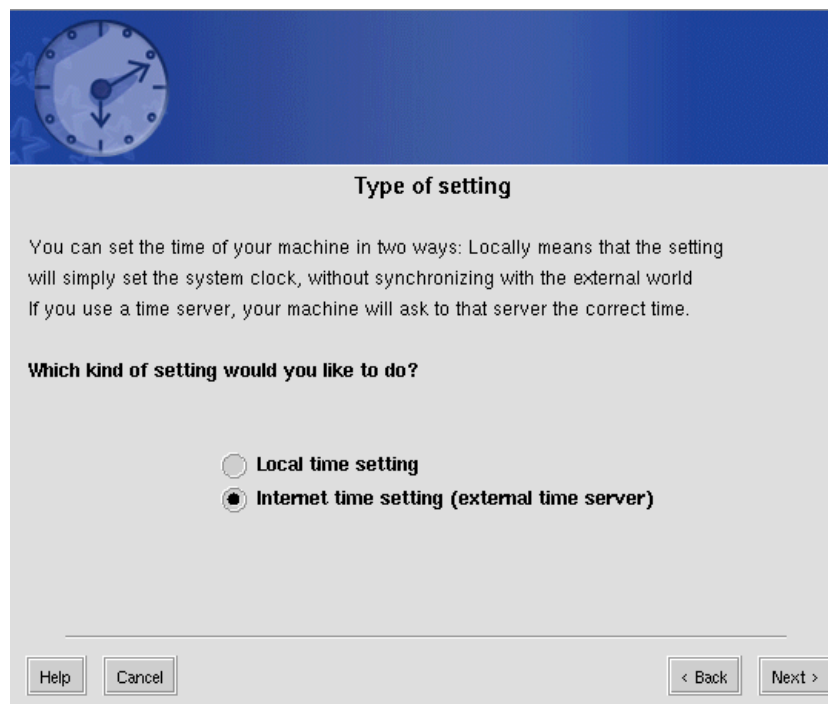


Figure 1-29. What method do you want for time synchronization?

This last wizard lets you set up a time server for your internal network. The protocol used is NTP. You will first have to choose whether you want to rely only on yourself or on an external server in order to get the right time (figure 1-29).

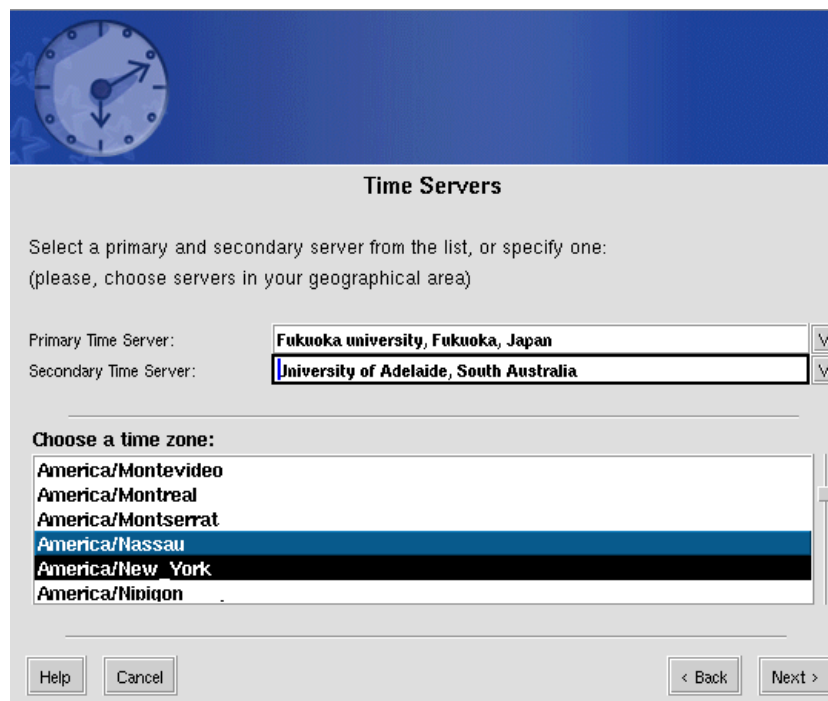


Figure 1-30. Choose your time servers

If you choose to rely on an external server, the wizard will ask you two time servers to query, in the order of preference. As the help text says, choose a server which is closest to you geographically! You will also have to set your time zone, but normally you will not have to change it: by default, the time zone used during installation is selected (figure 1-30).



You can also enter an NTP name address by hand if you wish to use one not listed here.

Chapter 2. Configuring Masqueraded Clients

This chapter will show you how to make different operating systems use a *GNU/Linux* box with masquerading set up as a gateway to the outside world. The configuration tests on the following operating systems all proved successful:

- Apple Macintosh, with MacTCP or Open Transport;
- Commodore Amiga, with AmiTCP or AS225-stack;
- Digital VAX Stations 3520 and 3100, with UCX (TCP/IP stack for VMS);
- Digital Alpha/AXP, with Linux/Redhat;
- IBM AIX (on RS/6000), OS/2 (including Warp 3) and OS400 (on OS/400);
- Linux (of course!): any kernel release since 1.2.x;
- Microsoft DOS (with the NCSA Telnet package, partial DOS Trumpet support), Windows 3.1 (with the Netmanage Chameleon package) and Windows For Workgroup 3.11 (with TCP/IP package);
- Microsoft Windows 95, Windows 95 OSR2, Windows 98, Windows 98se;
- Microsoft Windows NT 3.51, 4.0 and 2000 (both workstation and server);
- Novell Netware 4.01 Server, with the TCP/IP service;
- SCO OpenServer (v3.2.4.2 and 5);
- Sun Solaris 2.51, 2.6 and 7.

Let's go through the configuration of a few of them. If your system is not listed, a simple way to proceed is to "just tell the OS which machine to use as a gateway". Note that our main focus here is the **gateway** side of the network: therefore, we won't touch on DNS, file sharing or connection schemes problems. Thus, for this chapter to be of any use to you, you need a well-configured local network. Refer to your system's documentation to set it up properly, paying special attention to the DNS settings.

What follows assumes that you are set up on a class C network: your different machines all have IP addresses like 192.168.0.x, with a netmask set to 255.255.255.0, and use eth0 as the network interface. We also take for granted that your gateway's IP address is set to 192.168.0.1, and that your machines can each "talk" to the gateway (test the latter with the ping command or its equivalent in your environment).

2.1. Linux Box

There are (at least) three ways to go about this.

2.1.1. On-The-Fly Configuration

This is probably the fastest way to proceed. However, when you next restart your network layer or your whole system, any configuration change you made will have disappeared!

If eth0 is the network interface through which you access the gateway, (as root) issue this simple command: `route add default gw 192.168.0.1 eth0` That's it! If the gateway is properly configured and connected to the Internet, the whole world is now within your reach through your favorite web browser.

2.1.2. Permanent, Manual Configuration

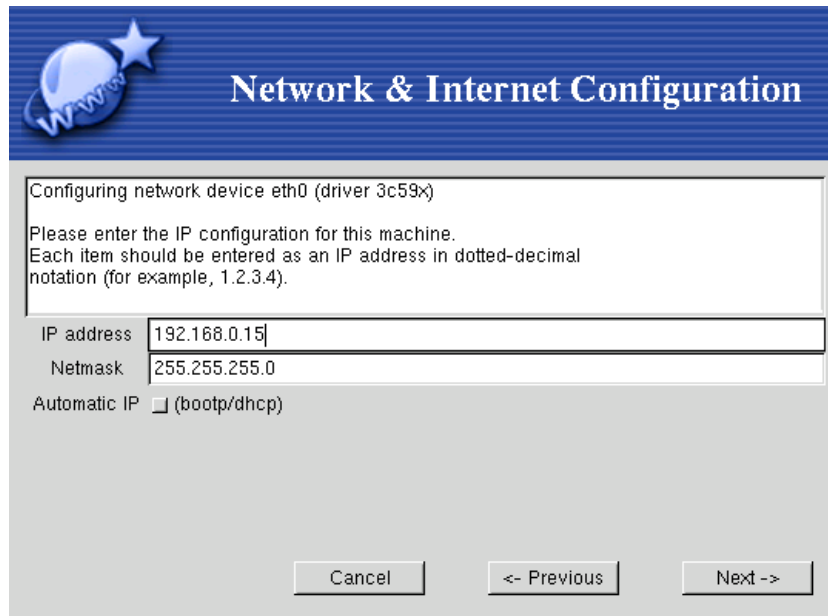
To maintain the configuration each time the system is shut down and restarted, we need to edit a configuration file. Its name is `/etc/sysconfig/network` on a **Mandrake Linux** machine (it may be different on yours). Open it with your usual text editor, then add the following lines:

```
GATEWAYDEV="eth0" GATEWAY="192.168.0.1"
```

You may now restart your network layer with: `service network restart`

2.1.3. Permanent, Automatic Configuration

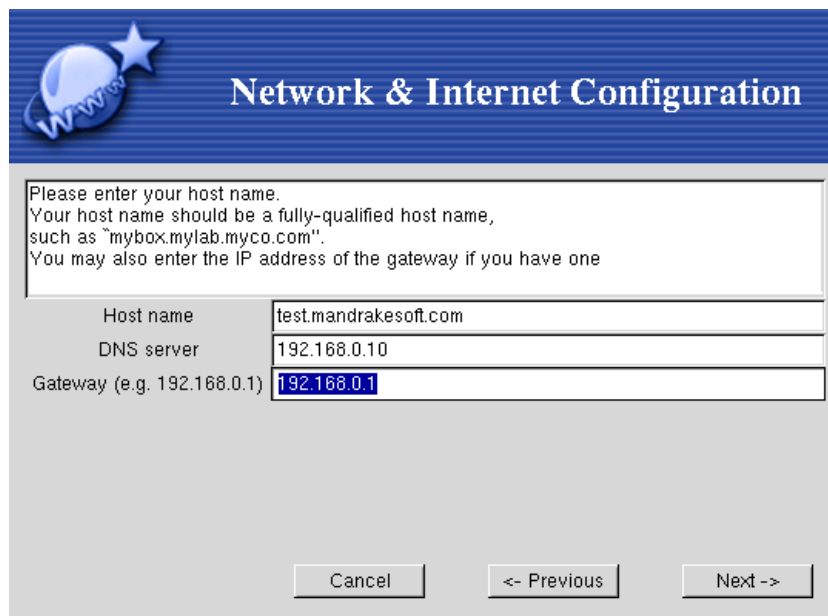
To install the configuration automatically, it's just about putting the right parameters in the configuration wizard. Refer to the *User Guide's Internet Configuration*. When you are configuring a local network Internet connection, the first step offers to configure the network in manual or automated mode (DHCP):



The screenshot shows a window titled "Network & Internet Configuration" with a blue header and a star icon. The main area contains the text: "Configuring network device eth0 (driver 3c59x)", "Please enter the IP configuration for this machine.", and "Each item should be entered as an IP address in dotted-decimal notation (for example, 1.2.3.4).". Below this are three input fields: "IP address" with the value "192.168.0.15", "Netmask" with the value "255.255.255.0", and "Automatic IP" with a checked checkbox and the text "(bootp/dhcp)". At the bottom are three buttons: "Cancel", "<- Previous", and "Next ->".

Figure 2-1. Reconfiguring The Local Network With draknet

Simply put the right information in it. If you have a bootp or DHCP server on your local network, simply check the **Automatic IP** box, and your configuration is done. If you have a static IP address for your machine, enter it in the first field after making sure the **Automatic IP** check box is deactivated. Then click on the **Next ->** button.



The screenshot shows a window titled "Network & Internet Configuration" with a blue header and a star icon. The main area contains the text: "Please enter your host name.", "Your host name should be a fully-qualified host name, such as 'mybox.mylab.myco.com'.", and "You may also enter the IP address of the gateway if you have one". Below this are three input fields: "Host name" with the value "test.mandrakesoft.com", "DNS server" with the value "192.168.0.10", and "Gateway (e.g. 192.168.0.1)" with the value "192.168.0.1". At the bottom are three buttons: "Cancel", "<- Previous", and "Next ->".

Figure 2-2. Setting up The Gateway With draknet

Here, you must write in the correct IP addresses for the gateway and DNS server. Once this is done, follow the wizard's steps and restart the network when proposed. And that's it. Your network is properly configured and ready to run. The configuration is now permanent.

2.2. Windows XP Box

We will assume here that you already have a configured network connection. The following snapshot shows the three different steps to get to the desired dialog.

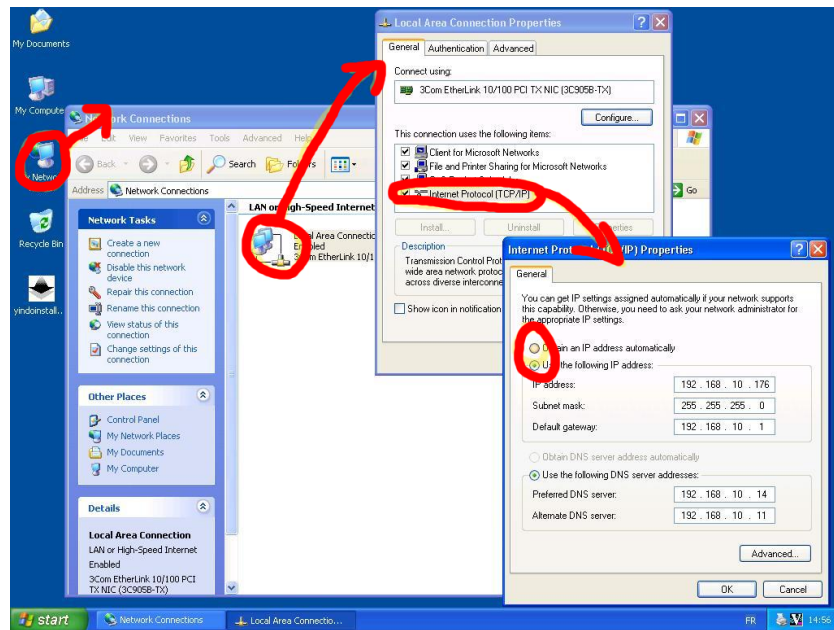


Figure 2-3. Setting up The Gateway With Windows XP

Here are the actions to take to go from one window to another:

1. On the desktop, right-click on the **My network places** icon, and select **Properties** in the menu that appears.
2. In the **Network Connections** window, do the same with the connection linked to the network where the gateway is located.
3. In the next dialog, select the **Internet Protocol (TCP/IP)** entry and click the **Properties** button.
4. In this dialog, you can choose to check **Obtain an IP address automatically** if you have a DHCP server on your network. Then, the gateway should also be automatically configured. If not, check **Use the following IP address** and fill in the associated fields.

2.3. Windows 95 or Windows 98 Box



Figure 2-4. The Network Icon Under Windows 95

Start by going in the Control Panel (**Start+Settings→Control Panel**) and find the network icon as shown. Double-click on it: the network configuration panel comes up.

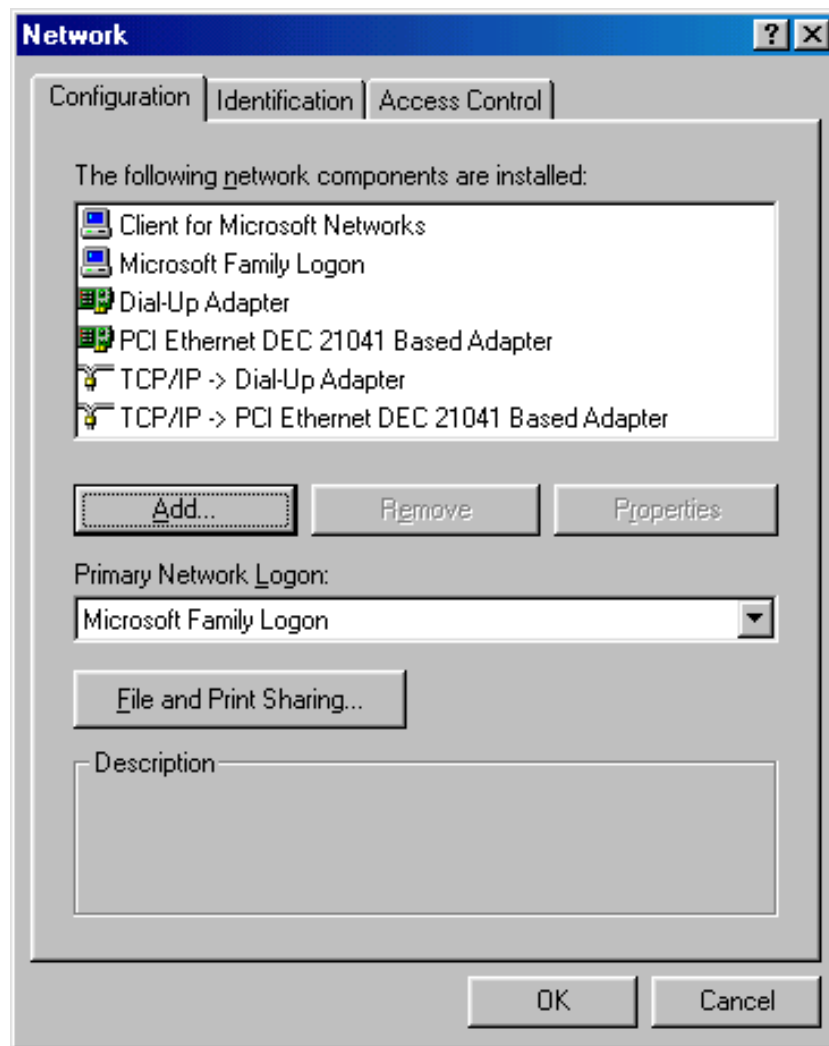


Figure 2-5. The Network Configuration Panel Under Windows 95

In the displayed list, you should find a protocol named TCP/IP . If not, you will have to refer to your system documentation to find out how to install it. If it is already there, select it and click on “Properties”.

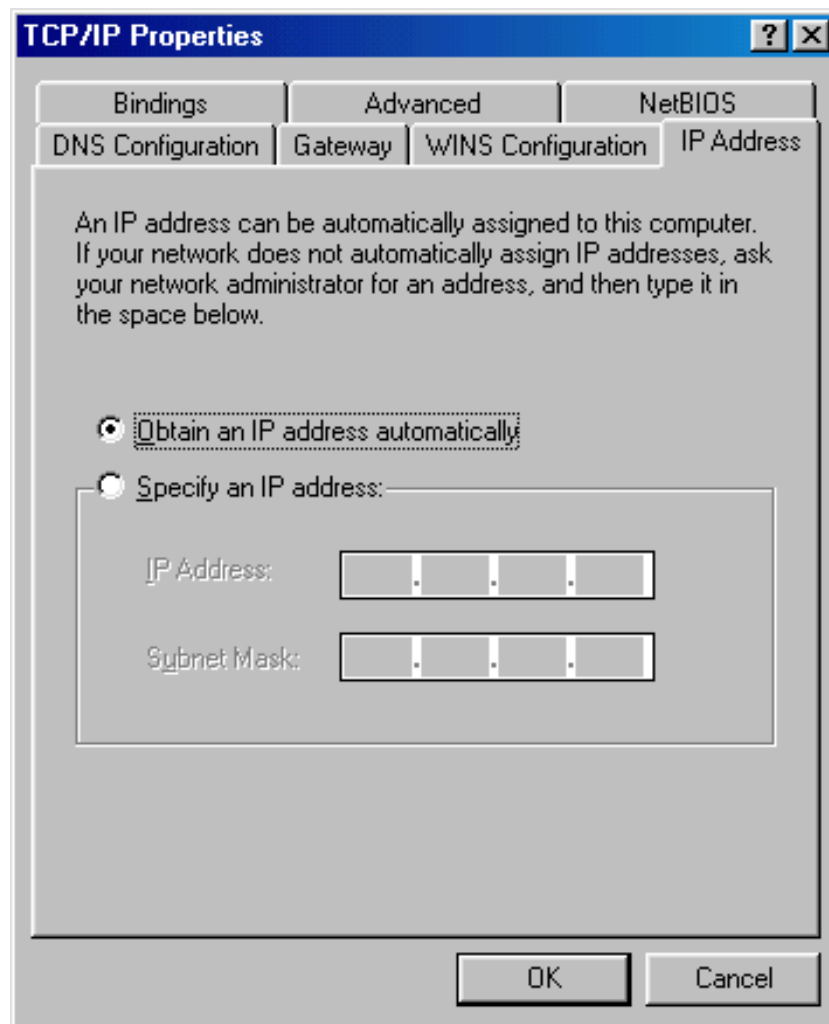


Figure 2-6. The TCP/IP Configuration Panel Under Windows 95

This window will enable you to set up your TCP/IP parameters. Your system administrator will tell you if you have a static IP address or if you are using DHCP (automatic IP address). Click on the Gateway tab.

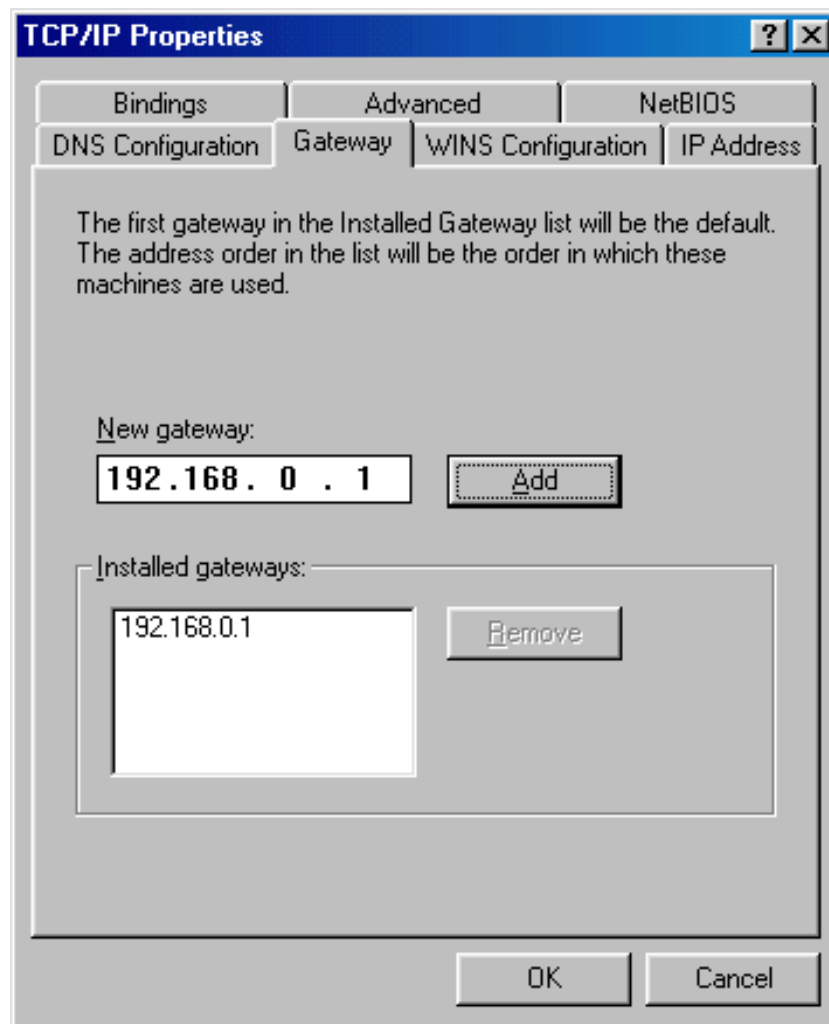


Figure 2-7. The Gateway Configuration Panel Under Windows 95

The rest is child's play! Fill in the blanks with your gateway's IP address (i.e. 192.168.0.1, in our example). Click the Add then the OK buttons.

You will need to reboot your computer, of course. Once this is done, find out if you can reach the rest of the world.

2.4. Windows NT or Windows 2000 Box

To configure these OSs, follow these simple steps:

1. Go to Control Panel+Network→Protocol.

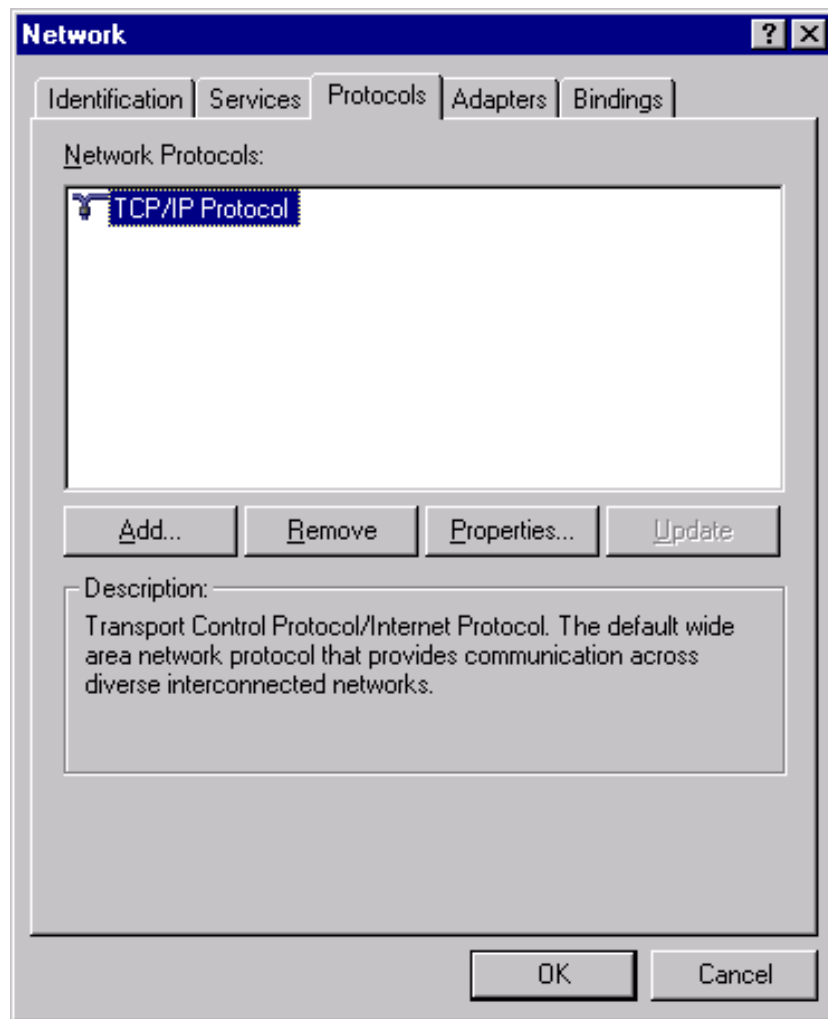


Figure 2-8. The Protocol Configuration Panel Under Windows NT

2. First, select the **TCP/IP Protocol** in the list of network protocols. Then, click on the **Properties** button, and select the network card connected to the local network (figure 2-9). In this example, we show a configuration with the DHCP server activated on the *mandrakesecurity* server: the **Obtain an IP address from a DHCP server** option is checked.

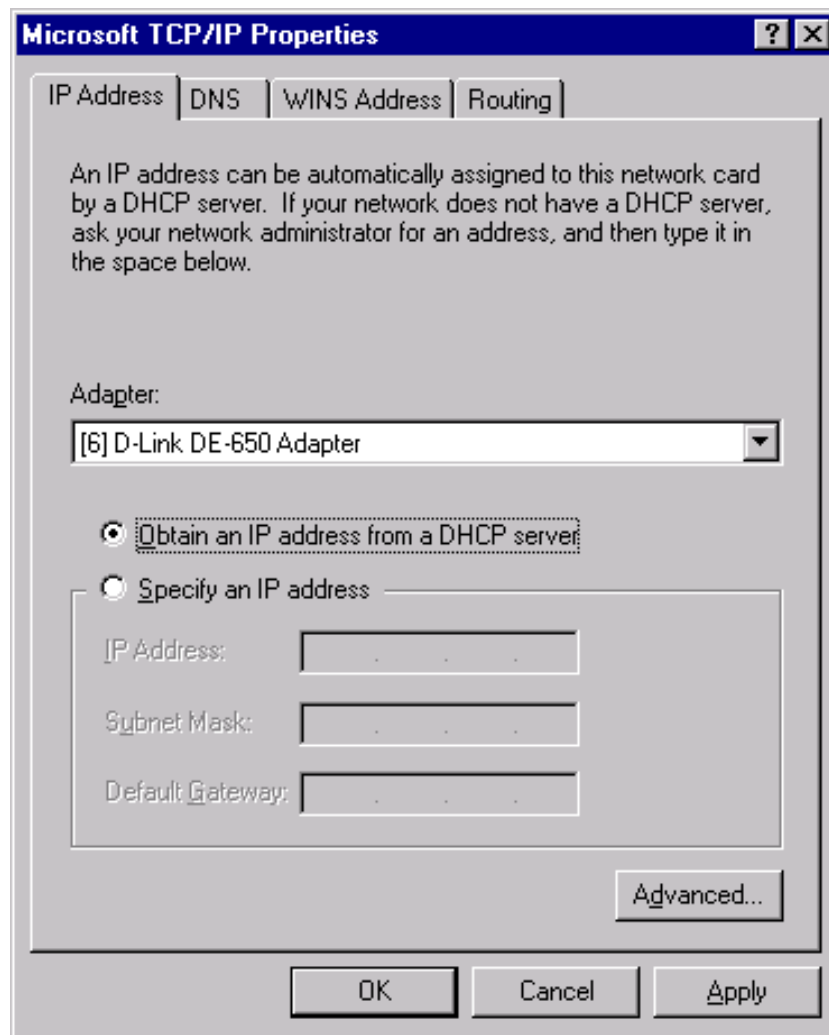


Figure 2-9. The Network Software Panel Under Windows NT

If this is your case, you just need to confirm all those choices and reboot. Otherwise, follow the following steps.

3. If you have no DHCP server, you need to manually set all parameters. Begin by checking the **Specify an IP address** option (figure 2-10).

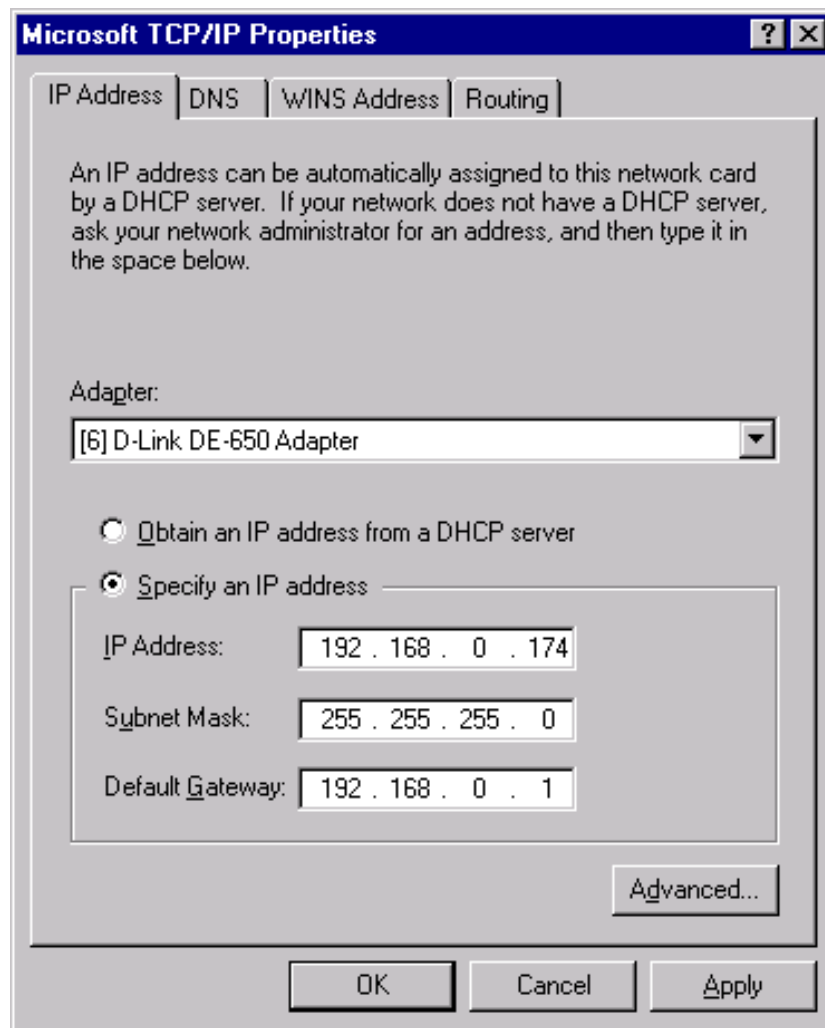


Figure 2-10. The TCP/IP Configuration Panel Under Windows NT

Select the appropriate adapter, the IP address should already be correct.

4. Simply fill in the **Default Gateway** field with 192.168.0.1 (the address of the *Linux* box sharing the connection in our example).
5. Finally, you will need to specify the DNS servers you use in the **DNS** tab as shown in figure 2-11.

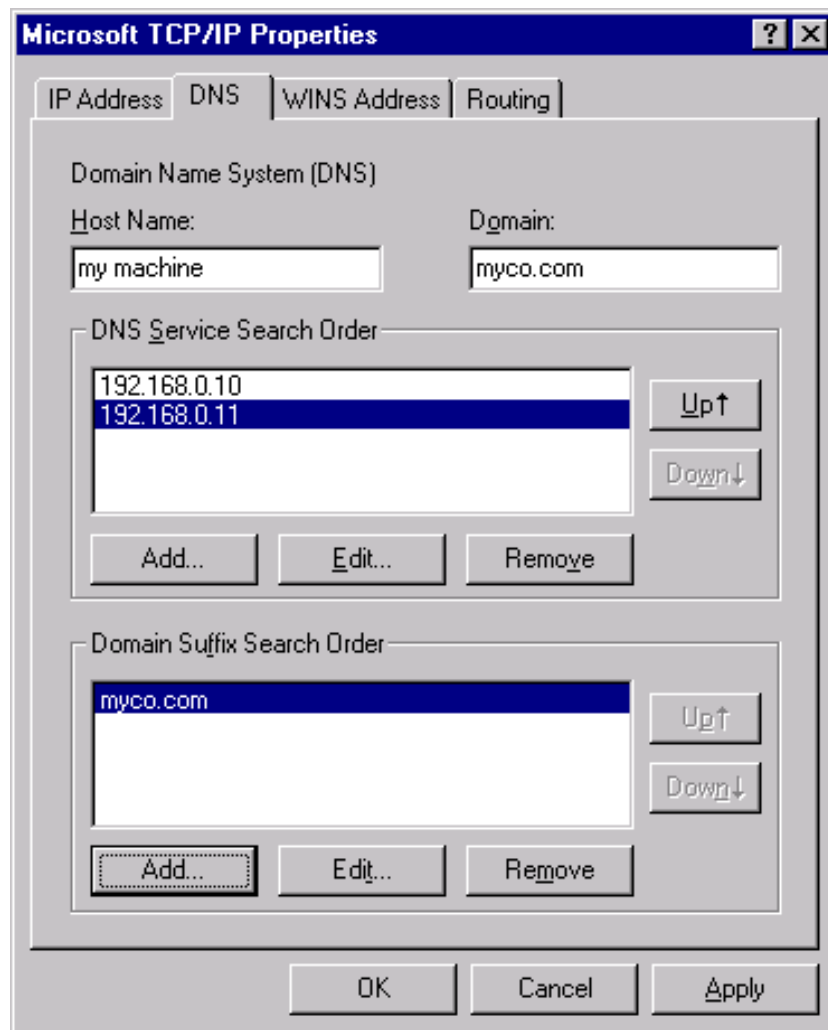


Figure 2-11. The DNS Configuration Panel Under Windows NT

You must also provide a host name and an associated domain name.



Unless you know exactly what you are doing, proceed with utmost care with the following steps:

- leave the **Automatic DHCP configuration** field blank unless you have a DHCP server somewhere on your network;
- leave all the **WINS Server** fields blank as well unless you have one or more WINS servers;
- do not place a check the **Enable IP Forwarding** field unless your NT machine is used for routing and, once again, you know perfectly what you are doing;
- please disable **DNS for Windows Name Resolution** and **Enable LMHOSTS** lookup.

Click on **OK** in the dialog boxes which then appear and restart your computer to put test the configuration.

2.5. DOS Box Using The NCSA Telnet Package

In the directory which hosts the NCSA package, you will find a file called `config.tel`. Edit it with your favorite editor and add the following lines:

```
name=default
host=yourlinuxhostname
hostip=192.168.0.1
```



```
gateway=1
```

Of course, write the name of your *Linux* box instead of `yourlinuxhostname` and change the gateway address given here (192.168.0.1), which is only an example.

Now save the file, try to telnet your *Linux* box, then a machine somewhere out there...

2.6. Windows For Workgroup 3.11

The TCP/IP 32b package should already be installed. Go to the **Main+Windows Setup+Network Setup→Drivers** menu entry and select **Microsoft TCP/IP-32 3.11b** in the **Network Drivers** section, then click **Setup**.

From here, the procedure is quite similar to the one described in the *Windows NT* section.

2.7. MacOS Box

2.7.1. MacOS 8/9

First of all, you need to open the **TCP/IP Control Panel** as shown below in the Apple menu.

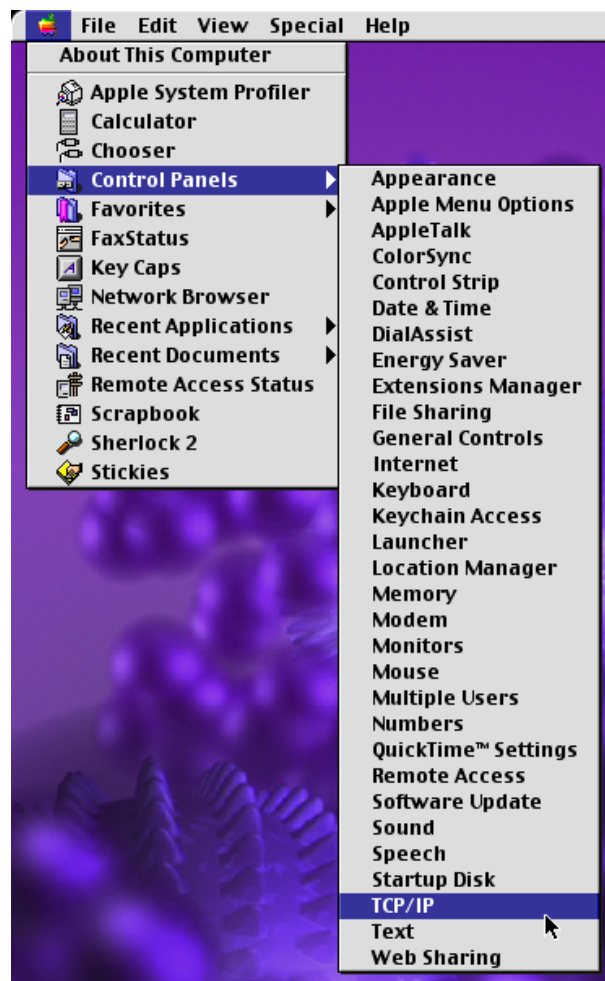


Figure 2-12. Accessing The TCP/IP Control Panel

2.7.1.1. With an Automatic DHCP Configuration

If you configured your firewall to be a DHCP server, follow this very procedure, otherwise go to the next section.

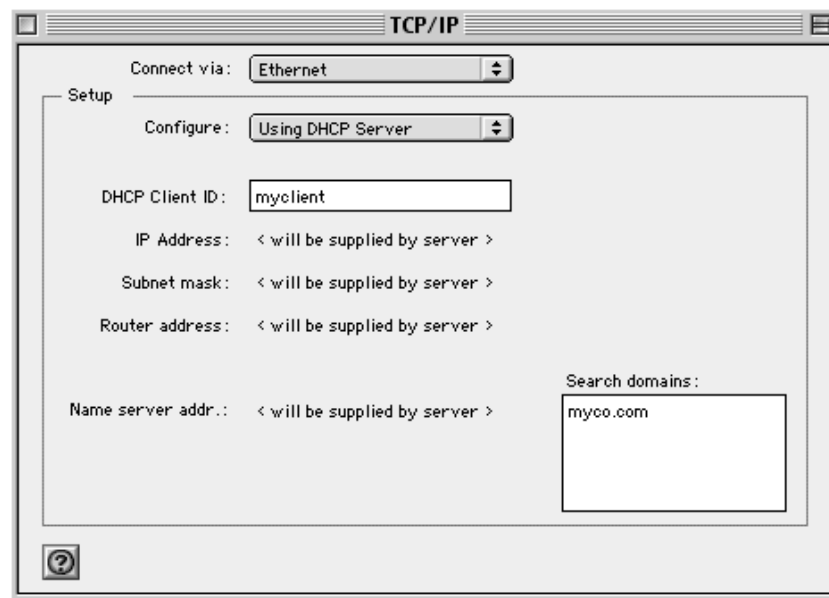


Figure 2-13. Automatic Configuration of Internet Access For MacOS

In the dialog that appears, fill the fields as shown hereafter:

- Connect via: **Ethernet**;
- Configure: **Using DHCP server**;
- DHCP Client ID: **192.168.0.1**.

2.7.1.2. For a Manual Configuration

If you have no DHCP server on your local network, follow this procedure:

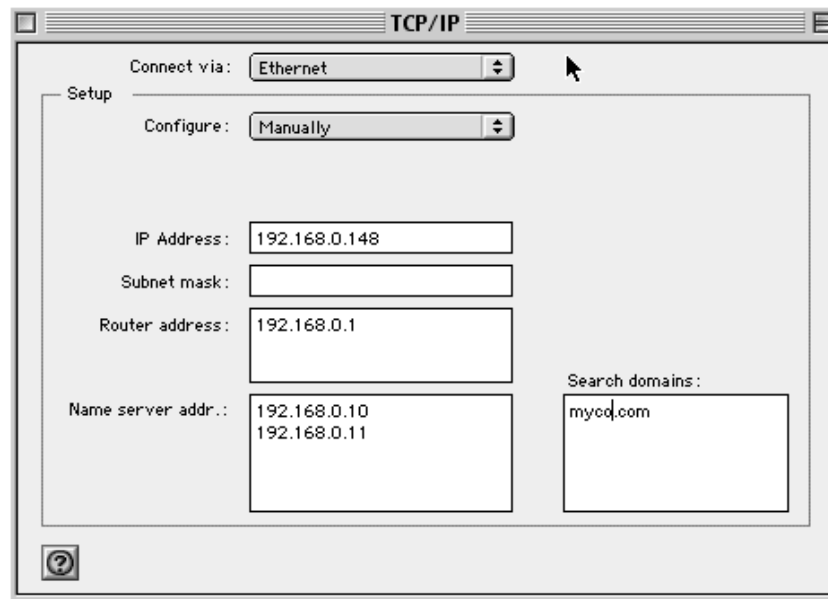


Figure 2-14. Manual Configuration of Internet Access For MacOS

In the dialog that appears fill the fields as shown here:

- Connect via: **Ethernet**;
- Configure: **Manually**;
- IP address: **192.168.0.248**;
- Subnet Mask: **255.255.255.0**;
- Router Address: **192.168.0.1**;
- Name Servers Addresses: **192.168.0.10**; **192.168.0.11**
- Search Domain: **myco.com**;



The name servers addresses may be the address of the internal DNSs or those of your Internet Service Provider's servers.

2.7.2. MacTCP

1. In the **MacTCP** control panel, select the Ethernet network driver (caution, it's not EtherTalk) then click the **More...** button.
2. Under **Gateway Address**, enter the address of the *Linux* box sharing the connection (192.168.0.1 in our example).
3. Click **OK** to save the settings. You may have to restart your system to test these settings.

2.8. OS/2 Warp Box

The TCP/IP protocol should already be installed. If not, install it.

1. Go in **Programs**, then **TCP/IP (LAN)**, then **TCP/IP Settings**.
2. Under **Routing**, choose **Add**. In **Type**, select **default**.
3. Fill the **Router address** field with the address of your *Linux* box sharing the Internet connection (192.168.0.1 in our example).

4. Now close the TCP/IP control panel, answer **Yes** to all questions, then reboot your system before testing the settings.

II. In-Depth Configuration of Common Services

Introduction to The Configuration of Services

This part will detail the most common services a system administrator may need for both Internet and intranet uses. We will try to document the most common packages for middle-size companies. All services will be configured using the *Webmin* tool, which will be briefly introduced next.

1. Introduction to Webmin

The *Webmin* tool allows to perform remote administration of your machine using nothing more than a web browser which supports the HTTPS (HTTP over SSL) protocol. Hence, all traffic to/from it is secure.

This makes *Webmin* ideal for system administrators because all major platforms have web browsers which meet or exceed the above requirements. Moreover, *Webmin* has its own “web server” so it does not need 3rd party software (like a web server) to work. Everything is included.

Webmin has a modular architecture, allowing you to write your own configuration modules if you need to. It comes with modules for all the services described here, and more. Almost all aspects of your machine can be configured with it.

2. Services

The services covered in this part are:

- Internet/Intranet web site hosting (HTTP). We will speak about the *Apache* web server. See the “*Internet/Intranet Web Server*”, page 41 chapter.
- Mail Management (SMTP). This chapter focuses on sending mail with the *Postfix* mail server. Refer to the “*Postfix Mail Server*”, page 47 chapter.
- Mail Retrieving (POP and IMAP). We will speak about getting mail with the *IMAP-2000* mail server. See the “*Incoming Mail Server: POP And IMAP*”, page 51 chapter.
- Sharing of files and printers (NFS, SMB and FTP). Sharing resources is the main topic in this chapter, using NFS tools, *Samba* and *WU-FTPD*. Refer to the “*Resource Sharing*”, page 55 chapter.
- Database. We will detail the usage of the *MySQL* database server. See the “*MySQL Database Server*”, page 63 chapter.
- Home hosting (NIS). Distributed user management is the main subject in the “*NIS Client And Server*”, page 67 chapter.
- Domain Name System (DNS). We will speak about the *BIND* name server in the “*BIND DNS Server*”, page 71 chapter.

Please note that **all** the tools used for the above services are open-source software and are already included in your **Mandrake Linux** distribution.

So, on with the services!

Chapter 3. Internet/Intranet Web Server

Apache allows your company to create a web site and serve web pages to client browsers such as *Mozilla*. *Apache* is powerful and freely available. You can design static or dynamic sites using, for example, *PHP*. It's one of the most popular server application on the Internet.

3.1. Installation

The first step is to check that the *Apache* web server is installed on your computer. If it is not, please use *RpmDrake* or type `urpmi apache` in a terminal, as root.

The server configuration is done through the **Apache web server**. You will find it in the **Servers** category (accessible by the **Servers** tab). If you don't find it, you could get the `apache.wbm` file on the Webmin site (<http://www.webmin.com/webmin/standard.html>) and install it by using the **Webmin configuration** module in the corresponding category. If you click on the **Webmin modules** icon, you will be directed to a page where you can tell *Webmin* the path to the module to be installed (or removed).


3.2. Step-by-Step Configuration Example


[Webmin Index](#)
[Module Config](#)


Apache Webserver


[Apply Changes](#)
[Stop Apache](#)
[Search docs..](#)


Global Configuration



[Processes and Limits](#)



[Networking and Addresses](#)



[Apache Modules](#)



[MIME Types](#)


[Miscellaneous](#)


[CGI Programs](#)


[Per-Directory Options Files](#)


[Re-Configure Known Modules](#)


[Edit Defined Parameters](#)

Virtual Servers

Type	Address	Port	Server Name	Document Root
Default Server	Any	Any	Automatic	/var/www/html
Virtual Server	Any	443	Automatic	/var/www/html

Create a New Virtual Server

Address

☐ Any

☒ Add name virtual server address (if needed)

Port

☐ Default ☐ Any

Document Root

Server Name

☐ Automatic

Copy directives from

Create

[Return to index](#)

Figure 3-1. Webmin's Main Apache Module Screen

Webmin's Apache module considers all sites hosted on your computer as virtual servers and the default one is your main site. The default configuration of the web server is localized in the **Default** server entry of the **Virtual server** section. Click on **Default** server and you will get a screen which has two main sections. On top, you have the options' icons and at the bottom, the per directory configuration options.

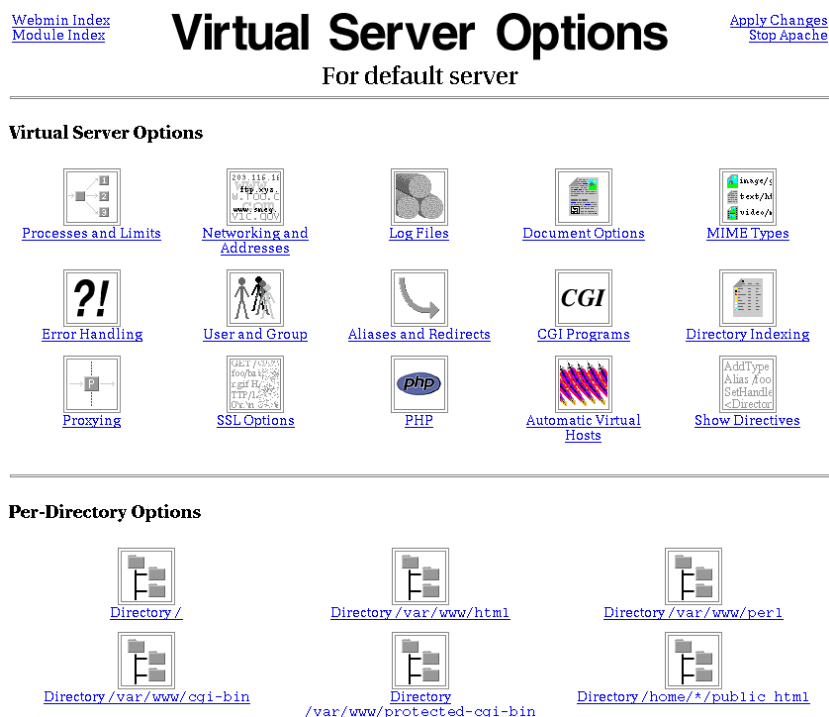


Figure 3-2. Apache' Default Server Configuration Screen

The **Networking and addresses** section contains some important options. In it, you can specify the webmaster's mail address in the **Server admin email address** field. To avoid false requests on your web server, configure the following options: **Lookup hostnames** and **Use hostname supplied by browser**, respectively to yes and no.

Apache writes errors and accessed pages into a journal. Options in the section **Log files** manage this journal. The first one, **Error log to**, allows you to choose between **Syslog** (the centralized system log), a specific log file or a program. The same is possible for **Browser log**. You can specify the information format in the log file. If you plan to use a log parser, you must specify the log format that *Apache* should generate. The last important option, **Don't log references from**, is useful to avoid some hosts to appear in the log file. For example, if you are accessing your web site from your internal network, you could prefer not to fill the log with those requests.

In the **Document options** section, specify your site's **Document root**. This is the path to the directory containing your web pages. **User WWW directory** specifies the name of the directory in each user account containing user web pages. Moreover, you can restrict this option for some user. **Directory options** contains common options for your web server, like the ability to execute a CGI program or to follow *UNIX* symbolic links in a web tree (**Execute CGI programs** and **Follow symbolic links**).

[Webmin Index](#)
[Module Index](#)

Document Options

For default server

[Apply Changes](#)
[Stop Apache](#)

Document Options for default server

Document root directory

Default ☒ /var/www/html

User WWW directory

Default ☒ public_html

All users accessible

All users except

Only users

Per-directory options file

Default ☒ htaccess

Directory options

Default ☒ Selected below..

Option	Set for directory	Merge with parent
Execute CGI programs	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input type="radio"/> Enable <input type="radio"/> Disable
Follow symbolic links	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input type="radio"/> Enable <input type="radio"/> Disable
Server-side includes and execs	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input type="radio"/> Enable <input type="radio"/> Disable
Server-side includes	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input type="radio"/> Enable <input type="radio"/> Disable
Generate directory indexes	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input type="radio"/> Enable <input type="radio"/> Disable
Generate Multiviews	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input type="radio"/> Enable <input type="radio"/> Disable
Follow symbolic links if owners match	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input type="radio"/> Enable <input type="radio"/> Disable

Generate MD5 digests

☐ Yes ☐ No ☒ Default

Virtual server path

Default

Error message footer

Email address ☐ Server name ☐ None ☒ Default

Default base for imagemaps

Default

Default action for imagemaps

Default

Action on incorrect use of imagemaps

Default

Save

[Return to server index](#)

Figure 3-3. Document Options Section

If your web site contains some directories, you can specify aliases in the **Alias and redirects** section. For example, if you have a complex directory tree containing your web pages, you can simplify the navigation by creating aliases. Instead of pointing your browser to `http://www.example.com/foo/bar/again/and/more`, just use `http://www.example.com/morestuff` in **Document directory aliases**. The second part of the screen is dedicated to redirections, which let you redirect a part of your web address to a specific web page or directory.

[Webmin Index](#)
[Module Index](#)

Aliases and Redirects

For default server

[Apply Changes](#)
[Stop Apache](#)

Aliases and Redirects for default server

Document directory aliases

From	To
/icons/	/var/www/icons/
/foo	/var/www/html/foo

Regexp document directory aliases

From	To
------	----

URL redirects

From	Status	To
------	--------	----

Regexp URL redirects

From	Status	To
------	--------	----

Permanent URL redirects

From	To
------	----

Temporary URL redirects

From	To
------	----

Map local to remote URLs

Local URL path	Remote URL
----------------	------------

Map remote Location: headers to local

Remote URL	Local URL path
------------	----------------

Save

[Return to server index](#)

Figure 3-4. Alias And Redirection Section

If you plan to use *Common Gateway Interface* programs, the **CGI programs** section lets you specify which directory contains your CGIs, and configure some variables passed to the executables. The default values allow you to directly use your script in your web site.

If you want to increase security on your site by using cryptography, the **SSL options** section allows you to enable SSL connections. Sessions opened using SSL will be encrypted, hence secure. You can choose the log file in **SSL log file** or the protocols with **SSL protocols**. This choice depends on the clients you will be serving. Moreover, you can indicate to *Apache* which certificate to use in order to authenticate your site to the client (Certificate/private key file).

Figure 3-5. SSL Options Section

The **PHP** icon lets you configure specific default *PHP* values for the site. For example, if you have one *PHP* script used by many sites, you can differentiate the latter using *PHP* flags.

Read more about the other icons of this screen in *Advanced Configuration*, page 44.

All configuration points we talked about can be accessed through the **Show directives** section. It presents the part of the *Apache* configuration file associated with the virtual host. Each link sends you to the appropriate *Webmin* section in order for you to configure it.

On the other hand, the second section manages your site's directories. As you can see, if you click on one directory name, you can specify the same general options for each directory. For example, you can configure specific **Mime Types** for your download directory, or specific directories when people request a precise, private directory.

At this point, you can apply any of your changes directly by clicking on **Apply Changes** link in the top right corner. In the same area, the **Stop apache** link allows you to stop your web server.

3.3. Advanced Configuration

Webmin's main screen, *Apache* Module, is composed of many sections. The first, **Processes and limits**, allows you to tune your *Apache* server. You can configure the number of initial instances of *Apache* (**Initial server processes** and **Maximum spare server processes**), the header and request line sizes (**Maximum headers in request** and **Maximum request line size**) or the number of clients per process (**Maximum requests per server process**).

Figure 3-6. The Configuration Screen of Apache Processes

The **Listen on addresses and port** option in the **Networking and addresses** section could be important for your web server configuration. You can specify the default port *Apache* listens on for regular and encrypted sessions (by default, 80 and 443, respectively). **Multiple requests per connection** sets up the number of requests that can be served for a browser, and **Keep-alive timeout** configures the timeout of web browser requests.

In the **Apache modules** section, you will find all detected *Apache* modules. Here, you can select modules you don't want to be loaded by *Apache*.

The **Miscellaneous** section contains some useful options. The first one, **Core dump directory**, configures the directory in which *Apache* will store its memory dump in the unlikely case of crashing. In the same way, **Server lock file** and **Server PID file** specifies, respectively, the lock file and the one containing *Apache*'s *UNIX* process number. You can configure the HTTP header returned by *Apache* in the **Server HTTP header** field and the **Server execution mode**.

Apache can run in stand-alone mode. If so, it will be launched and wait for connections. *Apache* can also be launched with the *inetd* command when a browser sends a request to the computer. Generally, *Apache* with *inetd* is used for small web sites.

The next section concerns **CGI programs**. If you use CGI programs with your web server, you can specify where to put the CGI output log file and its maximum size.

Per-directory options files allows to redefine common options to a particular directory of your web site. Moreover, this file (generally *.htaccess*) permits to limit access to directories. In this section, select the directory and click on **Create options file**. You will get a new page with many icons, similar to the common configuration pages. Here, you can specify some default values for *PHP* (click on the **PHP** icon), some access restrictions by user or group (click on the **Access control** icon) or configure the directory listing format (click on **Directory indexing** icon).

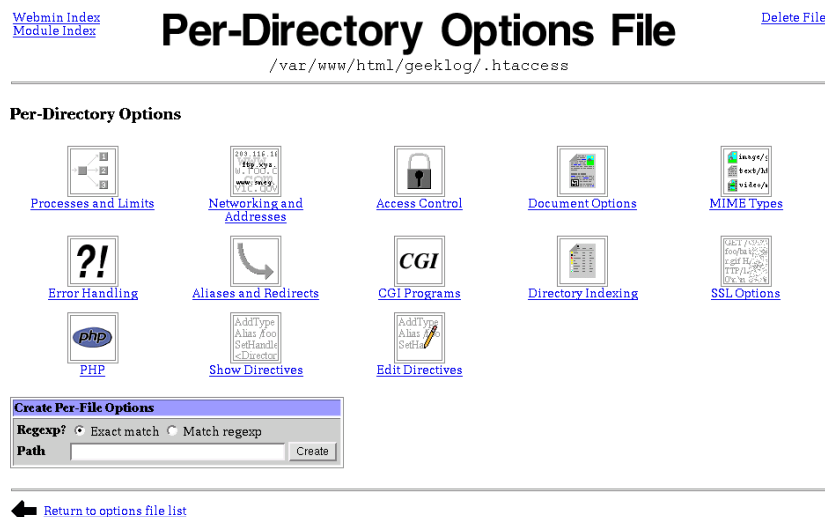


Figure 3-7. Directory Limitations Using *.htaccess*

The **Re-configure known modules** section contains a list of possible installed *Apache* modules. *Webmin* tries to find automatically which ones are installed, but you can configure them by hand in this section.

The **Edit defined parameters** section enables to modify the parameters passed to *Apache* when the server is launched.

Using the **Virtual servers** section, you can set up a multi-domain web server. For example, your company owns *foo.com* and *bar.net*. With virtual servers, you can handle the two domains with one computer and one *Apache* server. You just have to specify the IP address and eventually the server's port number, the document root, where your site's files are stored, and the name of the virtual server. If you are managing multiple sites, you can copy configuration directives from other virtual servers. This can save you lots of time.

For each site represented by a virtual server, you have other options. In the **Processes and limits** section, you will find options to limit system resources. You can set up the memory, CPU and process resources limit to prevent the system to fall down because of misbehaving clients. The **Error handling** section enables to specify which web

page is called if a specific error appears during the request treatment. For example, if *Apache* cannot find a requested web page, it displays a 404 error. The **User and group** icon allows you to specify which user owns the *Apache* processes corresponding to the current virtual server.

The **MIME types** take care of file associations and char sets in *Apache*. You can set up which command will be executed when a request is made for certain file types. In the same way, you can tell *Apache* how to find multi-lingual pages. If you have many sites, you can tell it to change the owner of the processes to the values you set. The icon **Directory indexing** allows you to do so. If you want to create a cache server for some web pages, the **Proxying** section contain some options which will help you out.

Chapter 4. Postfix Mail Server

With *Postfix*, you can set up and configure a mail server in order to send and receive mails. This server can communicate directly with other mail servers on the Internet through the SMTP protocol. With the right configuration, *Postfix* can handle all the mail sent to the domain of your company.

4.1. Installation

The first step is to check that the *Postfix* server is installed on your computer. If it is not, please use the `rpmdrake` application or type `urpmi apache` in a terminal to install it.

The server configuration is done through the **Postfix configuration** button. You will find it in the **Servers** category. If you don't find it, you can recuperate the `postfix.wbm` file on the Webmin site (<http://www.webmin.com/webmin/standard.html>) and install it by using the **Webmin configuration** module in the **Webmin** category. If you click on the **Webmin modules** icon, you will end up on a page where you can tell *Webmin* the path to the module to be installed (or removed).

4.2. Step-by-Step Configuration Example

Each *Postfix* option in the *Webmin* module is documented. Just click on the option's name and a new window will appear, explaining the relevant option.

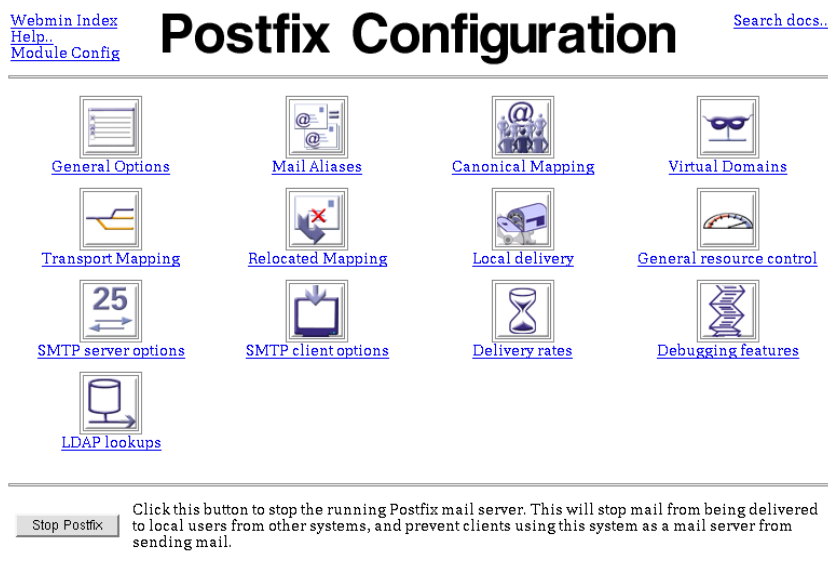


Figure 4-1. Postfix Module's Start-Up Screen

Postfix's configuration begins with the first icon: **General options**. The **Most useful general options** part is relevant and interesting. The other part will be described in the *Advanced Configuration*, page 48 section of this chapter. The first option concerns outgoing mail. You should specify the mail domain. Leave it to domain-name if the computer's domain name has the same value as your mail domain name. Through the second option, *Postfix* knows which domains and host names to manage. For example, your computer name is `gateway.example.com` and your mail domain is `example.com`. The outgoing mail domain should be `example.com` and the incoming domains would be `example.com`, `localhost.example.com`, `localhost.localdomain`, `gateway.example.com`: these are *Webmin* default values. The last important option is the **Postmaster address**. In that field, you must specify the administrator's e-mail address, whom will receive mail reports in case of problems.

[Webmin Index](#)
[Module Index](#)
[Help](#)

General Options

Most Useful General options [What is it?](#)

What domain to use in outbound mail ☐ Use hostname ☐ Use domainname

What domains to receive mail for ☐ Local machine ☐ Whole domain

What trouble to report to the postmaster ☐ Default

Other General Options

Send outgoing mail via host ☐ Deliver directly

Address that receives bcc of each message ☐ None

Timeout on handling requests [Default database type](#)

Default message delivery transport [Sender address for bounce mail](#)

Number of subdirs below the queue dir [Name of queue dirs split across subdirs](#)

Max number of Received: headers [Time in hours before sending a warning for no delivery](#) ☐ Disabled

[Network interfaces for receiving mail](#) ☐ All

[Idle time after internal IPC client disconnects](#) [Timeout for I/O on internal comm channels](#)

[Mail system name](#) [Mail owner](#)

[Official mail system version](#) [Max service requests handled before exiting](#)

[Time to wait for next service request](#)

[Internet hostname of this mail system](#) ☐ Default (provided by system)

[Local internet domain name](#) ☐ Default (provided by system)

[Local networks](#) ☐ Default (all attached networks)

[Send postmaster notice on bounce to...](#) ☐ Default [Send postmaster notice on 2bounce to...](#) ☐ Default

Figure 4-2. Postfix's Main Configuration Screen

The options contained in the second part, **Other general options**, must be configured. The **Internet hostname of this mail system** must be set up with the correct hostname. A wrong entry could lead your e-mails to be rejected from other mail servers. The **Local Internet domain name** option must be specified. Finally, by specifying the correct **Local networks**, you could avoid becoming the victim of illegal spammers. This option tells *Postfix* from which IP address it should accept e-mails. Leave all the notice options to the postmaster. He will receive all error messages from *Postfix*.

The **Mail aliases** section configures the mail redirection to valid existing mailboxes. For example, you can configure it in order for all mail sent to the postmaster to be redirected to root's mailbox. You can also configure e-mail aliases for your users. The first part specifies where *Postfix* should look for in the database file.

The next step concerns **General resource control**. Two options are interesting: **Max size of a message** and **Max size of bounced message**. The first one configures the maximum size of e-mails managed by *Postfix*. It restricts the file size of attachments users of your network could try to send. The second option avoids to fill your hard drive when an e-mail cannot be delivered locally. It bounces in *Postfix*'s queue, waiting to be delivered.

In the **SMTP server options** section, configure **Restrict mail relaying** with these values: `$mydestination,$mynetworks`. It will prevent *Postfix* from relaying e-mails from any client.

4.3. Advanced Configuration

The **General options** section contains a lot of fields, but not all are important. You can choose to send all outgoing mails to a relay server, or to deliver them directly through **Send outgoing mail via host**. If you want to keep trace of all your e-mails, put an address in **Address that receives bcc of each message**, and the latter will receive a copy of each mail. For advanced administration, you could change the *Postfix* database type in the **Default database type** field from hash to dbm. You can specify the **Time in hours before sending a warning for no delivery**. If you plan to use this mail server only to receive mails from the Internet, you could specify the public network interface in the **Network interfaces for receiving mail** field. If you want to change the system user who is running *Postfix*, change it in **Mail owner**. The other options are system-specific, and are not important for configuring *Postfix*.

In the **Canonical mapping** section, you can specify mapping table files, which are used to rewrite e-mail headers managed by *Postfix*. For example, in **Address mapping lookup tables**, you could associate the name of employees with their e-mails: John.Doe@example.com and jdoe@example.com.

In the **Virtual domains** section, you can specify domain table files, which are used by *Postfix* to redirect specific e-mails or entire domains to another server.

The **Transport mapping** section tells *Postfix* which files contain domains to relay. In this file, specify all e-mails from a domain which is accepted by your mail server, which then has to be resent to another host.

The **Relocating mapping** section tells *Postfix* what to do with nonexistent e-mails or domains.

The **Local delivery** section contains options to help you configure the “life” of e-mails after *Postfix* receives them. It looks for a `.forward` file in user’s home directories (which indicate the e-mail address to send the mail back to); with **Search list for forward**, give the e-mail to `procmail` for filtering (**External command to use instead of mailbox delivery**), and then delivered to the **Spool directory**.

In the **SMTP server options** section, you can prevent receiving spam mail by configuring the **DNS domains for blacklist lookups** field. Some Internet servers run public DNSs with blacklisted hosts. These mail hosts are relaying spam mail. So configuring this option allows *Postfix* to look in these databases before accepting mails. All *Postfix* response in the bottom of the page should be kept to the default value.

The **Debugging features** section contains two options. The first one, **List of domain/network patterns for which verbose log is enabled**, specifies a list of hosts and domains, for which *Postfix* logs have to be verbose. The second one configures the log’s verbosity level.

If LDAP is installed on your system, you could access and configure options in the **LDAP lookups** section.

Chapter 5. Incoming Mail Server: POP And IMAP

By using POP (Post Office Protocol) and/or IMAP (Internet Message Access Protocol), users can access their electronic mailboxes and get their e-mails.

5.1. Foreword

If you've done a standard **Mandrake Linux** installation, mail access servers (POP3 or IMAP) are launched on demand by the *xinetd* super daemon. When a connection is done on the POP port (or IMAP), the *xinetd* daemon launches the appropriate program to answer the request.

A POP3 user will fetch his e-mails on his computer and will read them with a mail reader like *kMail* or *Evolution*, while the IMAP protocol allows users to leave their mails on the server and manage it remotely. IMAP is really adapted for mobile users, but, because mails consume a lot of disk space, the system administrator should check regularly his server or set-up quota policies.

5.2. Installation

To configure a POP/IMAP server, the *xinetd* and *imap-2000c* packages must be installed.

The server configuration is done through the **Extended Internet Service** module. You can find this module in the **Servers** category. If you don't find it, you can get the file *xinetd.wbm* from the Webmin site (<http://www.webmin.com/webmin/standard.html>) and install it by using the **Webmin configuration** module of the **Webmin** category. If you click on the **Webmin modules** icon, you will end up on a page where you can tell *Webmin* the path to the module to be installed (or removed).

5.3. Step-by-Step Configuration Example

When you click on the **Extended Internet Service** icon, the program will list all the accessible services on your computer which are managed by *xinetd*. These services can be up (activated) or down (stopped).

[Webmin](#)
[Index](#)
[Module](#)
[Config](#)

Extended Internet Services

[Search docs..](#)

Service name	Type	Port / number	Protocol	User	Server program	Enabled?
chargen	Internet	19	TCP	root	Internal to Xinetd	No
chargen-udp	Internet	19	UDP	root	Internal to Xinetd	No
daytime	Internet	13	TCP	root	Internal to Xinetd	No
daytime-udp	Internet	13	UDP	root	Internal to Xinetd	No
echo	Internet	7	TCP	root	Internal to Xinetd	No
echo-udp	Internet	7	UDP	root	Internal to Xinetd	No
time	Internet	37	TCP	root	Internal to Xinetd	No
time	Internet	37	UDP	root	Internal to Xinetd	No
sgi_fam	RPC		TCP	root	/usr/bin/fam	No
rsvnc	Internet	873	TCP	root	/usr/bin/rsync	No
swat	Internet	901	TCP	root	/usr/sbin/swat	No
cvspserver	Internet	2401	TCP	root	/usr/sbin/cvspserver	No
nntp	Internet	119	TCP	news	/usr/sbin/leafnode	No
ftp	Internet	21	TCP	root	/usr/sbin/in.ftpd	No
cvspserver	Internet	2401	TCP	root	/usr/sbin/cvspserver	No
imap	Internet	143	TCP	root	/usr/sbin/imapd	Yes
imaps	Internet	993	TCP	root	/usr/sbin/imapd	No
ftp	Internet	21	TCP	root	/usr/sbin/in.ftpd	Yes
pop2	Internet	109	TCP	root	/usr/sbin/ipop2d	No
pop3	Internet	110	TCP	root	/usr/sbin/ipop3d	Yes
pop3s	Internet	995	TCP	root	/usr/sbin/ipop3sd	No
imap	Internet	143	TCP	root	/usr/sbin/imapd	Yes
ftp	Internet	21	TCP	root	/usr/sbin/in.ftpd	Yes

[Create a new internet service](#)

Edit Defaults

Click this button to edit default options that apply to all internet and RPC services

Apply Changes

Click this button to apply the current configuration by sending a SIGHUP signal to the running xinetd process



[Return to index](#)

Figure 5-1. xinetd Module's Start-Up Screen

As we have previously seen, the two most used protocols for fetching mails are POP and IMAP. Both have an associated protocol dedicated to security: POP3S and IMAPS, which crypt data flow.

The `imap-2000c` package sets up services with standard options. Click on a service name to configure it. For the service to be accessible, check yes in the **Service enabled?** option. Then, you can restrict access to your service through the third table: **Service access control**. Enter the IP addresses of computers allowed to retrieve mail in the box **Allow access from** and select **Only listed hosts**....

[Webmin Index](#)
[Module Index](#)

Edit Internet Service

Service network options	
Service name	pop3
Bind to address	<input checked="" type="radio"/> All <input type="radio"/> <input type="text"/>
Socket type	Stream
Service enabled?	<input checked="" type="radio"/> Yes <input type="radio"/> No
Port number	<input checked="" type="radio"/> Standard <input type="radio"/> <input type="text"/>
Protocol	Default

Service program options	
Service handled by	<input type="radio"/> Internal to Xinetd <input checked="" type="radio"/> Server program /usr/sbin/pop3d <input type="radio"/> Redirect to host <input type="text"/> port <input type="text"/>
Run as user	root
Run as group	<input checked="" type="radio"/> From user <input type="radio"/> <input type="text"/>
Wait until complete?	<input type="radio"/> Yes <input checked="" type="radio"/> No
Max concurrent servers	<input checked="" type="radio"/> Unlimited <input type="radio"/> <input type="text"/>
Nice level for server	<input checked="" type="radio"/> Default <input type="radio"/> <input type="text"/>
Maximum connections per second	<input checked="" type="radio"/> Unlimited <input type="radio"/> <input type="text"/>
Delay if maximum is reached	<input type="text"/> seconds

Service access control	
Allow access from	<input checked="" type="radio"/> All hosts <input type="radio"/> Only listed hosts..
Deny access from	<input checked="" type="radio"/> No hosts <input type="radio"/> Only listed hosts..
Allow access at times	<input checked="" type="radio"/> Anytime <input type="radio"/> <input type="text"/>

[Return to service list](#)

Figure 5-2. POP3 Configuration Module

Then, **Save** your changes and click on **Apply changes** to tell the *xinet* daemon to apply the new configuration.

5.4. Advanced Configuration

There are many other options which are not required for a standard configuration. In **Service Network options**, the **Bind to address** and **Port number** options allow to force the daemon to listen to a specific address-and-port couple. If you have many network interfaces and you want mail traffic to pass only through a specific one, you can specify it here by entering the the interface's IP address.

In the second table named **Service program options**, you can redirect all your requests to another computer. In **Service handled by**, check **Redirect to host** and enter the IP address and port of the machine. The **Run as user** and **Run as group** options both allow the service to be run as a specific user.

Xinetd allows to set up boundary options for each services. The **Max concurrent servers** option specifies the maximum number of daemon instances to be launched at the same time. **Maximum connection per second** specifies the number of connection requests the server can handle. If the maximum is reached, then the **Delay if maximum is reached** option specifies the time interval until the instance of the service daemon will be reachable again. In the POP3 example, you can specify that only three POP3 servers can be launched and respond to five connection requests per second.

The last useful option is **Nice level for server**, which indicates the program's system priority.

For very large mail servers, you can use stand-alone pop-imap (not managed by *xinetd*).

Chapter 6. Resource Sharing

6.1. Resource Sharing: Samba

The *Samba* server allows you to easily integrate a **Mandrake Linux** computer in an heterogeneous network. Through *Samba*, your computer can appear in other people's network neighborhood and act as a **Microsoft Windows** server by sharing files, distant user accounts, printers, etc.

6.1.1. How to Install

In order to be able to configure *Samba* with *Webmin*'s interface, you have to check that *samba* packages are installed on your system. Then, the module called **Samba Windows FileSharing** allows you to configure the server and devices. You can find this modules in the **Servers** category (by clicking on the **Servers** tab). If this module is unreachable, you need to install it from the Webmin site (<http://www.webmin.com/webmin/standard.html>) using the module **Webmin Configuration**. Click on the **Webmin Module** icon and indicate the location of the file you wish to install (or delete).

6.1.2. Step-by-Step Configuration Example

[Webmin Index](#)
[Module Config](#)


Samba Share Manager


[Search docs.](#)


Share Name	Path	Security
homes	All Home Directories	Read/write to all known users
printers	All Printers	Printable to everyone


[Create a new file share](#) [Create a new printer share](#) [Create a new copy](#) [View All Connections](#)


Global Configuration



[Unix Networking](#)



[Windows Networking](#)



[Authentication](#)


[Windows to Unix Printing](#)


[Miscellaneous Options](#)


[File Share Defaults](#)


[Printer Share Defaults](#)


[SWAT](#)

Encrypted Passwords

- [Edit Samba users and passwords.](#)
- [Convert Unix users to Samba users.](#)
- [Configure automatic Unix and Samba user synchronisation.](#)

Restart Samba Servers

Click this button to restart the running Samba servers on your system. This will force the current configuration to be applied. This will also disconnect any connections to the server, so if you do not want the current configuration to be applied immediately you should just wait 1 minute until Samba reloads the configuration automatically.

Figure 6-1. The Samba Module's Main Window

Once in the *Samba* configuration section of *Webmin*, click on the **Windows Networking** icon. Define a **Workgroup** for your server. At the same time, you have the possibility to modify the **Server Name** and the **Server Description**. Then, select the security level **User level** and validate your choice by clicking on **Save**.

[Webmin Index](#)
[Module Index](#)

Windows Networking Options

Figure 6-2. Configuring The Common Networking Options

If you plan to use the file server with a *Windows 9x* client, you have to modify the default value in the Authentication section: set Use encrypted passwords? to No.

[Webmin Index](#)
[Module Index](#)

Password Options

Figure 6-3. Setting The Authentication Method

Create a new shared directory in your file tree.

[Webmin Index](#)
[Module Index](#)

Create File Share

Figure 6-4. Configuring Your Sharing Entries

At last, click on the **Start Samba Server** button to activate your choices. If your *Samba* server is already running, you have to click on **Restart Samba Server**.

6.1.3. Advanced Configuration

The main part of the configuration screen is ruled by the **Share List** section. When you want to share a new directory from your tree, you can either create a new sharing with **Create a new file share**, or create a copy of an already existing sharing with **Create a new copy**. This allows you to configure all options for one specific shared directory and to create new sharing with the same options, like a template. In the same time, you can watch all current connections to your *Samba* server.

The **Authentication** section deals with *Samba* passwords. In the **File Share Defaults** section, *Samba* waits for you to configure the default directory that user will find in their sharing list. You can tune options concerning file names (case specificity for example) or file permissions (like `umask` or group permissions).

You should set to **Printer Share Defaults** the local printers you wish to be shared. Indicate the name and the availability of the printer. Moreover, all locally accessible printers you configured will be available through *Samba*. You can configure some parameters about this option in the **Windows to Unix Printing** section.

In the **Encrypted Passwords** section, you can manage all your *Samba* users. You can either manage them by hand or tell the *Webmin* module to automatically synchronize itself with the local-users base.

If you find this interface isn't what you await for, try the *SWAT interface*, which is the official *Samba* web interface.

6.2. Resource Sharing: FTP

WU-FTP allows you to create and set up an FTP server. With the latter, your company can share files with people connected to the Internet (or to your intranet). According to your configuration, they could eventually upload files on your server.

6.2.1. Installation

The first operation is to install *WU-FTP* on your system. Use `rpmrake` or type `urpmi wu-ftp` in a terminal.

The server's configuration is done through *Webmin*'s **FTP server** module. You will find it in the **Servers** category. If you can't find it, you can get the `wuftpd.wbm` file on the *Webmin* site (<http://www.webmin.com/webmin/standard.html>) and install it by using the **Webmin configuration** module in the **Webmin** category. If you click on the **Webmin modules** icon, you will end up on a page where you can tell *Webmin* the path to the module to be installed (or removed).

6.2.2. Step-by-Step Configuration Example

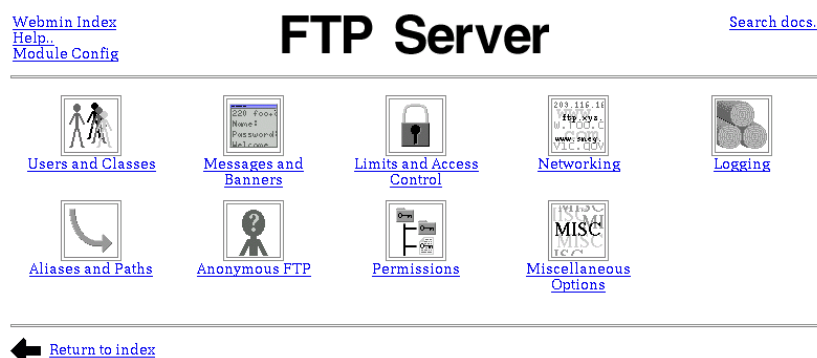


Figure 6-5. WU-FTP's Main Configuration Page

In the **Message and Banners** section, specify the e-mail address of the FTP administrator in **Owner's email address**. This e-mail will be seen by users logging on your server. They can therefore send you a report if they have any problem.

[Webmin Index](#)
[Module Index](#)
[Help..](#)

Messages and Banners

Messages, banners and README files			
Message files	Path	When to display	Classes to display for
	/welcome.msg	<input checked="" type="radio"/> At login <input type="radio"/> Entering any dir <input type="radio"/> Entering dir	
	message	<input type="radio"/> At login <input checked="" type="radio"/> Entering any dir <input type="radio"/> Entering dir	
		<input type="radio"/> At login <input type="radio"/> Entering any dir <input type="radio"/> Entering dir	
README files	Path	When to display last modified date	Classes to display for
	README*	<input checked="" type="radio"/> At login <input type="radio"/> Entering any dir <input type="radio"/> Entering dir	
	README*	<input type="radio"/> At login <input checked="" type="radio"/> Entering any dir <input type="radio"/> Entering dir	
		<input type="radio"/> At login <input type="radio"/> Entering any dir <input type="radio"/> Entering dir	
Greeting level	<input checked="" type="radio"/> Hostname and version <input type="radio"/> Hostname <input type="radio"/> Neither		
Pre-login banner	<input checked="" type="radio"/> None <input type="radio"/> From file <input type="text"/>		
Hostname for messages	<input checked="" type="radio"/> System hostname <input type="radio"/> <input type="text"/>		
Owner's email address	<input type="radio"/> Default <input checked="" type="radio"/> root@localhost <input type="text"/>		

Save

← [Return to FTP server options](#)

Figure 6-6. Wu-FTP Banner And Messages

If you plan to maintain a public FTP server, you should look at the options in the **Anonymous** access section. Anonymous access allows people with accounts on your computer to login with an FTP client and to retrieve files. Specify the **Anonymous root directories** and/or the **Guest root directories** with the path of the anonymous/guest FTP directories. You can specify a group for all files for anonymous user through **Unix groups for anonymous users**. The last option allows you to activate the check (and eventually deny) on the anonymous password (**Anonymous FTP password check**). Anonymous users don't have passwords to access to your FTP server. In general, they enter their e-mail address as a password. For example, you can force users to enter a password with the @ character. Moreover, you can deny some specific anonymous passwords through **Anonymous FTP passwords to deny**.

[Webmin Index](#)
[Module Index](#)
[Help..](#)

Anonymous FTP

Anonymous FTP options							
Anonymous FTP root directories	<table border="1"> <thead> <tr> <th>Directory</th> <th>For class</th> </tr> </thead> <tbody> <tr> <td>/var/ftp</td> <td>Any</td> </tr> <tr> <td></td> <td>Any</td> </tr> </tbody> </table>	Directory	For class	/var/ftp	Any		Any
Directory	For class						
/var/ftp	Any						
	Any						
Guest root directories	<table border="1"> <thead> <tr> <th>Directory</th> <th>For Unix users</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> </tr> </tbody> </table>	Directory	For Unix users				
Directory	For Unix users						
Unix groups for anonymous users	<table border="1"> <thead> <tr> <th>Switch to group</th> <th>For classes</th> </tr> </thead> <tbody> <tr> <td></td> <td>all</td> </tr> </tbody> </table>	Switch to group	For classes		all		
Switch to group	For classes						
	all						
Anonymous FTP password check	<input checked="" type="radio"/> Default <input type="radio"/> Must be RFC822 email address <input type="radio"/> Warn only						
Anonymous FTP passwords to deny	<input type="text"/>						

Save

← [Return to FTP server options](#)

Figure 6-7. Anonymous FTP Configuration Page

The **Permissions** section allows you to restrict use of some FTP commands with **Command restrictions**. By default, anonymous and guest users don't have the rights to use dangerous commands like `chmod` or `delete`. To avoid security problems, you can deny some file names to be uploaded with **Disallowed upload filenames**.

The last step is to create new aliases in **Aliases and Paths**. If you want your users to be allowed to access specific directories (like `/home/project/marketing`), specify an **alias** (like `marketing`) that can be reached by FTP users.

6.2.3. Advanced Configuration

The first section, **Users and Classes**, contains options about user access. You can specify the users who will be treated as guests. For example, if you have many local users and you want to allow them to access your FTP tree, add their user *ID* to **guest**. The same thing could be done for the **guest** group. In the same way, the **Use class** allows to force a specific access type according to the client's source address. By creating a user class, you can configure specific options in other sections applied to this class. For example, you can create a specific class for a department of your company, which has permissions to post data on your server. In the same time, another department would be represented by another class and would have only read and get access. The last point is to deny access. You can (and actually have to) deny some users like `daemon` or `shutdown`. The default value contains system users. You can specify users or groups *ID* to deny.

The **Messages and Banners** section allows you to configure the message the server will display. You can specify the welcome message **At login** or in each directory: **Entering any directory**. The default value tells *WU-FTP* to look for a `.message` file in each directory and to display it. The same configuration idea is repeated with `README` files. The next option allows you to choose the **Greeting level**, if you want to display the hostname or any other banner (**Hostname for messages**). You can display a message before a user logs in in the **Pre-login banner** field.

The options in the **Limits and Access Control** section enable to deny access from a specific IP class. FTP requests from these IP addresses will get an error message through the **Deny access from** option. You can limit concurrent access to your server for a specific class (**Concurrent user limits**) with an error message in case of problem, or limit the number of files (or bytes) transferred in an FTP session (**File and data transfer limits**). This options is quite detailed, because you can choose if the transfer limit is applied to data or text, and to upload and/or download transfers. You can choose access rights to a specific file with **Deny access to files** and **Allow access to files even if denied**. The last option configures the FTP session limit for some particular users: **Anonymous session limit** and **Guest session limit**, the number of time a user can type a wrong login/password before being rejected (**Maximum login failures**) and the user's ability to change group (**Can switch groups?**).

The main option in the **Networking** section is used to configure passive FTP connections. You can specify which IP addresses have the right to establish passive FTP connections (**Addresses for PASV connections**) and on which server ports (**Ports for PASV connections**).

The **Logging** section is used to configure which users are logged. Each specific action: **All commands**, **Transfer** and **Security violations** can be recorded in the session journal.

At last, options in the **Miscellaneous Options** section allow you to specify the command launched by *WU-FTPD* to display each listing mode and the FTP server's process priority with the **Service process nice level**. The last option configures the `umask` of uploaded files (**Default umask for uploaded files**). You can configure this file-permission mask to allow particular rights on uploaded files.

6.3. Resource Sharing: NFS

The **Network File System** service allows you to easily export directories of your computer to others through the network. Through NFS, you can share data among several users. This type of sharing is really easier to set up than *Samba*, but it is only used on *GNU/Linux* and *UNIX* systems. NFS is very insecure and must be used only in a secure local network.

6.3.1. How to Install

To configure your **NFS Exports** system using *Webmin*, you have to install the `nfs-utils` and `nfs-utils-clients` packages.

6.3.2. Step-by-Step Configuration Example

The **NFS Exports** configuration button is in the **System** index. You just have to click on the **Add a new export** link and the configuration page will open.



Figure 6-8. Starting The NFS Configuration

This module is very simple to use because you can have a lot of explanations about each parameter. Simply click on a parameter you do not understand and a pop-up will answer your question.

Once in the **Create Export** page, you just have to write the directory name you want to allow access from other computers. If you are not sure of the name of the directory you want to export, you can click on the icon with three dots and you will be able to browse your local system tree.

By default, the export is accessible to **Everyone**, this should be changed to the sub-network you actually use (for example: `192.168.1.0/255.255.255.0`) or a netgroup.



Figure 6-9. Creating NFS Export

You can restrict a little more the access to your exported directories. For example, you can choose the ID that you want to trust or not, you can also choose to allow to read or read and write in your directories.

6.3.3. How to Access The Exported Directory

To configure your client computer (192.168.50.92 in our example), you have to log on it and use the **Disk and Network Filesystems** button in the **System** index.

On a *UNIX* system, making a filesystem or a directory accessible is called “mounting” this remote directory. A mount point is the local directory where the remote one is going to be accessible.

To mount the `/home/myhome/sharing_dir` exported directory, you need to add a new entry in the **Disk and Network File System** icon of your system. To do so, you must **Add a mount** by selecting the **Network Filesystem (nfs)** type.

[Webmin Index](#)
[Search docs...](#)

Disk and Network Filesystems

Mounted As	Type	Location	In use?	Permanent?
/	New Linux Native Filesystem	IDE device A partition 5	Yes	Yes
/dev/pts	PTS Filesystem	none	Yes	Yes
/dev/shm	RAM Disk	none	Yes	Yes
/home	New Linux Native Filesystem	IDE device A partition 8	Yes	Yes
/media	New Linux Native Filesystem	IDE device A partition 9	Yes	Yes
/mnt/cdrom	SUPERMOUNT	/mnt/cdrom	Yes	Yes
/mnt/cdrom2	SUPERMOUNT	/mnt/cdrom2	Yes	Yes
/mnt/floppy	SUPERMOUNT	/mnt/floppy	Yes	Yes
/proc	Kernel Filesystem	proc	Yes	Yes
/usr	New Linux Native Filesystem	IDE device A partition 7	Yes	Yes
Virtual Memory	Virtual Memory	IDE device A partition 6	Yes	Yes
/dev	DEVFS	none	Yes	No
/mnt/save	New Linux Native Filesystem	IDE device B partition 1	Yes	No
/proc/bus/usb	USB Devices	none	Yes	No
... c/sys/fs/binfmt_misc	BINFMT_MISC	none	Yes	No

Add mount Type: Network Filesystem (nfs)

[Return to index](#)

Figure 6-10. Creating NFS Mount Points

In this example, we choose to mount the exported directory in `/mnt/disk`. This means you will be able to browse the `/home/myhome/sharing_dir` subdirectories of the 192.168.50.92 remote server using your preferred file manager in the `/mnt/disk` local directory.

[Webmin Index](#)
[Module Index](#)

Create Mount

Network Filesystem Mount Details

Mounted As: /mnt/disk

Save Mount? ☒ Save and mount at boot ☐ Save ☐ Don't save

Mount now? ☒ Mount ☐ Don't mount

NFS Hostname: 192.168.50.1 NFS Directory: /home/myhome/sharing_c...

Advanced Mount Options

Read-only? ☐ Yes ☒ No

Allow device files? ☐ Yes ☒ No

Disallow setuid programs? ☐ Yes ☒ No

Retry mounts in background? ☐ Yes ☒ No

Timeout: ☐ Default ☐

NFS version: ☐ Highest ☐

Allow user interrupt? ☐ Yes ☒ No

Buffer writes to filesystem? ☐ Yes ☒ No

Allow execution of binaries? ☐ Yes ☒ No

Allow users to mount this filesystem? ☐ Yes ☒ No

Return error on timeouts? ☐ Yes ☒ No

Number of Retransmissions: ☐ Default ☐

NFS Port: ☐ Default ☐

RPC Protocol: ☐ TCP ☒ UDP

[Return to filesystems list](#)

Figure 6-11. Configuring NFS Mount Point

If you select **Save and mount at boot**, this remote directory will be mounted each time you reboot. The server must be accessible each time you boot. If not, you will get warning messages.



It is very dangerous to modify the other mount points of local partitions on your computer. It may become impossible to reboot!

Chapter 7. MySQL Database Server

MySQL is a true multi-user, multi-threaded SQL (Structured Query Language) database server. MySQL is a client/server implementation that consists of a server daemon (`mysqld`) and many different client programs/libraries. The main goals of MySQL are speed, robustness and ease of use.

7.1. Getting Started

To configure your *MySQL Server* using *Webmin*, you must install the following **Mandrake Linux** RPM packages: `MySQL`, `MySQL-client` and `libmysql10`.

The **MySQL Database Server** configuration button is in the **Servers** index. Notice that there are two default databases (`mysql` and `test`). You should **not** neither modify, nor erase these databases.

There are a lot of other parameters which are interesting to use and manage (like permissions). If you do not understand the objective of a few configuration pages, you can use the **Help** link which will pop up an explanation window.

7.2. Creating a User For The Database

A database user has nothing to do with a *UNIX* user. This means that you have to manage the users differently. From **MySQL Database's** index page, click on the **User Permissions** and then on **Create new user**. The database users can have different permissions, which are all listed in that window.

Webmin Index
Module Index
Help..

Create User

MySQL user details

Username ☐ Anonymous user ☒ db_user

Password ☐ None ☒ Set to.. DB-V5er

Hosts ☒ Any ☐ [text field]

Permissions

- Select table data
- Insert table data
- Update table data
- Delete table data
- Create tables
- Drop tables
- Reload grants
- Shutdown database
- Manage processes
- File operations

Save

← [Return to user list](#) | [Return to database list](#)

Figure 7-1. Creating a MySQL User

7.3. Creating a Database

First of all, let's create the database which will hold our tables. Click on **Create a new database** from the main page and write in a name for your database.

[Webmin Index](#)
[Module Index](#)
[Help..](#)

Create Database

New database options

Database name:

Initial table: ☐ None

Initial table structure

Field name	Data type	Type width
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>

[← Return to database list](#)

Figure 7-2. Creating a MySQL Database

7.4. Creating a Table

Once the database is created, click on its icon to access the **Edit Database** page. Notice that there are no tables for the moment, but we can create new tables and drop or **back up** the database. Before clicking on the **Create a new table** label, you can select the number of fields the table will have (4 by default).

In the **Create Table** page, you must write the table's name and the field parameters.

[Webmin Index](#)
[Module Index](#)
[Help..](#)

Create Table

New table options

Table name:

Copy fields from table:

Type:

Initial fields

Field name	Data type	Type width	Part of primary key?
Number	int	<input type="text"/>	<input checked="" type="checkbox"/> Yes
Name	tinytext	<input type="text"/>	<input type="checkbox"/> Yes
Birth	date	<input type="text"/>	<input type="checkbox"/> Yes
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> Yes

[← Return to table list](#) | [Return to database list](#)

Figure 7-3. Creating a MySQL Table

If you want to modify the table parameters or add new fields, you can click on the name of the table from the **Edit Database** page.

[Webmin Index](#)
[Module Index](#)
[Help..](#)

Edit Table

Table MyTable in database test_db

Field name	Type	Allow nulls?	Key	Default value	Extras
Number	int(11)	No	Primary	0	
Name	tinytext	Yes	None	NULL	
Birth	date	Yes	None	NULL	

Add field of type:

[Return to table list](#) | [Return to database list](#)

Figure 7-4. Modifying a MySQL Table

7.5. Managing Data in a Table

We created everything needed to use the MySQL database. There are a lot of client programs you can use to connect to the MySQL server and there are a lot of programs that need such databases. You can also use *Webmin* to manage your data. In the **Edit Table** page, you can click on the **View Data** button to add, modify, or remove data.

[Webmin Index](#)
[Module Index](#)

Table Data

Table MyTable in database test_db

	Number	Name	Birth
<input type="checkbox"/>	1	William	1982-02-24
<input type="checkbox"/>	2	Arnold	1978-11-02
<input type="checkbox"/>	3	Michael	1980-06-05

[Select all](#) [Invert selection](#)

[Return to field list](#) | [Return to table list](#) | [Return to database list](#)

Figure 7-5. Managing Your Data

Chapter 8. NIS Client And Server

To simplify the management of users on your local network, it is possible to centralize network information such as user and password lists on a NIS domain (Network Information System).

With NIS, users can connect on any computer using the same login and password. Information sharing allows to distribute files such as `/etc/passwd`, `/etc/shadow` or `/etc/hosts` to share machine passwords or aliases. To distribute the data, you will have to configure a **Ressource Sharing** server like NFS (see *Resource Sharing: NFS*, page 59) or *Samba* (see *Resource Sharing: Samba*, page 55)

8.1. Installation

The configuration of the NIS server is done in two steps: the first one is the configuration of the **NIS tables**¹ of the server; the second one is the configuration of each client.

- on the server, you need to install the RPM package named `ypserv`;
- for each clients, you will need `portmap`, `yp-tools` and `ypbind`.

In order to use the *Webmin* module **NIS Client and Server**, you will have to select the category **System**, then the **NIS Client and Server** button.

8.2. Step-by-Step Configuration

8.2.1. NIS Server

You need to configure the NIS domain using your domain name (such as `mydomain.test`). And after clicking on the **NIS Server** icon, you must choose the **NIS tables to serve**. For our example, we will select the files `passwd`, `group` and `shadow` (you use the `Ctrl` key to select several files in the list).

1. The NIS tables are the files you have chosen to export.

[Webmin Index](#)
[Module Index](#)

NIS Server

NIS server options

Enable NIS server?
☒ Yes ☐ No

Serve NIS domain
☐ Same as client ☒ mydomain.test

Server type
☒ Master NIS server ☐ Slave of server

Master NIS server options

Lookup missing hosts in DNS?
☐ Yes ☒ No

Push updates to slaves?
☐ Yes ☒ No

NIS tables to serve
passwd
group
hosts
rpc
services

Minimum UID for 'Unix user' table records
500

Minimum GID for 'Unix group' table records
500

Slave servers

Master NIS files

File for 'Unix users'	/etc/passwd	File for 'Unix shadow passwords'	/etc/shadow
File for 'Unix groups'	/etc/group	File for 'Unix group passwords'	/etc/gshadow
File for 'Extra user information'	/etc/passwd.adjunct	File for 'Sendmail aliases'	/etc/aliases
File for 'Ethernet addresses'	/etc/ethers	File for 'Boot parameters'	/etc/bootparams
File for 'Host addresses'	/etc/hosts	File for 'Network addresses'	/etc/networks
File for 'Printers'	/etc/printcap	File for 'Network protocols'	/etc/protocols
File for 'Public keys'	/etc/publickey	File for 'RPC programs'	/etc/rpc
File for 'Network services'	/etc/services	File for 'Netgroups'	/etc/netgroup
File for 'NetIDs'	/etc/netid	File for 'Automounter maps'	/etc/auto.master
File for 'Automounter home'	/etc/auto.home		

Save and Apply

Return to NIS menu

Figure 8-1. NIS Server

You do not need to modify the file description done in the **Master NIS files** section. In the **NIS menu**, the **NIS Tables** icon allows to modify the tables that are served. The **Server Security** icon allows to select the clients you want to serve, or not.

8.2.2. NIS Client

For each client, you have to configure the NIS domain name parameter with the domain name used by the server. You must also enter IP address.

[Webmin Index](#)
[Module Index](#)

NIS Client

NIS client configuration

NIS domain
☐ None (NIS disabled) ☒ mydomain.test

NIS servers
☐ Find by broadcast ☒ Listed below ..
192.168.1.1

Save and Apply

Return to NIS menu

Figure 8-2. NIS Client

That’s all.

8.3. Client Advanced Configuration

Among all the exported data, some can be redundant with regards to the local configuration. So it is possible to choose the priority to be allocated to every source (local, NIS or other). To do so, you can use the **Client Services** button on each NIS client. It allows to choose the preferred order to search for data. For example, you can choose to resolve the host's addresses using 1) the file `/etc/hosts`, 2) then the NIS hosts served (if you selected it in the NIS table), 3) and finally (if the client cannot resolve any more), use the DNS server.

To verify that the client communicates with the server, you can use the `ypcat passwd` command to read the password data served by the server.

If you export your user's directories (using NFS for example), and if you start the **autofs** service on your NIS client, users should be able to automatically mount their own home directory when they log onto the client. This way, everybody can automatically log on every client, having all their personal data and configuration files.

Chapter 9. BIND DNS Server

The *BIND DNS Server* module creates and edits domains, DNS records and *BIND* options for the 8.x and 9.x releases. *BIND* (Berkeley Internet Name Domain) is an implementation of the Domain Name System (DNS) protocol and provides an open, redistributable reference implementation of the major components of the Domain Name System. A DNS server allows to associate a name to an IP address and vice versa. For example: `www.mandrakesoft.com` ("Name") is associated to `63.209.80.236` ("Address").

BIND is very useful for simple configurations, but there are a few differences between the 8.x and 9.x releases. However, you must be careful with this *Webmin* module because not all *BIND* 9.x options are supported yet. Hence, if you try to use the advanced options, you will have to look at the log files more carefully than with the other *Webmin* modules, in order to make sure *BIND* is working properly.

9.1. Installation

First of all, you must install the RPM package which contains the DNS server. The server we will use is *BIND*. The first thing to do is to install the three following packages: `bind`, `bind-utils`, and `caching-nameserver`.

In order to use the *Webmin* module, *BIND DNS Server*, you will have to select the **Servers** category and then the **BIND DNS Server** button (with the number 8 in the icon, **not number 4**).

9.2. Step-by-Step Configuration Example

Now, you should change, in the default **Files and Directories** category, the **Path to zone transfer program** field and enter `/var/named/`

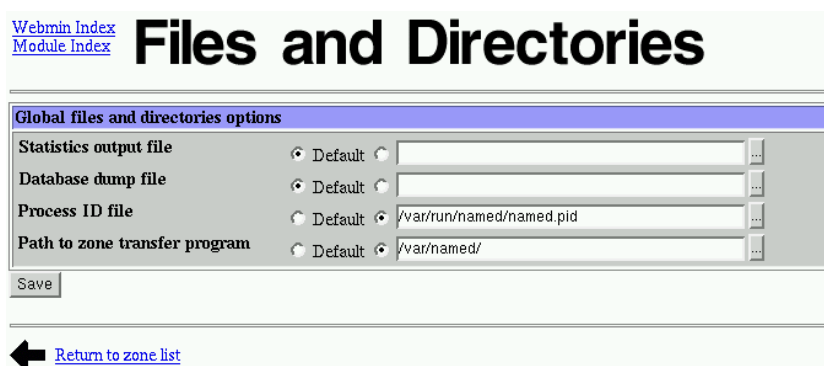


Figure 9-1. Files And Directories

You will notice that through `caching-nameserver`, there are two **Existing DNS Zones**; the **Root Zone** and the master zone, "127.0.0". The former is used by the DNS server and contacts the root servers on the Internet, so it can resolve domain names not handled by your DNS server. Unless your DNS server is used on an internal network (no access to the Internet) or if you are forwarding all queries to another server, you should not delete this root zone.

9.2.1. Configuring The DNS

In order to use each network services properly, you need to create a master zone which will describe your local network. We will concentrate on the **Existing DNS Zones** part. Notice that the **Root zone** is already created.

Master 127.0.0 describes the loopback network.

We want to create the **Root zone** which will describe all our local network machines. Select **Create a new master zone** and complete the page, as shown in figure 9-2.

[Webmin Index](#)
[Module Index](#)

Create Master Zone

New master zone options

Zone type	<input checked="" type="radio"/> Forward (Names to Addresses) <input type="radio"/> Reverse (Addresses to Names)		
Domain name / Network	mydomain.test		
Records file	<input checked="" type="radio"/> Automatic <input type="radio"/> ...		
Master server	mycomputer.mydomain.test	<input checked="" type="checkbox"/> Add NS record for master server?	
Email address	myemail@mydomain.test		
Use zone template?	<input type="radio"/> Yes <input checked="" type="radio"/> No		
	IP address for template records		
Refresh time	10800	seconds	Transfer retry time
			3600 seconds
Expiry time	604800	seconds	Default time-to-live
			38400 seconds

Create

[Return to zone list](#)

Figure 9-2. Creating a Forward Master Zone

A new page with many icons will be displayed: don't worry since most of them can be ignored if you do not need advanced configuration. You will be able to add all network machine names through this page, but you should create the reverse part of your master zone first. In fact, a DNS zone is composed of two parts; one for name-to-address conversion (i.e.: forward) and another for address-to-name conversion (i.e.: reverse).

Then, select **Return to the zone list** and choose **Create a new master zone** once more but this time, you must change the selection from **Forward** to **Reverse**. Instead of writing your domain name, you must write the network class like described: for a 192.168.1.0/24 network, you should write 192.168.1.

[Webmin Index](#)
[Module Index](#)

Create Master Zone

New master zone options

Zone type	<input type="radio"/> Forward (Names to Addresses) <input checked="" type="radio"/> Reverse (Addresses to Names)		
Domain name / Network	192.168.1		
Records file	<input checked="" type="radio"/> Automatic <input type="radio"/> ...		
Master server	mycomputer.mydomain.test	<input checked="" type="checkbox"/> Add NS record for master server?	
Email address	myemail@mydomain.test		
Use zone template?	<input type="radio"/> Yes <input checked="" type="radio"/> No		
	IP address for template records		
Refresh time	10800	seconds	Transfer retry time
			3600 seconds
Expiry time	604800	seconds	Default time-to-live
			38400 seconds

Create

[Return to zone list](#)

Figure 9-3. Creating a Reverse Master Zone

9.2.2. Recording Your Network's Computers

This step is the only one you have to restart each time you add a new machine in your network; all other parameters are configured only once, as long as your network does not change and you do not add other DNS servers.

Return to the zone list and select one of the new **Existing DNS Zones** (in our example, they are named 192.168.1 or mydomain.test). If you select mydomain.test, click on **Address** and you can add as many machine **Names** as your IP class allows you to (254 machine names in our example). Notice that the **Update reverse?** option is selected by default. Through this option, the **Reverse** part of your DNS is updated automatically.

[Webmin Index](#)
[Module Index](#)

Address Records

In mydomain.test

Add Address Record

Name

machine6

Time-To-Live

☒ Default
 ☐

seconds

Address

192.168.1.16

Update reverse?

☒ Yes
 ☐ Yes (and replace existing)
 ☐ No

Create

Name	TTL	Address
machine1.mydomain.test	Default	192.168.1.11
machine2.mydomain.test	Default	192.168.1.12
machine3.mydomain.test	Default	192.168.1.13

Name	TTL	Address
machine4.mydomain.test	Default	192.168.1.14
machine5.mydomain.test	Default	192.168.1.15

←

[Return to record types](#)

Figure 9-4. Adding Machine Names

9.2.3. Starting The Service

We created a very simple DNS. To start it and load the new configuration, you must go back to the **zone list** and click on **Start Name Server**.

Start Name Server

Click this button to start the BIND server, and load the current configuration

Figure 9-5. Starting Bind

If the button is not replaced by a new one named **Apply Changes**, then the server did not start because of a configuration error. In this case, you should read the next section.

Apply Changes

Click this button to restart the running BIND server. This will cause the current configuration to become active

Figure 9-6. Apply Changes to Bind

9.2.4. Configuring The Client

In order to use your local network to resolve Internet addresses, you have to configure the client to access the DNS. Go back to *Webmin's* index, and select the **Hardware** tab, then click on the **Network Configuration** icon. Then, select the **DNS Client** and type your DNS's IP if it is a remote client, or 127.0.0.1 if you are on the server.

Webmin Index
Module Index

DNS Client

DNS Client Options

Hostname: mycomputer.mydomain.test

Resolution order: Hosts NIS+ NIS DNS

DNS servers: 127.0.0.1

Search domains: ☐ None ☒ Listed..

mydomain.test

Save

Figure 9-7. Configuring The Client

9.3. Advanced Configuration

9.3.1. How to Debug

If the service did not start, you should look at the `/var/log/messages` file to read the debug output of *BIND*. If you do not find the error, you can use the `named-checkconf` program and `named-checkzone` to check your configuration.

Through the `bind-utils` package, you can use many utilities and therefore, test your DNS (`nslookup` or `dig`). To use them, you should add the DNS server's IP, preceded by the word "nameserver", in your local `/etc/resolv.conf` file.

9.3.2. The `rndc` Command

BIND's configuration is not done yet, since `rndc` is not yet configured to work in all your network. `rndc` is the name server control utility. You can, for example, have statistics (using the `rndc stats` command), or you can ask for status (using the `rndc status` command). For more information about this powerful command, you should read the `man` page: `rndc`.

`rndc` communicates with the nameserver over a TCP connection, sending commands authenticated with digital signatures; you should change the default key using the `/usr/sbin/new_key.pl` command. If you installed the `caching-nameserver` RPM package, a default `rndc` configuration will be done and you can use it. If not, you will have to add the key which is in the `/etc/rndc.conf` file to your `/etc/named.conf` file.

The control statement declares control channels to be used by system administrators in order to affect the operation of the local nameserver. These control channels are used by the `rndc` utility, to send commands to and retrieve non-DNS results from a nameserver. To modify it, you should not select the **Control Interface Options** icon. Indeed, *Webmin* removes the key options, which is in the "controls" statement. You should modify the `/etc/named.conf` file using a text editor. Here is an example of local-only `rndc` use with key:

```
// secret must be the same as in /etc/rndc.conf
key "key" {
    algorithm      hmac-md5;
    secret         "c0b0cEDYZIQKNXDjnRJLmcTuZiXADGfVBahwsAn0d0yJbTmzPeHW00LTTeCt";
};

controls {
    inet 127.0.0.1
    allow { 127.0.0.1;
    }
    keys { "key";
    };
};
```

For security reasons, the **Mandrake Linux** *BIND* server is started with limited user privileges. If you create a secondary DNS in your network, the *BIND* server could have to create its own files using the master DNS.

That's why you should change the owner of the "zones files" created by *Webmin*. The command to use is `chown named -R /var/named/`.

9.3.3. Documentation

If you want to do more with *BIND*, it is strongly recommended you read the *BIND 9 Administrator Reference Manual*, which is available in PDF format on the official *BIND* web site. More documentation is available in HTML format, if you click on **Search docs**, in the upper-right corner of *Webmin*'s **Network Configuration** or **BIND DNS Server** page. At their bottom, there are a lot of very interesting Internet links. Notice that the *Reference Manual* is available in HTML if you click on `bind-9.2.0/html/Bv9ARM.html`.

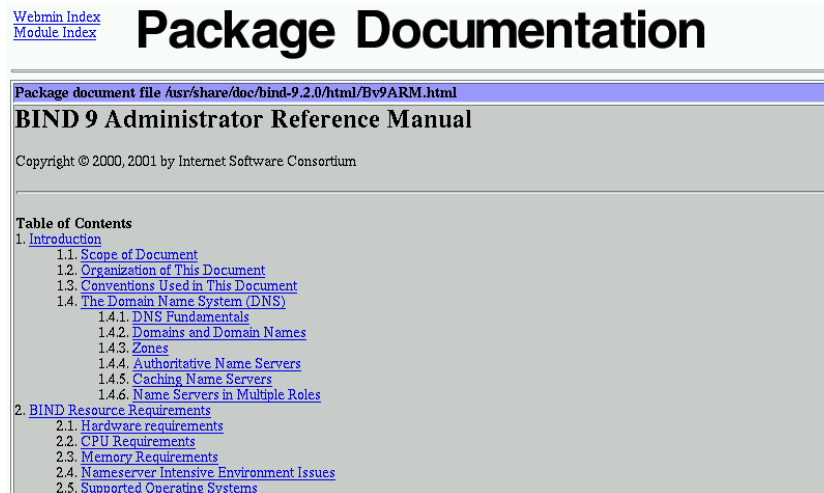


Figure 9-8. The BIND 9 Administrator Reference Manual Through Webmin

9.3.4. A Few More Options

For all the options which are not detailed in this document, we advise you to leave the default ones, unless you really know what you are doing. Each time you change the configuration, you should click on **Apply Changes** to verify that the server configuration is not broken by your change.

9.3.4.1. Global Server Options

- **Logging and Errors:** you can add logging channels or categories to allow you to sort out your logs the way you want to;
- **Access Control Lists:** the ACL statement assigns a symbolic name to an address match list (the IPs are separated by a "space");
- **Forwarding and Transfers:** the forwarding facility can be used to create a large, site-wide cache on a few servers, therefore reducing traffic over links to external nameservers. It can also be used to allow queries by servers that do not have direct access to the Internet, but wish to look up exterior names anyhow. Forwarding occurs only on those queries for which the server is not authoritative and does not have the answer in its cache.

9.3.4.2. Creating a New View

The view statement is a powerful new feature of *BIND* 9 which lets a nameserver answer a DNS query differently, depending on who is asking. It is particularly useful for implementing split DNS setups without having to run multiple servers.

III. Applied Theory

Why Theoretic Stuff in a Practical Guide?

Up until now in this manual, you have been reading very practical information. You should be able to efficiently configure your server and be happy with it.

However, all that is just a glimpse of your **Mandrake Linux** system's possibilities. In order to fully comprehend it, we chose to add two chapters to complete your **Mandrake Linux** knowledge:

- *"Security Under GNU/Linux"*, page 81: this is a must read for any system administrator. Even though you can make your **Mandrake Linux** system quite secure with default tools, efficient security can only be achieved through active administration, taking care of both physical and logical global system security;
- *"Networking Overview"*, page 119: a server is meant to bring services to a network. This manual would have been incomplete without a chapter dedicated to networking. The configuration of the network itself and the different protocols are tackled.

Chapter 10. Security Under GNU/Linux

This document is a general overview of security issues that face the administrator of *GNU/Linux* systems. It covers general security philosophy and a number of specific examples of how to better secure your *GNU/Linux* system from intruders. Also included are pointers to security-related material and programs.



1. The original document (see below) has been adapted to **Mandrake Linux** distribution, removing parts, changing others, etc.
2. This chapter is tightly linked with another one: msec – Mandrake Security Tools of the *Reference Manual*. Many of the aspects covered here are handled by the msec package. We suggest you carefully read this chapter afterwards.

10.1. Preamble

This chapter is based on a *HOWTO* by Kevin Fenzi and Dave Wreski which original is hosted by the Linux Documentation Project (<http://linuxdoc.org>)

10.1.1. Copyright Information

This document is copyrighted (c) 1998 - 2002 Kevin Fenzi and Dave Wreski

Modifications from v1.3.1, 11 February 2002, (C)opyright 2000-2002 MandrakeSoft

10.1.2. Introduction

This chapter covers some of the main issues that affect *GNU/Linux* security. General philosophy and net-born resources are discussed.

A number of other *HOWTO* documents overlap with security issues, and those documents have been pointed to wherever appropriate.

This chapter is **not** meant to be an up-to-date exploits document. Large numbers of new exploits happen all the time. This chapter will tell you where to look for such up-to-date information, and will give you some general methods to prevent such exploits from taking place.

10.2. Overview

This chapter will attempt to explain some procedures and commonly-used software to help your *GNU/Linux* system be more secure. It is important to discuss some of the basic concepts first, and create a security foundation, before we get started.

10.2.1. Why Do we Need Security?

In the ever-changing world of global data communications, inexpensive Internet connections, and fast-paced software development, security is becoming more and more of an issue. Security is now a basic requirement because global computing is inherently insecure. As your data goes from point A to point B on the Internet, for example, it may pass through several other points along the way, giving other users the opportunity to intercept, and even alter, it. Even other users on your system may maliciously transform your data into something you did not intend for. Unauthorized access to your system may be obtained by intruders, also known as “crackers”, who then use advanced knowledge to impersonate you, steal information from you, or even deny you access to your own resources. If you’re wondering what the difference is between a “hacker” and a “cracker”, see Eric Raymond’s document, How to Become a Hacker (<http://www.tuxedo.org/~esr/faqs/hacker-howto.html>).

10.2.2. How Secure Is Secure?

First, keep in mind that no computer system can ever be completely secure. All you can do is make it increasingly difficult for someone to compromise your system. For the average home *GNU/Linux*, not much is required to keep the casual cracker at bay. However, for high profile *GNU/Linux* users (banks, telecommunications companies, etc.), much more work is required.

Another factor to take into account is that the more secure your system is, the more intrusive your security becomes. You need to decide where in this balancing act your system will still be usable, and yet secure for your purposes. For instance, you could require everyone dialing into your system to use a call-back modem to call them back at their home number. This is more secure, but if someone is not at home, it makes it difficult for them to log in. You could also set up your *GNU/Linux* system with no network nor connection to the Internet, but this limits its usefulness.

If you are a medium to large-size site, you should establish a security policy stating how much security is required by your site and what auditing is in place to check it. You can find a well-known security policy example at faqs.org (<http://www.faqs.org/rfcs/rfc2196.html>). It has been recently updated, and contains a great framework for establishing a security policy for your company.

10.2.3. What Are You Trying to Protect?

Before you attempt to secure your system, you should determine what level of threat you have to protect against, what risks you should or should not take, and how vulnerable your system is as a result. You should analyze your system to know what you're protecting, why you're protecting it, what value it has, and who has responsibility for your data and other assets.

- **Risk** is the possibility that an intruder may be successful in attempting to access your computer. Can an intruder read or write files, or execute programs that could cause damage? Can they delete critical data? Can they prevent you or your company from getting important work done? Don't forget: someone gaining access to your account, or your system, can also impersonate you.

Additionally, having one insecure account on your system can result in your entire network being compromised. If you allow a single user to log in using a `.rhosts` file, or to use an insecure service, such as `tftp`, you risk an intruder getting "his foot in the door". Once the intruder has a user account on your system, or someone else's system, it can be used to gain access to another system, or another account.

- **Threat** is typically from someone with motivation to gain unauthorized access to your network or computer. You must decide who you trust to have access to your system, and what threat they could pose.

There are several types of intruders, and it is useful to keep their different characteristics in mind as you are securing your systems.

- **The Curious** – This type of intruder is basically interested in finding out what type of system and data you have.
 - **The Malicious** – This type of intruder is out to either bring down your systems, or deface your web page, or otherwise force you to spend time and money recovering from the damage he has caused.
 - **The High-Profile Intruder** – This type of intruder is trying to use your system to gain popularity and infamy. He might use your high-profile system to advertise his abilities.
 - **The Competition** – This type of intruder is interested in what data you have on your system. It might be someone who thinks you have something that could benefit him, financially or otherwise.
 - **The Borrowers** – This type of intruder is interested in setting up shop on your system and using its resources for their own purposes. He typically will run chat or IRC servers, porn archive sites, or even DNS servers.
 - **The Leapfrogger** – This type of intruder is only interested in your system to use it to get into other systems. If your system is well-connected or a gateway to a number of internal hosts, you may well see this type trying to compromise your system.
- **Vulnerability** describes how well-protected your computer is from another network, and the potential for someone to gain unauthorized access.

What's at stake if someone breaks into your system? Of course the concerns of a dynamic PPP home user will be different from those of a company connecting their machine to the Internet, or another large network.

How much time would it take to retrieve/recreate any data that was lost? An initial time investment now can save ten times more time later if you have to recreate data that was lost. Have you checked your backup strategy and verified your data lately?

10.2.4. Developing a Security Policy

Create a simple, generic policy for your system that your users can readily understand and follow. It should protect the data you're safeguarding as well as the privacy of the users. Some things to consider adding are: who has access to the system (can my friend use my account?), who's allowed to install software on the system, who owns what data, disaster recovery, and appropriate use of the system.

A generally-accepted security policy starts with the phrase

" That which is not permitted is prohibited "

This means that unless you grant access to a service for a user, that user shouldn't be using that service until you do grant access. Make sure the policies work on your regular user account. Saying, "Ah, I can't figure out this permissions problem, I'll just do it as root" can lead to security holes that are very obvious, and even ones that haven't been exploited yet.

RFC 1244 (<http://www.faqs.org/rfcs/rfc1244.html>) is a document that describes how to create your own network security policy.

rfc1281 (<http://www.faqs.org/rfcs/rfc1281.html>) is a document that shows a security policy example with detailed descriptions of each step.

Finally, you might want to look at the COAST policy archive (<ftp://coast.cs.purdue.edu/pub/doc/policy>) to see what some real-life security policies look like.

10.2.5. Means of Securing Your Site

This section will discuss various means with which you can secure the assets you have worked hard for: your local computer, your data, your users, your network, even your reputation. What would happen to your reputation if an intruder deleted some of your users' data? Or defaced your web site? Or published your company's corporate project plan for the next quarter? If you are planning a network installation, there are many factors you must take into account before adding a single computer to your network.

Even if you have a single dial up PPP account, or just a small site, this does not mean intruders won't be interested in your systems. Large, high-profile sites are not the only targets – many intruders simply want to exploit as many sites as possible, regardless of their size. Additionally, they may use a security hole in your site to gain access to other sites you're connected to.

Intruders have a lot of time on their hands, and can avoid guessing how you've obscured your system just by trying all the possibilities. There are also a number of reasons an intruder may be interested in your systems, which we will discuss later.

10.2.5.1. Host Security

Perhaps the area of security on which administrators concentrate most is host-based security. This typically involves making sure your own system is secure, and hoping everyone else on your network does the same. Choosing good passwords, securing your host's local network services, keeping good accounting records, and upgrading programs with known security exploits are among the things the local security administrator is responsible for doing. Although this is absolutely necessary, it can become a daunting task once your network becomes larger than a few computers.

10.2.5.2. Local Network Security

Network security is as necessary as local host security. With hundreds, thousands, or more computers on the same network, you can't rely on each one of those systems being secure. Ensuring that only authorized users can use your network, building firewalls, using strong encryption, and ensuring there are no "rogue" (that is, unsecured) computers on your network are all part of the network security administrator's duties.

This document will discuss some of the techniques used to secure your site, and hopefully show you some of the ways to prevent an intruder from gaining access to what you are trying to protect.

10.2.5.3. Security Through Obscurity

One type of security that must be discussed is "security through obscurity". This means, for example, moving a service that has known security vulnerabilities to a non-standard port in hopes that attackers won't notice it's there and thus won't exploit it. Rest assured that they can determine that it's there and will exploit it. Security through obscurity is no security at all. Simply because you may have a small site, or a relatively low profile, does not mean an intruder won't be interested in what you have. We'll discuss what you're protecting in the next sections.

10.2.6. Organization of this Chapter

This chapter has been divided into a number of sections. They cover several broad security issues. The first, *Physical Security*, page 84, covers how you need to protect your physical machine from tampering. The second, *Local Security*, page 88, describes how to protect your system from tampering by local users. The third, *Files and File-System Security*, page 90, shows you how to setup your file systems and permissions on your files. The next, *Password Security and Encryption*, page 94, discusses how to use encryption to better secure your machine and network. *Kernel Security*, page 100 discusses what kernel options you should set or be aware of for a more secure system. *Network Security*, page 103, describes how to better secure your *GNU/Linux* system from network attacks. *Security Preparation (before you go on-line)*, page 109, discusses how to prepare your machine(s) before bringing them on-line. Next, *What To Do During and After a Breaking*, page 110, discusses what to do when you detect a system compromise in progress or detect one that has recently happened. In *Security Sources*, page 112, some primary security resources are enumerated. The Q and A section *Frequently Asked Questions*, page 114, answers some frequently-asked questions, and finally a conclusion in *Conclusion*, page 115.

The two main points to realize when reading this chapter are:

- Be aware of your system. Check system logs such as `/var/log/messages` and keep an eye on your system, and
- Keep your system up-to-date by making sure you have installed the current versions of software and have upgraded per security alerts. Just doing this will help make your system markedly more secure.

10.3. Physical Security

The first layer of security you need to take into account is the physical security of your computer systems. Who has direct physical access to your computer? Should they? Can you protect your computer from their tampering? Should you?

How much physical security you need on your system is very dependent on your situation, and/or budget.

If you are a home user, you probably don't need a lot (although you might need to protect your computer from tampering by children or annoying relatives). If you are in a lab, you need considerably more, but users will still need to be able to get work done on the computers. Many of the following sections will help out. If you are in an office, you may or may not need to secure your computer off-hours or while you are away. At some companies, leaving your console unsecured is a termination offense.

Obvious physical security methods such as locks on doors, cables, locked cabinets, and video surveillance are all good ideas, but beyond the scope of this chapter :-)

10.3.1. Computer Locks

Many modern *PC* cases include a “locking” feature. Usually this will be a socket on the front of the case that allows you to turn an included key to a locked or unlocked position. Case locks can help prevent someone from stealing your PC, or opening up the case and directly manipulating/stealing your hardware. They can also sometimes prevent someone from rebooting your computer from their own floppy or other hardware.

These case locks do different things according to the support in the motherboard and how the case is constructed. On many *PC*'s, they make it so you have to break the case to get the case open. On some others, they will not let you plug in new keyboards or mice. Check your motherboard or case instructions for more information. This can sometimes be a very useful feature, even though the locks are usually very low quality and can easily be defeated by attackers with locksmithing.

Some computers (most notably SPARCs and Macs) have a dangle on the back: if you put a cable through attackers would have to cut the cable or break the case to get into it. Just putting a padlock or combo lock through these can be a good deterrent to someone stealing your computer.

10.3.2. BIOS Security

The *BIOS* is the lowest level of software that configures or manipulates your x86-based hardware. *grub* and other *GNU/Linux* boot methods access the *BIOS* to determine how to boot up your *GNU/Linux* computer. Other hardware that *GNU/Linux* runs on has similar software (Open Firmware on Macs and new Suns, Sun boot PROM, etc...). You can use your *BIOS* to prevent attackers from rebooting your computer and manipulating your *GNU/Linux* system.

Many *PC BIOS*s let you set a boot password. This doesn't provide all that much security (the *BIOS* can be reset, or removed if someone can get into the case), but might be a good deterrent (i.e. it will take time and leave traces of tampering). Similarly, on *S/Linux* (*GNU/Linux* for *SPARC*(tm) processor computers), your EEPROM can be set to require a boot-up password. This might slow attackers down.

Le mot de passe par défaut s'avère un autre risque lorsque vous faites confiance au mot de passe du *BIOS* pour sécuriser votre système. La plupart des fabricants de *BIOS* ne s'attendent pas à ce que les utilisateurs ouvrent leur ordinateur et déconnectent les batteries s'ils oublient leur mot de passe et ont muni leurs *BIOS* de mots de passe par défaut qui fonctionnent nonobstant le mot de passe que vous avez choisi. Voici certains des mots de passe les plus communs :

```
j262 AWARD_SW AWARD_PW lkwpeter Biostar AMI Award
  bios BIOS setup cmos AMI!SW1 AMI?SW1 password hewittrand shift + s y
  x z
```

J'ai testé un *BIOS* Award et AWARD_PW a fonctionné. Ces mots de passe sont assez faciles à obtenir sur les sites Web des fabricants et sur astalavista (<http://astalavista.box.sk>) et en tant que tel, un mot de passe de *BIOS* ne peut pas être considéré comme une protection adéquate contre les attaquants informés.

Many *x86 BIOS*s also allow you to specify various other good security settings. Check your *BIOS* manual or look at it the next time you boot up. For example, some *BIOS*s disallow booting from floppy drives and some require passwords to access some *BIOS* features.



If you have a server computer, and you set up a boot password, your computer will not boot up unattended. Keep in mind that you will need to come in and supply the password in the event of a power failure ;(

10.3.3. OpenBoot Security

The *PROM* is the lowest level of software that configures or manipulates your sparc-based hardware. *SILO* and other *GNU/Linux* boot methods access the *PROM* to determine how to boot up your *GNU/Linux* computer. Other hardware that *GNU/Linux* runs on has similar software (OpenFirmware on Macs and new Suns, *x86 BIOS*, etc...). You can use your *PROM* to prevent attackers from rebooting your computer and manipulating your *GNU/Linux* system.

OpenBoot is much more advanced than a *PC BIOS* when it comes to security (consult the “*Installation Guide*” on how to access and use *OpenBoot*).



It is important to set your password before setting the security mode, as you would be unable to set it any more. Moreover, SUN claims you need to contact your vendor's customer support service to make your computer bootable again.

This is an interaction example on how to set your boot password:

```
> password
> New password (only first 8 chars are used):
> Retype new password:
>
```

2. You can choose between three security levels setting the `security-mode` variable:

- a. Full: all commands except for `go` require the password.
- b. Command: all commands except for `boot` and `go` require the password.
- c. None: no password required (default).

This is an interaction example on how to set your security mode:

```
> setenv security-mode full
>
```



If you have a server computer, and you set up a boot password, your computer will not boot up unattended. Keep in mind that you will need to come in and supply the password in the event of a power failure ;(

10.3.4. Boot Loader Security

Keep in mind when setting all these passwords that you need to remember them :-). Also remember that these passwords will only slow the determined attacker. They won't prevent someone from booting from a floppy and mounting your root partition.

If you are using security in conjunction with a boot loader, you might as well disable booting from a floppy in your computer's *BIOS*, and password-protect the *BIOS*.

Also keep in mind that the `/etc/lilo.conf` will need to be mode "600" (readable and writing for root only), or others will be able to read your passwords!

If you are using security in conjunction with a boot loader, you might as well password-protect the *PROM*.



Once again, if you have a server computer, and you set up a boot password, your computer will not boot up unattended. Keep in mind that you will need to come in and supply the password in the event of a power failure;{(

10.3.4.1. With GRUB

The various *GNU/Linux* boot loaders also can have a boot password set. *grub* is quite flexible in that sense: your default configuration file `/boot/grub/menu.lst` may contain a line allowing the loading of a new config file with different options (this new file may contain a new password to access another third config file and so on).

So you must add a line in your `/boot/grub/menu.lst` file, something like:

```
password very_secret /boot/grub/menu2.lst
```

and of course generate a new `/boot/grub/menu2.lst` config file where you move insecure entries previously removed from `/boot/grub/menu.lst`.

>From the grub info page:

- Command: `password passwd new-config-file`
 Disable all interactive editing control (menu entry editor and command line). If the password `PASSWD` is entered, it loads the `NEW-CONFIG-FILE` as a new config file and restarts the GRUB Stage 2.

10.3.4.2. With LILO

LILO has password and restricted settings; password requires password at boot time, whereas restricted requires a boot-time password only if you specify options (such as `single`) at the *LILO* prompt.

From the `lilo.conf` man page:

```
password=password
    The per-image option 'password=...' (see below)
    applies to all images.

restricted
    The per-image option 'restricted' (see below)
    applies to all images.

password=password
    Protect the image by a password.

restricted
    A password is only required to boot the image if
    parameters are specified on the command line
    (e.g. single).
```

10.3.4.3. With SILO

The *SILO* boot loader may also have a boot password: password requires password at boot time, whereas restricted requires a boot-time password only if you specify options (such as `single`) at the *SILO* prompt.

>From the `silos.conf` man page:

```
password=password
    Protect booting by a password. The password is
    given in cleartext in the configuration file.
    Because of that the configuration file should be
    only readable by the super user and the password
    should differ if possible from other passwords on
    the system.

restricted
    A password is only required to boot the image spec-
    ified in /etc/silo.conf if parameters are specified
    on the command line or if the image is not speci-
    fied in the configuration file at all (i.e. arbi-
    trary file load).
```

10.3.5. xlock and vlock

If you wander away from your computer from time to time, it is nice to be able to “lock” your console so that no one can tamper with or look at your work. Two programs that do this are: `xlock` and `vlock`.

`xlock` is a *X* display locker. You can run `xlock` from any `xterm` on your console and it will lock the display and require your password to unlock. Most desktop environment also propose this feature in their respective menus.

`vlock` is a simple little program that allows you to lock some or all of the virtual consoles on your *GNU/Linux* box. You can lock just the one you are working in or all of them. If you just lock one, others can come in and use the console; they will just not be able to use your virtual console until you unlock it.

Of course, locking your console will prevent someone from tampering with your work, but won't prevent them from rebooting your computer or otherwise disrupting your work. It also does not prevent them from accessing your computer from another computer on the network and causing problems.

More importantly, it does not prevent someone from switching out of the *X Window System* entirely, and going to a normal virtual console login prompt, or to the VC that X11 was started from, and suspending it, thus obtaining your privileges. For this reason, you might consider only using it while under control of *KDM* (or other).

10.3.6. Security of local devices

If you have a webcam or a microphone attached to your system, you should consider if there is some danger of an attacker gaining access to those devices. When not in use, unplugging or removing such devices might be an option. Otherwise you should carefully read and look at any software with provides access to such devices.

10.3.7. Detecting Physical Security Compromises

The first thing to always note is when your computer was rebooted. Since *GNU/Linux* is a robust and stable OS, the only times your computer should reboot is when **you** take it down for OS upgrades, hardware swapping, or the like. If your computer has rebooted without you doing it, that may be a sign that an intruder has compromised it. Many of the ways that your computer can be compromised require the intruder to reboot or power off your computer.

Check for signs of tampering on the case and computer area. Although many intruders clean traces of their presence out of logs, it's a good idea to check through them all and note any discrepancy.

It is also a good idea to store log data at a secure location, such as a dedicated log server within your well-protected network. Once a computer has been compromised, log data becomes of little use as it most likely has also been modified by the intruder.

The *syslog* daemon can be configured to automatically send log data to a central *syslog* server, but this is typically sent in unencrypted, allowing an intruder to view data as it is being transferred. This may reveal information about your network that is not intended to be public. There are *syslog* daemons available that encrypt the data as it is being sent.

Also be aware that faking *syslog* messages is easy – with an exploit program having been published. *syslog* even accepts net log entries claiming to come from the local host without indicating their true origin.

Some things to check for in your logs:

- Short or incomplete logs.
- Logs containing strange timestamps.
- Logs with incorrect permissions or ownership.
- Records of reboots or restarting of services.
- Missing logs.
- `su` entries or logins from strange places.

We will discuss system log data *Keep Track of Your System Accounting Data*, page 110 in this chapter.

10.4. Local Security

The next thing to take a look at is the security in your system against attacks from local users. Did we just say **local** users? Yes!

Getting access to a local user account is one of the first things that system intruders attempt while on their way to exploiting the root account. With lax local security, they can then “upgrade” their normal user access to root access using a variety of bugs and poorly setup local services. If you make sure your local security is tight, then the intruder will have another hurdle to jump.

Local users can also cause a lot of havoc with your system even (especially) if they really are who they say they are. Providing accounts to people you don’t know or for whom you have no contact information for is a very bad idea.

10.4.1. Creating New Accounts

You should make sure you provide user accounts with only the minimal requirements for the task they need to do. If you provide your son (age 10) with an account, you might want him to only have access to a word processor or drawing program, but be unable to delete data that is not his.

Several good rules of thumb when allowing other people legitimate access to your *GNU/Linux* computer:

- Give them the minimal amount of privileges they need.
- Be aware when/where they log in from, or should be logging in from.
- Make sure you remove inactive accounts, which you can determine by using the `last` command and/or checking log files for any activity by the user.
- The use of the same `userid` on all computers and networks is advisable to ease account maintenance, and permits easier analysis of log data.
- The creation of group `user-ids` should be absolutely prohibited. User accounts also provide accountability, and this is not possible with group accounts.

Many local user accounts that are utilized in security compromises have not been used in months or years. Since no one is using them, they provide the ideal attack vehicle.

10.4.2. Root Security

The most sought-after account on your computer is the root (superuser) account. It has authority over the entire computer, which may also include authority over other computers on the network. Remember that you should only use the root account for very short, specific tasks, and should mostly run as a normal user. Even small mistakes made while logged in as the root user can cause problems. The less time you are on with root privileges, the safer you will be.

Several tricks to avoid messing up your own box as root:

- When doing some complex command, try running it first in a non-destructive way... especially commands that use globbing: e.g., if you want to do `rm -f foo*.bak`, first do `ls foo*.bak` and make sure you are going to delete the files you think you are. Using `echo` in place of destructive commands also sometimes works.
- Only become root to do single specific tasks. If you find yourself trying to figure out how to do something, go back to a normal user *shell* until you are **sure** what needs to be done by root.
- The command path for the root user is very important. The command path (that is, the `PATH` environment variable) specifies the directories in which the *shell* searches for programs. Try to limit the command path for the root user as much as possible, and **never** include `.` (which means “the current directory”) in your `PATH`. Additionally, never have writable directories in your search path, as this can allow attackers to modify or place new binaries in your search path, allowing them to run as root the next time you run that command.
- Never use the `rlogin/rsh/rexec` suite of tools (called the “r-utilities”) as root. They are subject to many sorts of attacks, and are downright dangerous when run as root. Never create a `.rhosts` file for root.
- The `/etc/securetty` file contains a list of terminals that root can login from. By default, this is set to only the local virtual consoles (ttys). Be very wary of adding anything else to this file. You should be able to log

in remotely as your regular user account and then `su` if you need to (hopefully over `ssh` or other encrypted channel), so there is no need to be able to login directly as `root`.

- Always be slow and deliberate running as `root`. Your actions could affect a lot of things. Think before you type!

If you absolutely, positively need to allow someone (hopefully very trusted) to have `root` access to your computer, there are a few tools that can help. `sudo` allows users to use their password to access a limited set of commands as `root`. This would allow you to, for instance, let a user be able to eject and mount removable media on your *GNU/Linux* box, but have no other `root` privileges. `sudo` also keeps a log of all successful and unsuccessful `sudo` attempts, allowing you to track down who used what command to do what. For this reason, `sudo` works well even in places where a number of people have `root` access, because it helps you to keep track of changes made.

Although `sudo` can be used to give specific users special privileges for particular tasks, it does have several shortcomings. It should be used only for a limited set of tasks, like restarting a server, or adding new users. Any program that offers a *shell* escape will give `root` access to a user invoking it via `sudo`. This includes most editors, for example. Also, a program as innocuous as `/bin/cat` can be used to overwrite files, which could allow `root` to be exploited. Consider `sudo` as a means for accountability, and don't expect it to replace the `root` user and still be secure.

10.5. Files and File-System Security

A few minutes of preparation and planning ahead before putting your systems on-line can help protect them and the data stored in them.

- There should never be a reason for users' home directories to allow SUID/SGID programs to be run from there. Use the `nosuid` option in `/etc/fstab` for partitions that are writable by others than `root`. You may also wish to use `nodev` and `noexec` on users' home partitions, as well as `/var`, thus prohibiting execution of programs, and creation of character or block devices, which should never be necessary anyway.
- If you are exporting file systems using NFS, be sure to configure `/etc/exports` with the most restrictive access possible. This means not using wild cards, not allowing `root` write access, and exporting read-only wherever possible.
- Configure your users' file-creation `umask` to be as restrictive as possible. See *umask Settings*, page 91.
- If you are mounting file systems using a network file system such as NFS, be sure to configure `/etc/fstab` with suitable restrictions. Typically, using `nodev`, `nosuid`, and perhaps `noexec`, are desirable.
- Set file system limits instead of allowing `unlimited` as default. You can control the per-user limits using the resource-limits PAM module and `/etc/pam.d/limits.conf`. For example, limits for group users might look like this:

```
@users    hard  core   0
@users    hard  nproc  50
@users    hard  rss    5000
```

This says to prohibit the creation of core files, restrict the number of processes to 50, and restrict memory usage per user to 5MB.

You can also use the `/etc/login.defs` configuration file to set the same limits.

- The `/var/log/wtmp` and `/var/run/utmp` files contain the login records for all users on your system. Their integrity must be maintained because they can be used to determine when and from where a user (or potential intruder) has entered your system. These files should also have 644 permissions, without affecting normal system operation.
- The immutable bit can be used to prevent accidentally deleting or overwriting a file that must be protected. It also prevents someone from creating a hard link to the file. See the `chattr(1)` man page for information on the immutable bit.
- *suid* and SGID files on your system are a potential security risk, and should be monitored closely. Because these programs grant special privileges to the user who is executing them, it is necessary to ensure that

insecure programs are not installed. A favorite trick of crackers is to exploit SUID-root programs, then leave a SUID program as a back door to get in the next time, even if the original hole is plugged.

Find all SUID/SGID programs on your system, and keep track of what they are, so you are aware of any changes which could indicate a potential intruder. Use the following command to find all SUID/SGID programs on your system:

```
root# find / -type f \( -perm -04000 -o -perm -02000 \)
```

You can remove the *suid* or SGID permissions on a suspicious program with *chmod*, then restore them back if you absolutely feel it is necessary.

- World-writable files, particularly system files, can be a security hole if a cracker gains access to your system and modifies them. Additionally, world-writable directories are dangerous, since they allow a cracker to add or delete files as he wishes. To locate all world-writable files on your system, use the following command:

```
root# find / -perm -2 ! -type l -ls
```

and be sure you know why those files are writable. In the normal course of operation, several files will be world-writable, including some from */dev*, and symbolic links, thus the *! -type l* which excludes these from the previous *find* command.

- Unowned files may also be an indication that an intruder has accessed your system. You can locate files on your system that have no owner or belong to no group with the command:

```
root# find / -nouser -o -nogroup -print
```

- Finding *.rhosts* files should be a part of your regular system administration duties, as they should not be permitted on your system. Remember, a cracker only needs one insecure account to potentially gain access to your entire network. You can locate all *.rhosts* files on your system with the following command:

```
root# find /home -name .rhosts -print
```

- Finally, before changing permissions on any system files, make sure you understand what you are doing. Never change permissions on a file because it seems like the easy way to get things working. Always determine why the file has that permission before changing it.

10.5.1. umask Settings

The *umask* command can be used to determine the default file-creation mode on your system. It is the octal complement of the desired file mode. If files are created without any regard to their permission settings, the user could inadvertently give read or write permission to someone that should not have it. Typical *umask* settings include 022, 027, and 077 (which is the most restrictive). Normally, the *umask* is set in */etc/profile*, so it applies to all users on the system. The file creation mask can be calculated by subtracting the desired value from 777. In other words, a *umask* of 777 would cause newly-created files to contain no read, write or execute permission for anyone. A mask of 666 would cause newly-created files to have a mask of 111. For example, you may have a line that looks like this:

```
# Set the user's default umask
umask 033
```

Be sure to make *root*'s *umask* 077, which will disable read, write, and execute permission for other users, unless explicitly changed using *chmod*. In this case, newly-created directories would have 744 permissions, obtained by subtracting 033 from 777. Newly-created files using the 033 *umask* would have permissions of 644.



In Mandrake Linux, it is only necessary to use 002 for a *umask*. This is due to the fact that the default configuration is one user per group.

10.5.2. File Permissions

It's important to ensure that your system files are not open for casual editing by users and groups who shouldn't be doing such system maintenance.

UNIX separates access control on files and directories according to three characteristics: owner, group, and other. There is always exactly one owner, any number of members of the group, and everyone else.

A quick explanation of *UNIX* permissions:

Ownership - Which user(s) and group(s) retain(s) control of the permission settings of the node and parent of the node

Permissions - Bits capable of being set or reset to allow certain types of access to it. Permissions for directories may have a different meaning than the same set of permissions on files.

Read:

- To be able to view contents of a file
- To be able to read a directory

Write:

- To be able to add to or change a file
- To be able to delete or move files in a directory

Execute:

- To be able to run a binary program or *shell* script
- To be able to search in a directory, combined with read permission

Save Text Attribute: (For directories)

The "sticky bit" also has a different meaning when applied to directories than when applied to files. If the sticky bit is set on a directory, then a user may only delete files that he owns or for which he has explicit write permission granted, even when he has write access to the directory. This is designed for directories like */tmp*, which are world-writable, but where it may not be desirable to allow any user to delete files at will. The sticky bit is seen as a *t* in a long directory listing.

suid Attribute: (For Files)

This describes set-user-id permissions on the file. When the set user *ID* access mode is set in the owner permissions, and the file is executable, processes which run it are granted access to system resources based on user who owns the file, as opposed to the user who created the process. This is the cause of many "buffer overflow" exploits.

SGID Attribute: (For Files)

If set in the group permissions, this bit controls the "*set group id*" status of a file. This behaves the same way as *suid*, except the group is affected instead. The file must be executable for this to have any effect.

SGID Attribute: (For directories)

If you set the SGID bit on a directory (with `chmod g+s directory`), files created in that directory will have their group set to the directory's group.

You - The owner of the file

Group - The group you belong to

Everyone - Anyone on the system that is not the owner or a member of the group

File Example:

```
-rw-r--r-- 1 queen users      114 Aug 28 1997 .zlogin
1st bit - directory?          (no)
2nd bit - read by owner?      (yes, by queen)
3rd bit - write by owner?     (yes, by queen)
4th bit - execute by owner?   (no)
5th bit - read by group?      (yes, by users)
6th bit - write by group?     (no)
7th bit - execute by group?   (no)
8th bit - read by everyone?   (yes, by everyone)
9th bit - write by everyone?  (no)
10th bit - execute by everyone? (no)
```

The following lines are examples of the minimum sets of permissions that are required to perform the access described. You may want to give more permission than what's listed here, but this should describe what these minimum permissions on files do:

```
-r----- Allow read access to the file by owner
--w----- Allows the owner to modify or delete the file
           (Note that anyone with write permission to the directory
           the file is in can overwrite it and thus delete it)
---x----- The owner can execute this program, but not shell scripts,
           which still need read permission
--s----- Will execute with effective User ID = to owner
-----s- Will execute with effective Group ID = to group
-rw-----T No update of "last modified time". Usually used for swap
           files
---t----- No effect. (formerly sticky bit)
```

Directory Example:

```
drwxr-xr-x 3 queen users      512 Sep 19 13:47 .public_html/
1st bit - directory?          (yes, it contains many files)
2nd bit - read by owner?      (yes, by queen)
3rd bit - write by owner?     (yes, by queen)
4th bit - execute by owner?   (yes, by queen)
5th bit - read by group?      (yes, by users)
6th bit - write by group?     (no)
7th bit - execute by group?   (yes, by users)
8th bit - read by everyone?   (yes, by everyone)
9th bit - write by everyone?  (no)
10th bit - execute by everyone? (yes, by everyone)
```

The following lines are examples of the minimum sets of permissions that are required to perform the access described. You may want to give more permission than what's listed, but this should describe what these minimum permissions on directories do:

```
dr----- The contents can be listed, but file attributes can't be read
d--x----- The directory can be entered, and used in full execution
           paths
dr-x----- File attributes can be read by owner
d-wx----- Files can be created/deleted, even if the directory
           isn't the current one
d-----x-t Prevents files from deletion by others with write
           access. Used on /tmp
d---s---s-- No effect
```

System configuration files (usually in `/etc`) are usually mode 640 (`-rw-r----`), and owned by root. Depending on your site's security requirements, you might adjust this. Never leave any system files writable by a group or everyone. Some configuration files, including `/etc/shadow`, should only be readable by root, and directories in `/etc` should at least not be accessible by others.

suid shell Scripts

suid shell scripts are a serious security risk, and for this reason the kernel will not honor them. Regardless of how secure you think the *shell* script is, it can be exploited to give the cracker a root *shell*.

10.5.3. Integrity Checking

Another very good way to detect local (and also network) attacks on your system is to run an integrity checker like *Tripwire*, *Aide* or *Osiris*. These integrity checkers run number of checksums on all your important binaries and config files and compares them against a database of former, known-good values as a reference. Thus, any changes in the files will be flagged.

It's a good idea to install these sorts of programs onto a floppy, and then physically set the write protect on the floppy. This way intruders can't tamper with the integrity checker itself or change the database. Once you have something like this setup, it's a good idea to run it as part of your normal security administration duties to see if anything has changed.

You can even add a crontab entry to run the checker from your floppy every night and mail you the results in the morning. Something like:

```
# set mailto
MAILTO=queen
# run Tripwire
15 05 * * * root /usr/local/adm/tcheck/tripwire
```

will mail you a report each morning at 5:15am.

Integrity checkers can be a godsend to detecting intruders before you would otherwise notice them. Since a lot of files change on the average system, you have to be careful what is cracker activity and what is your own doing.

You can find the freely available unsupported version of Tripwire at TripWire (<http://www.tripwire.org>) free of charge. Manuals and support can be purchased.

Aide can be found at <http://www.cs.tut.fi/~rammer/aide.html> (<http://www.cs.tut.fi/~rammer/aide.html>).

Osiris can be found at <http://www.shmoo.com/osiris/> (<http://www.shmoo.com/osiris/>).

10.5.4. Trojan Horses

"Trojan Horses" are named after the fabled ploy in Homer's "Iliad". The idea is that a cracker distributes a program or binary that sounds great, and encourages other people to download it and run it as root. Then the program can compromise their system while they are not paying attention. While they think the binary they just pulled down does one thing (and it might very well), it also compromises their security.

You should take care of what programs you install on your computer. **MandrakeSoft** provides MD5 checksums and PGP signatures on its RPM files so you can verify you are installing the real thing. You should never run any unfamiliar binary, for which you don't have the source, as root! Few attackers are willing to release source code to public scrutiny.

Although it can be complex, make sure you are getting the source for a program from its real distribution site. If the program is going to run as root, make sure either you or someone you trust has looked over the source and verified it.

10.6. Password Security and Encryption



Most of encryption programs described in this chapter are available in your **Mandrake Linux** distribution.

One of the most important security features used today are passwords. It is important for both you and all your users to have secure, unguessable passwords. Your **Mandrake Linux** distributions include `passwd` program that do not allow you to set a easily guessable password. Make sure your `passwd` program is up to date.

In-depth discussion of encryption is beyond the scope of this chapter, but an introduction is in order. Encryption is very useful, possibly even necessary in this day and age. There are all sorts of methods of encrypting data, each with its own set of characteristics.

Most *UNIX* systems (and *GNU/Linux* is no exception) primarily use a one-way encryption algorithm, called DES (Data Encryption Standard) to encrypt your passwords. This encrypted password is then stored in `/etc/shadow`. When you attempt to login, the password you type in is encrypted again and compared with the entry in the file that stores your passwords. If they match, it must be the same password, and you are allowed access. Although DES is a two-way encryption algorithm (you can code and then decode a message, given the right keys), the variant that most Unixes use is one-way. This means that it should not be possible to reverse the encryption to get the password from the contents of `/etc/shadow`.

Brute force attacks, such as “Crack” or “John the Ripper” (see Section “Crack” and “John the Ripper”, page 99) can often guess passwords unless your password is sufficiently random. PAM modules (see below) allow you to use a different encryption routine with your passwords (MD5 or the like). You can use Crack to your advantage, as well. Consider periodically running Crack against your own password database, to find insecure passwords. Then contact the offending user, and instruct him to change his password.

You can go to CERN (http://consult.cern.ch/writeup/security/security_3.html) for information on how to choose a good password.

10.6.1. PGP and Public-Key Cryptography

Public-key cryptography, such as that used for PGP, uses one key for encryption, and one key for decryption. Traditional cryptography, however, uses the same key for encryption and decryption; this key must be known to both parties, and thus somehow transferred from one to the other securely.

To alleviate the need to securely transmit the encryption key, public-key encryption uses two separate keys: a public key and a private key. Each person’s public key is available by anyone to do the encryption, while at the same time each person keeps his or her private key to decrypt messages encrypted with the correct public key.

There are advantages to both public key and private key cryptography, and you can read about those differences in the RSA Cryptography FAQ (<http://www.rsasecurity.com/rsalabs/faq/>), listed at the end of this section.

PGP (Pretty Good Privacy) is well-supported on *GNU/Linux*. Versions 2.6.2 and 5.0 are known to work well. For a good primer on *PGP* and how to use it, take a look at the different *PGP* FAQs at [faqs.org/faqs/pgp-faq/](http://www.faqs.org/faqs/pgp-faq/)

Be sure to use the version that is applicable to your country. Due to export restrictions by the US Government, strong-encryption is prohibited from being transferred in electronic form outside the country.

US export controls are now managed by EAR (Export Administration Regulations). They are no longer governed by ITAR.

There is also a step-by-step guide for configuring *PGP* on *GNU/Linux* available at LinuxFocus (<http://mercury.chem.pitt.edu/~angel/LinuxFocus/English/November1997/article7.html>). It was written for the international version of *PGP*, but is easily adaptable to the United States version. You may also need a patch for some of the latest versions of *GNU/Linux*; the patch is available at <ftp://metalab.unc.edu/pub/Linux/apps/crypto> (<ftp://metalab.unc.edu/pub/Linux/apps/crypto>).

There is a project maintaining a free re-implementation of *PGP* with open source. GnuPG is a complete and free replacement for *PGP*. Because it does not use IDEA or RSA it can be used without any restrictions. *GnuPG* is in compliance with OpenPGP (<http://www.faqs.org/rfcs/rfc2440.html>). See the GNU Privacy Guard web page for more information: <http://www.gnupg.org/> (<http://www.gnupg.org/>).

More information on cryptography can be found in the RSA cryptography FAQ, available at <http://www.rsasecurity.com/rsalabs/faq/> (<http://www.rsa.com/rsalabs/newfaq/>). Here you will find information on such terms as “Diffie-Hellman”, “public-key cryptography”, “digital certificates”, etc.

10.6.2. SSL, S-HTTP and S/MIME

Often users ask about the differences between the various security and encryption protocols, and how to use them. While this isn’t an encryption document, it is a good idea to explain briefly what each protocol is, and where to find more information.

- **SSL:** - SSL, or Secure Sockets Layer, is an encryption method developed by Netscape to provide security over the Internet. It supports several different encryption protocols, and provides client and server authentication. SSL operates at the transport layer, creates a secure encrypted channel of data, and thus can seamlessly encrypt data of many types. This is most commonly seen when going to a secure site to view a secure online document with *Communicator*, and serves as the basis for secure communications with *Communicator*, as well as many other Netscape Communications data encryption. More information can be found at Openssl.org (<http://www.openssl.org>). Information on Netscape’s other security implementations, and a good starting point for these protocols is available at netscape (<http://home.netscape.com/info/security-doc.html>). It’s also worth noting that the SSL protocol can be used to pass many other common protocols, “wrapping” them for security. See quiltaholic (<http://www.quiltaholic.com/rickk/sslwrap/>)
- **S-HTTP:** - S-HTTP is another protocol that provides security services across the Internet. It was designed to provide confidentiality, authentication, integrity, and non-repudiability [cannot be mistaken for someone else] while supporting multiple key-management mechanisms and cryptographic algorithms via option negotiation between the parties involved in each transaction. S-HTTP is limited to the specific software that is implementing it, and encrypts each message individually. [From RSA Cryptography FAQ, page 138]
- **S/MIME:** - S/MIME, or Secure Multipurpose Internet Mail Extension, is an encryption standard used to encrypt electronic mail and other types of messages on the Internet. It is an open standard developed by RSA, so it is likely we will see it on *GNU/Linux* one day soon. More information on S/MIME can be found at netscape (<http://developer.netscape.com/tech/security/email/smime.html>).

10.6.3. IPSEC Implementations

Along with CIPE, and other forms of data encryption, there are also several other implementations of IPSEC for *GNU/Linux*. IPSEC is an effort by the IETF to create cryptographically-secure communications at the IP network level, and to provide authentication, integrity, access control, and confidentiality. Information on IPSEC and Internet draft can be found at <http://www.ietf.org/html.charters/ipsec-charter.html> (<http://www.ietf.org/html.charters/ipsec-charter.html>). You can also find links to other protocols involving key management, and an IPSEC mailing list and archives.

The x-kernel *GNU/Linux* implementation, which was (project is now closed) being developed at the University of Arizona, uses an object-based framework for implementing network protocols called x-kernel, and can be found at <http://www.cs.arizona.edu/xkernel/hpcc-blue/linux.html> (<http://www.cs.arizona.edu/xkernel/hpcc-blue/linux.html>). Most simply, the x-kernel is a method of passing messages at the kernel level, which makes for an easier implementation.

Another freely-available IPSEC implementation is the *GNU/Linux* FreeS/WAN IPSEC. Their web page states, “These services allow you to build secure tunnels through untrusted networks. Everything passing through the untrusted net is encrypted by the IPSEC gateway computer and decrypted by the gateway at the other end. The result is Virtual Private Network or VPN. This is a network which is effectively private even though it includes computers at several different sites connected by the insecure Internet.”

It’s available for download from <http://www.xs4all.nl/~freeswan/> (<http://www.xs4all.nl/~freeswan/>).

As with other forms of cryptography, it is not distributed with the kernel by default due to export restrictions.

10.6.4. ssh (Secure SHell) and stelnet

ssh and stelnet are suites of programs that allow you to login to remote systems and have a encrypted connection.

openssh is a suite of programs used as a secure replacement for rlogin, rsh and rcp. It uses public-key cryptography to encrypt communications between two hosts, as well as to authenticate users. It can be used to securely login to a remote host or copy data between hosts, while preventing man-in-the-middle attacks (session hijacking) and DNS spoofing. It will perform data compression on your connections, and secure X11 communications between hosts.

There are several ssh implementations now. The original commercial implementation by Data Fellows can be found at The ssh home page available at <http://www.datafellows.com> (<http://www.datafellows.com>).

The excellent *Openssh* implementation is based on a early version of the *datafellows ssh* and has been totally reworked to not include any patented or proprietary pieces. It is free and under a BSD license. It can be found at: <http://www.openssh.com> (<http://www.openssh.com>).

There is also a open source project to re-implement ssh from the ground up called "psst...". For more information see: <http://www.net.lut.ac.uk/psst/> (<http://www.net.lut.ac.uk/psst/>).

You can also use ssh from your *Windows* workstation to your *GNU/Linux* ssh server. There are several freely available *Windows* client implementations, including PuTTY (<http://www.chiark.greenend.org.uk/~sgtatham/putty/>) and the one at therapy ssh (<http://guardian.htu.tuwien.ac.at/therapy/ssh/>) as well as a commercial implementation from DataFellows, at datafellows (<http://www.datafellows.com>).

SSLeay (outdated, see OpenSSL below) is a free implementation of Netscape's Secure Sockets Layer, developed by Eric Young. It includes several applications, such as "Secure telnet", a module for *Apache*, several databases, as well as several algorithms including DES, IDEA and "Blowfish".

Using this library, a secure telnet replacement has been created that does encryption over a telnet connection. Unlike SSH, stelnet uses SSL, the Secure Sockets Layer protocol developed by Netscape. You can find Secure telnet and Secure FTP by starting with the SSlEay FAQ, available at <http://www.psy.uq.oz.au/~ftp/Crypto/> (<http://www.psy.uq.oz.au/~ftp/Crypto/>).



The *OpenSSL* Project is based on *SSLeay* and is meant to develop a robust, commercial-grade, full-featured, and Open Source toolkit implementing the Secure Sockets Layer (SSL v2/v3) and Transport Layer Security (TLS v1) protocols as well as a full-strength general purpose cryptography library. For more information about this project, consult the OpenSSL home page (www.openssl.org). There is a large list of applications based on *OpenSSL* at OpenSSL related applications (<http://www.openssl.org/related/apps.html>).

SRP is another secure telnet/ftp implementation. From their web page:

"The SRP project is developing secure Internet software for free worldwide use. Starting with a fully-secure *Telnet* and *FTP* distribution, we hope to supplant weak networked authentication systems with strong replacements that do not sacrifice user-friendliness for security. Security should be the default, not an option!"

For more information, go to stanford.edu (<http://www-cs-students.stanford.edu/~tjw/srp/>).

10.6.5. PAM - Pluggable Authentication Modules

Your version of **Mandrake Linux** distribution ships with a unified authentication scheme called PAM. PAM allows you to change your authentication methods and requirements on the fly, and encapsulate all local authentication methods without recompiling any of your binaries. Configuration of PAM is beyond the scope of this chapter, but be sure to take a look at the PAM web site for more information. <http://www.kernel.org/pub/linux/libs/pam/index.html> (<http://www.kernel.org/pub/linux/libs/pam/index.html>).

Just a few of the things you can do with PAM:

- Use encryption other than DES for your passwords. (Making them harder to brute-force decode)
- Set resource limits on all your users so they can't perform denial-of-service attacks (number of processes, amount of memory, etc.)

- Enable shadow passwords (see below) on the fly
- allow specific users to login only at specific times from specific places

Within a few hours of installing and configuring your system, you can prevent many attacks before they even occur. For example, use PAM to disable the system-wide usage of `.rhosts` files in user's home directories by adding these lines to `/etc/pam.d/rlogin`:

```
#
# Disable rsh/rlogin/rexec for users
#
login auth required pam_rhosts_auth.so no_rhosts
```

10.6.6. Cryptographic IP Encapsulation (CIPE)

The primary goal of this software is to provide a facility for secure (against eavesdropping, including traffic analysis, and faked message injection) subnetwork interconnection across an insecure packet network such as the Internet.

CIPE encrypts the data at the network level. Packets traveling between hosts on the network are encrypted. The encryption engine is placed near the driver which sends and receives packets.

This is unlike SSH, which encrypts the data by connection, at the socket level. A logical connection between programs running on different hosts is encrypted.

CIPE can be used in tunneling, in order to create a Virtual Private Network. Low-level encryption has the advantage that it can be made to work transparently between the two networks connected in the VPN, without any change to application software.

Summarized from the CIPE documentation:

"The IPSEC standards define a set of protocols which can be used (among other things) to build encrypted VPNs. However, IPSEC is a rather heavyweight and complicated protocol set with a lot of options, implementations of the full protocol set are still rarely used and some issues (such as key management) are still not fully resolved. CIPE uses a simpler approach, in which many things which can be parameterized (such as the choice of the actual encryption algorithm used) are an install-time fixed choice. This limits flexibility, but allows for a simple (and therefore efficient, easy to debug...) implementation."

Further information can be found at <http://www.inka.de/~bigred/devel/cipe.html> (<http://www.inka.de/~bigred/devel/cipe.html>)

As with other forms of cryptography, it is not distributed with the kernel by default due to export restrictions.

10.6.7. Kerberos

Kerberos is an authentication system developed by the Athena Project at MIT. When a user logs in, *Kerberos* authenticates that user (using a password), and provides the user with a way to prove her identity to other servers and hosts scattered around the network.

This authentication is then used by programs such as `rlogin` to allow the user to login to other hosts without a password (in place of the `.rhosts` file). This authentication method can also be used by the mail system in order to guarantee that mail is delivered to the correct person, as well as to guarantee that the sender is who he claims to be.

Kerberos and the other programs that come with it, prevent users from "spoofing" the system into believing they are someone else. Unfortunately, installing *Kerberos* is very intrusive, requiring the modification or replacement of numerous standard programs.

You can find more information about Kerberos by looking at the Kerberos FAQ (<http://www.faqs.org/faqs/kerberos-faq/general/>), and the code can be found at <http://web.mit.edu/kerberos/www/> (<http://web.mit.edu/kerberos/www/>).

[From: Stein, Jennifer G., Clifford Neuman, and Jeffrey L. Schiller. "Kerberos: An Authentication Service for Open Network Systems." USENIX Conference Proceedings, Dallas, Texas, Winter 1998.]

Kerberos should not be your first step in improving security of your host. It is quite involved, and not as widely used as, say, SSH.

10.6.8. “Crack” and “John the Ripper”

If for some reason your `passwd` program is not enforcing hard-to-guess passwords, you might want to run a password-cracking program and make sure your users’ passwords are secure.

Password cracking programs work on a simple idea: they try every word in the dictionary, and then variations on those words, encrypting each one and checking it against your encrypted password. If they get a match they know what your password is.

There are a number of programs out there...the two most notable of which are *Crack* and *John the Ripper* (See OpenWall (<http://www.openwall.com/john/>)). They will take up a lot of your CPU time, but you should be able to tell if an attacker could get in using them by running them first yourself and notifying users with weak passwords. Note that an attacker would have to use some other hole first in order to read your `/etc/shadow` file, but such holes are more common than you might think.

Because security is only as strong as the most insecure host, it is worth mentioning that if you have any *Windows* computers on your network, you should check out *L0phtCrack*, a *Crack* implementation for *Windows*. It’s available from <http://www.atstake.com> (<http://www.atstake.com/research/lc3/>)

10.6.9. CFS - Cryptographic File System and TCFS - Transparent Cryptographic File System

CFS is a way of encrypting an entire directory tree and allowing users to store encrypted files on them. It uses an NFS server running on the local computer. More information and source code are available at [att](ftp://ftp.research.att.com/dist/mab/) (<ftp://ftp.research.att.com/dist/mab/>).

TCFS improves on CFS by adding more integration with the file system, so that it’s transparent to users that the file system is encrypted. More information at: [tcfs/](http://www.tcfs.it/) (<http://www.tcfs.it/>).

It also need not be used on entire file systems. It works on directory trees as well.

10.6.10. X11, SVGA and display security

10.6.10.1. X11

It’s important for you to secure your graphical display to prevent attackers from grabbing your passwords as you type them, reading documents or information you are reading on your screen, or even using a hole to gain root access. Running remote *X* applications over a network also can be fraught with peril, allowing sniffers to see all your interaction with the remote system.

X has a number of access-control mechanisms. The simplest of them is host-based: you use `xhost` to specify the hosts that are allowed access to your display. This is not very secure at all, because if someone has access to your computer, they can `xhost + their computer` and get in easily. Also, if you have to allow access from an untrusted computer, anyone there can compromise your display.

When using `xdm` (*X Display Manager*), or its *KDE* counterpart: *KDM*, to log in, you get a much better access method: MIT-MAGIC-COOKIE-1. A 128-bit “cookie” is generated and stored in your `.Xauthority` file. If you need to allow a remote computer access to your display, you can use the `xauth` command and the information in your `.Xauthority` file to provide access to only that connection. See the Remote-X-Apps mini-howto, available at <http://metalab.unc.edu/LDP/HOWTO/mini/Remote-X-Apps.html> (<http://metalab.unc.edu/LDP/HOWTO/mini/Remote-X-Apps.html>).

You can also use `ssh` (see *ssh (Secure SHell)* and *stelnet*, page 96, above) to allow secure *X* connections. This has the advantage of also being transparent to the end user, and means that no unencrypted data flows across the network.

You can also disable any remote connections to your *X* server by using the `-nolisten tcp` options to your *X* server. This will prevent any network connections to your server over tcp sockets.

Take a look at the `Xsecurity` man page for more information on *X* security. The safe bet is to use `xdm` to login to your console and then use `ssh` to go to remote sites on which you wish to run *X* programs.

10.6.10.2. SVGA

SVGAlib programs are typically *suid*-root in order to access all your *GNU/Linux* computer's video hardware. This makes them very dangerous. If they crash, you typically need to reboot your computer to get a usable console back. Make sure any *SVGAlib* programs you are running are authentic, and can at least be somewhat trusted. Even better, don't run them at all.

10.6.10.3. GGI (Generic Graphics Interface project)

The *GNU/Linux* GGI project is trying to solve several of the problems with video interfaces on *GNU/Linux*. GGI will move a small piece of the video code into the *GNU/Linux* kernel, and then control access to the video system. This means GGI will be able to restore your console at any time to a known good state. They will also allow a secure attention key, so you can be sure that there is no Trojan horse login program running on your console. <http://www.ggi-project.org/> (<http://www.ggi-project.org/>)

10.7. Kernel Security

This is a description of the kernel configuration options that relate to security, and an explanation of what they do, and how to use them.

As the kernel controls your computer's networking, it is important that it be very secure, and not be compromised. To prevent some of the latest networking attacks, you should try to keep your kernel version current. You can find new kernels at <ftp://ftp.kernel.org> (<ftp://ftp.kernel.org>) or from packages updates available through MandrakeUpdate.

There is also a international group providing a single unified crypto patch to the mainstream *GNU/Linux* kernel. This patch provides support for a number of cryptographic subsystems and things that cannot be included in the mainstream kernel due to export restrictions. For more information, visit their web page at: [kerneli](http://www.kerneli.org) (<http://www.kerneli.org>).

10.7.1. Kernel Compile Options

When this document was written, kernel 2.2 was state-of-the-art. Still today, most firewalls still run 2.2. However, with kernel 2.4, a lot of things have changed. Most of the compile options in this chapter are still valid, but the Masquerading and port forwarding have been replaced by iptables. You can have more information on iptables at [linuxguruz.org](http://www.linuxguruz.org) (<http://www.linuxguruz.org/iptables/howto/iptables-HOWTO.html>).

For 2.2.x kernels, the following options apply. You should see these options during the kernel configuration process. Many of the comments here are from `/usr/src/linux/Documentation/Configure.help`, which is the same document that is referenced while using the Help facility during the `make config` stage of compiling the kernel. Please consult the chapter Compiling And Installing New Kernels of the *Reference Manual* to a full description of the compilation of a brand new kernel.

- Network Firewalls (CONFIG_FIREWALL)

This option should be on if you intend to run any firewalling or masquerading on your *GNU/Linux* computer. If it's just going to be a regular client computer, it's safe to say no.

- IP: forwarding/gatewaying (CONFIG_IP_FORWARD)

If you enable IP forwarding, your *GNU/Linux* box essentially becomes a router. If your computer is on a network, you could be forwarding data from one network to another, and perhaps subverting a firewall that was put there to prevent this from happening. Normal dial-up users will want to disable this, and other users should concentrate on the security implications of doing this. Firewall computers will want this enabled, and used in conjunction with firewall software.

You can enable IP forwarding dynamically using the following command:

```
root# echo 1 > /proc/sys/net/ipv4/ip_forward
```

and disable it with the command:

```
root# echo 0 > /proc/sys/net/ipv4/ip_forward
```

- IP: syn cookies (CONFIG_SYN_COOKIES)

a “SYN Attack” is a denial of service (DoS) attack that consumes all the resources on your computer, forcing you to reboot. We can’t think of a reason you wouldn’t normally enable this. In the 2.1 kernel series this config option merely allows syn cookies, but does not enable them. To enable them, you have to do:

```
root# echo 1 > /proc/sys/net/ipv4/tcp_syncookies <P>
```

- IP: Firewalling (CONFIG_IP_FIREWALL)

This option is necessary if you are going to configure your computer as a firewall, do masquerading, or wish to protect your dial-up workstation from someone entering via your PPP dial-up interface.

- IP: firewall packet logging (CONFIG_IP_FIREWALL_VERBOSE)

This option gives you information about packets your firewall received, like sender, recipient, port, etc.

- IP: Drop source routed frames (CONFIG_IP_NOSR)

This option should be enabled. Source routed frames contain the entire path to their destination inside of the packet. This means that routers through which the packet goes do not need to inspect it, and just forward it on. This could lead to data entering your system that may be a potential exploit.

- IP: masquerading (CONFIG_IP_MASQUERADE)

If one of the computers on your local network for which your *GNU/Linux* box acts as a firewall wants to send something to the outside, your box can “masquerade” as that host, i.e., it forwards the traffic to the intended destination, but makes it look like it came from the firewall box itself. See <http://www.indyramp.com/masq> (<http://www.indyramp.com/masq>) and the chapter “*Configuring Masqueraded Clients*”, page 23 for more information.

- IP: ICMP masquerading (CONFIG_IP_MASQUERADE_ICMP)

This option adds ICMP masquerading to the previous option of only masquerading TCP or UDP traffic.

- IP: transparent proxy support (CONFIG_IP_TRANSPARENT_PROXY)

This enables your *GNU/Linux* firewall to transparently redirect any network traffic originating from the local network and destined for a remote host to a local server, called a “transparent proxy server”. This makes the local computers think they are talking to the remote end, while in fact they are connected to the local proxy. See the IP-Masquerading *HOWTO* and <http://www.indyramp.com/masq> (<http://www.indyramp.com/masq>) for more information.

- IP: always defragment (CONFIG_IP_ALWAYS_DEFRAG)

Generally this option is disabled, but if you are building a firewall or a masquerading host, you will want to enable it. When data is sent from one host to another, it does not always get sent as a single packet of data, but rather it is fragmented into several pieces. The problem with this is that the port numbers are only stored in the first fragment. This means that someone can insert information into the remaining packets that isn’t supposed to be there. It could also prevent a teardrop attack against an internal host that is not yet itself patched against it.

- Packet Signatures (CONFIG_NCPFS_PACKET_SIGNING)

This is an option that will sign NCP packets for stronger security. Normally you can leave it off, but it is there if you do need it.

- IP: Firewall packet netlink device (CONFIG_IP_FIREWALL_NETLINK)

This is a really neat option that allows you to analyze the first 128 bytes of the packets in a user-space program, to determine if you would like to accept or deny the packet, based on its validity.

- Socket Filtering (CONFIG_FILTER)

For most people, it's safe to say no to this option. This option allows you to connect a user-space filter to any socket and determine if packets should be allowed or denied. Unless you have a very specific need and are capable of programming such a filter, you should say no. Also note that as of this writing, all protocols were supported except TCP.

- Port Forwarding

Port Forwarding is an addition to IP Masquerading which allows some forwarding of packets from outside to inside a firewall on given ports. This could be useful if, for example, you want to run a web server behind the firewall or masquerading host and that web server should be accessible from the outside world. An external client sends a request to port 80 of the firewall, the firewall forwards this request to the web server, the web server handles the request and the results are sent through the firewall to the original client. The client thinks that the firewall computer itself is running the web server. This can also be used for load balancing if you have a farm of identical web servers behind the firewall. Information about this feature is available from monmouth (<http://www.monmouth.demon.co.uk/ipsubs/portforwarding.html>). For general info, please see compsoc (<ftp://ftp.compsoc.net/users/steve/ipportfw/linux21/>).

- Socket Filtering (CONFIG_FILTER)

Using this option, user-space programs can attach a filter to any socket and thereby tell the kernel that it should allow or disallow certain types of data to get through the socket. *GNU/Linux* socket filtering works on all socket types except TCP for now. See the text file `/usr/src/linux/Documentation/networking/filter.txt` for more information.

- IP: Masquerading

The 2.2 kernel masquerading has been improved. It provides additional support for masquerading special protocols, etc. Be sure to read the IP Chains *HOWTO* for more information.

10.7.2. Kernel Devices

There are a few block and character devices available on *GNU/Linux* that will also help you with security.

The two devices `/dev/random` and `/dev/urandom` are provided by the kernel to provide random data at any time.

Both `/dev/random` and `/dev/urandom` should be secure enough to use in generating *PGP* keys, *ssh* challenges, and other applications where secure random numbers are required. Attackers should be unable to predict the next number given any initial sequence of numbers from these sources. There has been a lot of effort put in to ensuring that the numbers you get from these sources are random in every sense of the word.

The only difference between the two devices, is that `/dev/random` runs out of random bytes and it makes you wait for more to be accumulated. Note that on some systems, it can block for a long time waiting for new user-generated entropy to be entered into the system. So you have to use care before using `/dev/random`. (Perhaps the best thing to do is to use it when you're generating sensitive keying information, and you tell the user to pound on the keyboard repeatedly until you print out "OK, enough".)

`/dev/random` is high quality entropy, generated from measuring the inter-interrupt times etc. It blocks until enough bits of random data are available.

`/dev/urandom` is similar, but when the store of entropy is running low, it'll return a cryptographically strong hash of what there is. This isn't as secure, but it's enough for most applications.

You might read from the devices using something like:

```
root# head -c 6 /dev/urandom | mimencode
```

This will print six random characters on the console, suitable for password generation. You can find `mimencode` in the `metamail` package.

See `/usr/src/linux/drivers/char/random.c` for a description of the algorithm.

10.8. Network Security

Network security is becoming more and more important as people spend more and more time connected. Compromising network security is often much easier than compromising physical or local security, and is much more common.

There are a number of good tools to assist with network security, and more and more of them are shipped with your **Mandrake Linux** distribution, either in the main CD-ROM, contribs, or through the FTP crypto server (see above).

10.8.1. Packet Sniffers

One of the most common ways intruders gain access to more systems on your network is by employing a packet sniffer on a already compromised host. This "sniffer" just listens on the *Ethernet* port for things like `passwd` and `login` and `su` in the packet stream and then logs the traffic after that. This way, attackers gain passwords for systems they are not even attempting to break into. Clear-text passwords are very vulnerable to this attack.

Example: Host A has been compromised. Attacker installs a sniffer. Sniffer picks up admin logging into Host B from Host C. It gets the admin's personal password as they login to B. Then, the admin does a `su` to fix a problem. They now have the root password for Host B. Later the admin lets someone `telnet` from his account to Host Z on another site. Now the attacker has a password/*login* on Host Z.

In this day and age, the attacker doesn't even need to compromise a system to do this: they could also bring a laptop or PC into a building and tap into your net.

Using `ssh` or other encrypted password methods thwarts this attack. Things like APOP for POP accounts also prevents this attack. (Normal POP logins are very vulnerable to this, as is anything that sends clear-text passwords over the network.)

10.8.2. System services and `tcp_wrappers`

Before you put your *GNU/Linux* system on **ANY** network the first thing to look at is what services you need to offer. Services that you do not need to offer should be disabled so that you have one less thing to worry about and attackers have one less place to look for a hole.

There are a number of ways to disable services under *GNU/Linux*. You can look at your `/etc/inetd.conf` file and see what services are being offered by your `inetd`. Disable any that you do not need by commenting them out (# at the beginning of the line), and then restart your `inetd` service.

You can also remove (or comment out) services in your `/etc/services` file. This will mean that local clients will also be unable to find the service (i.e., if you remove `ftp`, and try and `ftp` to a remote site from that computer it will fail with an `unknown service` message). It's usually not worth the trouble to remove services from `/etc/services`, since it provides no additional security. If a local person wanted to use `ftp` even though you had commented it out, they would make their own client that use the common FTP port and would still work fine.

Some of the services you might want to leave enabled are:

- `ftp`
- `telnet` (or `ssh`)
- mail, such as `pop-3` or `imap`
- `identd`

If you know you are not going to use some particular package, you can also delete it entirely. `rpm -e packagename` will erase an entire package.

Additionally, you really want to disable the `rsh/rlogin/rcp` utilities, including `login` (used by `rlogin`), `shell` (used by `rcp`), and `exec` (used by `rsh`) from being started in `/etc/inetd.conf`. These protocols are extremely insecure and have been the cause of exploits in the past.

You should check your `/etc/rc.d/rc[0-9].d`, and see if any of the servers started in that directory are not needed. The files in that directory are actually symbolic links to files in the directory `/etc/rc.d/init.d`. Renaming the files in the `init.d` directory disables all the symbolic links that point to that file. If you only wish to disable a service for a particular run level, rename the appropriate symbolic link by replacing the `S` with a `K`, like this:

```
root# cd /etc/rc6.d
root# mv S45dhcpd K45dhcpd
```



You may also use a command line utility to do that: `chkconfig` or the graphical interface under *KDE*: `ksysv`.

Your **Mandrake Linux** distributions ships with a `tcp_wrapper` “wrapping” all your TCP services. The `tcp_wrapper` (`tcpd`) is invoked from `inetd` instead of the real server. `tcpd` then checks the host that is requesting the service, and either executes the real server, or denies access from that host. `tcpd` allows you to restrict access to your TCP services. You should edit `/etc/hosts.allow` and add in only those hosts that need to have access to your computer’s services.

If you are a home dial up user, we suggest you deny ALL. `tcpd` also logs failed attempts to access services, so this can alert you if you are under attack. If you add new services, you should be sure to configure them to use `tcp_wrappers` if they are TCP-based. For example, a normal dial-up user can prevent outsiders from connecting to his computer, yet still have the ability to retrieve mail, and make network connections to the Internet. To do this, you might add the following to your `/etc/hosts.allow`:

ALL: 127.

And of course `/etc/hosts.deny` would contain:

ALL: ALL

which will prevent external connections to your computer, yet still allow you from the inside to connect to servers on the Internet.

Keep in mind that `tcp_wrappers` only protects services executed from `inetd`, and a select few others. There very well may be other services running on your computer. You can use `netstat -ta` to find a list of all the services your computer is offering.

10.8.3. Verify Your DNS Information

Keeping up-to-date DNS information about all hosts on your network can help to increase security. If an unauthorized host becomes connected to your network, you can recognize it by its lack of a DNS entry. Many services can be configured to not accept connections from hosts that do not have valid DNS entries.

10.8.4. `identd`

`identd` is a small program that typically runs out of your `inetd` server. It keeps track of what user is running what TCP service, and then reports this to whoever requests it.

Many people misunderstand the usefulness of `identd`, and so disable it or block all off site requests for it. `identd` is not there to help out remote sites. There is no way of knowing if the data you get from the remote `identd` is correct or not. There is no authentication in `identd` requests.

Why would you want to run it then? Because it helps **you** out, and is another data-point in tracking. If your `identd` is uncompromised, then you know it’s telling remote sites the user-name or UID of people using TCP services. If the admin at a remote site comes back to you and tells you user so-and-so was trying to hack into their site, you can easily take action against that user. If you are not running `identd`, you will have to look at

lots and lots of logs, figure out who was on at the time, and in general take a lot more time to track down the user.

The `identd` that ships with most distributions is more configurable than many people think. You can disable it for specific users (they can make a `.noident` file), you can log all `identd` requests (We recommend it), you can even have `identd` return a UID instead of a user name or even `NO-USER`.

10.8.5. Configuring and Securing the Postfix MTA

The Postfix mail server was written by Wietse Venema, author of Postfix and several other staple Internet security products, as an “attempt to provide an alternative to the widely-used Sendmail program. Postfix attempts to be fast, easy to administer, and hopefully secure, while at the same time being sendmail compatible enough to not upset your users.”

Further information on postfix can be found at the Postfix home (<http://www.postfix.org>) and in the Configuring and Securing Postfix (http://www.linuxsecurity.com/feature_stories/feature_story-91.html).

10.8.6. SATAN, ISS, and Other Network Scanners

There are a number of different software packages out there that do port and service-based scanning of computers or networks. *SATAN*, *ISS*, *SAINT*, and *Nessus* are some of the more well-known ones. This software connects to the target computer (or all the target computers on a network) on all the ports they can, and try to determine what service is running there. Based on this information, you can tell if the computer is vulnerable to a specific exploit on that server.

SATAN (Security Administrator’s Tool for Analyzing Networks) is a port scanner with a web interface. It can be configured to do light, medium, or strong checks on a computer or a network of computers. It’s a good idea to get *SATAN* and scan your computer or network, and fix the problems it finds. Make sure you get the copy of *SATAN* from metalab (<http://metalab.unc.edu/pub/packages/security/Satan-for-Linux/>) or a reputable FTP or web site. There was a Trojan copy of *SATAN* that was distributed out on the net. trouble.org (<http://www.trouble.org/~zen/satan/satan.html>). Note that *SATAN* has not been updated in quite a while, and some of the other tools below might do a better job.

ISS (Internet Security Scanner) is another port-based scanner. It is faster than Satan, and thus might be better for large networks. However, *SATAN* tends to provide more information.

TriSentry (formerly Abacus) is a suite of tools to provide host-based security and intrusion detection. look at its home page on the web for more information. <http://www.psionic.com/products/> (<http://www.psionic.com/products/>)

SAINT is a updated version of *SATAN*. It is web based and has many more up to date tests than *SATAN*. You can find out more about it at: <http://www.wwdsi.com/~saint> (<http://www.wwdsi.com/saint>)

Nessus is a free security scanner. It has a GTK graphical interface for ease of use. It is also designed with a very nice plug in setup for new port-scanning tests. For more information, take a look at: <http://www.nessus.org> (<http://www.nessus.org/>)

10.8.6.1. Detecting Port Scans

There are some tools designed to alert you to probes by *SATAN* and *ISS* and other scanning software. However, if you liberally use `tcp_wrappers`, look over your log files regularly, you should be able to notice such probes. Even on the lowest setting, *SATAN* still leaves traces in the logs.

There are also “stealth” port scanners. A packet with the TCP ACK bit set (as is done with established connections) will likely get through a packet-filtering firewall. The returned RST packet from a port that **had no established session** can be taken as proof of life on that port. I don’t think TCP wrappers will detect this.

You might also look at SNORT (<http://www.snort.org>), which is a free IDS (Intrusion Detection System), which can detect other network intrusions.

10.8.7. sendmail, qmail and MTA's¹

One of the most important services you can provide is a mail server. Unfortunately, it is also one of the most vulnerable to attack, simply due to the number of tasks it must perform and the privileges it typically needs.

If you are using *sendmail* it is very important to keep up on current versions. *sendmail* has a long long history of security exploits. Always make sure you are running the most recent version from *sendmail* (<http://www.sendmail.org/>).

Keep in mind that *sendmail* does not have to be running in order for you to send mail. If you are a home user, you can disable *sendmail* entirely, and simply use your mail client to send mail. You might also choose to remove the `-bd` flag from the *sendmail* startup file, thereby disabling incoming requests for mail. In other words, you can execute *sendmail* from your startup script using the following instead:

```
# /usr/lib/sendmail -q15m
```

This will cause *sendmail* to flush the mail queue every fifteen minutes for any messages that could not be successfully delivered on the first attempt.

Many administrators choose not to use *sendmail*, and instead choose one of the other mail transport agents. You might consider switching over to *qmail*. *qmail* was designed with security in mind from the ground up. It's fast, stable, and secure. *Qmail* can be found at *qmail* (<http://www.qmail.org>)

In direct competition to *qmail* is *postfix*, written by Wietse Venema, the author of *tcp_wrappers* and other security tools. Formerly called *umailer*, and sponsored by **IBM**, this is also a mail transport agent written from the ground up with security in mind. You can find more information about *postfix* at *postfix* (<http://www.postfix.org>)



postfix is the default MTA shipped with **Mandrake Linux**. Please, refer to "*Postfix Mail Server*", page 47.

10.8.8. Denial of Service Attacks

A "Denial of Service" (DoS) attack is one where the attacker tries to make some resource too busy to answer legitimate requests, or to deny legitimate users access to your computer.

Denial of service attacks have increased greatly in recent years. Some of the more popular and recent ones are listed below. Note that new ones show up all the time, so this is just a few examples. Read the *GNU/Linux* security lists and the bugtraq list and archives for more current information.

- **SYN Flooding** - SYN flooding is a network denial of service attack. It takes advantage of a "loophole" in the way TCP connections are created. The newer *GNU/Linux* kernels (2.0.30 and up) have several configurable options to prevent SYN flood attacks from denying people access to your computer or services. See *Kernel Security*, page 100 for proper kernel protection options.
- **Ping Flooding** - Ping flooding is a simple brute-force denial of service attack. The attacker sends a "flood" of ICMP packets to your computer. If they are doing this from a host with better bandwidth than yours, your computer will be unable to send anything on the network. A variation on this attack, called "smurfing", sends ICMP packets to a host with **your** computer's return IP, allowing them to flood you less detectably. You can find more information about the "smurf" attack at *linuxsecurity.com* (http://www.linuxsecurity.com/articles/network_security_article-4258.html)

If you are ever under a ping flood attack, use a tool like *tcpdump* to determine where the packets are coming from (or appear to be coming from), then contact your provider with this information. Ping floods can most easily be stopped at the router level or by using a firewall.

- **Ping o' Death** - The Ping o' Death attack sends ICMP `ECHO REQUEST` packets that are too large to fit in the kernel data structures intended to store them. Because sending a single, large (65,510 bytes) "ping" packet to many systems will cause them to hang or even crash, this problem was quickly dubbed the "Ping o' Death". This one has long been fixed, and is no longer anything to worry about.

1. Mail Transport Agents

You can find code for most exploits, and a more in-depth description of how they work, at <http://www.insecure.org/spl0its.html> (<http://www.insecure.org/spl0its.html>) using their search engine.

10.8.9. NFS (Network File System) Security.

NFS is a very widely-used file sharing protocol. It allows servers running `nfsd` and `mountd` to “export” entire file systems to other computers using NFS file system support built in to their kernels (or some other client support if they are not *GNU/Linux* computers). `mountd` keeps track of mounted file systems in `/etc/mtab`, and can display them with `showmount`.

Many sites use NFS to serve home directories to users, so that no matter what computer in the cluster they login to, they will have all their home files.

There is some small amount of security allowed in exporting file systems. You can make your `nfsd` map the remote root user (UID=0) to the nobody user, denying them total access to the files exported. However, since individual users have access to their own (or at least the same UID) files, the remote root user can login or `su` to their account and have total access to their files. This is only a small hindrance to an attacker that has access to mount your remote file systems.

If you must use NFS, make sure you export to only those computers that you really need to. Never export your entire root directory; export only directories you need to export.

See the NFS *HOWTO* for more information on NFS, available at LDP (<http://www.ibiblio.org/mdw/HOWTO/NFS-HOWTO/>)

10.8.10. NIS (Network Information Service) (formerly YP).

Network Information service (formerly YP) is a means of distributing information to a group of computers. The NIS master holds the information tables and converts them into NIS map files. These maps are then served over the network, allowing NIS client computers to get login, password, home directory and *shell* information (all the information in a standard `/etc/passwd` file). This allows users to change their password once and have it take effect on all the computers in the NIS domain.

NIS is not at all secure. It was never meant to be. It was meant to be handy and useful. Anyone that can guess the name of your NIS domain (anywhere on the net) can get a copy of your `passwd` file, and use *crack* and *John the Ripper* against your users’ passwords. Also, it is possible to spoof NIS and do all sorts of nasty tricks. If you must use NIS, make sure you are aware of the dangers.

There is a much more secure replacement for NIS, called *NIS+*. Check out the NIS *HOWTO* for more information: NIS-HOWTO (<http://www.ibiblio.org/mdw/HOWTO/NIS-HOWTO/>)

10.8.11. Firewalls

Firewalls are a means of controlling what information is allowed into and out of your local network. Typically the firewall host is connected to the Internet and your local LAN, and the only access from your LAN to the Internet is through the firewall. This way the firewall can control what passes back and forth from the Internet and your LAN.

There are a number of types of firewalls and methods of setting them up. *GNU/Linux* computers make pretty good firewalls. Firewall code can be built right into 2.0 and higher kernels. The user-space tools `ipchains` for 2.2 kernels, and `iptables` for 2.4 kernels, allows you to change, on the fly, the types of network traffic you allow. You can also log particular types of network traffic.

Firewalls are a very useful and important technique in securing your network. However, never think that because you have a firewall, you don’t need to secure the computers behind it. This is a fatal mistake. Check out the very good Firewall-HOWTO at your latest metalab archive for more information on firewalls and *GNU/Linux*. Firewall-HOWTO (<http://www.ibiblio.org/mdw/HOWTO/Firewall-HOWTO.html>)

If you have no experience with firewalls, and plan to set up one for more than just a simple security policy, the Firewalls book by O’Reilly and Associates or other online firewall document is mandatory reading. Check out O’Reilly site (<http://www.ora.com>) for more information. The National Institute of Standards and Technology have put together an excellent document on firewalls. Although dated 1995, it is still quite good. You can find it at nist.gov (<http://csrc.nist.gov/publications/nistpubs/800-10/>). Also of interest includes:

- The Freefire Project -- a list of freely-available firewall tools, available at freefire (http://sites.inka.de/sites/lina/freefire-1/index_en.html)
- Mason -- the automated firewall builder for *GNU/Linux*. This is a firewall script that learns as you do the things you need to do on your network! More info at: mason (<http://www.pobox.com/~wstearns/mason/>)

10.8.12. IP Chains - GNU/Linux Kernel 2.2.x Firewalling

GNU/Linux IP Firewalling Chains is an update to the 2.0 *GNU/Linux* firewalling code for the 2.2 kernel. It has many more features than previous implementations, including:

- More flexible packet manipulations
- More complex accounting
- Simple policy changes possible automatically
- Fragments can be explicitly blocked, denied, etc.
- Logs suspicious packets.
- Can handle protocols other than ICMP/TCP/UDP.

Be sure to read the IP Chains *HOWTO* for further information. It is available at linuxdos.org (<http://www.linuxdoc.org/HOWTO/IPCHAINS-HOWTO.html>)

10.8.13. Netfilter - Linux Kernel 2.4.x Firewalling

In yet another set of advancements to the kernel IP packet filtering code, netfilter allows users to set up, maintain, and inspect the packet filtering rules in the new 2.4 kernel.

The netfilter subsystem is a complete rewrite of previous packet filtering implementations including ipchains and ipfwadm. Netfilter provides a large number of improvements, and it has now become an even more mature and robust solution for protecting corporate networks.

iptables is the command-line interface used to manipulate the firewall tables within the kernel.

Netfilter provides a raw framework for manipulating packets as they traverse through various parts of the kernel. Part of this framework includes support for masquerading, standard packet filtering, and now more complete network address translation. It even includes improved support for load balancing requests for a particular service among a group of servers behind the firewall.

The stateful inspection features are especially powerful. Stateful inspection provides the ability to track and control the flow of communication passing through the filter. The ability to keep track of state and context information about a session not only makes rules simpler but also helps to better interpret higher-level protocols.

Additionally, small modules can be developed to perform additional specific functions, such as passing packets to programs in userspace for processing then reinjecting back into the normal packet flow. The ability to develop these programs in userspace reduces the level of complexity that was previously associated with having to make changes directly at the kernel level.

Other IP Tables references include:

- Oskar Andreasson IP Tables Tutorial (http://www.linuxsecurity.com/feature_stories/feature_story-94.html) — Oskar Andreasson speaks with LinuxSecurity.com about his comprehensive IP Tables tutorial and how this document can be used to build a robust firewall for your organization.
- Hal Burgiss Introduces Linux Security Quick-Start Guides (http://www.linuxsecurity.com/feature_stories/feature_story-93.html) — Hal Burgiss has written two authoritative guides on securing Linux, including managing firewalling.
- Netfilter Home page (<http://netfilter.samba.org>) — The netfilter/iptables home page.
- Linux Kernel 2.4 Firewalling Matures: netfilter (http://www.linuxsecurity.com/feature_stories/kernel-netfilter.html) — This LinuxSecurity.com article describes the basics of packet filtering, how

to get started using iptables, and a list of the new features available in the latest generation of firewalling for Linux.

10.8.14. VPNs - Virtual Private Networks

VPN's are a way to establish a "virtual" network on top of some already existing network. This virtual network often is encrypted and passes traffic only to and from some known entities that have joined the network. VPN's are often used to connect someone working at home over the public Internet to a internal company network.

If you are running a *GNU/Linux* masquerading firewall and need to pass MS PPTP (Microsoft's VPN point-to-point product) packets, there is a linux kernel patch out to do just that. See: ip-masq-vpn (ftp://ftp.rubyriver.com/pub/jhardin/masquerade/ip_masq_vpn.html).

There are several *GNU/Linux* VPN solutions available:

- `vpnd`. See the <http://sunsite.auc.dk/vpnd/> (<http://sunsite.auc.dk/vpnd/>).
- Free S/Wan, available at <http://www.xs4all.nl/~freeswan/> (<http://www.xs4all.nl/~freeswan/>)
- `ssh` can be used to construct a VPN. See the VPN mini-howto for more information.
- `vps` (virtual private server) at strongcrypto (<http://www.strongcrypto.com>).
- `vtun` (virtual tunnel) at sourceforge (<http://vtun.sourceforge.net/>).
- `yavipin` (<http://yavipin.sourceforge.net>).

See also the section on IPSEC for pointers and more information.

10.9. Security Preparation (before you go on-line)

Ok, so you have checked over your system, and determined it's as secure as feasible, and you're ready to put it online. There are a few things you should now do in order to prepare for an intrusion, so you can quickly disable the intruder, and get back up and running.

10.9.1. Make a Full Backup of Your Computer

Discussion of backup methods and storage is beyond the scope of this chapter, but here are a few words relating to backups and security:

If you have less than 650MB of data to store on a partition, a CD-R copy of your data is a good way to go (as it's hard to tamper with later, and if stored properly can last a long time), you will of course need at least 650MB of space to make the image. Tapes and other re-writable media should be write-protected as soon as your backup is complete, and then verified to prevent tampering. Make sure you store your backups in a secure off-line area. A good backup will ensure that you have a known good point to restore your system from.

10.9.2. Choosing a Good Backup Schedule

A six-tape cycle is easy to maintain. This includes four tapes for during the week, one tape for even Fridays, and one tape for odd Fridays. Perform an incremental backup every day, and a full backup on the appropriate Friday tape. If you make some particularly important changes or add some important data to your system, a full backup might well be in order.

10.9.3. Testing your backups

You should do periodic tests of your backups to make sure they are working as you might expect them to. Restores of files and checking against the real data, sizes and listings of backups, and reading old backups should be done on a regular basis.

10.9.4. Backup Your RPM File Database

In the event of an intrusion, you can use your RPM database like you would use tripwire, but only if you can be sure it too hasn't been modified. You should copy the RPM database to a floppy, and keep this copy off-line at all times.

The files `/var/lib/rpm/fileindex.rpm` and `/var/lib/rpm/packages.rpm` most likely won't fit on a single floppy. But if compressed, each should fit on a separate floppy.

Now, when your system is compromised, you can use the command:

```
root# rpm -Va
```

to verify each file on the system. See the `rpm` man page, as there are a few other options that can be included to make it less verbose. Keep in mind you must also be sure your RPM binary has not been compromised.

This means that every time a new RPM is added to the system, the RPM database will need to be re-archived. You will have to decide the advantages versus drawbacks.

10.9.5. Keep Track of Your System Accounting Data

It is very important that the information that comes from `syslog` has not been compromised. Making the files in `/var/log` readable and writable by only a limited number of users is a good start.

Be sure to keep an eye on what gets written there, especially under the `auth` facility. Multiple login failures, for example, can indicate an attempted break-in.

You will want to look in `/var/log` and check `messages`, `mail.log`, and others.

You might also want to configure your log-rotating script to keep logs around longer so you have time to examine them. Take a look at the `logrotate` man page.

If your log files have been tampered with, see if you can determine when the tampering started, and what sort of things appeared to be tampered with. Are there large periods of time that cannot be accounted for? Checking backup tapes (if you have any) for untampered log files is a good idea.

Intruders typically modify log files in order to cover their tracks, but they should still be checked for strange happenings. You may notice the intruder attempting to gain entrance, or exploit a program in order to obtain the root account. You might see log entries before the intruder has time to modify them.

You should also be sure to separate the `auth` facility from other log data, including attempts to switch users using `su`, login attempts, and other user accounting information.

If possible, configure `syslog` to send a copy of the most important data to a secure system. This will prevent an intruder from covering his tracks by deleting his `login/su/ftp` etc attempts. See the `syslog.conf` man page, and refer to the `@` option.

There are several more advanced `syslogd` programs out there. Take a look at <http://www.core-sdi.com/ssyslog/> (<http://www.core-sdi.com/ssyslog/>) for Secure Syslog. Secure Syslog allows you to encrypt your `syslog` entries and make sure no one has tampered with them.

Another `syslogd` with more features is `syslog-ng` (<http://www.balabit.hu/en/downloads/syslog-ng/>). It allows you a lot more flexibility in your logging and also can crypt your remote `syslog` streams to prevent tampering.

Finally, log files are much less useful when no one is reading them. Take some time out every once in a while to look over your log files, and get a feeling for what they look like on a normal day. Knowing this can help make unusual things stand out.

10.9.6. Apply All New System Updates.

Due to the fast-paced nature of security fixes, new (fixed) programs are always being released. Before you connect your computer to the network, it's a good idea to run `MandrakeUpdate` (on another computer connected to the Internet and get all the updated packages since you received your distribution CD-ROM. Many times these packages contain important security fixes, so it's a good idea to get them installed.

10.10. What To Do During and After a Breaking

So you have followed some of the advice here (or elsewhere) and have detected a break-in? The first thing to do is to remain calm. Hasty actions can cause more harm than the attacker would have.

10.10.1. Security Compromise Underway.

Spotting a security compromise under way can be a tense undertaking. How you react can have large consequences.

If the compromise you are seeing is a physical one, odds are you have spotted someone who has broken into your home, office or lab. You should notify your local authorities. In a lab, you might have spotted someone trying to open a case or reboot a computer. Depending on your authority and procedures, you might ask them to stop, or contact your local security people.

If you have detected a local user trying to compromise your security, the first thing to do is confirm they are in fact who you think they are. Check the site they are logging in from. Is it the site they normally log in from? No? Then use a non-electronic means of getting in touch. For instance, call them on the phone or walk over to their office/house and talk to them. If they agree that they are on, you can ask them to explain what they were doing or tell them to cease doing it. If they are not on, and have no idea what you are talking about, odds are this incident requires further investigation. Look into such incidents, and have lots of information before making any accusations.

If you have detected a network compromise, the first thing to do (if you are able) is to disconnect your network. If they are connected via modem, unplug the modem cable; if they are connected via *Ethernet*, unplug the *Ethernet* cable. This will prevent them from doing any further damage, and they will probably see it as a network problem rather than detection.

If you are unable to disconnect the network (if you have a busy site, or you do not have physical control of your computers), the next best step is to use something like `tcp_wrappers` or `ipfwadm` to deny access from the intruder's site.

If you can't deny all people from the same site as the intruder, locking the user's account will have to do. Note that locking an account is not an easy thing. You have to keep in mind `.rhosts` files, FTP access, and a host of possible back doors.

After you have done one of the above (disconnected the network, denied access from their site, and/or disabled their account), you need to kill all their user processes and log them off.

You should monitor your site well for the next few minutes, as the attacker will try to get back in. Perhaps using a different account, and/or from a different network address.

10.10.2. Security Compromise has already happened

So you have either detected a compromise that has already happened or you have detected it and locked (hopefully) the offending attacker out of your system. Now what?

10.10.2.1. Closing the Hole

If you are able to determine what means the attacker used to get into your system, you should try to close that hole. For instance, perhaps you see several FTP entries just before the user logged in. Disable the FTP service and check and see if there is an updated version, or if any of the lists know of a fix.

Check all your log files, and make a visit to your security lists and pages and see if there are any new common exploits you can fix. You can find your **Mandrake Linux** security fixes by running the `MandrakeUpdate` regularly.

There is now a *GNU/Linux* security auditing project. They are methodically going through all the user-space utilities and looking for possible security exploits and overflows. From their announcement:

"We are attempting a systematic audit of *GNU/Linux* sources with a view to being as secure as *OpenBSD*. We have already uncovered (and fixed) some problems, but more help is welcome. The list is unmoderated and also a useful resource for general security discussions. The list address is: `security-audit@ferret.lmh.ox.ac.uk`. To subscribe, send a mail to: `security-audit-subscribe@ferret.lmh.ox.ac.uk`"

If you don't lock the attacker out, they will likely be back. Not just back on your computer, but back somewhere on your network. If they were running a packet sniffer, odds are good they have access to other local computers.

10.10.2.2. Assessing the Damage

The first thing is to assess the damage. What has been compromised? If you are running an Integrity Checker like *Tripwire*, you can use it to perform an integrity check; and it should help to tell you what has been compromised. If not, you will have to look around at all your important data.

Since *GNU/Linux* systems are getting easier and easier to install, you might consider saving your config files, wiping your disk(s), reinstalling, then restoring your user files and your config files from backups. This will ensure that you have a new, clean system. If you have to backup files from the compromised system, be especially cautious of any binaries that you restore, as they may be Trojan horses placed there by the intruder.

Re-installation should be considered mandatory upon an intruder obtaining root access. Additionally, you'd like to keep any evidence there is, so having a spare disk in the safe may make sense.

Then you have to worry about how long ago the compromise happened, and whether the backups hold any damaged work. More on backups later.

10.10.2.3. Backups, Backups, Backups!

Having regular backups is a godsend for security matters. If your system is compromised, you can restore the data you need from backups. Of course, some data is valuable to the attacker too, and they will not only destroy it, they will steal it and have their own copies; but at least you will still have the data.

You should check several backups back into the past before restoring a file that has been tampered with. The intruder could have compromised your files long ago, and you could have made many successful backups of the compromised file!

Of course, there are also a raft of security concerns with backups. Make sure you are storing them in a secure place. Know who has access to them. (If an attacker can get your backups, they can have access to all your data without you ever knowing it.)

10.10.2.4. Tracking Down the Intruder.

Ok, you have locked the intruder out, and recovered your system, but you're not quite done yet. While it is unlikely that most intruders will ever be caught, you should report the attack.

You should report the attack to the admin contact at the site from which the attacker attacked your system. You can look up this contact with *whois* or the Internic database. You might send them an email with all applicable log entries and dates and times. If you spotted anything else distinctive about your intruder, you might mention that too. After sending the email, you should (if you are so inclined) follow up with a phone call. If that admin in turn spots your attacker, they might be able to talk to the admin of the site where they are coming from and so on.

Good crackers often use many intermediate systems, some (or many) of which may not even know they have been compromised. Trying to track a cracker back to their home system can be difficult. Being polite to the admins you talk to can go a long way to getting help from them.

You should also notify any security organizations you are a part of (CERT (<http://www.cert.org/>) or similar), as well as **MandrakeSoft** (<http://www.mandrakesecure.net/en/>).

10.11. Security Sources

There are a **lot** of good sites out there for *UNIX* security in general and *GNU/Linux* security specifically. It's very important to subscribe to one (or more) of the security mailing lists and keep current on security fixes. Most of these lists are very low volume, and very informative.

10.11.1. LinuxSecurity.com References

The LinuxSecurity.com web site has numerous Linux and open source security references written by the LinuxSecurity staff and people collectively around the world.

- Linux Advisory Watch (<http://www.linuxsecurity.com/vuln-newsletter.html>) — A comprehensive newsletter that outlines the security vulnerabilities that have been announced throughout the week. It includes pointers to updated packages and descriptions of each vulnerability.
- Linux Security Week (<http://www.linuxsecurity.com/newsletter.html>) — The purpose of this document is to provide our readers with a quick summary of each week's most relevant Linux security headlines.
- Linux Security Discussion List (<http://www.linuxsecurity.com/general/maillinglists.html>) — This mailing list is for general security-related questions and comments.
- Linux Security Newsletters (<http://www.linuxsecurity.com/general/maillinglists.html>) — Subscription information for all newsletters.
- comp.os.linux.security FAQ (<http://www.linuxsecurity.com/docs/colsfaq.html>) — Frequently Asked Questions with answers for the comp.os.linux.security newsgroup.
- Linux Security Documentation (<http://www.linuxsecurity.com/docs/>) — A great starting point for information pertaining to Linux and Open Source security.

10.11.2. FTP Sites

CERT is the Computer Emergency Response Team. They often send out alerts of current attacks and fixes. See <ftp://ftp.cert.org> (<ftp://ftp.cert.org>) for more information.

ZEDZ (formerly Replay) (<http://www.zedz.net> (<http://www.zedz.net>)) has archives of many security programs. Since they are outside the US, they don't need to obey US crypto restrictions.

Matt Blaze is the author of CFS and a great security advocate. Matt's archive is available at <ftp://ftp.research.att.com/pub/mab> (<ftp://ftp.research.att.com/pub/mab>)

[tue.nl](ftp://ftp.win.tue.nl/pub/security/) is a great security FTP site in the Netherlands. [tue.nl](ftp://ftp.win.tue.nl/pub/security/) (<ftp://ftp.win.tue.nl/pub/security/>)

10.11.3. web Sites

- The Hacker FAQ is a FAQ about hackers: The Hacker FAQ (<http://www.plethora.net/~seebs/faqs/hacker.html>)
- The COAST archive has a large number of *UNIX* security programs and information: COAST (<http://www.cerias.purdue.edu/coast/>)
- SuSe Security Page: <http://www.suse.de/security/> (<http://www.suse.de/security/>)
- Rootshell.com is a great site for seeing what exploits are currently being used by crackers: <http://www.rootshell.com/> (<http://www.rootshell.com/>)
- BUGTRAQ puts out advisories on security issues: BUGTRAQ archives (<http://online.securityfocus.com/archive/1>)
- CERT, the Computer Emergency Response Team, puts out advisories on common attacks on *UNIX* platforms: CERT home (<http://www.cert.org/>)
- Dan Farmer is the author of *SATAN* and many other security tools. His home site has some interesting security survey information, as well as security tools: <http://www.trouble.org/security> (<http://www.trouble.org/security>)
- The *GNU/Linux* security WWW is a good site for *GNU/Linux* security information: Linux Security WWW (<http://www.aoy.com/Linux/Security/>)
- Infilsec has a vulnerability engine that can tell you what vulnerabilities affect a specific platform: <http://www.infilsec.com/vulnerabilities/> (<http://www.infilsec.com/vulnerabilities/>)
- CIAC sends out periodic security bulletins on common exploits: <http://ciac.llnl.gov/cgi-bin/index/bulletins> (<http://ciac.llnl.gov/cgi-bin/index/bulletins>)
- A good starting point for *GNU/Linux* Pluggable Authentication modules can be found at kernel.org (<http://www.kernel.org/pub/linux/libs/pam/>).
- WWW Security FAQ, written by Lincoln Stein, is a great web security reference. Find it at <http://www.w3.org/Security/Faq/www-security-faq.html> (<http://www.w3.org/Security/Faq/www-security-faq.html>)

10.11.4. Mailing Lists

Mandrake Linux security list: you can be informed for each security fix by subscribing to oursecurity mailing-list (<http://www.mandrakesecure.net/en/mlist.php>).

Bugtraq: To subscribe to bugtraq, send mail to listserv@netspace.org containing the message body "subscribe bugtraq". (see links above for archives).

CIAC: Send e-mail to majordomo@tholia.llnl.gov. In the BODY (not subject) of the message put: "subscribe ciac-bulletin"

10.11.5. Books – Printed Reading Material

There are a number of good security books out there. This section lists a few of them. In addition to the security specific books, security is covered in a number of other books on system administration.

References

D. Brent Chapman, Elizabeth D. Zwicky, *Building Internet Firewalls*, 1st Edition September 1995, ISBN 1-56592-124-0.

Simson Garfinkel, Gene Spafford, *Practical UNIX & Internet Security*, 2nd Edition April 1996, ISBN 1-56592-148-8.

Deborah Russell, G.T. Gangemi, Sr., *Computer Security Basics*, 1st Edition July 1991, ISBN 0-937175-71-4.

Olaf Kirch, *Linux Network Administrator's Guide*, 1st Edition January 1995, ISBN 1-56592-087-2.

Simson Garfinkel, *PGP: Pretty Good Privacy*, 1st Edition December 1994, ISBN 1-56592-098-8.

David Icove, Karl Seger, William VonStorch, *Computer Crime A Crimefighter's Handbook*, 1st Edition August 1995, ISBN 1-56592-086-4.

John S. Flowers, *Linux Security*, New Riders, March 1999, ISBN 0735700354.

Anonymous, *Maximum Linux Security : A Hacker's Guide to Protecting Your Linux Server and Network*, July 1999, ISBN 0672313413.

Terry Escamilla, *Intrusion Detection*, John Wiley and Sons, September 1998, ISBN 0471290009.

Donn Parker, *Fighting Computer Crime*, John Wiley and Sons, September 1998, ISBN 0471163783.

10.12. Frequently Asked Questions

Q: Is it more secure to compile driver support directly into the kernel, instead of making it a module?

A: Some people think it is better to disable the ability to load device drivers using modules, because an intruder could load a Trojan module or a module that could affect system security.

However, in order to load modules, you must be root. The module object files are also only writable by root. This means the intruder would need root access to insert a module. If the intruder gains root access, there are more serious things to worry about than whether he will load a module.

Modules are for dynamically loading support for a particular device that may be infrequently used. On server computers, or firewalls for instance, this is very unlikely to happen. For this reason, it would make more sense to compile support directly into the kernel for machines acting as servers. Modules are also slower than support compiled directly in the kernel.

Q: Why does logging in as root from a remote machine always fail?

A: See *Root Security*, page 89. This is done intentionally to prevent remote users from attempting to connect via telnet to your computer as root, which is a serious security vulnerability, because then the root password would be transmitted, in clear text, across the network. Don't forget: potential intruders have time on their side, and can run automated programs to find your password. Additionally, this is done to keep a clear record of who logged in, not just root.

Q: How can I enable the *Apache* SSL extensions?

A: Simply install the package `mod_ssl`, and consult the documentation at `mod_ssl` home page (www.modssl.org).



You should also consider the `mod_sxnet` module, which is a plugin for `mod_ssl` and allows the activation of the "Thawte Secure Extranet". `mod_ssl` encrypt communications, but `mod_ssl-sxnet` goes further and allows to securely authenticate the user of the web page thanks to a personal certificate. You have more info on this application on Thawte (<http://www.thawte.com/certs/strongextranet/>) or install the `mod_sxnet` module from your Mandrake distribution and read the included package documentation.

You might also try ZEDZ net (<http://www.zedz.net>) which has many pre-built packages, and is located outside of the United States.

Q: How can I manipulate user accounts, and still retain security?

A: Your **Mandrake Linux** distribution contains a great number of tools to change the properties of user accounts.

- The `pwconv` and `unpwconv` programs can be used to convert between shadowed and non-shadowed passwords.
- The `pwck` and `grpck` programs can be used to verify proper organization of the `/etc/passwd` and `/etc/group` files.
- The `useradd`, `usermod`, and `userdel` programs can be used to add, delete and modify user accounts. The `groupadd`, `groupmod`, and `groupdel` programs will do the same for groups.
- Group passwords can be created using `gpasswd`.

All these programs are "shadow-aware" – that is, if you enable shadow they will use `/etc/shadow` for password information, otherwise they won't.

Q: How can I password-protect specific HTML documents using *Apache*?

A: I bet you didn't know about <http://www.apacheweek.org> (<http://www.apacheweek.com>), did you?

You can find information on user authentication at <http://www.apacheweek.com/features/userauth> (<http://www.apacheweek.com/features/userauth>) as well as other web server-security tips from http://www.apache.org/docs/misc/security_tips.html (http://www.apache.org/docs/misc/security_tips.html)

10.13. Conclusion

By subscribing to the security alert mailing lists, and keeping current, you can do a lot towards securing your computer. If you pay attention to your log files and run something like tripwire regularly, you can do even more.

A reasonable level of computer security is not difficult to maintain on a home computer. More effort is required on business computers, but *GNU/Linux* can indeed be a secure platform. Due to the nature of *GNU/Linux* development, security fixes often come out much faster than they do on commercial operating systems, making *GNU/Linux* an ideal platform when security is a requirement.

Security-related terms

Included below are several of the most frequently used terms in computer security. A comprehensive dictionary of computer security terms is available in the LinuxSecurity.com Dictionary (<http://www.linuxsecurity.com/dictionary/>)

authentication

The process of knowing that the data received is the same as the data that was sent, and that the claimed sender is in fact the actual sender.

bastion Host

A computer system that must be highly secured because it is vulnerable to attack, usually because it is exposed to the Internet and is a main point of contact for users of internal networks. It gets its name from the highly fortified projects on the outer walls of medieval castles. Bastions overlook critical areas of defense, usually having strong walls, room for extra troops, and the occasional useful tub of boiling hot oil for discouraging attackers. Some reasonable definition here.

buffer overflow

Common coding style is to never allocate large enough buffers, and to not check for overflows. When such buffers overflow, the executing program (daemon or set-uid program) can be tricked in doing some other things. Generally this works by overwriting a function's return address on the stack to point to another location.

denial of service

An attack that consumes the resources on your computer for things it was not intended to be doing, thus preventing normal use of your network resources for legitimate purposes.

dual-homed Host

A general-purpose computer system that has at least two network interfaces.

firewall

A component or set of components that restricts access between a protected network and the Internet, or between other sets of networks.

host

A computer system attached to a network.

IP spoofing

IP Spoofing is a complex technical attack that is made up of several components. It is a security exploit that works by tricking computers in a trust relationship into thinking that you are someone that you really aren't. There is an extensive paper written by daemon9, route, and infinity in the Volume Seven, Issue Forty-Eight of Phrack Magazine.

non-repudiation

The property of a receiver being able to prove that the sender of some data did in fact send the data even though the sender might later deny ever having sent it.

packet

The fundamental unit of communication on the Internet.

packet filtering

The action a device takes to selectively control the flow of data to and from a network. Packet filters allow or block packets, usually while routing them from one network to another (most often from the Internet to an internal network, and vice-versa). To accomplish packet filtering, you set up rules that specify what types of packets (those to or from a particular IP address or port) are to be allowed and what types are to be blocked.

perimeter network

A network added between a protected network and an external network, in order to provide an additional layer of security. A perimeter network is sometimes called a DMZ.

proxy server

A program that deals with external servers on behalf of internal clients. Proxy clients talk to proxy servers, which relay approved client requests to real servers, and relay answers back to clients.

superuser

An informal name for root.

Chapter 11. Networking Overview

11.1. Copyright

This chapter is based on a *HOWTO* by Joshua D. Drake {POET} which original is hosted by the thelinuxreview.com (<http://www.thelinuxreview.com/>) web site.

The NET-3/4-HOWTO, NET-3, and Networking-HOWTO, information on how to install and configure networking support for Linux. Copyright (c) 1997 Terry Dawson, 1998 Alessandro Rubini, 1999 Joshua D. Drake {POET} - thelinuxreview.com/ (<http://www.thelinuxreview.com/>) is a FREE document. You may redistribute it under the terms of the GNU General Public License.

Modifications from v1.6.9, July 03, 2000, (C)opyright 2000 - 2002 MandrakeSoft

11.2. How to Use This Chapter

This document is organized top-down. The first sections include informative material and can be skipped if you are not interested; what follows is a generic discussion of networking issues, and you must ensure you understand this before proceeding to more specific parts. The rest, technology-specific information, is grouped in three main sections: Ethernet and IP-related information, technologies pertaining to widespread PC hardware and seldom-used technologies.

The suggested path through the document is thus the following:

Read the generic sections

These sections apply to every, or nearly every, technology described later and so are very important for you to understand. On the other hand, I expect many of the readers to be already confident with this material.

Consider your network

You should know how your network is, or will be, designed and exactly what hardware and technology types you will be implementing.

Read the *Ethernet Information*, page 125 section if you are directly connected to a LAN or the Internet:

This section describes basic Ethernet configuration and the various features that Linux offers for IP networks, like firewalling, advanced routing and so on.

Read the next section if you are interested in low-cost local networks or dial-up connections

The section describes PLIP, PPP, SLIP and ISDN, the widespread technologies used on personal workstations.

Read the technology-specific sections related to your requirements

If your needs differ from IP and/or common hardware, the final section covers details specific to non-IP protocols and peculiar communication hardware.

Do the configuration work

You should actually try to configure your network and take careful note of any problems you have.

Look for further help if needed

If you experience problems that this document does not help you to resolve then read the section related to where to get help or where to report bugs.

Have fun!

Networking is fun, enjoy it.

11.2.1. Conventions Used in This Document

No special convention is used here, but you must be warned about the way commands are shown. Following the classic *UNIX* documentation, any command you should type to your shell is prefixed by a prompt. This howto shows “user%” as the prompt for commands that do not require superuser privileges, and “root#” as the prompt for commands that need to be run as root. I chose to use “root#” instead of a plain “#” to prevent confusion with snapshots from shell scripts, where the hash mark is used to define comment lines.

When “Kernel Compile Options” are shown, they are represented in the format used by **menuconfig**. They should be understandable even if you (like me) are not used to **menuconfig**. If you are in doubt about the options’ nesting, running the program once can’t do anything, but help.

11.3. General Information About Linux Networking

11.3.1. Linux Networking Resources

There are a number of places where you can find good information about Linux networking.

There is a wealth of consultants available. A searchable listing can be found at [thelinuxreview.com/](http://www.thelinuxreview.com/) (<http://www.thelinuxreview.com/>) web site.

Alan Cox, the current maintainer of the Linux kernel networking code, maintains a world wide web page that contains highlights of current and new developments in Linux Networking at: [www.linux.org.uk](http://www.linux.org.uk/NetNews.html) (<http://www.linux.org.uk/NetNews.html>).

There is a newsgroup in the Linux news hierarchy dedicated to networking and related matters, it is: `comp.os.linux.networking` (`news:comp.os.linux.networking`)

There is a mailing list to which you can subscribe where you may ask questions relating to Linux networking. To subscribe you should send a mail message:

```
To: majordomo@vger.rutgers.edu
Subject: anything at all
Message:
subscribe linux-net
```

Please remember when reporting any problem to include as much relevant detail about the problem as you can. Specifically you should identify the versions of software that you are using, especially the kernel version, the version of tools such as `pppd/` or `dip` and the exact nature of the problem you are experiencing. This means taking note of the exact syntax of any error message you receive and of any command you are issuing.

11.3.2. Where to Get Some Non Linux-Specific Network Information

If you are after some basic tutorial information on TCP/IP networking generally, then I recommend you take a look at the following documents:

TCP/IP Introduction

This document comes as both a text version (<ftp://athos.rutgers.edu/runet/tcp-ip-intro.doc>) and a postscript version (<ftp://athos.rutgers.edu/runet/tcp-ip-intro.ps>).

TCP/IP Administration

This document comes both as a text version (<ftp://athos.rutgers.edu/runet/tcp-ip-admin.doc>) and a postscript version (<ftp://athos.rutgers.edu/runet/tcp-ip-admin.ps>).

If you are looking for some more detailed information on TCP/IP networking, then I highly recommend:

“**Internet working with TCP/IP, Volume 1: principles, protocols and architecture**, by Douglas E. Comer, ISBN 0-13-227836-7, Prentice Hall publications, Third Edition, 1995.”

If you want to learn about how to write network applications in a *UNIX*-compatible environment, then I also highly recommend:

“ **Unix Network Programming**, by W. Richard Stevens, ISBN 0-13-949876-1, Prentice Hall publications, 1990.”

A second edition of this book is appearing on the bookshelves; the new book is made up of three volumes: check Prentice-Hall’s web site (<http://www.phptr.com/>) for more information.

You might also try the `comp.protocols.tcp-ip` (`news:comp.protocols.tcp-ip`) newsgroup.

An important source of specific technical information relating to the Internet and the TCP/IP suite of protocols are RFC’s. RFC is an acronym for “Request For Comment” and is the standard means of submitting and documenting Internet protocol standards. There are many RFC repositories. Many of these sites are ftp sites and others provide World Wide Web access with an associated search engine that allows you to search the RFC database for particular keywords.

One possible source for RFC’s is at Nexor RFC database (<http://pubweb.nexor.co.uk/public/rfc/index/rfc.html>).

11.4. Generic Network Configuration Information

You will need to know and understand the following subsections before you actually try to configure your network. They are fundamental principles that apply regardless of the exact nature of the network you wish to deploy.

11.4.1. What do I Need to Start?

Before you start building or configuring your network, you will need some things. The most important of these are:

11.4.1.1. Current Kernel Source (Optional)



Your **Mandrake Linux** distribution comes with networking enabled, therefore it may not be required to recompile the kernel. If you are running well known hardware you should be just fine. For example: 3COM NIC, NE2000 NIC, or an Intel NIC. However if you find yourself in the position that you do need to update the kernel, the following information is provided.

Because the kernel you are running now might not yet have support for the network types or cards that you wish to use, you will probably need the kernel source so that you can recompile the kernel with the appropriate options.

However, as long as you stay within the mainstream of hardware, there should be no need to recompile your kernel unless there is a very specific feature that you need.

You can always obtain the latest kernel source from [sunsite.unc.edu](ftp://sunsite.unc.edu/pub/linux/kernel.org/pub/linux/kernel/) (<ftp://sunsite.unc.edu/pub/linux/kernel.org/pub/linux/kernel/>). This is not the official site, but they have LOTS of bandwidth and capacity. The official site is kernel.org but please use the above if you can. Please remember that [ftp.kernel.org](http://kernel.org) is seriously overloaded. Use a mirror.

Normally, the kernel source will be untarred into the `/usr/src/linux` directory. For information on how to apply patches and build the kernel, you should read the Kernel-HOWTO (<http://linuxdoc.org/HOWTO/Kernel-HOWTO.html>). For information on how to configure kernel modules you should read the “Modules mini-HOWTO”. Also, the README file found in the kernel sources and the Documentation directory are very informative for the brave reader.

Unless specifically stated otherwise, I recommend you stick with the standard kernel release (the one with the even number as the second digit in the version number). Development release kernels (the ones with the odd second digit) may have structural or other changes that may cause problems working with the other software on your system. If you are uncertain that you could resolve those sorts of problems in addition to the potential for there being other software errors, then don’t use them.

11.4.1.2. IP Addresses, an Explanation

Internet Protocol Addresses are composed of four bytes. The convention is to write addresses in what is called “dotted decimal notation”. In this form, each byte is converted to a decimal number, (0-255), dropping any leading zeroes unless the number is zero and written with each byte separated by a ‘.’ character. By convention each interface of a host or router has an IP address. It is legal for the same IP address to be used on each interface of a single machine in some circumstances but usually, each interface will have its own address.

Internet Protocol Networks are contiguous sequences of IP addresses. All addresses within a network have a number of digits within the address in common. The portion of the address that is common amongst all addresses within the network is called the “network portion” of the address. The remaining digits are called the “host portion”. The number of bits that are shared by all addresses within a network is called the netmask and it is the latter’s role to determine which addresses belong to the network it is applied to and which don’t. For example, consider the following:

Host Address	192.168.110.23
Network Mask	255.255.255.0
Network Portion	192.168.110.
Host Portion	.23
Network Address	192.168.110.0
Broadcast Address	192.168.110.255

Any address that is “bitwise anded” with its netmask will reveal the address of the network it belongs to. The network address is therefore always the lowest numbered address within the range of addresses on the network and always has the host portion of the address coded all zeroes.

The broadcast address is a special one which every host on the network listens to, in addition to its own unique address. This address is the one that datagrams are sent to if every host on the network is meant to receive it. Certain types of data like routing information and warning messages are transmitted to the broadcast address so that every host on the network can receive it simultaneously. There are two commonly used standards for what the broadcast address should be. The most widely accepted one is to use the highest possible address on the network as the broadcast address. In the example above, this would be 192.168.110.255. For some reason, other sites have adopted the convention of using the network address as the broadcast address. In practice, it doesn’t matter very much which you use but you must make sure that every host on the network is configured with the same broadcast address.

For administrative reasons, some time early in the development of the IP protocol, some arbitrary groups of addresses were formed into networks and these networks were grouped into what are called classes. These classes provide a number of standard size networks that could be allocated. The ranges allocated are:

Network Class	Netmask	Network Addresses
A	255.0.0.0	0.0.0.0 - 127.255.255.255
B	255.255.0.0	128.0.0.0 - 191.255.255.255
C	255.255.255.0	192.0.0.0 - 223.255.255.255
Multicast	240.0.0.0	224.0.0.0 - 239.255.255.255

What addresses you should use depends on exactly what it is that you are doing. You may have to use a combination of the following activities to get all the addresses you need:

Installing a Linux machine on an existing IP network

If you wish to install a Linux machine onto an existing IP network, then you should contact whoever administers the network and ask them for the following information:

- Host IP Address
- IP Network Address
- IP Broadcast Address
- IP Netmask

- Router Address
- Domain Name Server Address

You should then configure your Linux network device with those details. You can not make them up and expect your configuration to work.

Building a brand new network that will never connect to the Internet

If you are building a private network and you never intend that network to be connected to the Internet then you can choose whatever addresses you like. However, for safety and consistency reasons there have been some IP network addresses that have been reserved specifically for this purpose. These are specified in RFC1597 and are as follows:

Network Class	Netmask	Network Addresses
A	255.0.0.0	10.0.0.0 - 10.255.255.255
B	255.255.0.0	172.16.0.0 - 172.31.255.255
C	255.255.255.0	192.168.0.0 - 192.168.255.255

Table 11-1. Reserved Private Network Allocations

You should first decide how large you want your network to be, and then choose as many addresses as you require.

11.4.2. Routing

Routing is a big topic. It is easily possible to write large volumes of text about it. Most of you will have fairly simple routing requirements, some of you will not. I will cover some basic fundamentals of routing only. If you are interested in more detailed information, then I suggest you refer to the references provided at the start of the document.

Let's start with a definition. What is IP routing? Here is one that I'm using:

"IP routing is the process by which a host with multiple network connections decides where to deliver IP datagrams it has received."

It might be useful to illustrate this with an example. Imagine a typical office router, it might have a PPP link off the Internet, a number of Ethernet segments feeding the workstations and another PPP link off to another office. When the router receives a datagram on any of its network connections, routing is the mechanism that it uses to determine which interface it should send the datagram to next. Simple hosts also need to route, all Internet hosts have two network devices, one is the loopback interface described above and the other is the one it uses to talk to the rest of the network, perhaps an Ethernet, perhaps a PPP or SLIP serial interface.

Ok, so how does routing work? Each host keeps a special list of routing rules, called a routing table. This table contains rows which typically contain at least three fields: the first is a destination address, the second is the name of the interface to which the datagram is to be routed, and the third is optionally the IP address of another machine which will carry the datagram on its next step through the network. With Linux, you can see this table by using the following command:

```
user% cat /proc/net/route
```

or by using either one of the following commands:

```
user% /sbin/route -n
user% netstat -r
```

The routing process is fairly simple: an incoming datagram is received, the destination address (who it is for) is examined and compared with each entry in the table. The entry that best matches that address is selected

and the datagram is forwarded to the specified interface. If the gateway field is filled, then the datagram is forwarded to that host via the specified interface. Otherwise, the destination address is assumed to be on the network supported by the interface.

11.4.2.1. What Does The Routed Program Do?

The routing configuration described above is best suited for simple network arrangements where there is only one possible path to a determined destination. When you have a more complex network arrangement, things get a little more complicated. Fortunately for most of you, this won't be an issue.

The big problem with "manual routing" or "static routing" as described, is that if a machine or link fails in your network then the only way you can direct your datagrams another way, if another way exists, is by manually intervening and executing the appropriate commands. Naturally this is clumsy, slow, impractical and hazard prone. Various techniques have been developed to automatically adjust routing tables in the event of network failures where there are alternate routes. All of these techniques are loosely grouped by the term "dynamic routing protocols".

You may have heard of some of the more common dynamic routing protocols. The most common are probably RIP (Routing Information Protocol) and OSPF (Open Shortest Path First Protocol). The Routing Information Protocol is very common on small networks such as small-medium sized corporate networks or building networks. OSPF is more modern and more capable of handling large network configurations and better suited to environments where there is a large number of possible paths through the network. Common implementations of these protocols are: **routed** - RIP and **gated** - RIP, OSPF and others. The routed program is normally supplied with your Linux distribution or is included in the "NetKit" package detailed above.

An example of where and how you might use a dynamic routing protocol might look something like figure 11-1.

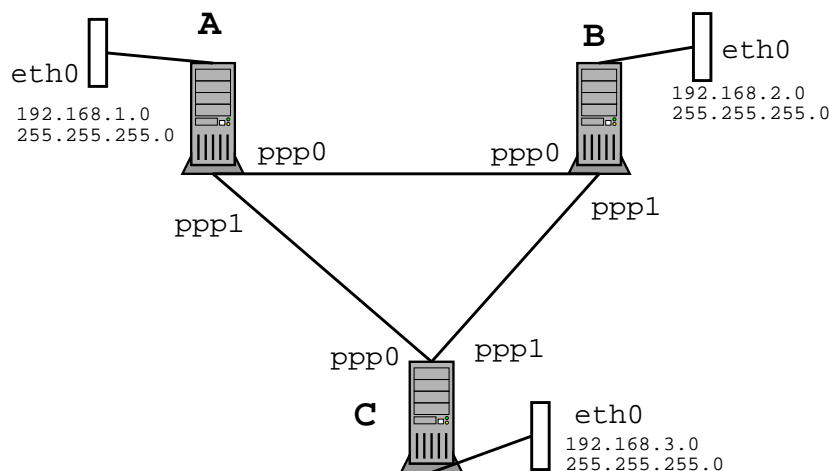


Figure 11-1. A Dynamic Routing Example

We have three routers A, B and C. Each one supports one Ethernet segment with a Class C IP network (netmask 255.255.255.0). Each router also has a PPP link to each of the other routers. The network forms a triangle.

It should be clear that the routing table at router A could look like:

```
root# route add -net 192.168.1.0 netmask 255.255.255.0 eth0
root# route add -net 192.168.2.0 netmask 255.255.255.0 ppp0
root# route add -net 192.168.3.0 netmask 255.255.255.0 ppp1
```

This would work just fine until the link between router A and B should fail. If that link failed, then with the routing entry shown above, hosts on the Ethernet segment of A could not reach hosts on the Ethernet segment on B because their datagram would be directed to router A's ppp0 link which is broken. They could still continue to talk to hosts on the Ethernet segment of C and hosts on the C's Ethernet segment could still talk to hosts on B's Ethernet segment, because the link between B and C is still intact.

But wait, if A can talk to C and C can still talk to B, why shouldn't A route its datagrams for B via C and let C send them to B? This is exactly the sort of problem that dynamic routing protocols like RIP were designed to solve. If each of the routers A, B and C were running a routing daemon, then their routing tables would be automatically adjusted to reflect the new state of the network should any one of the links in the network fail. To configure such a network is simple, for each router, you only need to do two things. In this case for router A:

```
root# route add -net 192.168.1.0 netmask 255.255.255.0 eth0
root# /usr/sbin/routed
```

The **routed** routing daemon automatically finds all active network ports when it starts and sends and listens for messages on each of the network devices to allow it to determine and update the routing table on the host.

This has been a very brief explanation of dynamic routing and where you would use it. If you want more information, then you should refer to the suggested references listed at the top of the document.

The important points relating to dynamic routing are:

1. You only need to run a dynamic routing protocol daemon when your Linux machine has the possibility of selecting multiple possible routes to a destination. An example of this would be if you plan to use IP Masquerading.
2. The dynamic routing daemon will automatically modify your routing table to adjust to changes in your network.
3. RIP is suited for small-to-medium size networks.

11.5. Ethernet Information

This section covers information specific to Ethernet and the configuring of Ethernet cards.

11.5.1. Supported Ethernet Cards

11.5.1.1. 3Com

- 3Com 3c501 - (avoid like the plague) (3c501 driver);
- 3Com 3c503 (3c503 driver), 3c505 (3c505 driver), 3c507 (3c507 driver), 3c509/3c509B (ISA) / 3c579 (EISA);
- 3Com Etherlink III Vortex Ethercards (3c590, 3c592, 3c595, 3c597) (PCI), 3Com Etherlink XL Boomerang (3c900, 3c905) (PCI) and Cyclone (3c905B, 3c980) Ethercards (3c59x driver) and 3Com Fast EtherLink Ethercard (3c515) (ISA) (3c515 driver);
- 3Com 3ccfe575 Cyclone Cardbus (3c59x driver);
- 3Com 3c575 series Cardbus (3c59x driver) (most PCMCIA cards should be detected)

11.5.1.2. AMD, ATT, Allied Telesis, Ansel, Apricot

- AMD LANCE (79C960) / PCnet-ISA/PCI (AT1500, HP J2405A, NE1500/NE2100);
- ATT GIS WaveLAN;
- Allied Telesis AT1700;
- Allied Telesis LA100PCI-T;
- Allied Telesyn AT2400T/BT ("ne" module);
- Ansel Communications AC3200 (EISA);
- Apricot Xen-II / 82596.

11.5.1.3. Cabletron, Cogent, Crystal LAN

- Cabletron E21xx;
- Cogent EM110;
- Crystal LAN CS8920, Cs8900.

11.5.1.4. Danpex, DEC, Digi, DLink

- Danpex EN-9400;
- DEC DE425 (EISA) / DE434/DE435 (PCI) / DE450/DE500 (DE4x5 driver);
- DEC DE450/DE500-XA (dc21x4x) (Tulip driver);
- DEC DEPCA and EtherWORKS;
- DEC EtherWORKS 3 (DE203, DE204, DE205);
- DECchip DC21x4x "Tulip";
- DEC QSilver's (Tulip driver);
- Digi International RightSwitch;
- DLink DE-220P, DE-528CT, DE-530+, DFE-500TX, DFE-530TX.

11.5.1.5. Fujitsu, HP, ICL, Intel

- Fujitsu FMV-181/182/183/184;
- HP PCLAN (27245 and 27xxx series);
- HP PCLAN PLUS (27247B and 27252A);
- HP 10/100VG PCLAN (J2577, J2573, 27248B, J2585) (ISA/EISA/PCI);
- ICL EtherTeam 16i / 32 (EISA);
- Intel EtherExpress;
- Intel EtherExpress Pro.

11.5.1.6. KTI, Macromate, NCR NE2000/1000, Netgear, New Media

- KTI ET16/P-D2, ET16/P-DC ISA (work jumperless and with hardware-configuration options);
- Macromate MN-220P (PnP or NE2000 mode);
- NCR WaveLAN;
- NE2000/NE1000 (be careful with clones);
- Netgear FA-310TX (Tulip chip);
- New Media Ethernet.

11.5.1.7. PureData, SEEQ, SMC

- PureData PDUC8028, PDI8023;
- SEEQ 8005;
- SMC Ultra / EtherEZ (ISA);
- SMC 9000 series;
- SMC PCI EtherPower 10/100 (DEC Tulip driver);
- SMC EtherPower II (epic100.c driver).

11.5.1.8. Sun Lance, Sun Intel, Schneider, WD, Zenith, IBM, Enyx

- Sun LANCE adapters (kernel 2.2 and newer);
- Sun Intel adapters (kernel 2.2 and newer);
- Schneider and Koch G16;
- Western Digital WD80x3;
- Zenith Z-Note / IBM ThinkPad 300 built-in adapter;
- Znyx 312 etherarray (Tulip driver);

11.5.2. General Ethernet Information

Ethernet device names are `eth0`, `eth1`, `eth2` etc. The first card detected by the kernel is assigned `eth0` and the rest are assigned sequentially in the order they are detected.

Once you have your kernel properly built to support your Ethernet card then configuration of the card is easy.

Typically you would use something like (which most distributions already do for you, if you configured them to support your Ethernet):

```
root# ifconfig eth0 192.168.0.1 netmask 255.255.255.0 up
root# route add -net 192.168.0.0 netmask 255.255.255.0 eth0
```

Most of the Ethernet drivers were developed by Donald Becker (<mailto:becker@CESDIS.gsfc.nasa.gov>)

11.5.3. Using 2 or More Ethernet Cards in The Same Machine

The module will typically detect all of the installed cards.

Detection information is stored in the `/etc/conf.modules` file.

Consider that a user has 3 NE2000 cards, one at 0x300, one at 0x240, and one at 0x220. You would add the following lines to the `/etc/conf.modules` file:

```
alias eth0 ne
alias eth1 ne
alias eth2 ne
options ne io=0x220,0x240,0x300
```

What this does is tell the **modprobe** program to look for 3 NE based cards at the following addresses. It also states in which order they should be found and the device they should be assigned.

Most ISA modules can take multiple comma separated I/O values. For example:

```
alias eth0 3c501
alias eth1 3c501
options eth0 -o 3c501-0 io=0x280 irq=5
```

```
options eth1 -o 3c501-1 io=0x300 irq=7
```

The `-o` option allows for a unique name to be assigned to each module. The reason for this is that you can not have two copies of the same module loaded.

The `irq=` option is used to specify the hardware IRQ and the `io=` to specify the different io ports.

By default, the Linux kernel only probes for one Ethernet device, you need to pass command-line arguments to the kernel in order to force detection of further boards.

To learn how to make your Ethernet card(s) working under Linux you should refer to the Ethernet-HOWTO (<http://linuxdoc.org/HOWTO/Ethernet-HOWTO.html>).

11.6. IP-Related Information

These sections cover information specific to IP.

11.6.1. DNS

DNS stands for Domain Name System. It is the system responsible for mapping a machine name as `www.mandrakesoft.com` with the IP address of that machine, in this case: `216.71.116.162` at the time of writing. With DNS, mapping is available in both directions; from name to IP and vice-versa.

The DNS is composed of a great number of machines all over the Internet responsible for a certain number of names. Each machine is attributed a DNS server to which it can ask to map a particular name with its address. If that server does not have the answer, then it asks to another one and so on. You can also have a local DNS responsible for mapping addresses on your LAN.

We can differentiate two major DNS classes: caching DNS and master DNS server. The first one only “remembers” a previous request and then can answer it without asking a master DNS server once more. The latter servers are really responsible as a last resort to map an address with a name - or possibly tell that this name does not map any address.

11.6.2. DHCP and DHCPD

DHCP is an acronym for Dynamic Host Configuration Protocol. The creation of DHCP has made configuring the network on multiple hosts extremely simple. Instead of having to configure each host separately, you can assign all of the commonly used parameters by the hosts using a DHCP server.

Each time the host boots up it will broadcast a packet to the network. This packet is a call to any DHCP servers located on the same segment to configure the host.

DHCP is extremely useful in assigning items such as the IP address, netmask, and gateway of each host.

11.7. Using Common PC Hardware

11.7.1. ISDN

The Integrated Services Digital Network (ISDN) is a series of standards that specify a general purpose-switched digital data network. An ISDN “call” creates a synchronous point-to-point data service to the destination. ISDN is generally delivered on a high speed link that is broken down into a number of discrete channels. There are two different types of channels, the “B Channels” which will actually carry the user data, and a single channel called the “D channel” which is used to send control information to the ISDN exchange to establish calls and other functions. In Australia for example, ISDN may be delivered on a 2Mbps link that is broken into 30 discrete 64kbps B channels with one 64kbps D channel. Any number of channels may be used at a time and in any combination. You could for example establish 30 separate calls to 30 different destinations at 64kbps each, or you could establish 15 calls to 15 different destinations at 128kbps each (two channels used per call), or just a small number of calls and leave the rest idle. A channel may be used for either incoming or outgoing calls. The original intention of ISDN was to allow Telecommunications companies to provide a

single data service which could deliver either telephone (via digitized voice) or data services to your home or business without requiring you to make any special configuration changes.

There are a few different ways to connect your computer to an ISDN service. One way is to use a device called a “Terminal Adaptor” which plugs into the Network Terminating Unit that your telecommunications carrier will have installed when you got your ISDN service and presents a number of serial interfaces. One of the latter is used to enter commands to establish calls and configuration and the others are actually connected to the network devices that will use the data circuits when they are established. Linux will work in this sort of configuration without modification, you just treat the port on the Terminal Adaptor like you would treat any other serial device. Another way, which is the way the kernel ISDN support is designed for, allows you to install an ISDN card into your Linux machine and then has your Linux software handle the protocols and make the calls itself.

Kernel Compile Options:

```
ISDN subsystem --->
<*> ISDN support
[ ] Support synchronous PPP
[ ] Support audio via ISDN
< > ICN 2B and 4B support
< > PCBIT-D support
< > Teles/NICCY1016PC/Creatix support
```

The Linux implementation of ISDN supports a number of different types of internal ISDN cards. Hereunder are those listed in the kernel configuration options:

- ICN 2B and 4B;
- Octal PCBIT-D;
- Teles ISDN-cards and compatibles.

Some of these cards require software to be downloaded in order to make them operational. There is a separate utility to do this with.

Full details on how to configure the Linux ISDN support is available from the `/usr/src/linux/Documentation/isdn/` directory and a FAQ dedicated to **isdn4linux** is available at [www.lrz-muenchen.de](http://www.lrz-muenchen.de/~ui161ab/www/isdn/) (<http://www.lrz-muenchen.de/~ui161ab/www/isdn/>). (You can click on the English flag to get an English version).



About PPP. The PPP suite of protocols will operate over either asynchronous or synchronous serial lines. The commonly distributed PPP daemon for Linux, `pppd`, supports only asynchronous mode. If you wish to run the PPP protocol over your ISDN service, you need a specially modified version. Details of where to find it are available in the documentation referred to above.

11.7.2. PLIP

During development of the 2.1 kernel versions, support for the parallel port was changed to a better setup.

Kernel Compile Options:

```
General setup --->
[*] Parallel port support
Network device support --->
<*> PLIP (parallel port) support
```

The new code for PLIP behaves like the old one. Use the same **ifconfig** and **route** commands as in the previous section, but initialization of the device is different due to the advanced parallel port support.

The “first” PLIP device is always called `plip0`, where first is the first device detected by the system, similarly to what happens for Ethernet devices. The actual parallel port being used is one of the available ports, as

shown in `/proc/parport`. For example, if you have only one parallel port, you'll only have a directory called `/proc/parport/0`.

If your kernel didn't detect the IRQ number used by your port, `insmod plip` will fail; in this case, just write the right number to `/proc/parport/0/irq` and reinvoke **insmod**.

Complete information about parallel port management is available in the `Documentation/parport.txt` file, part of your kernel sources.

11.7.3. PPP

Due to the nature of PPP, its size, complexity, and flexibility it has been moved to its own HOWTO. The PPP-HOWTO is still a Linux Documentation Project document (<http://www.linuxdoc.org>) but its official home is at the [thelinuxreview.com](http://www.thelinuxreview.com) web site (<http://www.thelinuxreview.com>), PPP section (<http://www.thelinuxreview.com/howto/ppp>).

11.8. Other Network Technologies

The following subsections are specific to particular network technologies. The information contained in these sections does not necessarily apply to any other type of network technology. The topics are sorted alphabetically.

11.8.1. ARCNet

ARCNet device names are `arc0e`, `arc1e`, `arc2e` etc. or `arc0s`, `arc1s`, `arc2s` etc. The first card detected by the kernel is assigned `arc0e` or `arc0s` and the rest are assigned sequentially in the order they are detected. The letter at the end signifies whether you selected the Ethernet encapsulation packet format or the RFC1051 packet format.

Kernel Compile Options:

```
Network device support --->
  [*] Network device support
  <*> ARCnet support
    [ ] Enable arc0e (ARCnet "Ether-Encap" packet format)
    [ ] Enable arc0s (ARCnet RFC1051 packet format)
```

Once you have your kernel properly built to support your Ethernet card then configuration of the card is easy. Typically you would use something like:

```
root# ifconfig arc0e 192.168.0.1 netmask 255.255.255.0 up
root# route add -net 192.168.0.0 netmask 255.255.255.0 arc0e
```

Please refer to the `/usr/src/linux/Documentation/networking/arcnet.txt` and `/usr/src/linux/Documentation/networking/arcnet-hardware.txt` files for further information.

ARCNet support was developed by Avery Pennarun, apenwarr@foxnet.net.

11.8.2. Appletalk (AF_APPLETALK)

The Appletalk support has no special device names as it uses existing network devices.

Kernel Compile Options:

```
Networking options --->
  <*> Appletalk DDP
```

Appletalk support allows your Linux machine to interwork with Apple networks. An important use for this is to share resources such as printers and disks between both your Linux and Apple computers. Additional

software is required, this is called **netatalk**. Wesley Craig netatalk@umich.edu represents a team called the “Research Systems Unix Group” at the University of Michigan and they have produced the **netatalk** package which provides software that implements the Appletalk protocol stack and some useful utilities. The **netatalk** package will either have been supplied with your Linux distribution, or you will have to ftp it from its home site at the University of Michigan (<ftp://terminator.rs.itd.umich.edu/unix/netatalk/>)

To build and install the package, you should do:

```
user% tar xvfz ../netatalk-1.4b2.tar.Z
user% make
root# make install
```

You may want to edit the “Makefile” before calling **make** to actually compile the software. Specifically, you might want to change the DESTDIR variable which defines where the files will be installed later. The default of /usr/local/atalk is fairly safe.

11.8.2.1. Configuring The Appletalk Software

The first thing you need to do to make it all work is to ensure that the appropriate entries in the /etc/services file are present. The entries you need are:

```
rtmp 1/ddp # Routing Table Maintenance Protocol
nbp 2/ddp # Name Binding Protocol
echo 4/ddp # AppleTalk Echo Protocol
zip 6/ddp # Zone Information Protocol
```

The next step is to create the Appletalk configuration files in the /usr/local/atalk/etc directory (or wherever you installed the package).

The first file to create is the /usr/local/atalk/etc/atalkd.conf file. Initially, this file needs only one line that gives the name of the network device that supports the network that your Apple machines are on:

```
eth0
```

The Appletalk daemon program will add extra details after it is run.

11.8.2.2. Exporting a Linux Filesystem Via Appletalk

You can export filesystems from your Linux machine to the network so that Apple machines on the network can share them.

To do this you must configure the /usr/local/atalk/etc/AppleVolumes.system file. There is another configuration file called /usr/local/atalk/etc/AppleVolumes.default which has exactly the same format and describes which filesystems users connecting with guest privileges will receive.

Full details on how to configure these files and what the various options are can be found in the `afpd` man page: `afpd`.

A simple example might look like...

```
/tmp Scratch
/home/ftp/pub "Public Area"
```

... which would export your /tmp filesystem as AppleShare Volume “Scratch” and your ftp public directory as AppleShare Volume “Public Area”. The volume names are not mandatory, the daemon will choose some for you, but it won’t hurt to specify them anyway.

11.8.2.3. Sharing Your Linux Printer Across Appletalk

You can share your Linux printer with your Apple machines quite simply. You need to run the **papd** program which is the Appletalk Printer Access Protocol Daemon. When you run this program, it will accept requests from your Apple machines and spool the print jobs to your local line printer daemon for printing.

You need to edit the `/usr/local/ataalk/etc/papd.conf` file to configure the daemon. The syntax of this file is the same as that of your usual `/etc/printcap` file. The name you give to the definition is registered with the Appletalk naming protocol, NBP.

A sample configuration might look like...

```
TricWriter:\  
:pr=lp:op=cg:
```

... which would make a printer named “TricWriter” available to your Appletalk network, and all accepted jobs would be printed to the `lp` linux printer (as defined in the `/etc/printcap` file), using **lpd**. The entry `op=cg` says that the linux user `cg` is the printer operator.

11.8.2.4. Starting The AppleTalk Software

Ok, you should now be ready to test this basic configuration. There is an **rc.ataalk** file supplied with the **netatalk** package that should work well for you, so all you should have to do is:

```
root# /usr/local/ataalk/etc/rc.ataalk
```

and all should start up and run correctly. You should see no error messages and the software will send messages to the console indicating each stage as it starts.

11.8.2.5. Testing The AppleTalk Software

To test that the software is functioning properly, go to one of your Apple machines, pull down the Apple menu, select the Chooser, click on AppleShare, and your Linux box should appear.

11.8.2.6. Caveats of The AppleTalk Software

1. You may need to start the Appletalk support before you configure your IP network. If you have problems starting the Appletalk programs, or if after you start them you have trouble with your IP network, then try starting the Appletalk software before you run your `/etc/rc.d/rc.inet1` file.
2. The **afpd** (Apple Filing Protocol Daemon) severely messes up your hard disk. Below the mount points, it creates a couple of directories called `.AppleDesktop` and `Network Trash Folder`. Then, for each directory you access, it will create a `.AppleDouble` below it so it can store resource forks, etc. So think twice before exporting /, you will have a great time cleaning up afterwards.
3. The **afpd** program expects clear text passwords from the Macs. Security could be a problem, so be very careful when you run this daemon on a machine connected to the Internet. You have yourself to blame if somebody nasty does something bad.
4. The existing diagnostic tools such as **netstat** and **ifconfig** don't support Appletalk. The raw information is available in the `/proc/net/` directory if you need it.

11.8.2.7. More Information

For a much more detailed description of how to configure Appletalk for Linux, refer to Anders Brownworth **Linux Netatalk-HOWTO** page on TheHamptons.com (<http://thehamptons.com/anders/netatalk/>) web site.

11.8.3. ATM

Werner Almesberger <werner.almesberger@lrc.di.epfl.ch> manages a project which goal is to provide Asynchronous Transfer Mode support for Linux. Current information on the status of the project may be obtained the ATM on Linux (<http://linux-atm.sourceforge.net/>) web site.

11.8.4. AX25 (AF_AX25)

AX.25 device names are sl0, sl1, etc. in 2.0.* kernels or ax0, 'ax1, etc. in 2.1.* kernels.

Kernel Compile Options:

```
Networking options --->
  [*] Amateur Radio AX.25 Level 2
```

The AX25, Netrom and Rose protocols are covered by the AX25-HOWTO (<http://linuxdoc.org/HOWTO/AX25-HOWTO.html>). These protocols are used by Amateur Radio Operators world wide in packet radio experimentation.

Most of the work for implementation of these protocols has been done by Jonathon Naylor, jsn@cs.nott.ac.uk, and the new *HOWTO* maintener is Jeff Tranter (tranter@pobox.com).

11.8.5. DECNet

Support for DECNet is now included in current stable kernel (2.4). With **Mandrake Linux**, it has been available since kernel 2.2.

11.8.6. FDDI

FDDI device names are fddi0, fddi1, fddi2 etc. The first card detected by the kernel is assigned fddi0 and the rest are assigned sequentially in the order they are detected.

Larry Stefani, lstefani@ultranet.com, has developed a driver for the Digital Equipment Corporation FDDI EISA and PCI cards.

Kernel Compile Options:

```
Network device support --->
  [*] FDDI driver support
  [*] Digital DEFEA and DEFPA adapter support
```

If your kernel is built to support the FDDI driver and installed, configuration of the FDDI interface is almost identical to that of an Ethernet interface. You just specify the appropriate FDDI interface name in the **ifconfig** and **route** commands.

11.8.7. Frame Relay

The Frame Relay device names are `dlci00`, `dlci01` etc for the DLCI encapsulation devices and `sdla0`, `sdla1` etc for the FRAD(s).

Frame Relay is a new networking technology that is designed to suit data communications traffic that is of a “bursty” or intermittent nature. You connect to a Frame Relay network using a Frame Relay Access Device (FRAD). The Linux Frame Relay supports IP over Frame Relay as described in RFC-1490.

Kernel Compile Options:

```
Network device support --->
  <*> Frame relay DLCI support (EXPERIMENTAL)
    (24) Max open DLCI
    (8)  Max DLCI per device
  <*> SDLA (Sangoma S502/S508) support
```

Mike McLagan, mike.mclagan@linux.org, developed the Frame Relay support and configuration tools.

Currently, the only FRAD I know of that are supported are the Sangoma Technologies (<http://www.sangoma.com/>) S502A, S502E and S508, as well as the Emerging Technologies (<http://www.etinc.com/>).

To configure the FRAD and DLCI devices after you have rebuilt your kernel, you will need the Frame Relay configuration tools. These are available from [ftp.invlogic.com](ftp://ftp.invlogic.com) (<ftp://ftp.invlogic.com/pub/linux/fr/>).

Compiling and installing the tools is straightforward, but the lack of a top level Makefile makes it a fairly manual process:

```
user% tar xvfz ../frad-0.15.tgz
user% cd frad-0.15
user% for i in common dlci frad; do make -C $i clean; make -C $i; done
root# mkdir /etc/frad
root# install -m 644 -o root -g root bin/*.sfm /etc/frad
root# install -m 700 -o root -g root frad/fradcfg /sbin
rppt# install -m 700 -o root -g root dlci/dlcicfg /sbin
```

Note that the previous commands use **sh** syntax, if you use a **cs**h flavor instead (like **tcsh**), the **for** loop will look different.

After installing the tools, you need to create an `/etc/frad/router.conf` file. You can use this template, which is a modified version of one of the example files:

```
# /etc/frad/router.conf
# This is a template configuration for frame relay.
# All tags are included. The default values are based on the code
# supplied with the DOS drivers for the Sangoma S502A card.
#
# A '#' anywhere in a line constitutes a comment
# Blanks are ignored (you can indent with tabs too)
# Unknown [] entries and unknown keys are ignored
#

[Devices]
Count=1                # number of devices to configure
Dev_1=sdla0             # the name of a device
Dev_2=sdla1             # the name of a device

# Specified here, these are applied to all devices and can be overridden for
# each individual board.
#
Access=CPE
Clock=Internal
KBaud=64
Flags=TX
#
# MTU=1500               # Maximum transmit IFrame length, default is 4096
# T391=10                # T391 value      5 - 30, default is 10
# T392=15                # T392 value      5 - 30, default is 15
# N391=6                 # N391 value      1 - 255, default is 6
# N392=3                 # N392 value      1 - 10, default is 3
```

```

# N393=4                # N393 value    1 - 10, default is 4

# Specified here, these set the defaults for all boards
# CIRfwd=16            # CIR forward   1 - 64
# Bc_fwd=16            # Bc forward   1 - 512
# Be_fwd=0             # Be forward   0 - 511
# CIRbak=16            # CIR backward 1 - 64
# Bc_bak=16            # Bc backward  1 - 512
# Be_bak=0             # Be backward  0 - 511

#
#
# Device specific configuration
#
#
#
# The first device is a Sangoma S502E
#
[sdla0]
Type=Sangoma            # Type of the device to configure, currently only
                        # SANGOMA is recognized
#
# These keys are specific to the 'Sangoma' type
#
# The type of Sangoma board - S502A, S502E, S508
Board=S502E
#
# The name of the test firmware for the Sangoma board
# Testware=/usr/src/frad-0.10/bin/sdla_tst.502
#
# The name of the FR firmware
# Firmware=/usr/src/frad-0.10/bin/frm_rel.502
#
Port=360                # Port for this particular card
Mem=C8                 # Address of memory window, A0-EE, depending on card
IRQ=5                  # IRQ number, do not supply for S502A
DLCIs=1                # Number of DLCI's attached to this device
DLCI_1=16              # DLCI #1's number, 16 - 991
# DLCI_2=17
# DLCI_3=18
# DLCI_4=19
# DLCI_5=20
#
# Specified here, these apply to this device only,
# and override defaults from above
#
# Access=CPE           # CPE or NODE, default is CPE
# Flags=TXIgnore,RXIgnore,BufferFrames,DropAborted,Stats,MCI,AutoDLCI
# Clock=Internal        # External or Internal, default is Internal
# Baud=128              # Specified baud rate of attached CSU/DSU
# MTU=2048              # Maximum transmit IFrame length, default is 4096
# T391=10               # T391 value    5 - 30, default is 10
# T392=15               # T392 value    5 - 30, default is 15
# N391=6                # N391 value    1 - 255, default is 6
# N392=3                # N392 value    1 - 10, default is 3
# N393=4                # N393 value    1 - 10, default is 4

#
# The second device is some other card
#
# [sdla1]
# Type=FancyCard        # Type of the device to configure.
# Board=                # Type of Sangoma board
# Key=Value             # values specific to this type of device

#
# DLCI Default configuration parameters
# These may be overridden in the DLCI specific configurations
#
CIRfwd=64              # CIR forward   1 - 64
# Bc_fwd=16            # Bc forward   1 - 512
# Be_fwd=0             # Be forward   0 - 511
# CIRbak=16            # CIR backward 1 - 64
# Bc_bak=16            # Bc backward  1 - 512
# Be_bak=0             # Be backward  0 - 511

```

```
#
# DLCI Configuration
# These are all optional. The naming convention is
# [DLCI_D<devicenum>_<DLCI_Num>]
#

[DLCI_D1_16]
# IP=
# Net=
# Mask=
# Flags defined by Sangoma: TXIgnore,RXIgnore,BufferFrames
# DLCIFlags=TXIgnore,RXIgnore,BufferFrames
# CIRfwd=64
# Bc_fwd=512
# Be_fwd=0
# CIRbak=64
# Bc_bak=512
# Be_bak=0

[DLCI_D2_16]
# IP=
# Net=
# Mask=
# Flags defined by Sangoma: TXIgnore,RXIgnore,BufferFrames
# DLCIFlags=TXIgnore,RXIgnore,BufferFrames
# CIRfwd=16
# Bc_fwd=16
# Be_fwd=0
# CIRbak=16
# Bc_bak=16
# Be_bak=0
```

When you have built your `/etc/frad/router.conf` file, the only step remaining is to configure the actual devices themselves. This is only a little trickier than a normal network device configuration, you need to remember to bring up the FRAD device before the DLCI encapsulation devices. These commands are best hosted in a shell script, due to their number:

```
#!/bin/sh
# Configure the frad hardware and the DLCI parameters
/sbin/fradcfg /etc/frad/router.conf || exit 1
/sbin/dlcicfg file /etc/frad/router.conf
#
# Bring up the FRAD device
ifconfig sdla0 up
#
# Configure the DLCI encapsulation interfaces and routing
ifconfig dlci00 192.168.10.1 pointopoint 192.168.10.2 up
route add -net 192.168.10.0 netmask 255.255.255.0 dlci00
#
ifconfig dlci01 192.168.11.1 pointopoint 192.168.11.2 up
route add -net 192.168.11.0 netmask 255.255.255.0 dlci00
#
route add default dev dlci00
#
```


11.8.8. IPX (AF_IPX)

The IPX protocol is most commonly utilized in Novell NetWare^(tm) local area network environments. Linux includes support for this protocol and may be configured to act as a network endpoint, or as a router for IPX.

Kernel Compile Options:

```
Networking options --->
  [*] The IPX protocol
  [ ] Full internal IPX network
```

The IPX protocol and the NCPFS are covered in greater depth in the IPX-HOWTO (<http://linuxdoc.org/HOWTO/IPX-HOWTO.html>).

11.8.9. NetRom (AF_NETROM)

NetRom device names are nr0, nr1, etc.

Kernel Compile Options:

```
Networking options --->
  [*] Amateur Radio AX.25 Level 2
  [*] Amateur Radio NET/ROM
```

The AX25, Netrom and Rose protocols are covered by the AX25-HOWTO (<http://linuxdoc.org/HOWTO/AX25-HOWTO.html>). These protocols are used by Amateur Radio Operators world wide in packet radio experimentation.

Most of the work for implementation of these protocols has been done by Jonathon Naylor, jsn@cs.nott.ac.uk.

11.8.10. Rose Protocol (AF_ROSE)

Rose device names are rs0, rs1, etc. in 2.1.* kernels. Rose is available since the 2.1.* kernels.

Kernel Compile Options:

```
Networking options --->
  [*] Amateur Radio AX.25 Level 2
  <*> Amateur Radio X.25 PLP (Rose)
```

The AX25, Netrom and Rose protocols are covered by the AX25-HOWTO (<http://linuxdoc.org/HOWTO/AX25-HOWTO.html>). These protocols are used by Amateur Radio Operators world wide in packet radio experimentation.

Most of the work for implementation of these protocols has been done by Jonathon Naylor, jsn@cs.nott.ac.uk.

11.8.11. Samba - NetBEUI, NetBios, CIFS support.

Samba is an implementation of the Session Management Block protocol (SMB). It allows *Windows* and other systems to mount and use your disks and printers.

Samba and its configuration are covered in detail in the SMB-HOWTO (<http://linuxdoc.org/HOWTO/SMB-HOWTO.html>).

11.8.12. STRIP Support (Starmode Radio IP)

STRIP device names are 'st0', 'st1', etc.

Kernel Compile Options:

```
Network device support --->
[*] Network device support
....
[*] Radio network interfaces
< > STRIP (Metricom starmode radio IP)
```

STRIP is a protocol designed specifically for a range of Metricom radio modems for a research project being conducted by Stanford University called the MosquitoNet Project (<http://mosquitonet.Stanford.edu>). There is a lot of interesting reading here, even if you aren't directly interested in the project.

The Metricom radios connect to a serial port, employ spread spectrum technology and are typically capable of about 100kbps. Information on the Metricom radios is available from the Metricom Web Server (<http://www.metricom.com/>).



Metricom went bankrupt, but another company might buy the technology and re-activate the web site.

At present, the standard network tools and utilities do not support the STRIP driver, so you will have to download some customized tools from the MosquitoNet web server. Details on what software you need is available at the MosquitoNet Software Page (<http://mosquitonet.Stanford.edu/software.html>).

A summary of configuration is that you use a modified **slattach** program to set the line discipline of a serial tty device to STRIP and then configure the resulting st [0-9] device as you would for Ethernet with one important exception: for technical reasons, STRIP does not support the ARP protocol, so you must manually configure the ARP entries for each of the hosts on your subnet. This should not prove too onerous.

11.8.13. Token Ring

Token ring device names are 'tr0', 'tr1' etc. Token Ring is an IBM standard LAN protocol that avoids collisions by providing a mechanism that gives only one station on the LAN the right to transmit at a time. A "token" is held by one station at a time and the station holding the token is the only station allowed to transmit. When it has transmitted its data, it passes the token onto the next station. The token loops amongst all active stations, hence the name "Token Ring".

Kernel Compile Options:

```
Network device support --->
[*] Network device support
....
[*] Token Ring driver support
< > IBM Tropic chipset based adaptor support
```

Configuration of Token Ring is identical to that of Ethernet, with the exception of the network device name to configure.

11.8.14. X.25

X.25 is a circuit-based packet switching protocol defined by the C.C.I.T.T. (a standards body recognized by Telecommunications companies in most parts of the world). An implementation of X.25 and LAPB are being worked on and recent kernels (from 2.1.*) include the work in progress.

Jonathon Naylor <jsn@cs.nott.ac.uk> is leading the development and a mailing list has been established to discuss Linux X.25-related matters. To subscribe, send a message to majordomo@vger.rutgers.edu with the text subscribe linux-x25 in the body of the message.

11.8.15. WaveLan Card

WaveLan device names are `eth0`, `eth1`, etc.

Kernel Compile Options:

```
Network device support --->
[*] Network device support
....
[*] Radio network interfaces
....
<*> WaveLAN support
```

The WaveLAN card is a spread spectrum wireless LAN card. The card looks very much like an Ethernet card in practice, and is configured in much the same way.

You can get information on the WaveLan card on the ORiNOCCO (<http://www.orinocowireless.com/>) web site.

11.9. Cables And Cabling

Those of you handy with a soldering iron may want to build your own cables to interconnect two Linux machines. The following cabling diagrams should assist you in this.

11.9.1. Serial NULL Modem cable

Not all NULL modem cables are alike. Many null modem cables do little more than trick your computer into thinking all the appropriate signals are present and swap transmit and receive data. This is ok, but it means you must use software flow control (XON/XOFF) which is less efficient than hardware flow control. The following cable provides the best possible signalling between machines and allows you to use hardware (RTS/CTS) flow control.

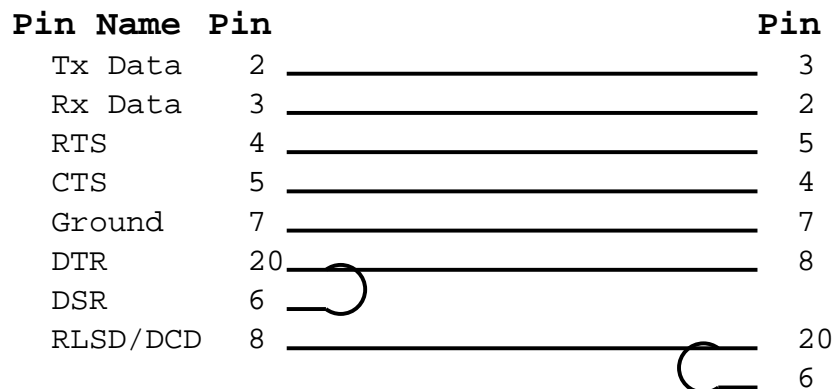


Figure 11-2. The NULL-Modem Cabling

11.9.2. Parallel Port Cable (PLIP Cable)

If you intend to use the PLIP protocol between two machines, then this cable will work for you notwithstanding what sort of parallel ports you have installed.

Pin Name	pin	pin
STROBE	1*	
D0->ERROR	2	15
D1->SLCT	3	13
D2->PAPOUT	4	12
D3->ACK	5	10
D4->BUSY	6	11
D5	7*	
D6	8*	

D7	9*	
ACK->D3	10	----- 5
BUSY->D4	11	----- 6
PAPOUT->D2	12	----- 4
SLCT->D1	13	----- 3
FEED	14*	
ERROR->D0	15	----- 2
INIT	16*	
SLCTIN	17*	
GROUND	25	----- 25

Notes:

- do not connect the pins marked with an asterisk “*”;
- extra grounds are 18,19,20,21,22,23 and 24;
- if the cable you are using has a metallic shield, it should be connected to the metallic DB-25 shell at **one end only**.



A badly wired PLIP cable can destroy your controller card. Be very careful and double-check every connection to ensure you don't cause yourself any unnecessary work or heartache.

While you may be able to run PLIP cables for long distances, you should avoid it if you can. The specifications for the cable allow for a cable length of about 1 meter or so. Please be very careful when running long PLIP cables as sources of strong electromagnetic fields such as lightning, power lines and radio transmitters can interfere with and sometimes even damage your controller. If you really want to connect two of your computers over a large distance, you really should be looking at obtaining a pair of thin-net Ethernet cards and running some coaxial cable.

11.9.3. 10base2 (Thin Coax) Ethernet Cabling

10base2 is an Ethernet cabling standard that specifies the use of 50 ohm coaxial cable with a diameter of about 5 millimeters. There are a couple of important rules to remember when interconnecting machines with 10base2 cabling. The first is that you must use terminators at **both ends** of the cabling. A terminator is a 50 ohm resistor that helps to ensure that the signal is absorbed and not reflected when it reaches the end of the cable. Without a terminator at each end of the cabling, you may find that the Ethernet is unreliable or doesn't work at all. Normally, you should use “T pieces” to interconnect the machines, so that you end up with something that looks like:

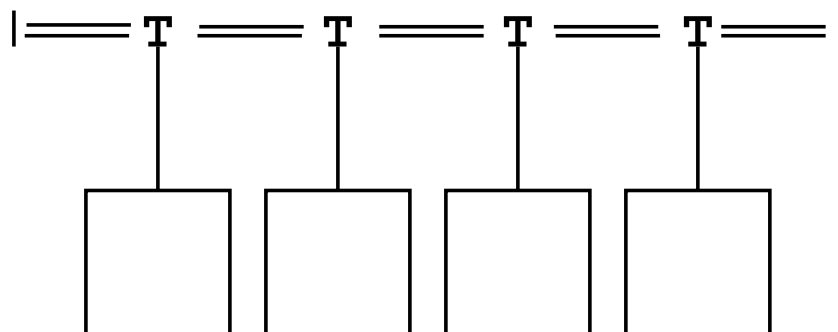


Figure 11-3. 10base2 Ethernet Cabling

where the | at either end represents a terminator, the '=====' represents a length of coaxial cable with BNC plugs at either end, and the T represents a "T piece" connector. You should keep the length of cable between the "T piece" and the actual Ethernet card in the PC as short as possible, ideally the "T piece" will be plugged directly into the Ethernet card.

11.9.4. Twisted Pair Ethernet Cable

If you only have two twisted pair Ethernet cards and you wish to connect them, you do not require a hub. You can cable the two cards directly together. A diagram showing how to do this is included in the Ethernet-HOWTO (<http://linuxdoc.org/HOWTO/Ethernet-HOWTO.html>)

Appendix A. GNU Free Documentation License

A.1. GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or proces-

sing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See Copyleft (<http://www.gnu.org/copyleft/>).

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

A.2. How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (c) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software

Foundation; with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Glossary

APM

Advanced Power Management. A feature used by some *BIOS* es in order to make the machine enter a standby state after a given period of inactivity. On laptops, APM is also responsible for reporting the battery status and, if it is supported, the estimated remaining battery life.

ASCII

American Standard Code for Information Interchange. The standard code used for storing characters, including control characters, on a computer. Many 8-bit codes (such as ISO 8859-1, the Linux default character set) contain ASCII as their lower half.

See Also: ISO 8859.

BSD

Berkeley Software Distribution. A *Unix* variant developed at the Berkeley University computing department. This version has always been considered more advanced technically than the others, and has brought many innovations to the computing world in general and to *Unix* in particular.

CHAP

Challenge-Handshake Authentication Protocol: protocol used by ISP s to authenticate their clients. In this scheme, a value is sent to the client (the machine who connects), the client calculates a *hash* from this value which it sends to the server, and the server compares the *hash* with the one it has calculated. It is different from PAP in that it re-authenticates on a periodic basis after the initial authentication.

See Also: PAP.

CIFS

Common Internet FileSystem The predecessor of the SMB filesystem, used on *DOS* systems.

DHCP

Dynamic Host Configuration Protocol. A protocol designed for machines on a local network to dynamically get an IP address from a DHCP server.

DMA

Direct Memory Access. A facility used on the *PC* architecture which allows for a peripheral to read or write from main memory without the help of the CPU . PCI peripherals use bus mastering and do not need DMA .

DNS

Domain Name System. The distributed name/address mechanism used in the Internet. This mechanism allows you to map a domain name to an IP address, which is what lets you look up a site by domain name without knowing the IP address of the site. DNS also allows reverse lookup, that is you can get a machine's IP address from its name.

DPMS

Display Power Management System. Protocol used by all modern monitors in order to manage power saving features. Monitors supporting these features are commonly called "green monitors".

ELF

Executable and Linking Format. This is the binary format used by most *GNU/Linux* distributions nowadays.

ext2

short for the "Extended 2 filesystem". This is *GNU/Linux* ' native filesystem and has all characteristics of any *Unix* filesystem: support for special files (character devices, symbolic links...), file permissions and ownership, and so on.

FAQ

Frequently Asked Questions. A document containing a series of questions/answers about a specific topic. Historically, FAQ s appeared in newsgroups, but this sort of document now appears on various web sites, and even commercial products have their FAQ . Generally, they are very good sources of information.

FAT

File Allocation Table. Filesystem used by *DOS* and *Windows* .

FDDI

Fiber Distributed Digital Interface. A high-speed network physical layer, which uses optical fiber for communication. Only used on big networks, mainly because of its price.

FHS

Filesystem Hierarchy Standard. A document containing guidelines for a coherent file tree organization on *Unix* systems. **Mandrake Linux** complies with this standard in most aspects.

FIFO

First In, First Out. A data structure or hardware buffer from which items are taken out in the order they were put in. *Unix* pipes are the most common examples of FIFO s.

FTP

File Transfer Protocol. This is the standard Internet protocol used to transfer files from one machine to another.

GFDL

The GNU Free Documentation License. It is the license that applies to all **Mandrake Linux** documentation.

GIF

Graphics Interchange Format. An image file format, widely used on the web . GIF images may be compressed or animated. Due to copyright problems it is a bad idea to use them, replace them as much as possible by the far advanced PNG format instead.

GNU

GNU's Not Unix. The GNU project has been initiated by Richard Stallman at the beginning of the 80s, and aimed at developing a free operating system ("free" as in "free speech"). Currently, all tools are there, except... the kernel. The GNU project kernel, *Hurd* , is not rock solid yet. *Linux* borrows, among others, two things from GNU : its *C* compiler, *gcc* , and its license, the GPL .

See Also: GPL.

GPL

General Public License. The license of the *GNU/Linux* kernel, it goes the opposite way of all proprietary licenses in that it gives no restriction as to copying, modifying and redistributing the software, as long as the source code is made available. The only restriction, if one can call it that, is that the persons to which you redistribute it must also benefit from the same rights.

GUI

Graphical User Interface. Interface to a computer consisting of windows with menus, buttons, icons and so on. The vast majority prefer a GUI over a CLI (*Command Line Interface*) for ease of use, even though the latter is more versatile.

HTML

HyperText Markup Language. The language used to create web documents.

HTTP

HyperText Transfer Protocol. The protocol used to connect to websites and retrieve HTML documents or files.

IDE

Integrated Drive Electronics. The most widely used bus on today's *PC* s for hard disks. An IDE bus can contain up to two devices, and the speed of the bus is limited by the device on the bus which has the slower command queue (and not the slower transfer rate!).

See Also: ATAPI.

IP masquerading

is when you use a firewall to hide your computer's true IP address from the outside. Typically any outside network connections you make beyond the firewall will inherit the firewall's IP address. This is useful in situations where you may have a fast Internet connection with only one IP address but wish to use more than one computer that have internal network IP addresses assigned.

IRC

Internet Relay Chat. One of the few Internet standards for live speech. It allows for channel creation, private talks, and also file exchange. It is also designed to be able to make servers connect to each other, which is why several IRC networks exist today: **Undernet**, **DALnet**, **EFnet** to name a few.

IRC channels

are the "places" inside IRC servers where you can chat with other people. Channels are created in IRC servers and users join those channels so they can communicate with each other. Messages written on an

channel are only visible to those people connected to that channel. Two or more users can also create a “private” channel so they don’t get disturbed by other users. Channel names begin with a #.

ISA

Industry Standard Architecture. The very first bus used on *PC* s, it is slowly being abandoned in favor of the PCI bus. Some hardware manufacturers still use it, though. It is still very common that SCSI cards supplied with scanners, CD writers, ... are ISA . Too bad.

ISDN

Integrated Services Digital Network. A set of communication standards for allowing a single wire or optical fiber to carry voice, digital network services and video. It has been designed in order to eventually replace the current phone system, known as PSTN (*Public Switched Telephone Network*) or POTS (*Plain Ole Telephone Service*). Technically ISDN is a circuit switched data network.

ISO

International Standards Organization. A group of companies, consultants, universities and other sources which enumerates standards in various topics, including computing. The papers describing standards are numbered. The standard number iso9660, for example, describes the filesystem used on CD-ROM s.

ISP

Internet Service Provider. A company which sells Internet access to its customers, whether the access is over telephone lines or dedicated lines.

JPEG

Joint Photographic Experts Group. Another very common image file format. JPEG is mostly suited for compressing real-world scenes, and does not work very well on non-realistic images.

LAN

Local Area Network. Generic name given to a network of machines connected to the same physical wire.

LDP

Linux Documentation Project. A nonprofit organization which maintains *GNU/Linux* documentation. Its mostly known documents are *HOWTOs* , but it also maintains FAQ s, and even a few books.

MBR

Master Boot Record. Name given to the first sector of a bootable hard drive. The MBR contains the code used to load the operating system into memory or a bootloader (such as *LILO*), and the partition table of that hard drive.

MIME

Multipurpose Internet Mail Extensions. A string of the form type/subtype describing the contents of a file attached in an e-mail. This allows MIME -aware mail clients to define actions depending on the type of the file.

MPEG

Moving Pictures Experts Group. An ISO committee which generates standards for video and audio compression. MPEG is also the name of their algorithms. Unfortunately, the license for this format is very restrictive, and as a consequence there are still no *Open Source* MPEG players...

NCP

NetWare Core Protocol. A protocol defined by **Novell** to access Novell NetWare file and print services.

NFS

Network FileSystem. A network filesystem created by **Sun Microsystems** in order to share files across a network in a transparent way.

NIC

Network Interface Controller. An adapter installed in a computer which provides a physical connection to a network, such as an *Ethernet* card.

NIS

Network Information System. NIS was also known as “Yellow Pages”, but **British Telecom** holds a copyright on this name. NIS is a protocol designed by **Sun Microsystems** in order to share common information across a NIS **domain**, which can gather a whole LAN , part of this LAN or several LAN s. It can export password databases, service databases, groups information and more.

PAP

Password Authentication Protocol. A protocol used by many ISP s to authenticate their clients. In this scheme, the client (you) sends an identifier/password pair to the server, which is not encrypted.
See Also: CHAP.

PCI

Peripheral Components Interconnect. A bus created by **Intel** and which is today the standard bus for *PC* architectures, but other architectures use it too. It is the successor of ISA , and it offers numerous services: device identification, configuration information, IRQ sharing, bus mastering and more.

PCMCIA

Personal Computer Memory Card International Association. More and more commonly called "PC Card" for simplicity reasons, this is the standard for external cards attached to a laptop: modems, hard disks, memory cards, *Ethernet* cards, and more. The acronym is sometimes humorously expanded to *People Cannot Memorize Computer Industry Acronyms...*

PNG

Portable Network Graphics. Image file format created mainly for web use, it has been designed as a patent-free replacement for GIF and also has some additional features.

Plug'n'Play

Plug'n'Play. First an add-on for ISA in order to add configuration information for devices, it has become a more widespread term which groups all devices able to report their configuration parameters. As such, all PCI devices are Plug'n'Play.

POP

Post Office Protocol. The common protocol used for retrieving mail from an ISP .

PPP

Point to Point Protocol. This is the protocol used to send data over serial lines. It is commonly used to send IP packets to the Internet, but it can also be used with other protocols such as Novell's IPX protocol.

RAID

Redundant Array of Independent Disks. A project initiated at the computing science department of Berkeley University, in which the storage of data is spread along an array of disks using different schemes. At first, this was implemented using floppy drives, which is why the acronym originally stood for *Redundant Array of Inexpensive Disks*.

RAM

Random Access Memory. Term used to identify a computer's main memory. The "Random" here means that any part of the memory can be directly accessed...

RFC

Request For Comments. RFC s are the official Internet standard documents, published by the IETF (*Internet Engineering Task Force*). They describe all protocols, their usage, their requirements and so on. When you want to learn how a protocol works, pick up the corresponding RFC .

RPM

Redhat Package Manager. A packaging format developed by **Red Hat** in order to create software packages, it is used in many *GNU/Linux* distributions, including **Mandrake Linux** .

SCSI

Small Computers System Interface. A bus with a high throughput designed to allow for several types of peripherals. Unlike IDE , a SCSI bus is not limited by the speed at which the peripherals accept commands. Only high-end machines integrate a SCSI bus directly on the motherboard, *PC* s need add-on cards.

SMB

Server Message Block. Protocol used by *Windows* machines (*9x* or *NT*) for file and printer sharing across a network.
See Also: CIFS.

SMTP

Simple Mail Transfer Protocol. This is the common protocol for transferring email. Mail Transfer Agents such as *sendmail* or *postfix* use SMTP . They are sometimes also called SMTP servers.

SVGA

Super Video Graphics Array. The video display standard defined by VESA for the *PC* architecture. The resolution is 800x 600 x 16 colors.

TCP

Transmission Control Protocol. This is the most common reliable protocol that uses IP to transfer network packets. TCP adds the necessary checks on top of IP to make sure that packets are delivered. Unlike UDP, TCP works in connected mode, which means that two machines must establish a connection before exchanging data.

URL

Uniform Resource Locator. A string with a special format used to identify a resource on the Internet in a unique way. The resource can be a file, a server or other item. The syntax for a URL is
protocol://server.name[:port]/path/to/resource.

When only a machine name is given and the protocol is http://, it defaults to retrieving the file index.html on the server.

VESA

Video Electronics Standards Association. An industry standards association aimed at the *PC* architecture. It is the author of the SVGA standard, for example.

WAN

Wide Area Network. This network, although similar to a LAN connects computers on a network that is not physically connected to the same wires and are separated by a greater distance.

account

on a *Unix* system, the combination of a name, a personal directory, a password and a *shell* which allows a person to connect to this system.

alias

mechanism used in a *shell* in order to make it substitute one string for another before executing the command. You can see all aliases defined in the current session by typing alias at the prompt.

arp

Address Resolution Protocol. The Internet protocol used to dynamically map an Internet address to physical (hardware) addresses on local area networks. This is limited to networks that support hardware broadcasting.

assembly language

is the programming language that is closest to the computer, which is why it's called a "low level" programming language. Assembly has the advantage of speed since assembly programs are written in terms of processor instructions so little or no translation is needed when generating executables. Its main disadvantage is that it is processor (or architecture) dependent. Writing complex programs is very time-consuming as well. So, assembly is the fastest programming language, but it isn't portable between architectures.

ATAPI

("AT Attachment Packet Interface") An extension to the ATA specification ("Advanced Technology Attachment", more commonly known as IDE, *Integrated Drive Electronics*) which provides additional commands to control CDROM drives and magnetic tape drives. IDE controllers equipped with this extension are also referred to as EIDE (*Enhanced IDE*) controllers.

ATM

This is an acronym for **Asynchronous Transfer Mode**. An ATM network packages data into standard size blocks (53 bytes: 48 for the data and 5 for the header) which it can convey efficiently from point to point. ATM is a circuit switched packet network technology oriented towards high speed (multi-megabits) optical networks.

atomic

a set of operations is said to be atomic when it executes all at once, and cannot be preempted.

background

in *shell* context, a process is running in the background if you can type commands while said process is running.

See Also: job, foreground.

backup

is a means of saving your important data to a safe medium and location. Backups should be done regularly, especially with more critical information and configuration files (the prime directories to backup are `/etc`, `/home` and `/usr/local`). Traditionally, many people use `tar` with `gzip` or `bzip2` to backup directories and files. You can use these tools or programs like `dump` and `restore`, along with many other free or commercial backup solutions.

batch

is a processing mode where jobs are submitted to the processor, and then the processor executes them one after the other till it executes the last one and it's ready for another list of processes.

beep

is the little noise your computer's speaker does to warn you of some ambiguous situation when you're using command completion and, for example, there's more than one possible choice for completion. There might be other programs that make beeps to let you know of some particular situation.

beta testing

is the name given to the process of testing the beta version of a program. Programs usually get released in alpha and beta states for testing prior to final release.

bit

stands for *BI*nary *di*giT. A single digit which can take the values 0 or 1, because calculation is done in base two.

block mode files

files whose contents are buffered. All read/write operations for such files go through buffers, which allows for asynchronous writes on the underlying hardware, and for reads, not to read again what is already in a buffer.

See Also: buffer, buffer cache, character mode files.

boot

the procedure taking place when a computer is switched on, where peripherals are recognized one after the other, and where the operating system is loaded into memory.

bootdisk

a bootable floppy disk containing the code necessary to load the operating system from the hard disk (sometimes it is self-sufficient).

bootloader

is a program that starts the operating system. Many bootloaders give you the opportunity to load more than one operating system by letting you choose between them at a boot menu. Bootloaders like *grub* are popular because of this feature and are very useful in dual- or multi-boot systems.

buffer

a small portion of memory with a fixed size, which can be associated with a block mode file, a system table, a process and so on. The coherency of all buffers is maintained by the buffer cache.

See Also: buffer cache.

buffer cache

a crucial part of an operating system kernel, it is in charge of keeping all buffers up-to-date, shrinking the cache when needed, clearing unneeded buffers and more.

See Also: buffer.

bug

illogical or incoherent behavior of a program in a special case, or a behavior which does not follow the documentation or accepted standards issued for the program. Often, new features introduce new bugs in a program. Historically, this term comes from the old days of punch cards: a bug (the insect!) slipped into a hole of a punch card and, as a consequence, the program misbehaved. Ada Lovelace, having discovered this, declared "It's a bug!", and since then the term has remained.

byte

eight consecutive bits, interpreted in base two as a number between 0 and 255.

See Also: bit.

case

when taken in the context of strings, the case is the difference between lowercase letters and uppercase (or capital) letters.

character mode files

files whose content is not buffered. When associated to physical devices, all input/output on these devices is performed immediately. Some special character devices are created by the operating system (`/dev/zero`, `/dev/null` and others). They correspond to data flows.

See Also: block mode files.

client

program or computer that periodically connects to another program or computer to give it orders or ask for information. In the case of **peer to peer** systems such as SLIP or PPP the client is taken to be the end that initiates the connection and the remote end, being called, is taken to be the server. It is one of the components of a **client/server system**.

client/server system

system or protocol consisting of a **server** and one or several **clients**.

command line

what is provided by a shell and allows the user to type commands directly. Also subject of an eternal “flame war” between its supporters and its detractors :-)

command mode

under *Vi* or one of its clones, it is the state of the program in which pressing a key (this above all regards letters) will not insert the character in the file being edited, but instead perform an action specific to the said key (unless the clone has remappable commands and you have customized your configuration). You may get out of it typing one of the “back to insertion mode” commands: **i**, **I**, **a**, **A**, **s**, **S**, **o**, **O**, **c**, **C**, ...

compilation

is the process of translating source code that is human readable (well, with some training) and that is written in some programming language (*C*, for example) into a binary file that is machine readable.

completion

ability of a *shell* to automatically expand a substring to a filename, user name or other, as long as there is a match.

compression

is a way to shrink files or decrease the number of characters sent over a communications connection. Some file compression programs include *compress*, *zip*, *gzip*, and *bzip2*.

console

is the name given to what used to be called terminals. They were the users machines (a screen plus a keyboard) connected to one big central mainframe. On *PC* s, the physical terminal is the keyboard and screen.

See Also: virtual console.

cookies

temporary files written on the local hard disk by a remote web server. It allows for the server to be aware of a user's preferences when this user connects again.

datagram

A datagram is a discrete package of data and headers which contain addresses, which is the basic unit of transmission across an IP network. You might also hear this called a “packet”.

dependencies

are the stages of compilation that need to be satisfied before going on to other compilation stages in order to successfully compile a program.

desktop

If you're using the X Window System, the desktop is the place on the screen inside which you work and upon which your windows and icons are displayed. It is also called the background, and is usually filled with a simple color, a gradient color or even an image.

See Also: virtual desktops.

directory

Part of the filesystem structure. Within a directory, files or other directories are stored. Sometimes there are sub-directories (or branches) within a directory. This is often referred to as a directory tree. If you want to see what's inside another directory, you will either have to list it or change to it. Files inside a directory are referred to as leaves while sub-directories are referred to as branches. Directories follow the same restrictions as files although the permissions mean different things. The special directories `.` and `..` refer to the directory itself and to the parent directory respectively.

discrete values

are values that are non-continuous. That is, there's some kind of "spacing" between two consecutive values.

distribution

is a term used to distinguish one *GNU/Linux* manufacturers product from another. A distribution is made up of the core Linux kernel and utilities, as well as installation programs, third-party programs, and sometimes proprietary software.

DLCI

The DLCI is the Data Link Connection Identifier and is used to identify a unique virtual point to point connection via a Frame Relay network. The DLCI's are normally assigned by the Frame Relay network provider.

echo

is when the characters you type in a username entry field, for example, are shown "as is", instead of showing "*" for each one you type.

editor

is a term typically used for programs that edit text files (aka text editor). The most well-known *GNU/Linux* editors are the GNU Emacs (*Emacs*) editor and the *Unix* editor *Vi*.

email

stands for Electronic Mail. This is a way to send messages electronically between people on the same network. Similar to regular mail (aka snail mail), email needs a destination and sender address to be sent properly. The sender must have an address like "sender@senders.domain" and the recipient must have an address like "recipient@recipients.domain." Email is a very fast method of communication and typically only takes a few minutes to reach anyone, regardless of where in the world they are located. In order to write email, you need an email client like *pine* or *mutt* which are text-mode clients, or GUI clients like *kmail*.

environment

is the execution context of a process. It includes all the information that the operating system needs to manage the process and what the processor needs to execute the process properly.

See Also: process.

environment variables

a part of a process' environment. Environment variables are directly viewable from the *shell*.

See Also: process.

escape

in the shell context, is the action of surrounding some string between quotes to prevent the shell from interpreting that string. For example, when you need to use spaces in some command line and pipe the results to some other command you have to put the first command between quotes ("escape" the command) otherwise the shell will interpret it wrong and won't work as expected.

filesystem

scheme used to store files on a physical media (hard drive, floppy) in a consistent manner. Examples of filesystems are FAT, *GNU/Linux*' ext2fs, iso9660 (used by CD-ROMs) and so on. An example of a virtual filesystem is the `/proc` filesystem.

firewall

a machine or a dedicated piece of hardware which, in the topology of a local network, is the unique connecting point to the outside network, and which filters, or controls the activity on some ports, or makes sure only some specific interfaces may have access to them.

flag

is an indicator (usually a bit) which is used to signal some condition to a program. For example, a filesystem has, among others, a flag indicating if it has to be dumped in a backup, so when the flag is active the filesystem gets backed up, and when it's inactive it doesn't.

focus

the state for a window to receive keyboard events (such as key-presses, key-releases and mouse clicks) unless they are trapped by the window manager.

foreground

in shell context, the process in the foreground is the one which is currently running. You have to wait for such a process to finish in order to be able to type commands again.

See Also: job, background.

Frame Relay

Frame Relay is a network technology ideally suited to carrying traffic that is of bursty or sporadic nature. Network costs are reduced by having many Frame Relay customer sharing the same network capacity and relying on them wanting to make use of the network at slightly different times.

framebuffer

projection of a video card's RAM into the machine's address space. This allows for applications to access the video RAM without the chore of having to talk to the card. All high-end graphical workstations use framebuffers, for example.

full-screen

This term is used to refer to applications that take up the whole visible area of your display.

gateway

link connecting two IP networks.

globbing

in the *shell*, the ability to group a certain set of filenames with a globbing pattern.

See Also: globbing pattern.

globbing pattern

a string made of normal characters and special characters. Special characters are interpreted and expanded by the *shell*.

hardware address

This is a number that uniquely identifies a host in a physical network at the media access layer. Examples of this are **Ethernet Addresses** and **AX.25 Addresses**.

hidden file

is a file which can't be "seen" when doing a `ls` command with no options. Hidden files' filenames begin with a `.` and are used to store the user's personal preferences and configurations for the different programs (s)he uses. For example, *bash*'s command history is saved into `.bash_history`, which is a hidden file.

home directory

often abbreviated by "home", this is the name for the personal directory of a given user.

See Also: account.

host

refers to a computer and is commonly used when talking about computers that are connected on a network.

icon

is a little drawing (normally sized 16x 16, 32x 32, 48x 48 and sometimes 64x 64 pixels) which represents, under a graphical environment, a document, a file or a program.

inode

entry point leading to the contents of a file on a *Unix*-like filesystem. An inode is identified in a unique way by a number, and contains meta-information about the file it refers to, such as its access times, its type, its size, **but not its name!**

insert mode

under *Vi* or one of its clones, it is the state of the program in which pressing a key will insert that character in the file being edited (except pathological cases like the completion of an abbreviation, right justify at the end of the line, ...). One gets out of it pressing the key **Esc** (or **Ctrl-I**).

Internet

is a huge network that connects computers around the world.

IP address

is a numeric address consisting of four parts which identifies your computer on the Internet. IP addresses are structured in a hierarchical manner, with top level and national domains, domains, subdomains and each machine's personal address. An IP address would look something like 192.168.0.1. A machine's personal address can be one of two types: static or dynamic. Static IP addresses are addresses that never change, but rather are permanent. Dynamic IP addresses mean your IP address will change with each new connection to the network. Dial-up and cable modem users typically have dynamic IP addresses while some DSL and other high-speed connections provide static IP addresses.

ISO 8859

The ISO 8859 standard includes several 8-bit extensions to the ASCII character set. Especially important is ISO 8859-1, the "Latin Alphabet No. 1", which has become widely implemented and may already be seen as the de facto standard ASCII replacement.

ISO 8859-1 supports the following languages: Afrikaans, Basque, Catalan, Danish, Dutch, English, Faroese, Finnish, French, Galician, German, Icelandic, Irish, Italian, Norwegian, Portuguese, Scottish, Spanish, and Swedish.

Note that the ISO 8859-1 characters are also the first 256 characters of ISO 10646 (Unicode). However, it lacks the EURO symbol and does not fully cover Finnish and French. ISO 8859-15 is a modification of ISO 8859-1 that covers these needs.

See Also: ASCII.

job

in *shell* context, a job is a process running in the background. You can have several jobs in the same shell and control these jobs.

See Also: foreground, background.

kernel

is the guts of the operating system. The kernel is responsible for allocating resources and separating processes from each other. It handles all of the low-level operations that allow programs to talk directly to the hardware on your computer, manages the buffer cache and so on.

kill ring

under *Emacs*, it is the set of text areas cut or copied since the beginning of the editor, which may be recalled to be inserted again, and which is organized like a ring.

launch

is the action of invoking, or starting, a program.

library

is a collection of procedures and functions in binary form to be used by programmers in their programs (as long as the library's license allows them to do so). The program in charge of loading shared libraries at run time is called the dynamic linker.

link

reference to an inode in a directory, therefore giving a (file) name to the inode. Examples of inodes which don't have a link (and hence have no name) are: anonymous pipes (as used by the shell), sockets (aka network connections), network devices and so on.

linkage

last stage of the compile process, which consists in linking together all object files in order to produce an executable file, and matches unresolved symbols with dynamic libraries (unless a static linkage has been asked, in which case the code of these symbols will be included in the executable).

Linux

is a *Unix*-like operating system which runs on a variety of different computers, and is free for anyone to use and modify. Linux (the kernel) was written by Linus Torvalds.

login

connection name for a user on a *Unix* system, and the action to connect.

lookup table

is a table that puts in correspondance codes (or tags) and their meaning. It is often a data file used by a program to get further information about a particular item.

For example, *harddrake* uses such a table to know what a manufacturer's product code means. This is one line from the table, giving information about item CTL0001

```
CTL0001 sound    sb      Creative Labs    SB16 \
                HAS_OPL3|HAS_MPU401|HAS_DMA16|HAS_JOYSTICK
```

loopback

virtual network interface of a machine to itself, allowing the running programs not to have to take into account the special case where two network entities are in fact the same machine.

major

number specific to the device class.

manual page

a small document containing the definition of a command and its usage, to be consulted with the `man` command. The first thing one should (learn how to) read when hearing of a command he doesn't know :-)

minor

number identifying the specific device we are talking about.

mount point

is the directory where a partition or another device is attached to the *GNU/Linux* filesystem. For example, your CD-ROM is mounted in the `/mnt/cdrom` directory, from where you can explore the contents of any mounted CD s.

mounted

A device is mounted when it is attached to the *GNU/Linux* filesystem. When you mount a device you can browse its contents. This term is partly obsolete as with the "supermount" feature, users do not need any more to manually mount removable medias.

See Also: mount point.

MSS

The Maximum Segment Size (**MSS**) is the largest quantity of data that can be transmitted at one time. If you want to prevent local fragmentation MSS would equal MTU-IP header.

MTU

The Maximum Transmission Unit (**MTU**) is a parameter that determines the largest datagram than can be transmitted by an IP interface without it needing to be broken down into smaller units. The MTU should be larger than the largest datagram you wish to transmit unfragmented. Note, this only prevents fragmentation locally, some other link in the path may have a smaller MTU and the datagram will be fragmented there. Typical values are 1500 bytes for an ethernet interface, or 576 bytes for a SLIP interface.

multitasking

the ability for an operating system to share CPU time between several processes. At low level, this is also known as multiprogramming. Switching from one process to another requires that all the current process context be saved and restored when this process is elected again. This operation is called context switch, and on Intel, is done 100 times per second; therefore it's fast enough so that a user has the illusion that the operating system runs several applications at the same time. There are two types of multitasking: preemptive multitasking is where the operating system is responsible for taking away the CPU and pass it to another process; cooperative multitasking is where the process itself gives back the CPU. The first variant is, obviously, the better choice because no program can consume the entire CPU time and block other processes. *GNU/Linux* does preemptive multitasking. The policy to select which process should be run, depending on several parameters, is called scheduling.

multiuser

is used to describe an operating system which allows multiple users to log into and use the system at the exact same time, each being able to do their own work independent of other users. A multitasking operating system is required to provide multiuser support. *GNU/Linux* is both a multitasking and multiuser operating system, as any *Unix* system for that matter.

named pipe

a *Unix* pipe which is linked, as opposed to pipes used in shells.

See Also: pipe, link.

naming

a word commonly used in computing for a method to identify objects. You will often hear of “naming conventions” for files, functions in a program and so on.

newsgroups

discussion and news areas that can be accessed by a news or USENET client to read and write messages specific to the topic of the newsgroup. For example, the newsgroup `alt.os.linux.mandrake` is an alternate newsgroup (alt) dealing with the Operating System (os) *GNU/Linux*, and specifically, **Mandrake Linux** (mandrake). Newsgroups are broken down in this fashion to make it easier to search for a particular topic.

null, character

the character or byte number 0, it is used to mark the end of a string.

object code

is the code generated by the compilation process to be linked with other object codes and libraries to form an executable file. Object code is machine readable.

See Also: compilation, linkage.

on the fly

Something is said to be done “on the fly” when it’s done along with something else, without you noticing it or explicitly asking for it.

open source

is the name given to free source code of a program that is made available to development community and public at large. The theory behind this is that allowing source code to be used and modified by a broader group of programmers will ultimately produce a more useful product for everyone. Some popular open source programs include *Apache*, *sendmail* and *GNU/Linux*.

operating system

is the interface between the applications and the underlying hardware. The tasks for any operating system are primarily to manage all of the machine specific resources. On a *GNU/Linux* system, this is done by the kernel and loadable modules. Other well-known operating systems include *AmigaOS*, *MacOS*, *FreeBSD*, *OS/2*, *Unix*, *Windows NT*, and *Windows 9x*.

owner

in the context of users and their files, the owner of a file is the user who created that file.

owner group

in the context of groups and their files, the owner group of a file is the group to which the user who created that file belongs to.

pager

program displaying a text file one screenful at a time, and making it easy to move back and forth and search for strings in this file. We advise you to use `less`.

password

is a secret word or combination of words or letters that is used to secure something. Passwords are used in conjunction with user logins to multi-user operating systems, web sites, FTP sites, and so forth. Passwords should be hard-to-guess phrases or alphanumeric combinations, and should never be based on common dictionary words. Passwords ensure that other people cannot log into a computer or site with your account.

patch, to patch

file holding a list of corrections to issue to a source code in order to add new features, to remove bugs, or to modify it according to one’s wishes and needs. The action consisting of the application of these corrections to the archive of source code (aka “patching”).

path

is an assignment for files and directories to the filesystem. The different layers of a path are separated by the “slash” or `’/’` character. There are two types of paths on *GNU/Linux* systems. The **relative** path is the

position of a file or directory in relation to the current directory. The **absolute** path is the position of a file or directory in relation to the root directory.

pipe

a special *Unix* file type. One program writes data into the pipe, and another program reads the data at the other end. *Unix* pipes are FIFO s, so the data is read at the other end in the order it was sent. Very widely used with the shell. See also **named pipe**.

pixmap

is an acronym for “pixel map”. It’s another way of referring to bitmapped images.

plugin

add-on program used to display or play some multimedia content found on a web document. It can usually be easily downloaded if your browser is not yet able to display or play that kind of information.

porting

a program is translating that program in such a way that it can be used in a system it was not originally intended for, or it can be used in “similar” systems. For example, to be able to run a *Windows* -native program under *GNU/Linux* (natively), it must first be ported to *GNU/Linux* .

precedence

dictates the order of evaluation of operands in an expression. For example: If you have $4 + 3 * 2$ you get 10 as the result, since the product has more precedence than the addition. If you want to evaluate the addition first, then you have to add parenthesis like this $(4 + 3) * 2$, and you get 14 as the result since the parenthesis have more precedence than the addition and the product, so the operations in parenthesis get evaluated first.

preprocessors

are compilation directives that instruct the compiler to replace those directives for code in the programming language used in the source file. Examples of *C* ’s preprocessors are `#include`, `#define`, etc.

process

in the operating system context, a process is an instance of a program being executed along with its environment.

prompt

in a *shell* , this is the string before the cursor. When you see it, you can type your commands.

protocol

Protocols organize the communication between different machines across a network, either using hardware or software. They define the format of transferred data, whether one machine controls another, etc. Many well-known protocols include HTTP, FTP, TCP, and UDP.

proxy

a machine which sits between a network and the Internet , whose role is to speed up data transfers for the most widely used protocols (HTTP and FTP , for example). It maintains a cache of previous demands, which avoids the cost of asking for the file again if another machine asks for the same thing. Proxies are very useful on low bandwidth networks (such as modem connections). Sometimes the proxy is the only machine able to access outside the network.

pulldown menu

it is a menu that is “rolled” with a button in some of its corners. When you press that button, the menu “unrolls” itself showing you the full menu.

quota

is a method for restricting disk usage and limits for users. Administrators can restrict the size of home directories for a user by setting quota limits on specific filesystems.

read-only mode

for a file means that the file cannot be written to. You can read its contents but you can’t modify them.
See Also: read-write mode.

read-write mode

for a file, it means that the file can be written to. You can read its contents and modify them.
See Also: read-only mode.

regular expression

a powerful theoretical tool which is used to search and match text strings. It lets one specify patterns these strings must obey. Many *Unix* utilities use it: *sed* , *awk* , *grep* , *perl* among others.

root

is the superuser of any *Unix* system. Typically root (aka the system administrator) is the person responsible for maintaining and supervising the *Unix* system. This person also has complete access to everything on the system.

root directory

This is the top level directory of a filesystem. This directory has no parent directory, thus *'..'* for root points back to itself. The root directory is written as *'/'*.

root filesystem

This is the top level filesystem. This is the filesystem where *GNU/Linux* mounts its root directory tree. It is necessary for the root filesystem to reside in a partition of its own, as it is the basis for the whole system. It holds the root directory.

route

Is the path that your datagrams take through the network to reach their destination. Is the path between one machine and another in a network.

run level

is a configuration of the system software that only allows certain selected processes to exist. Allowed processes are defined, for each runlevel, in the file */etc/inittab*. There are eight defined runlevels: 0, 1, 2, 3, 4, 5, 6, S and switching among them can only be achieved by a privileged user by means of executing the commands *init* and *telinit*.

script

shell scripts are sequences of commands to be executed as if they were entered in the console one after the other. *shell* scripts are *Unix* 's (somewhat) equivalent of *DOS* batch files.

security levels

Mandrake Linux 's unique feature that allows you to set different levels of restrictions according to how secure you want to make your system. There are 6 predefined levels ranging from 0 to 5, where 5 is the tightest security. You can also define your own security level.

server

program or computer that provides a feature or service and awaits the connections from **clients** to execute their orders or give them the information they ask. In the case of **peer to peer** systems such as **slip** or **ppp** the server is taken to be the end of the link that is called and the end calling is taken to be the client. It is one of the components of a **client/ server system**.

shadow passwords

a password management suite on *Unix* systems in which the file containing the encrypted passwords is not world-readable, whereas it is when using the normal password system. It also offers other features such as password aging.

shell

The *shell* is the basic interface to the operating system kernel and is what provides the command line where users enter commands to run programs and system commands. All shells provide a scripting language which can be used to automate tasks or simplify often-used complex tasks. These *shell* scripts are similar to batch files from the *DOS* operating system, but are much more powerful. Some example shells are *bash* , *sh* , and *tcsh* .

single user

is used to describe a state of an operating system, or even an operating system itself, that only allows a single user to log into and use the system at any time.

site dependent

means that the information used by programs like *imake* and *make* to compile some source file depends on the site, the computer architecture, the computer's installed libraries, and so on.

socket

file type corresponding to any network connection.

soft links

see “symbolic links”.

standard error

the file descriptor number 2, opened by every process, used by convention to print error messages to the terminal screen by default.

See Also: standard input, standard output.

standard input

the file descriptor number 0, opened by every process, used by convention as the file descriptor from which the process receives data.

See Also: standard error, standard output.

standard output

the file descriptor number 1, opened by every process, used by convention as the file descriptor in which the process prints its output.

See Also: standard error, standard input.

streamer

is a device that takes “streams” (not interrupted or divided in shorter chunks) of characters as its input. A typical streamer is a tape drive.

switch

Switches are used to change the behavior of programs, and are also called command-line options or arguments. To determine if a program has optional switches that can be used, read the *man* pages or try to pass the `--help` switch to the program (ie. `program --help`).

symbolic links

special files, containing nothing but a string that makes reference to another file. Any access to them is the same as accessing the file whose name is the referenced string, which may or may not exist, and the path to which can be given in a relative or an absolute way.

target

is the object of compilation, i.e. the binary file to be generated by the compiler.

telnet

creates a connection to a remote host and allows you to log into the machine, provided you have an account. Telnet is the most widely-used method of remote logins, however there are better and more secure alternatives, like `ssh`.

theme-able

a graphical application is theme-able if it is able to change its appearance in real time. Many window managers are theme-able as well.

traverse

for a directory on a *Unix* system, this means that the user is allowed to go through this directory, and possibly to directories under it. This requires that the user has the execute permission on this directory.

username

is a name (or more generally a word) that identifies a user in a system. Each username is attached to a unique and single UID (user ID)

See Also: login.

variables

are strings that are used in *Makefile* files to be replaced by their value each time they appear. Usually they are set at the beginning of the *Makefile*. They are used to simplify *Makefile* and source files tree management.

More generally, variables in programming, are words that refer to other entities (numbers, strings, tables, etc.) that are likely to vary while the program is executing.

verbose

For commands, the verbose mode means that the command reports to standard (or possibly error) output all the actions it performs and the results of those actions. Sometimes, commands have a way to define the “verbosity level”, which means that the amount of information that the command will report can be controlled.

virtual console

is the name given to what used to be called terminals. On *GNU/Linux* systems, you have what are called virtual consoles which enable you to use one screen or monitor for many independently running sessions. By default, you have six virtual consoles which can be reached by pressing **ALT-F1** through **ALT-F6**. There is a seventh virtual console by default, **ALT-F7**, which will permit you to reach a running X Window System. In X, you can reach the text console by pressing **CTRL-ALT-F1** through **CTRL-ALT-F6**.
See Also: console.

virtual desktops

In the X Window System, the window manager may provide you several desktops. This handy feature allows you to organize your windows, avoiding the problem of having dozens of them stacked on top of each other. It works as if you had several screens. You can switch from one virtual desktop to another in a manner that depends on the window manager you're using.

See Also: window manager, desktop.

wildcard

The '*' and '?' characters are used as wildcard characters and can represent anything. The '*' represents any number of characters, including no characters. The '?' represents exactly one character. Wildcards are often used in regular expressions.

window

In networking, the **window** is the largest amount of data that the receiving end can accept at a given point in time.

window manager

the program responsible for the "look and feel" of a graphical environment, dealing with window bars, frames, buttons, root menus, and some keyboard shortcuts. Without it, it would be hard or impossible to have virtual desktops, to resize windows on the fly, to move them around, ...